The USTC-iFlytek Systems for CHiME-5 Challenge

Jun Du[1], *Tian Gao*[1], *Lei Sun*[1], *Feng Ma*[2], *Yi Fang*[2], *Di-Yuan Liu*[2], *Qiang Zhang*[2], *Xiang Zhang*[2], *Hai-Kun Wang*[2], *Jia Pan*[2], *Jian-Qing Gao*[2], *Chin-Hui Lee*[3], *Jing-Dong Chen*[4]

[1]University of Science and Technology of China, Hefei, Anhui, P. R. China
[2]iFlytek Research, iFlytek Co., Ltd., Hefei, Anhui, P. R. China
[3]Georgia Institute of Technology, Atlanta, Georgia, USA
[4]Northwestern Polytechnical University, Shanxi, P. R. China

jundu@ustc.edu.cn, gtian09@mail.ustc.edu.cn,sunlei17@mail.ustc.edu.cn,chl@ece.gatech.edu

Abstract

This report describes our submission to the fifth CHiME Challenge. The main technical points of our system include the deep learning based speech enhancement and separation, training data augmentation via different versions of the official training data, SNR-based array selection, front-end model fusion, acoustic model fusion, and language model fusion. Tested on the development test set, our best system for single-array track using official LM has yielded a 37.7% WER relative reduction over the results given by official baseline system.

1. System Overview

CHiME-5[1] challenge features a single-array track and a multiple-array track, and we participate both of them. A unified framework of training process is given in Figure 1. As we can see, it contains several main parts including deep-learning based speech saparation (SS Model), speech enhancement (SE Model), multi-channel based WPE denoising, beamforming and acoustic model training. For the front-end, we fisrt conduct data simulation to augment data size by estimating impulse responses between binaural data and far-field data. Meanwhile, we apply a conventional multi-channel noise reduction using log-spectral amplitude [2] which is based on generalized weighted prediction error (GWPE) [3] and independent vector analysis (IVA) [4]. With the denoised data, we can build the following speech enhancement model and speech separation model which are both based on deep-learning techniques. After all, each method of these frond-end techniques can provide processed data of official original training data, add increase the diversity of original data. Using the final augmented data, five types of acoustic model are trained as the back-end system.

The testing phase can be divided into two scenarios: single-array track and multiple-array track. For each track, two separate rankings will be produced: *Rank A* compares systems which are based on conventional acoustic modeling and using the supplied official language model, while *Rank B* has no such limitations. Speaking of single-track, the same conventional multi-channel preprocessing described above is first conducted. Then, speaker-dependent SS models are trained with the denoised data for each speaker among the test set. The outputs of SS model and SE model, are integrated together to provide necessary initialization information for beamforming. After that, the beamformed speech is sent to back-end acoustic models for recognition. What's more, several acoustic models are fused at the state-level. However, for *Rank B*, the first-pass decoding is performed with the HMM and 3-gram to generate the lattice as the hypotheses, which are served for the second-pass decoding with a LSTM-based LM.

In multiple-array track, we first use the SE Model to estimate the signal to noise ratio (SNR) for each array, two arrays with maximum SNRs are selected. The rest procedures are almost the same with single-arrary track, conventional multi-channel preprocessing is first used. Then, speaker-dependent SS model are trained with the denoised data for each speaker among the test set. The outputs of SS model and SE model are integrated together to provide necessary initialization information for beamforming. The beamformed speech is sent to back-end acoustic model for recognition. Multiple acoustic models of both two selected arrays are fused at the state-level. For *Rank B*, the first-pass decoding is performed with the HMM and 3-gram to generate the lattice as the hypotheses, which are served for the second-pass decoding with a LSTM-based LM.

More details will be introduced in the following subsections.

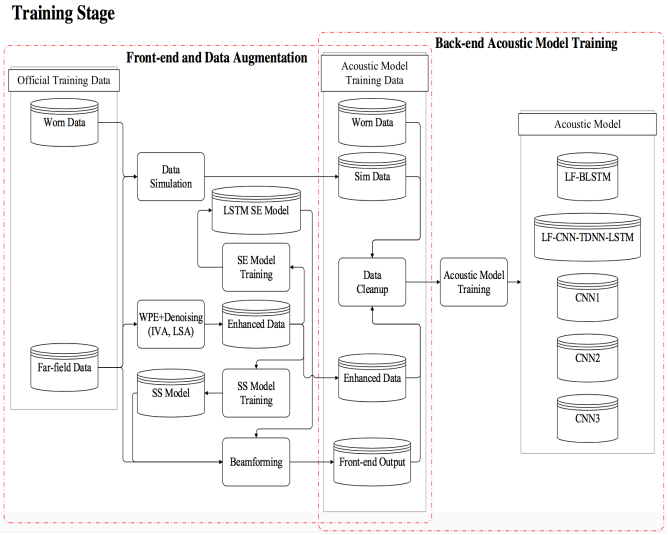

Figure 1: *An illustration of unified training stage, including front-end processing, data augmentation and acoustic modeling.*

2. Main contributions

First of all, due to rules defined by official[1], systems are allowed to exploit knowledge of the utterance start and end time, the utterance speaker label and the speaker location label. It's allowed to use binaural data and far-field data in the training set.

2.1. Training Stage

For acoustic model training, the procedures are the same for both single-array track and multi-array track. Here we introduce the details of training stage in some subsections separately.

2.1.1. Multi-channel preprocessing

For CHiME-5, we first utilize a multi-channle preprocessing step by traditional methods of signal process, which doesn't rely on training. It uses log-spectral amplitude [2] which is based on generalized weighted prediction error (GWPE) [3] and independent vector analysis (IVA) [4]. The goal of this step is to suppress some obvious noises and output the single-channel signals for the following stage. The preprocessing is simple but important for our entire system.

2.1.2. Speech enhancement model training

A deep-learning based speech enhancement method is adopted here, namely 'SE Model'. To simulate the training data, pure noise data is first extracted from official training set, in the guidence of human annotations. These segments are further filtered by using offical ASR model to make sure no textual signals within them. To be better consistent with the processed in testing phase, the multi-channel preprocessed data in Section 2.1.1 is taken as the target 'clean' data instead of binaural data. Then, target speech is corrupted with noise segments at different SNR levels. A densely connected progressive learning based speech enhancement model [5] is used to predict the ideal ratio masks (IRM) of speech. Since the speech quality is extremely low in far-field conditions, the output masks from speech enhancement models are only used as one kind of reference information to beamforming.

2.1.3. Speech separation model training

The recogniton of overlapping regions is one of the most critical problem in CHiME-5 challenge, however, overlapping speech occupies a large propotion. Firstly, non-overlapping regions of each speaker are detected and selected as the source data. They are mixed together to build speaker-dependent training data[6], where each target speaker is corrputed with other interference speakers among one same session. To overcome the low-resource problem of non-overlapping data, a two-stage speaker-dependent speech separation system is proposed. Similar to the discussion about different learning targets in [7], in the first stage the learning target is defined in an intermediate mapping form:

$$E_{\text{IM}} = \sum_{t,f} \left(\log \hat{z}^{\text{IRM}}(t,f) + x^{\text{LPS}}(t,f) - \bar{z}^{\text{LPS}}(t,f) \right)^2 \quad (1)$$

where $\hat{z}^{\text{IRM}}(t,f)$ is the estimated IRM with the logarithm operation and the input LPS features $x^{\text{LPS}}(t,f)$ to generate the masked LPS features. The trainded models, denoted as 'SS1' models, are applied to original data, including both non-overlapping and overlapping parts. After that, the new separated data is used as source data to generate new speaker-dependent training data. As for stage 2, we train the 'SS2' models with another learning target:

$$E_{\text{IRM}} = \sum_{t,f} \left(\hat{z}^{\text{IRM}}(t,f) - \bar{z}^{\text{IRM}}(t,f) \right)^2 \quad (2)$$

where $\hat{z}^{\text{IRM}}(t,f)$ and $\bar{z}^{\text{IRM}}(t,f)$ are the estimated and the reference IRMs, respectively. For model architecture in both stages, we utilize a two layer Bi-directional long short-term memory (BLSTM) as the speech separation model, each direction with 512 cells. 257-dimensional LPS feature are ultilized here as the acoustic feature to facilitate recovering waveforms, 7-frame expansion is used in the input. The computational network toolkit (CNTK) [8] is used for training. After separation stage, the resulting waveforms can be directly sent to back-end acoustic models, or provide only masks to the following beamforming.

2.1.4. Beamforming

Given the estimated masks from speech enhancement and separation models introduced above, we extend the complex Gaussian mixture model(CGMM) in[9] from 2 Gaussian mixtures to 3 Gaussian mixtures. Those mixtures indicate noises, target speaker and interference speakers, respectively. It's the first time to address those three factors simultaneously in realistic conditions. The masks are adopted as the initialization state for the EM algorithm. Final masks are sent to generalized eigenvalue decomposition (GEVD) beamformer[10]. So far, the entire frond-end stage finishes and outputs the separated waveform for recognition.

2.1.5. Data compilration

As shown in Figure 1, the final training data of acoustic model is largely augmented by our different processing methods. It mainly contains several parts as follows:

* Original binaural data
* The far-field data after multi-channel preprocessing described in Section 2.1.1
* The separated data from speech separation models described in Section2.1.3
* Data after beamforming described in Section2.1.4
* Simulated far-field data by using estimated impulse responses and binaural data

2.1.6. Acoustic model

In the back-end, we use five different kinds of acoustic models. The first two are based on lattice-free maximum mutual information (LF-MMI) training [11], including a conventional 5-layer BLSTM network and CNN-TDNN-LSTM (2-layer CNN + 9-layer TDNN + 3-layer LSTM) network. Both of them are trained on Kaldi Toolkit [12], with the input combining 40-dimensional MFCC feature and 100-dimensional i-vector.

Futhermore, three cross-entropy based acoustic models are trained by our self-developping tools, while the model input changes to the combination of 40-dimensional log mel-frequency filterbank (LMFB) feature and raw waveform. They are conventioanl CLDNN [13], 50-layer deep fully CNN [14] and 50-layer deep fully CNN with gate on feature map. They are listed as follows:

* LF-BLSTM: 5 layers BLSTM network with LF-MMI training
* LF-CNN-TDNN-LSTM: 2-layer CNN + 9-layer TDNN + 3-layer LSTM network with LF-MMI training
* CNN1: CLDNN (CNN-BLSTM-DNN) with CE training
* CNN2: 50-layer deep fully CNN with CE training
* CNN3: 50-layer deep fully CNN using gating mechanism with CE training

2.1.7. Language model

Besides the language models of official baseline, we build a conventional LSTM-based language model for *Rank B*.

2.2. Testing Stage

Depend on two different tracks in CHiME-5, the testing stages are different as well.

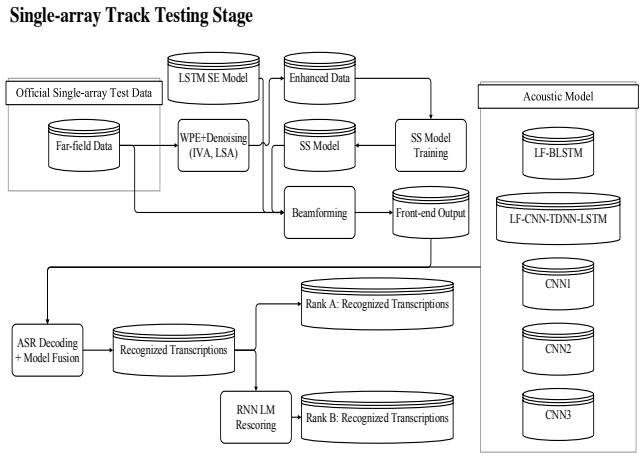

Figure 2: *An illustration of testing stage in single-array track.*

2.2.1. Single-array track

Figure 2 shows the flowchart of single-array testing stage. All those components remain the same with training stage in Section 2.1, including multi-channel processing, SE models and beamforming method, except for 'SS models'. Since SS models are speaker-dependent, they should be trained newly on testing sessions. In single-array track, the training of SS models only uses the reference array data. After all, the beamformed data is recognized by different acoustic models introduced in Section 2.1.6. Several acoustic models are fused at the state-level. However, for *Rank B*, the first-pass decoding is performed with the HMM and 3-gram to generate the lattice as the hypotheses, which are served for the second-pass decoding with a LSTM-based LM.

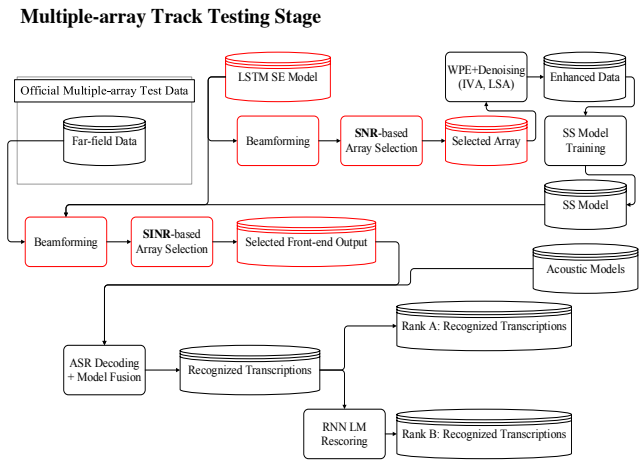

Figure 3: *An illustration of testing stage in multi-array track.*

2.2.2. Multi-array track

Different from single-array track, there is one additional step which is about array selection. Provided masks from deep-learning speech enhancement models, we select two best arrays in terms of signal to noise ratio (SNR) and signal to interference plus noise ratio (SINR). Two best arrays among all devices are selected. All precedures in single-array track are conducted on these two best arrays. At the end, different versions of output data are fused at the state-level after acoustic models. However, for *Rank B*, the first-pass decoding is performed with the HMM and 3-gram to generate the lattice as the hypotheses, which are served for the second-pass decoding with a LSTM-based LM.

3. Experimental evaluation

3.1. Front-end experiments

First of all, we present the front-end results on official baseline in single-array track. Time Delay Neural Neural Network (TDNN) recipe [11] using lattice-free maximum mutual information (LF-MMI) training, is used here. The trainining data keeps the same with official recipe in KALDI [12], which uses both binaural data and far-filed data with speech perturbation. The front-end uses a weighted delay-and-sum beamformer by BeamformIt toolkit [15] as a default multichannel speech enhancement approach. More details can be found in [1]. Compared with the official reported WER of 81.3% , our implemented version yields a comparable WER of 81.1%, as listed in Table 1. 'Single-channel' denotes the results of processed data which uses only single-channel speaker-dependent speech separation models described in Section 2.1.3. Futhermore, given the single-channel separation masks and enhancement masks, the beamformed data yields the results in the last row, namely 'Multi-channel'. It's observed that both stage are apparently effective in terms of WER reduction.

So far, testing data processed by those frond-end processing procedures is fixed in the rest of this paper, which denotes the output data after multi-channel beamforming.

Table 1: *WERs (%) for the development set using only the reference array with official acoustic baseline.*

WER(%) on Dev set	Kitchen	Dining	Living	Ave	Relative Gain(%)
Official	85.0	79.7	78.4	81.1	-
Single-channel	80.2	76.3	71.2	75.7	6.7
Multi-channel	72.3	67.7	63.3	67.7	16.5

3.2. Data augmentation

In Session 02, we evaluate different versions of training data in terms of WER. As listed in Table 2, the first row presents the subset of the best 'Multi-channel' results in Section 3.1. In the second row, we remove the speech perturbation and use only original binaural data (64h) and far-field data(110h). The WER increases from 65.3% to 66.7%. Then we add 120 hours of simulated data by the estimated impulse responses, namely 'Simu data'. It effectively reduces the WER to 64.9%. Futhermore, to better keep consistent with our speech separation models, we add separated speech data of far-field data in original training set, denoted as 'SS data'. The final performance yields the WER of 64.1%.

Hence, the training data is fixed, including binaural data,

far-field data, multi-channel processed data, simulated far-field data, and separated far-field data, which is about 530 hours.

Table 2: *WERs (%) for Session 02 which uses different versions of training data.*

WER(%) on S02	P05	P06	P06	P07	Ave
Baseline data	68.7	61.7	66.3	66.6	65.3
No perturbation	69.2	65.5	66.1	66.0	66.7
+ Simu data	68.1	63.6	64.0	63.9	64.9
+ SS data	68.4	62.4	62.2	63.4	64.1

3.3. Acoustic models

As shown in Figure 4, we have compared WERs of acoustic models and model ensembling on development set, for Rank A. The result of official baseline acoustic model is shown in blue bar. Motivated by the findings in Section 3.2, we first use the newly fixed training data with new model architectures instead of LF-TDNN, including LF-BLSTM and LF-CNN-TDNN-LSTM. They are all built with lattice-free maximum mutual information (LF-MMI) training method by KALDI tookit. As we can see, the LF-CNN-TDNN-LSTM yields better results than LF-BLSTM.

Performance of three CNNs is comparable due to their big architecture similaritities, so we directly present the ensembling results of three CNNs. Futhermore, we ensemble all five kinds of acoustic models via the state posterior averaging and lattice combination. Compared with official acoustic model, the final WER is reduced from 67.7% to 50.6%, indicating a relative reduction of 25.3%. This large improvement can be attributed to both data augmentation , acoustic modeling and ensembling. Compared with official baseline with a WER of 81.3% reported in [1], our system achieves a WER relative reduction of 37.7% in *Rank A*.

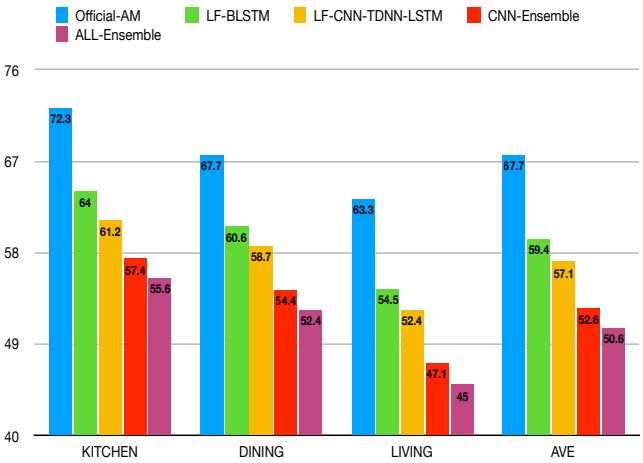

Figure 4: *WERs comparison between acoustic models and model ensembling on development set, for Rank A. Note the official-AM is trained with original training data, while others are trained using newly fixed training data in Section 3.2.*

3.4. Language model

Since the training material is extremely rare, our LSTM-based LM yields slightly better results in development set, as listed below.

Table 3: *WERs (%) of development set between official LM and our LSTM-based LM.*

WER(%) on Dev set	Kitchen	Dining	Living	Ave
Official	55.6	52.4	45.0	50.6
Ours	55.1	51.7	44.7	50.2

3.5. Results summary

To summarize, in the following tables we present the performance details of our best system tuned on the development test set, with its corresponding results on the evaluation test set. The only difference between *Rank A* and *Rank B* is the language model, which yields slightly better results when using LSTM-based model. It's surprising that utilizing multiple arrays doesn't bring any performance improvements on the evaluation test set, while it's significant when using single reference array. It's worth exploring the reason in the future. After all, the final results take the first place among all submitted system in all four tasks.

Table 4: *WERs (%) of the best system tuned on the development test set, with its corresponding performance on the evaluation test set, for Rank A.*

Track	Session		Kitchen	Dining	Living	Ave
Single-Array	Dev	S02	57.8	49.4	41.8	50.6
		S09	52.4	56.8	51.4	
	Eval	S01	56.6	38.7	56.7	46.4
		S21	50.4	41.4	42.8	
Multiple-Array	Dev	S02	46.3	46.0	41.1	45.6
		S09	46.1	48.6	50.5	
	Eval	S01	58.7	38.0	55.8	46.6
		S21	52.5	41.6	42.3	

Table 5: *WERs (%) of the best system tuned on the development test set, with its corresponding performance on the evaluation test set, for Rank B.*

Track	Session		Kitchen	Dining	Living	Ave
Single-Array	Dev	S02	57.4	48.5	41.5	50.2
		S09	51.8	56.4	51.1	
	Eval	S01	56.2	38.3	56.5	46.1
		S21	50.2	41.4	42.4	
Multiple-Array	Dev	S02	45.4	45.7	40.7	45.0
		S09	45.6	48.4	49.4	
	Eval	S01	58.1	37.1	55.1	46.1
		S21	52.5	41.1	42.2	

4. References

[1] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth CHiME Speech Separation and Recognition Challenge: Dataset, task and baselines," in *Proceedings of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH 2018)*, Hyderabad, India, Sep. 2018.

[2] I. Cohen, "Multichannel post-filtering in nonstationary noise environments," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1149–1160, 2004.

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

[4] N. Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*. IEEE, 2011, pp. 189–192.

[5] T. Gao, J. Du, L.-R. Dai, and C.-H. Lee, "Densely connected progresssive learning for LSTM-based speech enhancement," in *ICASSP*, 2018.

[6] ——, "A unified DNN approach to speaker-dependent simultaneous speech enhancement and speech separation in low SNR environments," *Speech Communication*, vol. 95, pp. 28–39, 2017.

[7] L. Sun, J. Du, L. Dai, and C. Lee, "Multiple-target deep learning for lstm-rnn based speech enhancement," in *Hands-free Speech Communications and Microphone Arrays (HSCMA), 2017*. IEEE, 2017, pp. 136–140.

[8] F. Seide and A. Agarwal, "Cntk: Microsoft's open-source deep-learning toolkit," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 2135–2135.

[9] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, "Robust mvdr beamforming using time-frequency masks for online/offline asr in noise," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 5210–5214.

[10] E. Warsitz and R. Haeb-Umbach, "Blind acoustic beamforming based on generalized eigenvalue decomposition," *IEEE Transactions on audio, speech, and language processing*, vol. 15, no. 5, pp. 1529–1539, 2007.

[11] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, and S. Wang, Y.and Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi." in *Interspeech*, 2016, pp. 2751–2755.

[12] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[13] T. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4580–4584.

[14] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *Acoustics, speech and signal processing (ICASSP), 2013 IEEE international conference on*. IEEE, 2013, pp. 8614–8618.

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