The OPPO System for CHiME-6 Challenge

Xiaoming Ren, Huifeng Zhu, Liuwei Wei, Linju Yang, Ming Yu, Chenxing Li, Dong Wei, Jie Hao

Beijing OPPO telecommunications corp., ltd., Beijing, China

renxiaoming@oppo.com, zhuhuifeng@oppo.com, weiliuwei@oppo.com, yanglinju@oppo.com, yuming@oppo.com, lichenxing1@oppo.com, weidong@oppo.com, haojie@oppo.com

Abstract

This paper describes our system and experimental results for the 6th CHiME Challenge. We participate in Track1(ASR only) on Category A and B.

Our system mainly include data preparation, frontend procesing, acoustic modeling, lattice rescoring with RNN Language Model(RNNLM) and system combination.

The frontend employs the baseline Guided Source Separation(GSS) [1]. For backend, we use TDNN-F and CNN-TDNNF [2] acoustic models, and finally apply Minimum Bayes Risk(MBR) [3] decoding for multiple lattices of different acoustic models.

Comparing with the offical baseline system, our system can get 20.44% and 18.07% relative Word Error Rate(WER) reduction on the dev and eval sets respectively.

1. Background

The system focus on Track1(ASR only) with conventional acoustic model, our submission system include data preparation, frontend, acoustic modeling, language modeling and system combination with MBR decoding. Figure 1 shows the framework of the submission system. With the proposed system, we finally achieve 41.18% and 42.02% WER on the dev and eval sets respectively. The rest of paper is organized as follows, section 2 describes the system in detail. The details of our expermental evaluation are given in section 3.

2. System Description

The overall framework of our system constain data preparation, frontend processing, acoustic modeling, language modeling and decoding, which is described in detail as follows:

2.1. Data Preparation

For the training data, comparing to official baseline, in addition we clean up and augment the data on the following aspects :

- For the worn(L+R) microphone training data, realign original utterance segmentation using ASR model
- We apply only speed perturbation for the traning data without the volume perturbation
- Clean up the training data by filtering out segments which are less than 1 second
- Remove some noises which can be recognized as words from noises used in Room Impuse Responses(RIR) [4] convolution

With the above data cleanup and data augmentation methods, we obtain about 1400 hours of data as the final training set, which contains the following dataset:

- The realigned worn(L+R)training data

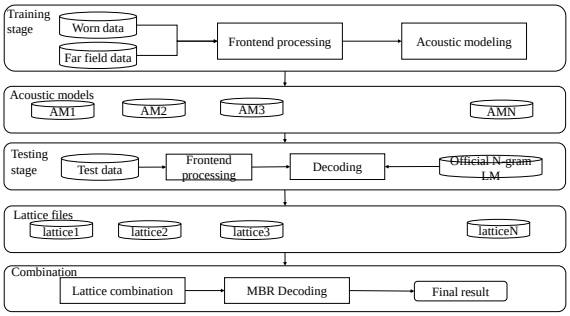

Figure 1: *Framework of system.*

- The far field data enhanced by GSS module
- The worn data and enhanced far field data both convolved with RIRs
- The augmented previous three datasets by speed perturbation

2.2. Frontend processing

For frontend processing, compared to the official baseline setup, we apply the GSS not only in testing stage but also in training stage.

2.3. Acoustic modeling

For acoustic model training, we use two different kinds of acoustic model structures based on lattice-free maximum mutual informaton (LF-MMI) training. They are TDNN-F network and CNN-TDNN-F network with the $\{40, 80\}$-dimenstion MFCC and 100-dimenstion online ivector. We train various acoustic models with different parameters and all the acoustic models are trained using Kaldi [5] toolkit. We use the following acoustic models:

- CNN-TDNN-F$\{1, 2, 3, 4\}$: GSS module with $\{10, 15\}$ context, 40-dim MFCC, 6-layer CNN + 19-layer TDNN-F with bottleneck-dim = 512, NUM-PDFS = $\{2500, 3500\}$, RIR augment
- CNN-TDNN-F$\{5, 6\}$: GSS module with $\{10, 15\}$ context, 40-dim MFCC, 6-layer CNN + 19-layer TDNN-F with bottleneck-dim = 768, NUM-PDFS = 3500, RIR augment
- CNN-TDNN-F7: GSS module with 15 context, 80-dim MFCC, 6-layer CNN + 19-layer TDNN-F with bottleneck-dim = 512, NUM-PDFS = 3500, RIR augment

- TDNN-F8: GSS module with 15 context, 40-dim MFCC, 25-layer TDNN-F with bottleneck-dim = 512, NUM-PDFS = 2500, RIR augment

2.4. Language modeling

For Category B, based on the offical transcription of the training data, we build a 2-layer LSTM-based language model and rescore the lattice using the score of LSTM-based LM and offical n-gram LM with a weighting of 0.55 and 0.45 respectively.

2.5. Decoding

In decoding phase, we use multiple acoustic models which are described in acoustic modeling section. Firstly, we get the lattices from each acoustic model. Then we combine all the lattices and apply MBR decoding to get the final result.

3. Expermental evaluation

3.1. Acoustic models

For acoustic models, we use official TDNNF model with 15 layers, deeper TDNNF model with 25 layers and CNN-TDNNF model which 6 convolution layers and followed by 19 TDNNF layers. Table 1 compare the three acoustic models using the official training data and frontend module.

Table 1: *WER(%) of different acoustic models on the dev and eval sets*

AM	dev	eval
tdnnf15	51.76	51.29
tdnnf25	50.77	50.30
cnn-tdnnf25	48.53	48.15

3.2. Frontend

In order to match the data in testing stage, we also apply GSS module for all multi-array data in the training stage, instead of randomly selecting 400k utterances from multi-array data in baseline. The result of WER is presented in Table 2. Compared to the official baseline, WER is reduced by 2% absolutely on dev set. We conjecture that the multi-array GSS training data are more compatible with dev and eval dataset. Furthemore, as a result of multi-array GSS in training stage, the amount of training data is reduced from 1500 hours to 240 hours. Consequently, it can speed up acoustic model training.

Table 2: *WER(%) of different frontends on the dev and eval sets*

Frontend	dev	eval
baseline	48.53	48.15
multi-array GSS in training stage	46.54	48.02

3.3. Data augmentation

By applying GSS module in training stage, it significantly reduce the amount of training data. In order to augment the training data, we repalce the L channel worn data with the L+R channel worn data and realigned (L+R) channels worn data respectively, WER can be reduced by 0.99% and 0.72% absolutely on dev and eval sets. After RIR data augmentation, the amount of training data increase by 4 times. It has a total of 1800 hours. After cleaning up, there are 1400 hours left which is equivalent to the baseline. RIR data augmentation greatly improves the performace of our system. We achive 0.49% and 1.53% WER absolutely reduction on the dev and eval sets.

Table 3: *WER(%) of different datasets on the dev and eval sets*

data	dev	eval
multi-array GSS + worn(L)	46.54	48.02
multi-array GSS + worn(L+R)	45.55	47.30
multi-array GSS + aligned worn(L+R)	45.80	47.34
multi-array GSS + aligned worn(L+R) + RIR	45.31	45.81

3.4. Feature

Comparing to model in the last line of Table 3 which used 40-dimenstion MFCC, we find that using 80-dimenstion MFCC can get 44.99% and 45.28% WER on the dev and eval sets.

3.5. System combination

Finally, we combine lattices produced by multiple acoustic models descibed in section 2.3, the WER of each acousti model are presented in Table 4 and then apply MBR decoding to get the final result. In Track1, for Category A, we get the lattice using offical N-gram LM, combine lattices and apply MBR decoding. At last we achieve the WER of 41.99% and 42.41% on the dev and eval sets. For Category B, The only difference is to rescore the lattices by using RNNLM, we get 41.18% and 42.02% of WER on dev and eval sets.

Table 4: *WER(%) of different acoustic models on the dev and eval sets*

AM	dev	eval
CNN-TDNN-F1(10,2500)	45.20	45.76
CNN-TDNN-F2(10,3500)	45.00	45.58
CNN-TDNN-F3(15,2500)	45.61	45.74
CNN-TDNN-F4(15,3500)	45.31	45.81
CNN-TDNN-F5(10)	45.46	45.80
CNN-TDNN-F6(15)	45.50	46.17
CNN-TDNN-F7	44.99	45.28
TDNN-F8	46.66	47.14

3.6. Results summary

To summarize, the final results of our system in detail on the development and evaluation sets are reported in Table 5.

Table 5: *WERs of the system in Track1(ASR only) for Category A and Category B*

Category	Session		Kitchen	Dining	Living	Ave
A	Dev	S02	47.42	45.57	38.31	41.99
		S09	39.28	43.22	38.80	
	Eval	S01	58.00	35.83	47.80	42.41
		S21	51.49	35.09	34.43	
B	Dev	S02	46.66	45.00	37.47	41.18
		S09	38.48	41.99	38.04	
	Eval	S01	57.56	35.47	47.76	42.02
		S21	50.95	34.95	33.75	

4. References

[1] C. Boeddeker, J. Heitkaemper, J. Schmalenstroeer, L. Drude, J. Heymann, R. Haeb-Umbach, "Front-end processing for the CHiME-5 dinner party scenario," *in Proc. CHiME-5*, pp35-40,2018.

[2] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohamadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks," *in Proceedings of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Hyderabad, India, Sep. 2018.

[3] 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 & Language*, vol. 25, pp. 802-828, 2011.

[4] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5220-5224, 2017

[5] 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 Proceedings IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2011.