The SJTU CHiME-4 system: Acoustic Noise Robustness for Real Single or Multiple Microphone Scenarios

Yanmin Qian *Tian Tan*

Key Lab. of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering, Shanghai Jiao Tong University, China
{yanminqian,tantian}@sjtu.edu.cn

Abstract

Noise robust speech recognition is one of the most challenging problems. This paper described the most important technical designs in the SJTU CHiME-4 Challenge system covering data usage, feature normalization, advanced acoustic model, auxiliary feature joint training, multi-model joint decoding and multi-pass decoding pipeline. The impacts on the final recognition accuracy from each technology are explored and compared. With the proposed technologies, our final system obtains a very large improvement compared to the formal released baseline system. The final average WERs of the real test set are 6.41%, 9.14%, 13.91% for 6-channel, 2-channel, and 1-channel, respectively.

1. Background

This paper describes the key points and contributions of the SJTU system (Shanghai Jiao Tong University) for the 4th CHiME Challenge [1]. We participate in all the evaluations for the challenge, including 6-ch / 2-ch / 1-ch tracks. Our works mainly focus on the acoustic modeling, so the front-end we used is the released baseline BeamformIt, the language model is the baseline RNNLM. In comparison to CHiME-3 challenge, our new progress mainly includes:

- Data augmentation using all channels with the beamformed data
- Feature normalization
- Advanced acoustic model including very deep CNN [2] and auxiliary feature joint training [3]
- System combination using the multi-model joint decoding and multi-pass decoding pipeline.

In the next section, we will describe these key technologies in detail.

2. Contributions

2.1. Data usage

Compared to the released baseline only using 18 hours noisy training data from channel 5, the training set is augmented with data from all channels (excluding the channel 2 located at the back of the device), and moreover the beamformed audio stream on these channels is also pooled together, which totally results 6×18=108 hours for training.

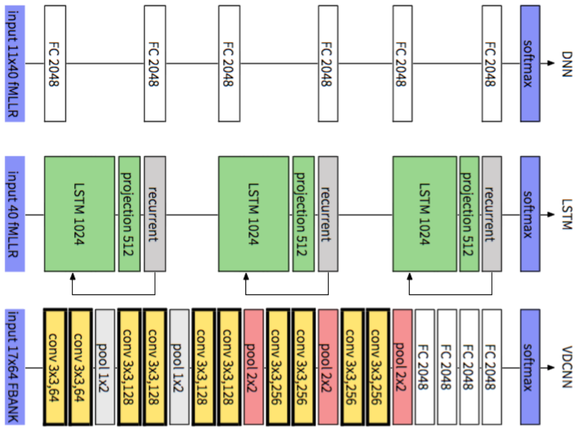


Figure 1: Model structures and configs used in our systems

2.2. Feature normalization

The appropriate feature normalization is very important for speech recognition in noisy scenarios. It can make the system more robust to the changes in environments and channels. CMN, CVN and CMVN are compared with FBANK, and the FBANK with CMN on per speaker shows the best performance.

2.3. Advanced acoustic models

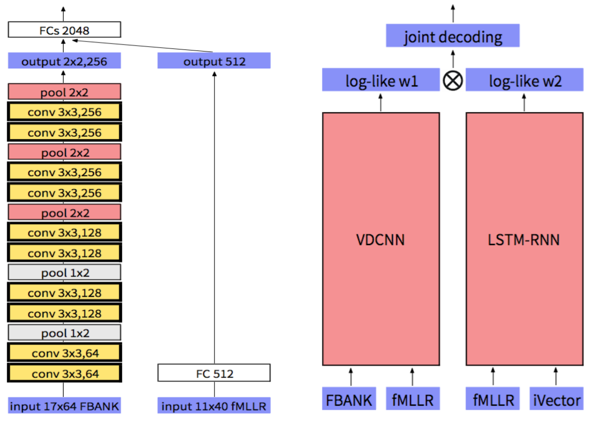
In addition to the basic DNN model, which is used in the released baseline, other advanced models are applied. One is named very deep CNN (VDCNN), which is proposed in our recent work [2, 4, 5], and particularly it shows the powerful potentiality on noise robustness [2]. Another is LSTM-RNN, and it has been verified effective on several tasks [6]. The model structures and configurations used in this work are illustrated in Figure 1, and more details can be referred to the work in [2].

2.4. Joint training with auxiliary features

The use of auxiliary features in factor-aware training is one type of adaptation popular for robust ASR [3, 7, 8, 9]. We use the same framework as our previous work for LSTM-RNN based speaker-aware training using i-vector [8], which concatenating the auxiliary feature with the original feature at the input layer.

In contrast, for the VDCNN usage, [5] proposed another auxiliary feature joint training architecture shown as the left part of Figure 2. Considering the auxiliary features, such as fMLLR and i-vector, are the non-topographical, they are separately

Thank Mr. Mengxiao Bi for the support on the computational cluster platform. This work was supported by Shanghai Sailing Program No. 16YF1405300, Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning, China NSFC projects (No. 61573241 and No. 61603252) and Interdisciplinary Program (14JCZ03) of Shanghai Jiao Tong University in China.



(a) Joint training of VDCNNs with auxiliary features

(b) Joint decoding of VDCNN & RNN

Figure 2: The architectures of VDCNN with auxiliary features joint training, and VDCNN & RNN joint decoding

transformed with a normal fully-connected layer first, and then the outputs are concatenated with those of the VDCNN block to be fed into the following shared MLP layers. Both fMLLR and i-vector can be used as auxiliary features for VDCNNs here.

2.5. Joint decoding with VDCNN and RNN

To explore the huge complementarity within VDCNN and LSTM-RNN, a joint decoding scheme shown as the right part of Figure 2 is implemented [5, 10]. It uses a weighted sum combination of acoustic log likelihoods from VDCNN and LSTM-RNN systems. Moreover, the DNN system also can be added into this framework to perform the multi-model (three) joint decoding.

2.6. Final multi-pass decoding system

Embedded with these above key features, our final submitted system is based on a multi-pass decoding framework, which is illustrated as Figure 3. It consists of 5 stages, shown as P1∼P5.

- **P1:** The front-end audio processing, including beamforming for multi-channel condition and feature extraction. In the 1-ch track, the single channel audio is used to extract all types of features directly.
- **P2:** Speaker-independent acoustic models are built individually, including DNN, VDCNN & LSTM-RNN. and auxiliary features based modeling are also constructed.
- **P3:** The DNN-SI system is adapted by the 2-pass mode, which uses 1-best from the first pass SI model. Then the 1-best from the adapted DNN-SA model is used to do the cross-adaptation for VDCNN and LSTM-RNN, named VDCNN-SA and LSTM-RNN-SA respectively.
- **P4:** Three speaker-adaptation models, including DNN-SA, VDCNN-SA and LSTM-RNN-SA are integrated to perform the proposed multi-model joint-decoding.
- **P5:** The RNNLM rescoring is applied on the lattices from the P4 stage to get the final results of the fusion system. If only considering the best single system, the lattices from VDCNN-SA in P3 are applied with RNNLM rescoring to generate the best single system results.

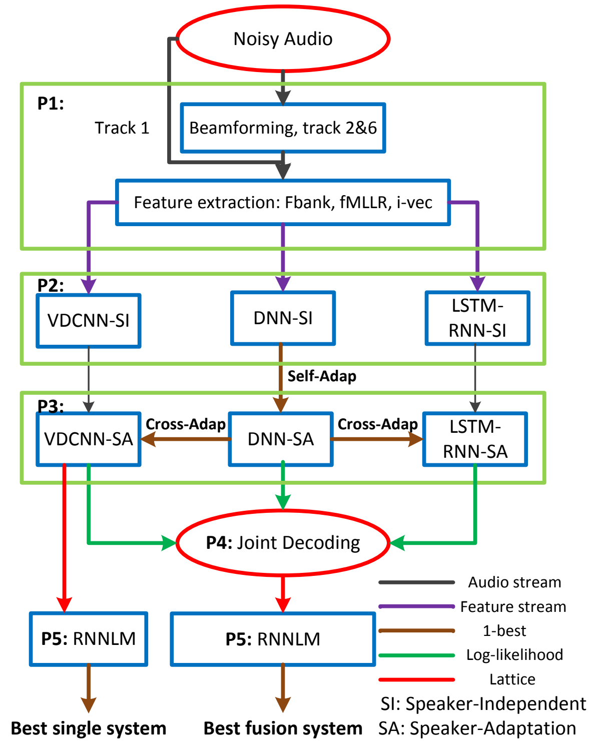


Figure 3: The multi-pass decoding for the CHiME4-Challenge

3. Experimental evaluation

The detailed results comparison in our system will be described in this section. The GMM-HMM system was trained using the released standard Kaldi [11] recipe. It is a MFCC-LDA-MLLT-FMLLR GMM-HMMs system. After that, a forced-alignment is performed to get the state level labels for NN training. In this work, all the DNN models are constructed using Kaldi [11], and other models are built using CNTK [12]. It is noted that except the results in Table 4 which used SMBR training and RNNLM rescoring, all the results in other tables used the CE criterion in training and the released trigram in decoding.

3.1. Data augmentation

Data augmentation was first evaluated, different amount of data, described in Section 2.1, were compared. In this experiment, DNN systems with fMLLR feature were used. As shown in Table 1, using more data always get better performance. For the fast investigation on the other system configuration, only the beamformed audio stream was used in training first (18 hours) in the following experiments, and the final submitted system will be retrained using all 108 hours data.

Table 1: WER (%) comparison of different training data usages for the 6ch-track, using fMLLR features in DNN models. The beamformed data on ch1-ch6 is used for testing in all setups

System	fMLLR	
	dev-real	dev-sim
Chan5	9.39	10.46
BF	9.30	10.51
Chan1-6	8.49	9.29
Chan1-6+BF	8.20	8.90

3.2. Acoustic models

Different acoustic models were then constructed, including DNN, LSTM, CNN and very deep CNN. As shown in Table 2, VDCNN get a 10% relative improvement on the real data over the DNN with the speaker dependent feature.

Table 2: WER (%) comparison of different acoustic models for the 6ch-track. Beamformed data on ch1-ch6 is used for both training and testing in all setups. **Feats** indicates the model input feature

System	Feats	dev-real	dev-sim
DNN	fMLLR	9.30	10.51
LSTM	fMLLR	10.26	11.69
CNN	FBANK	10.14	12.22
VDCNN		8.66	10.52

3.3. Auxiliary feature joint training

The auxiliary feature joint trainings in the VDCNN model and LSTM-RNN model are implemented. The different types of auxiliary features are explored and the related results are shown in Table 3. For the i-vector, a GMM with 2048 Gaussians is used to extract a 10-dimensional i-vector for each utterance, and these i-vectors were obtained using MFCC features. We can see that joint training with auxiliary features obtain consistent gains on both VDCNN and LSTM-RNN, and the improvement in VDCNN is especially large which demonstrats the superiority of the proposed new architecture.

Table 3: WER (%) comparison of the very deep CNNs and LSTM-RNNs with auxiliary features joint training for the 6ch-track. Beamformed data on ch1-ch6 is used for both training and testing in all setups. **Aux** indicates the auxiliary feature

System	Feats	Aux	dev-real	dev-sim
VDCNN	FBANK	—	8.66	10.52
		fMLLR	7.92	8.90
		fMLLR+ivec	7.69	8.83
LSTM	fMLLR	—	10.26	11.69
		ivec	10.23	11.52

3.4. Submitted system

At last, we give the final submitted results in Table 4. As stated above, the augmented 108 hours data was used for all model trainings, and the multi-pass decoding shown as the Figure 3 was performed to obtain the 1-best results. Considering we only want to focus on the acoustic modeling, so the released RNNLM was applied for the rescoring.

Due to the limited evaluation time, we can not finish the testing using the best fusion system on time. Accordingly the results from the best single system (applying RNNLM on VDCNN-SA in P3) are submitted as our final results for the challenge. All the results covering three tracks, including both dev and eval under different environments.

Table 4: WER (%) for the best submitted system.

Track	Envir.	Dev		Eval	
		real	sim	real	sim
1ch	BUS	8.32	6.87	22.25	9.64
	CAF	6.21	10.00	14.46	14.74
	PED	3.91	6.06	10.05	12.55
	STR	6.70	8.78	8.91	14.87
	AVG	6.28	7.93	**13.91**	12.95
2ch	BUS	5.92	4.71	14.06	6.33
	CAF	4.53	7.24	8.16	10.31
	PED	3.54	4.45	7.44	8.85
	STR	5.19	6.50	6.89	9.47
	AVG	4.79	5.73	**9.14**	8.74
6ch	BUS	4.31	3.88	8.45	4.59
	CAF	3.72	5.25	5.60	6.31
	PED	2.67	3.47	5.01	5.96
	STR	4.22	4.90	6.57	8.31
	AVG	3.73	4.37	**6.41**	6.29

4. References

[1] 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*, to appear.

[2] Y. Qian, M. Bi, T. Tan, and K. Yu, "Very deep convolutional neural networks for noise robust speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 12, pp. 2263–2276, 2016.

[3] 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.

[4] M. Bi, Y. Qian, and K. Yu, "Very deep convolutional neural networks for lvcsr," in *Proceedings of Interspeech*, 2015, pp. 3259–3263.

[5] Y. Qian and P. Woodland, "Very deep convolutional neural networks for robust speech recognition," in *Proceedings of SLT*, 2016.

[6] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." in *Proceedings of Interspeech*, 2014, pp. 338–342.

[7] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proceedings of ICASSP*, 2013, pp. 7398–7402.

[8] T. Tan, Y. Qian, D. Yu, S. Kundu, L. Lu, K. C. SIM, X. Xiao, and Y. Zhang, "Speaker-aware training of lstm-rnns for acoustic modelling," in *Proceedings of ICASSP*, 2016, pp. 5280–5284.

[9] Y. Qian, T. Tan, D. Yu, and Y. Zhang, "Integrated adaptation with multi-factor joint-learning for far-field speech recognition," in *Proceedings of ICASSP*, 2016, pp. 5770–5775.

[10] P. Woodland, X. Liu, Y. Qian, C. Zhang, M. Gales, P. Karanasou, P. Lanchantin, and L. Wang, "Cambridge university transcription systems for the multi-genre broadcast challenge," in *Proceedings of ASRU*, 2015, pp. 639–646.

[11] 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 *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, dec 2011, iEEE Catalog No.: CFP11SRW-USB.

[12] A. Agarwal, E. Akchurin, G. Basoglu, G. Chen, S. Cyphers, J. Droppo, A. Eversole, B. Guenter, M. Hillebrand, R. Hoens *et al.*, "An introduction to computational networks and the computational network toolkit," Tech. Rep. MSR-TR-2014-112, August 2014.[Online]. Available: http://research. microsoft. com/apps/pubs/default. aspx, Tech. Rep., 2014.