The USTC-iFlytek System for the First DIHARD Challenge

Lei Sun[1], Jun Du[1], Chao Jiang[2], Xueyang Zhang[2], Shan He[2], Bing Yin[2], and Chin-Hui Lee[3]

[1]University of Science and Technology of China, Hefei, Anhui, P. R. China
[2]iFlytek Research, Hefei, Anhui, P. R. China
[3]Georgia Institute of Technology, Atlanta, GA. USA

sunlei17@mail.ustc.edu.cn, jundu@ustc.edu.cn, chaojiang2@iflytek.com, xyzhang12@iflytek.com, shanhe2@iflytek.com, bingyin@iflytek.com, chl@ece.gatech.edu

Abstract

Our submitted systems for the first DIHARD challenge consist of several important modules of speech denoising, speech activity detection (SAD), i-vector design, and scoring strategy. One main contribution is the proposed long short-term memory (LSTM) based speech denoising model, which has been shown significant improvements over state-of-the-art diarization systems in highly mismatch conditions. The best diarization error rates (DERs) of our results on evaluation dataset are 24.56% and 36.05% , respectively in Track1 and Track2.

1. Database

A complete diarization system contains multiple sub-systems in charging of different aspects. In the section, we introduce all datasets for subsystems. First for speech enhancement, we have already explored its validity for realistic environments in [1]. Unlike only using English speech corpus WSJ0 [2], in this work we add 50-hour Chinese speech corpus from 863 Program to increase the diversity of clean speech data. Like WSJ0, the Chinese speech corpus is also reading-systle and recorded in quiet environment.

115 noise types are adopted here, including 100 noise types recorded in [3] and 15 home-made noise types. All clean speech files are corrupted with the above mentioned 115 noise types at three SNR levels (-5dB, 0dB and 5dB) to build a 400-hour training set, consisting of pairs of clean and noisy utterances.

Speaking of i-vector extractor, we choose the increasingly popular VoxCeleb corpus [4] to train the i-vector extractor based on universal background model (UBM). It is a large scale speaker identification dataset derived from YouTube, containing over 100,000 utterances for 1,251 celebrities. Moreover, we use another home-made corpus in iFlytek, which contains about 5,800 hours data from more than 38,000 persons. It is expected to provide enough data diversity which can enhance the performance of our residual CNN-based i-vector extractor.

For SAD training, 600-hour home-made realistic speech data in iFlytek was used. The speech quality is not very stable due to the complicated acoustic environments. Human annotations on each speech segment are set as the learning target.

The details of development set and evaluation set in DIHARD challenge can refer to [5, 6, 7].

2. System Description

The generic speaker diarization system often contains several main components: speech denoising, acoustic feature extraction, speech activity detection, speaker representation, speaker segmentation, speaker clustering and re-segmentation. In this section, we introduce each part in our system, as illustrated in Figure 1.

2.1. Speech denoising

Inspired by our previous work [8, 9], we adopt an advanced LSTM architecture with the novel design of hidden layers via densely connected progressive learning and output layer via multiple-target learning, as shown in Figure 2. The overall LSTM architecture aims to predict the clean log-power spectra (LPS) features and reference ideal ratio masks (IRMs) given the input noisy log-power spectra (LPS) features with acoustic context. All the target layers are designed to learn intermediate speech with higher SNRs or clean speech. For the input and multiple targets, LSTM layers are used to link between each other. This stacking style network can learn multiple targets progressively and efficiently. In order to make full use of the rich set of information from the multiple learning targets, we update the progressive learning in [9] with dense structures [10] in which the input and the estimations of intermediate target are spliced together to learn next target. Then, a weighted MMSE criterion in terms of multitask learning (MTL) is designed to optimize all network parameters randomly initialized with K target layers as follows:

$$\begin{aligned} E &= \sum_{k=1}^{K} \alpha_k E_k + E_{\text{IRM}} \\ E_k &= \frac{1}{N}\sum_{n=1}^{N} \|\mathcal{F}_k(\hat{\mathbf{x}}_n^0, \hat{\mathbf{x}}_n^1, ..., \hat{\mathbf{x}}_n^{k-1}, \mathbf{\Lambda}_k) - \mathbf{x}_n^k\|_2^2 \qquad (1) \\ E_{\text{IRM}} &= \frac{1}{N}\sum_{n=1}^{N} \|\mathcal{F}_{\text{IRM}}(\hat{\mathbf{x}}_n^0, \hat{\mathbf{x}}_n^1, ..., \hat{\mathbf{x}}_n^{K-1}, \mathbf{\Lambda}_{\text{IRM}}) - \mathbf{x}_n^{\text{IRM}}\|_2^2 \end{aligned}$$

where E_k is MSE corresponding to k^{th} target layer while E_{IRM} is MSE for MTL with IRM in the final output layer. $\hat{\mathbf{x}}_n^k$ and $\mathbf{x}_n^k$ are the n^{th} D-dimensional vectors of estimated and reference target LPS feature vectors for k^{th} target layer, respectively ($k > 0$), with N representing the mini-batch size. $\hat{\mathbf{x}}_n^0$ denotes the n^{th} vector of input noisy LPS features with acoustic context. $\mathcal{F}_k(\hat{\mathbf{x}}_n^0, \hat{\mathbf{x}}_n^1, ..., \hat{\mathbf{x}}_n^{k-1}, \mathbf{\Lambda}_k)$ is the neural network function for k^{th} target with the dense structure using the previously learned intermediate targets from $\hat{\mathbf{x}}_n^0$ to $\hat{\mathbf{x}}_n^{k-1}$, and $\mathbf{\Lambda}_k$ represents the parameter set of the weight matrices and bias vectors before k^{th} target layer, which are optimized in the manner of BPTT with gradient descent. $\mathbf{x}_n^{\text{IRM}}$, $\mathcal{F}_{\text{IRM}}(\hat{\mathbf{x}}_n^0, \hat{\mathbf{x}}_n^1, ..., \hat{\mathbf{x}}_n^{K-1}, \mathbf{\Lambda}_{\text{IRM}})$, and $\mathbf{\Lambda}_{\text{IRM}}$ are corresponding versions to IRM targets. α_k is the weighting factor for k^{th} target layer.

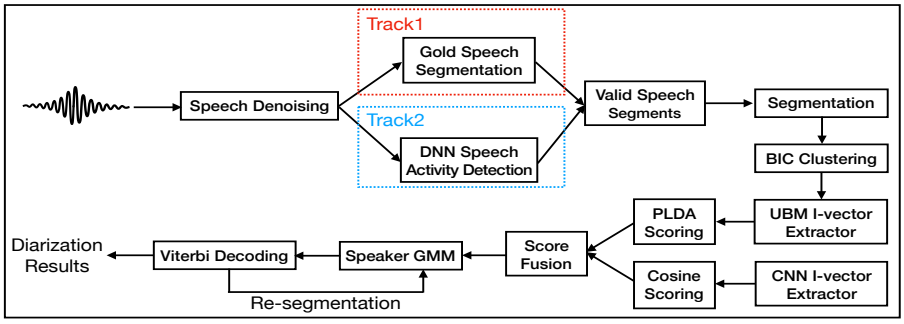

Figure 1: *Complete speaker diarization system diagram in both Track1 and Track2.*

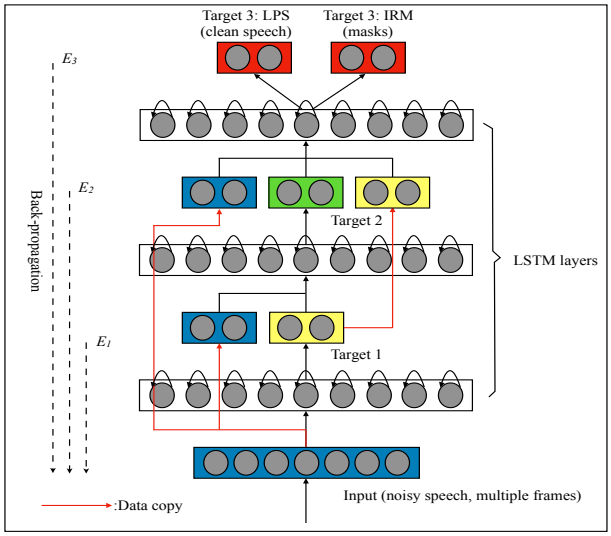

Figure 2: *A comparison of spectrograms for the proposed enhancement models with different training data setups.*

2.2. Speech activity detection

Here we train a framewise binary classification DNN of speech and non-speech. The features we use are 39-dimensional perceptual linear prediction (PLP) features (13-dimensional static PLP features with Δ and $\Delta\Delta$) and include an input context of 5 neighbouring frames (± 2), yielding a final dimensionality of 195 (39×5). Considering utility efficiency, the DNN model adopts a small and compact structure using 2 hidden layers with 256 and 128 hidden units in each layer and a final dual output layer, i.e. an architecture of 195-256-128-2. All training data is from realistic collected corpus.

2.3. Speaker segmentation and clustering

To fully utilize the effective information embedded in every stage, we propose a two-pass short-long term diarization system in this section.

2.3.1. Short-term diarization

Given the valid speech segments from SAD, it is important to split them into speaker homogeneous segments. It is also pivotal to prevent error accumulating in the very beginning. We use the Bayesian information criterion (BIC) [11] as the hypothesis testing metric. Then a global agglomerative hierarchical clustering (AHC) algorithm [12] is performed on all segments. At this step, every single segment is relatively short. The process is conducted iteratively, until a certain criterion is reached, upon which one separate cluster should arrive an upper limit or the number of clusters reaches a default maximum speaker number.

2.3.2. Long-term diarization

When the duration of each segment is relatively long, the i-vector can be a more powerful representative feature. We use an i-vector extraction system where the UBM includes 1024 Gaussians and the total variability (TV) matrix reduces the dimension to 400. The i-vectors are denoted as UBM i-vectors. Then all i-vectors are global mean subtracted, whitened, and length-normalized, then a PLDA scoring model is trained to measure the similarity between the i-vectors. In clustering, we repeatedly merge the closest two i-vectors based on a default PLDA score metric. Moreover, we retrain the UBM i-vector/PLDA model using the denoising data.

2.3.3. Residual CNN-based i-vector extractor

We train a residual CNN network for i-vector which is shown in Figure 3. For the input layer, 512 frames of 64 dimensional filterbank features which belong to the same person are grouped together as a feature map. At output layer, a 512 dimensional vector is generated as the identity vector of the specific person. During the first stage in training, we pre-train the network by predicting the speaker identity using softmax loss. Then triplet loss [13] is used as the second stage training criterion. Similarities between different CNN i-vectors are measured by cosine score.

2.3.4. Realignment

At the end, a realignment over frames is performed via Viterbi decoding on the GMM of each speaker. To make it more stable, we also use some smoothing strategy to prevent erroneously detected speaker turns [14].

3. Hardware Requirements

For models which need to be trained, several open-source tools are used for specific usage. First the computational network toolkit (CNTK) [15] was used for training our denoising model. Training one whole epoch needs 10 hours on a NVIDIA GeForce 1080Ti GPU card while we trained it for 30 epoches. CNN-based i-vector extractor are trained on Caffe [16], which needs 4 days on a NVIDIA GeForce M40 GPU card. As for the UBM/PLDA model, we use Hadoop [17] platform to shorten

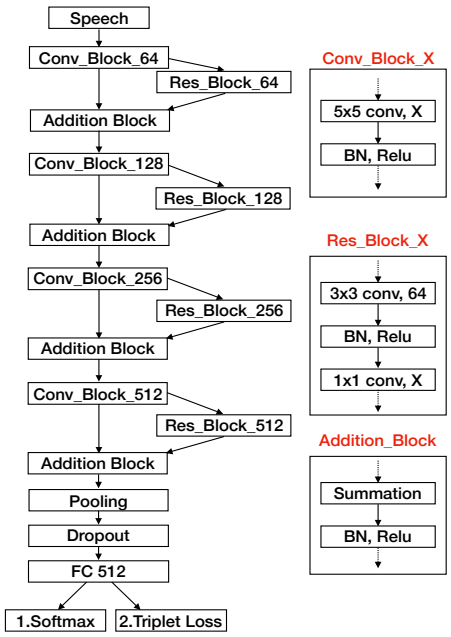

Figure 3: *The architecture of the residual CNN i-vector model.*

Table 1: *DER comparison of different speech inputs for UBM i-vector based diarization system on development set.*

DER(%)	Track1			
Speech	Miss	FA	SpkrErr	Overall
Original	8.50	0	11.76	20.26
Denoised	8.50	0	11.18	19.68
Retrained	8.50	0	11.01	19.51
DER(%)	Track2			
Speech	Miss	FA	SpkrErr	Overall
Original	18.60	6.10	8.50	33.20
Denoised	16.50	6.00	7.90	30.40
Retrained	16.50	6.00	7.60	30.10

the training time in only 4 hours. All other related tools are written and efficiency optimized in C++. For a 10 minutes recording, the diarization from scratch needs approximately 22 seconds.

4. Experiments Evaluation

4.1. Evaluation metric

We measure the performance of the diarization system by DER, which is defined by the evaluation campaigns organized by NIST. It compares the differences between the ground-truth reference segmentation and the generated diarization output. The final DER result is the sum of three types of errors: E_{Miss}, E_{FA} and E_{Spkr}, where each represents the percent of missed speech, false alarm error speech, and speaker misclassification error speech, respectively. Lower DER indicates better diarization performance. Note that, for DIHARD challenge, non-scoring collar is not permitted which means collar is set to zero in scoring script. Moreover, multiple speakers in overlap speech segments are counted.

Table 2: *DER comparison of different scoring strategies on development set.*

DER(%)	Track1			
Scoring	Miss	FA	SpkrErr	Overall
PLDA	8.50	0	11.01	19.51
PLDA+Cosine	8.50	0	8.90	17.40
DER(%)	Track2			
Scoring	Miss	FA	SpkrErr	Overall
PLDA	16.50	6.00	7.60	30.10
PLDA+Cosine	16.50	6.00	6.90	29.40

Table 3: *DER results of different scoring strategies on evaluation set.*

DER (%)	Track1	Track2
PLDA	24.96	36.39
PLDA+Cosine	24.56	36.05

4.2. Results on Development Set

First, we build a baseline speaker diarization system based on UBM i-vector extractor and PLDA model, which are both trained upon original VoxCeleb data. In Track1, we only use the gold speech segmentation, while Track2 uses the outputs of DNN-based SAD. As shown in Table 1, the DER on development set can benefit directly from denoised speech from 20.26% to 19.68% in Track1. Note that, our system does not tackle with overlap speech segments. That is to say, all overlap segments will be distributed to only one speaker, which generates inevitable missed error in both Tracks. Specifically, Miss is 8.5% in Track1 while FA is 0 with gold segmentation. In Track2, denoised speech can significantly reduce the percentage of Miss and FA, due to the removal of environmental interferences. Moreover, the valid speech segments can be less confusing, in terms of the reduction of SpkrErr. Furthermore, by retraining the i-vector extractor and PLDA model using denoised training data, additional improvements could be observed for both Track1 and Track2 as shown in the third row of each track.

System fusion [18, 19] is an effective strategy to improve the performance of speaker diarization system, including feature-level fusion [20], system output-level fusion [21], and multi-model fusion like audio-visual fusion [22]. To fully utilize the complementarity between UBM i-vector and CNN i-vector, in our fusion system we directly conduct a scoring fusion between PLDA score of UBM i-vector and cosine score of CNN i-vector. Comparing to single PLDA scoring, the fusion method obtains relative SpkrErr reductions of 19.2% in Track1 and 9.2% in Track2, respectively.

4.3. Results on Evaluation Set

Due to the limitation of uploading times each day, parameters are tuned on development set and then applied to evaluation data. Perfomance on both datasets has approcimately same tendency, but also has some differences. The results are shown in Table 3, the fusion method still can improve the performance but not as much as it on development set. This is partly because the parameters concerning the fusion process are very sensitive to data distributions, so proper parameters on evaluation set are not well found.

Besides the overall best results on evaluation set, here we briefly introduce the differences of all our submitted systems on

the challenge leaderboard [23], in order to provide more practical experimences. For Track 1, all these systems use the denoised speech:

- System1 intends to find method to do system fusion between PLDA score of UBM i-vector and CNN i-vector score, and finally achives the DER of 24.56%;
- System2 indicates a pure CNN i-vector based system, and achives the DER of 25.67%;
- System4 indicates an UBM i-vector based PLDA score system, and achives the DER of 24.96%;
- System3 is taken to adjust front-end stategy, but we upload wrong files by mistake and gets an exceptional result with the DER of 36.14%.

While in Track 2, the experimental route is not completely the same.

- System1 uses the original speech without front-end preprocessing, and achives the DER of 38.99%;
- System2 uses denoising model and UBM i-vector based PLDA score for clustering, and finally achives the DER of 36.39%;
- System3 intends to try some fusion methods between original speech and denoised speech in the front-end stage. The performance always can not exceed using only denoised speech. One of attempts yield the DER of 36.56%;
- System4 intends to tune score fusion strategy with only denoised speech, and finally achieves the DER of 36.05%.

Through we can not make sure all the final performance represents the best ability of every single system, current results are still meaningful. A method using front-end preprocessing and back-end score fusion yields the best results. Also, it's obviously observed that the performance gap between each system is too small to notice in real applications. The biggest the problem is still waiting to be solved, such as overlap detection, overlap attribution.

5. References

[1] L. Sun, J. Du *et al.*, "A Novel LSTM-Based Speech Preprocessor For Speaker Diarization In Realistic Mismatch Conditions," in *ICASSP*, 2018.

[2] D. Paul and J. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[3] G. Hu, "100 Nonspeech Sounds," in *web.cse.ohio-state.edu/pnl/corpus/HuNonspeech/HuCorpus.html*, 2004.

[4] A. Nagrani, J. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[5] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "First DIHARD Challenge Evaluation Plan," in *https://zenodo.org/record/1199638*, 2018.

[6] E. Bergelson, "Bergelson Seedlings HomeBank Corpus," doi:10.21415/T5PK6D.

[7] N. Ryant, "DIHARD Corpus," dLinguistic Data Consortium.

[8] L. Sun, J. Du, L.-R. Dai, and C.-H. 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.

[9] T. Gao, J. Du, L.-R. Dai, and C.-H. Lee, "SNR-Based Progressive Learning of Deep Neural Network for Speech Enhancement." in *INTERSPEECH*, 2016, pp. 3713–3717.

[10] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993*, 2016.

[11] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, 1978.

[12] K. Han and S. Narayanan, "A robust stopping criterion for agglomerative hierarchical clustering in a speaker diarization system," in *Interspeech 2010, September 26-30, Makuhari, Japan*, 2010, pp. Interspeech–2010.

[13] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[14] C. Fredouille, S. Bozonnet, and N. Evans, "The LIA-EURECOM RT'09 Speaker Diarization System," in *RT?9, NIST Rich Transcription Workshop*, vol. 15, 2009, pp. 17–23.

[15] D. Yu, A. Eversole, M. Seltzer, K. Yao, Z. Huang, B. Guenter, O. Kuchaiev, Y. Zhang, F. Seide, H. Wang *et al.*, "An introduction to computational networks and the computational network toolkit," Technical report, Tech. Rep. MSR, Microsoft Research, 2014, 2014. research. microsoft. com/apps/pubs, Tech. Rep., 2014.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[17] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on*. Ieee, 2010, pp. 1–10.

[18] S. Tranter, "Two-way cluster voting to improve speaker diarisation performance," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, vol. 1. IEEE, 2005, pp. I–753.

[19] D. Moraru, S. Meignier, C. Fredouille, L. Besacier, and J. Bonastre, "The ELISA consortium approaches in broadcast news speaker segmentation during the NIST 2003 rich transcription evaluation," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 1. IEEE, 2004, pp. I–373.

[20] A. Friedland, B. Vinyals, C. Huang, and D. Muller, "Fusing short term and long term features for improved speaker diarization," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 4077–4080.

[21] S. Bozonnet, N. Evans, X. Anguera, O. Vinyals, G. Friedland, and C. Fredouille, "System output combination for improved speaker diarization," in *Interspeech 2010, September 26-30, Makuhari, Japan*, 2010, pp. Interspeech–2010.

[22] I. Gebru, S. Ba, X. Li, and R. Horaud, "Audio-visual speaker diarization based on spatiotemporal bayesian fusion," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[23] Official, "Online Leaderboard of First DIHARD Challenge," in *https://coml.lscp.ens.fr/dihard/results.php*, 2018.