IberSPEECH 2018
21-23 November 2018, Barcelona, Spain

JHU Diarization System Description

Zili Huang, Paola García, Jesús Villalba, Daniel Povey, Najim Dehak

Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD, USA
{hzili1,pgarci27,jvillal7}@jhu.edu,dpovey@gmail.com,ndehak3@jhu.edu

Abstract

We present the JHU system for Iberspeech-RTVE Speaker Diarization Evaluation. This assessment combines Spanish language and broadcast audio in the same recordings, conditions in which our system has not been tested before. To tackle this problem, the pipeline of our general system, developed entirely in Kaldi, includes an acoustic feature extraction, a SAD, an embedding extractor, a PLDA and a clustering stage. This pipeline was used for both, the open and the closed conditions (described in the evaluation plan). All the proposed solutions use wide-band data (16KHz) and MFCCs as their input. For the closed condition, the system trains a DNN SAD using the Albayzin2016 data. Due to the small amount of data available, the i-vector embedding extraction was the only approach explored for this task. The PLDA training utilizes Albayzin data followed by an Agglomerative Hierarchical Clustering (AHC) to obtain the speaker segmentation. The open condition employs the DNN SAD obtained in the closed condition. Four types of embeddings were extracted, x-vector-basic, x-vector-factored, i-vector-basic and BNF-i-vector. The x-vector-basic is a TDNN trained on augmented Voxceleb1 and Voxceleb2. The x-vector-factored is a factored-TDNN (TDNN-F) trained on SRE12-micphn, MX6-micphn, VoxCeleb and SITW-dev-core. The i-vector-basic was trained on Voxceleb1 and Voxceleb2 data (no augmentation). The BNF-i-vector is a BNF-posterior i-vector trained with the same data as x-vector-factored. The PLDA training for the new scenario uses the Albayzin2016 data. The four systems were fused at the score level. Once again, the AHC computed the final speaker segmentation.

We tested our systems in the Albayzin2018 dev2 data and observed that the SAD is of importance to improve the results. Moreover, we noticed that x-vectors were better than i-vectors, as already observed in previous experiments.

Index Terms: speaker diarization, DNN, SAD, i-vectors, x-vectors, PLDA

1. Introduction

We present JHU's speaker diarization system for Iberspeech Diarization Evaluation. Our main goal is to test our current diarization system in other databases and possible scenarios. We provide a system solution for the open and closed condition scenario, using Kaldi. We follow a basic pipeline with specific characteristics for each scenario. The pipeline for our system is described as follows.

- Audio feature extraction
- Speech activity detection (SAD)
- Embedding extraction
- PLDA
- Score fusion (if possible)
- Clustering

Each of these items will be discussed in detail in the following sections.

2. Datasets

For the closed condition we only used the Albayzin2016 to train the SAD, the Universal Background Model (UBM), the i-vector extractor and PLDA models.

The datasets used for training the open condition are:

- Data set 1: Voxceleb1 and Voxceleb2 data (no augmentation). [1][2]
- Data set 2: SRE12-micphn, MX6-micphn, VoxCeleb and SITW-dev-core.
- Data set 3: Voxceleb1 and Voxceleb2 with augmentation.
- Data set 4: Fisher database (only database not in 16KHz)
- Albayzin2016: In-domain data set used in previous evaluations, which includes Aragon Radio database (20 hours) and 3/24 TV channel database (87 hours).
- Albayzin2018: RTVE broadcast data, which includes train, dev1, dev2 and test. The speaker diarization label is only offered for dev2, so the dev2 part is used for development purposes and tunning our system.

3. Feature Extraction and Speech Activity Detection

All the systems employ wide-band data (16KHz). [1] MFCCs were extracted for a 25ms window and 10ms frame rate. For the open condition, the MFCC feature dimension of x-vector-basic, x-vector-factored, i-vector-basic and BNF-i-vector is 30, 40, 24, 40 respectively. For the closed condition, the MFCC configuration is the same as i-vector-basic.

3.1. Speaker Activity Detection

After computing the MFCCs for each case, the system trained a TDNN SAD model on the Albayzin2016 labeled data following the Aspire recipe in Kaldi [3]. The network consists of 5 TDNN layers and 2 layers of statistics pooling[4]. The overall context of the neural network is around 1s, with around 0.8s of left context and 0.2s of right context. This approach is suitable for our purposes since it can include a wider context not affecting the number of parameters. For this special case, we trained the DNN with two classes: speech and non-speech. The speech segments include both the clean voice and the voice with noises. Other parts of the audio are considered as non-speech, which may include music, noise and silence. A simple Viterbi decoding on a HMM with duration constraints of 0.3s for speech and 0.1s for silence is used to get speech activity labels for the test

[1] except set 4 that is in 8 KHz to train bottleneck DNN

10.21437/IberSPEECH.2018-49

data recordings. The energy based SAD was also tried for our experiment, but the results were worse overall.

4. Embeddings

We computed two different sets of embeddings depending on the condition. For the closed condition we focused on the i-vectors. This approach computes i-vectors in the traditional way; it trains a T-matrix with Albayzin2016-only data. Afterwards, we obtained the i-vectors for the Albayzin2018 dev2 and test set. We tried other DNN possibilities, but due to the few amount of data available the results were not promising.

For the open condition we examined four types of embeddings. The i-vectors-basic, trained on data set 3, obtained baseline results for Albayzin2018 dev2 and test. These i-vectors are of dimension 400.

The BNF-i-vectors (of dimension 600) use the bottleneck feature computed from data set 2 to refine the GMM alignments. The rest of the i-vector pipeline remains the same; the T-matrix was also trained on data set 2.

We explored two types of DNN based embedding architectures. The first one, the deafult Kaldi recipe for Voxceleb, is a TDNN for x-vector-basic [5, 6]. In this approach, each MFCC frame is passed through a sequence of TDNN layers. Then, a pooling layer accounts for the utterance level process and computes the mean and standard deviation of the TDNN output over time in a pooling layer. This intermediate representation, known as embedding, is projected to a lower dimension (512 in this case). The DNN output are the posterior probabilities of the training speakers. The objective function is cross entropy. We employed data set 3 for training the TDNN. The augmentation is performed as described in [7] using MUSAN noises[2].

For the second x-vector approach the pre-pooling layers are changed to factorized TDNNs (TDNN-F) with skip connections [8]. This new architecture reduces the number of parameters in the network by factorizing the weight matrix of each TDNN layer into the product of two low-rank matrices. The first factor is forced to be semi-orthogonal that will prevent the lost of information when projecting from high to low dimension. As in other architectures, skip connections are an option for this TDNN-F. Some input layers receive as input the output of the previous layer and other prior layers. The best solution so far is to have skip connection between low-rank interior layers in the TDNN-F. The x-vectors are of dimension 600.

5. PLDA and Score Fusion

For the closed condition we observed that the number of speakers estimated by the current approach was very high for the Albayzin2018 dev2 set. We decided to use PCA as in [9]. With this tuning strategy the system was able to take into account every recording for PCA rotation, instead of only the global PCA. This strategy also maintained the number of speaker in a desirable range.

For the open condition, we used the traditional PLDA workflow, and the PLDA was trained on the Albayzin2016 data. We obtained 4 different types of scores that addressed the four type of embeddings. We fused the four systems with equal weights.

6. Clustering

The system performed an Agglomerative Hierarchical Clustering (AHC) to obtain a segmentation of the recordings following Callhome diarization recipe [3]. To obtain an accurate estimation of the number of speakers and have a better speaker segmentation, the system scans for several thresholds until it finds an optimum on a hold-out dataset. We evaluated this approach using the Albayzin2018 dev2 dataset.

7. Experiments

In this section, we describe some experiments that give us a clue of the overall performance of our system. We evaluated our systems with Diarization Error Rate (DER), which is the most common metric for speaker diarization. The diarization error can be decomposed into speaker error, false alarm speech, missed speech and overlap speaker. Our DER tolerated errors within 250ms of a speaker transition and only scored the non-overlapping part of the segments because our model outputs single label for each frame. Our systems were evaluated on the Albayzin2018 dev2.

We employed the Albayzin2018 dev2 set as the initial part of our experiments. This set was divided into two parts, and we tune the parameters on one part and compute the DER performance on the other, which was similar to the Kaldi Callhome diarization recipe [3].

The DER results of different systems for the open and closed condition are shown in Table 1 and Table 2 respectively. We compared three different calibration strategies: supervised calibration, oracle calibration and more than 10s. In the supervised calibration we chose the optimal thresholds on the held-out cross validation set. For oracle calibration, we used the oracle number of speakers for AHC. However, unlike traditional speaker diarization dataset like CALLHOME dataset, the utterances in the Albayzin2018 dev2 set were long and contained more speakers. The speech segments of some speakers were so limited that we didn't want to create a new cluster for these speakers. The third column shows the DER results when we clustered with the oracle number of speakers that have more than 10 seconds speech in the segment. It should be noted that since the number of speakers are unknown for the test set, our final submission only used supervised calibration and we reported the DER results of oracle calibration on the development set just for reference.

As shown in Table 1, x-vector based systems outperform the i-vector based ones, which is consistent with previous studies. Among the four systems, the TDNN-F based x-vector performs the best. It outperforms the basic x-vector and i-vector by 1.27% and 3.90% absolute. Equal weighted score fusion further reduces the DER to 9.39%. It is interesting that the DER performance of x-vector based systems degrades when clustering with the actual number of speakers while it improves for the i-vector based systems. This indicates that i-vector based systems require more prior knowledge of the number of speakers.

Table 2 shows the DER results for the closed condition. The i-vector system achieves a DER of 24.03%, which is further improved to 22.26% if clustering with the oracle number of speakers. However, the performance of the x-vector is not as good as i-vector. We believe the reason is that we cannot obtain enough data to train a discriminative neural network. Even after data augmentation with the music, noise and speech we extracted from Albayzin2016 dataset, the training set only contained 332 hours of speech which was much smaller than the usual amount of data to train the x-vector system. Besides, since the recordings were from TV programs, a large number of speakers didn't have enough corpus. The score fusion didn't improve the system performance for the closed condition.

[2] http://www.openslr.org/resources/17

Table 1: *DER (%) comparison of different systems for the open condition*

system	supervised calibration	oracle calibration	more than 10s
x-vector-basic	13.39	13.46	13.68
x-vector-factored	12.12	13.48	13.12
i-vector-basic	16.02	15.69	15.08
BNF-i-vector	16.40	15.88	14.83
fusion	**9.39**	10.83	11.87

Table 2: *DER (%) comparison of different systems for the closed condition*

system	supervised calibration	oracle calibration	more than 10s
i-vector	**24.03**	22.26	23.46
x-vector	34.69	35.58	34.85
fusion	25.34	22.39	22.76

From our experiment, we also observe that the SAD is of vital importance. Since we don't know the oracle SAD marks, the quality of the SAD is directly associated with the DER performance. Three different SAD models were evaluated, among which the 5-layer TDNN model trained on in-domain data performs the best. It outperforms the TDNN model trained on the Librispeech with same network architecture by 2.27% absolute. The energy based SAD is simple but the performance is worse than the TDNN models by a large margin. In our final system, we use the TDNN SAD trained on Albayzin2016 for both the open and closed condition.

Table 3: *DER (%) of basic x-vector system with different SAD*

SAD	DER
energy based SAD	21.52
TDNN SAD trained on Librispeech	15.66
TDNN SAD trained on Albayzin2016	**13.39**

8. Future Work

Although we largely reduce the diarization error with in-domain SAD and system fusion, there are still many problems to investigate. The first is the overlap problem. Our current system cannot handle the overlapping speech, since it predicts a single speaker label for each frame. However, solving this problem is not easy. The whole procedure of the diarization might change to predict multiple labels for one frame. Second, as discussed in the former part, the number of speakers estimated by the supervised calibration is not very close to the actual number. Besides, clustering with the oracle number of speakers sometimes even degrades the system. Whether there exists better methods to control the clustering process, especially for the condition with many speakers, requires further studies. Third, due to the time limit, we didn't include the re-segmentation process in our system. We will add this part later to see if it can further boost the system performance.

9. Conclusions

This is the submission for the JHU Diarization system. We tried our out-of-the-box system in this new scenario that contained broadcast news in a new language. Two main solutions were proposed for the closed and the open conditions: i-vector and x-vector. I-vector was showed to be best suited for the closed condition, due to the small amount of training data. For the open condition, the best results were obtained by the x-vector based system. However, having a score fusion, before the clustering gave noticeable improvements. We are still planning to do some re-segmentation in future versions.

10. Acknowledgements

The authors would like to thank David Snyder for his help in this project.

11. References

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

[2] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[3] 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 of the IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU2011*. Waikoloa, HI, USA: IEEE, dec 2011, pp. 1–4.

[4] P. Ghahremani, V. Manohar, D. Povey, and S. Khudanpur, "Acoustic modelling from the signal domain using cnns." in *INTERSPEECH*, 2016, pp. 3434–3438.

[5] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors : Robust DNN Embeddings for Speaker Recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018*. Alberta, Canada: IEEE, apr 2018, pp. 5329–5333.

[6] G. Sell, D. Snyder, A. Mccree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, and S. Khudanpur, "Diarization is Hard: Some Experiences and Lessons Learned for the JHU Team in the Inaugural DIHARD Challenge," in *Proceedings of the 19th Annual Conference of the International Speech Communication Association, INTERSPEECH 2018*, Hyderabad, India, sep 2018, pp. 2808—-2812. [Online]. Available: http://www.danielpovey.com/files/2018_interspeech_dihard.pdf http://dx.doi.org/10.21437/Interspeech.2018-1893

[7] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[8] 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*

2018, Hyderabad, India, sep 2018. [Online]. Available: http://danielpovey.com/files/2018_interspeech_tdnnf.pdf

[9] C. Vaquero, A. Ortega, J. Villalba, A. Miguel, and E. Lleida, "Confidence measures for speaker segmentation and their relation to speaker verification," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.