The CLIR-CLSP System for the IberSPEECH-RTVE 2020 Speaker Diarization and Identity Assignment Challenge

Carlos Rodrigo Castillo-Sanchez[1], *Leibny Paola Garcia-Perera*[2]

[1]Computational Learning and Imaging Research, Universidad Autónoma de Yucatán, Mexico
[2]Center for Language and Speech Processing, Johns Hopkins University, USA
{carloscastillomvc,leibny}@gmail.com

Abstract

This paper describes the Speaker Diarization system jointly developed by the Computational Learning and Imaging Research (CLIR) laboratory of the Universidad Autónoma de Yucatán and the Center for Language and Speech Processing (CLSP) of the Johns Hopkins University for the Albayzin Speaker Diarization and Identity Assignment Challenge organized in the IberSPEECH 2020 conference. The Speaker Diarization system follows an x-vector-PLDA-VBx pipeline built with the Kaldi toolkit. It uses a Time Delay Neural Network (TDNN)-based Speech Activity Detector (SAD), with x-vectors as acoustic features, clustered with Agglomerative Hierarchical Clustering (AHC) as initialization for variational Bayes clustering. The system was only evaluated in the Speaker Diarization condition.

Index Terms: speaker diarization, time delay neural network, x-vector, vbx

1. Introduction

IberSPEECH's Albaizyn evaluation challenges cover a wide range of speech processing technologies that include speech-to-text transcription, search on speech, and speaker diarization, with the latter being the subject of this paper. Speaker diarization is the process of grouping the same speaker's utterances in an audio recording under the same label with no prior knowledge of the number nor identity of the intervening speakers. It is an essential preprocessing step for many speech applications, such as Automatic Speech Recognition (ASR), spoken document retrieval, or audio indexing [1]. Therefore, the improvement of speaker diarization technologies is crucial to perform adequately in real-world conditions. The IberSPEECH-RTVE Speaker Diarization and Identity Assignment Challenge calls for robust speaker diarization systems for real TV broadcast shows from a range of topics on the Spanish public network [2, 3].

In the previous Albaizyn evaluation, five teams submitted systems for the open-set speaker diarization condition. The ODESSA team [4] explored three different segment representation embeddings: Binary key, Triplet-loss, and x-vectors; trained with the challenge's data [5], NIST SRE, and VoxCeleb1, respectively. Their primary submission consisted of fusing at similarity matrix level three systems, one for each embedding type and clustering with AHC. This was possible as they shared the same 1-second segmentation.

The JHU team [6] also leveraged score fusion at similarity matrix level. They addressed four types of embeddings extractors: x-vector-basic, x-vector-factored, i-vector-basic, and bottleneck features (BNF) i-vector. The first extractor was trained on VoxCeleb1 and 2 with augmentations; the second one with SRE12-micphn, MX6-micphn, VoxCeleb and, SITW-dev-core; the third one with VoxCeleb1 and 2 with no augmentations; and the last one with the same data as x-vector-factored. Their pipeline used a TDNN-based SAD and Probabilistic Linear Discriminant Analysis (PLDA) trained with Albayzin2016 data. The four embeddings' similarity scores were fused on equal weights and clustered with AHC.

Our system follows the conventional diarization pipeline [7, 8, 9], described as follows: (1) Segmentation: in this step, the non-speech portions of the recording are removed, and the remaining speech regions are further cut into short segments. The system leverages a pre-trained, publicly available SAD[1] based on a TDNN with stats pooling. (2) Embedding extraction: in this step, the system extracts speaker-discriminative embedding for each segment; the submitted system uses x-vectors. (3) Clustering: after an embedding is extracted from each segment, the segments are grouped into different clusters; our system was tested with three different PLDAs based on in- and out-domain data, with AHC as initialization for variational Bayes clustering.

The paper is organized as it follows: in Section 2 the used databases are described. Section 3 further describes the system characteristics, and Section 4.1 presents the computational resources used.

2. Datasets

Four datasets were used to develop our speaker diarization system:

- VoxCeleb1: a large-scale speaker identification dataset with 1,251 speakers and over 100,000 utterances, collected ”in the wild” [10].
- VoxCeleb2: a speaker recognition dataset that contains over a million utterances from over 6,000 speakers under noisy and unconstrained conditions [11].
- DIHARD II: focused on ”hard” speaker diarization, contains 5-10 minute English utterances selected from 11 conversational domains, each including approximately 2 hours of audio [12].
- The Corporación Radiotelevisión Española (RTVE) speaker diarization database [3]: consists of around 70 hours of audio documents annotated in terms of speaker turns. A 41% of the database is used for evaluation purposes with no a priori information provided about the number of speakers. The remaining 59% consists of a collection of 8 different TV shows by the Spanish TV station provided in 3 partitions that can be used for system training and development.

[1] http://kaldi-asr.org/models/m12

10.21437/IberSPEECH.2021-19

The x-vector extractor model for the speaker diarization condition was trained using VoxCeleb1+2 augmented with DIHARD II data. The three tested PLDA models were developed as follows: The first PLDA model uses a mixture of the DIHARD II dev and RTVE 2018 datasets, augmented with MUSAN [13] noises and reverberation; the second one was trained with VoxCeleb1+2 data, with the last PLDA being the weighted interpolation of the previous two.

3. System overview

This section describes the components that comprise the developed speaker diarization system.

3.1. Speech activity detection

In order to extract speech segments, our system uses a pre-trained TDNN-based SAD model[1]. The model was developed with the CHiME-6 training data [14]; such data was recorded in real-life conditions containing large amounts of background noise and overlapping speech. The SAD neural network architecture employs high-resolution Mel-Frequency Cepstral Coefficients (MFCC) as input, extracted for a 25ms window with a 10ms frame rate; with average log of energy, 40 mel-frequency bins, and a low cutoff frequency mel bins of 40. The network consists of 5 TDNN layers and two layers of statistics pooling [15]; trained with cross-entropy objective function to produce speech and non-speech labels. The speech labels include clean voice and voice with noises. Music, noise, and silence are categorized as non-speech. SAD labels are obtained by Viterbi decoding using an HMM with minimum duration constraints of 0.3 s for speech and 0.1 s for silence. We also tried energy-based SAD, but it was discarded as it performed worse overall.

3.2. Embeddings

We explored two types of embeddings. The first one, i-vectors; following the default Kaldi recipe for DIHARD, we trained a T-matrix with RTVE 2018-only data; afterward, we extracted i-vectors from the RTVE 2020 dataset and obtained baseline results. These i-vectors were of dimension 128.

The second type of embeddings we tested was x-vectors [16, 17]. We explored two methods to compute them; first, we followed the default Kaldi recipe for DIHARD, using VoxCeleb1+2 and RTVE 2018 with additional augmentation using MUSAN noises[2] as described in [13], to train the TDNN-based embedding extractor. This method passes each MFCC through a sequence of TDNN layers. A pooling layer computes the mean and standard deviation of the TDNN output over time, accounting for the utterance level process, with this internal representation (the x-vector) projected to a lower dimension. The DNN output is the training speakers' posterior probabilities, with the objective function being cross-entropy.

We used a TDNN-based extractor that uses 40-dimensional filterbanks with a 25ms window and 15ms frame shift as acoustic features for the second approach. These features are used for the embedding extraction as in [7]. The x-vector extractor model was trained using a TDNN with a 1.5s window with a frame shift of 0.25s; its architecture consists of four TDNN-ReLU layers, each of them followed by a dense-ReLU; afterward, two dense-ReLU layers are incorporated before a pooling layer; with a final dense-ReLU incorporated from which 512-dimensional embeddings are extracted. Then, a dense-softmax provides the output layer for this TDNN architecture.

Table 1: *x-vector extractor architecture [18].*

Layer	Layer context
frame 1	[t - 2, t - 1, t, t + 1, t + 2]
frame 2	[t]
frame 3	[t - 4, t - 2, t, t + 2, t + 4]
frame 4	[t]
frame 5	[t - 3, t, t + 3]
frame 6	[t]
frame 7	[t - 4, t, t + 4]
frame 8	[t]
frame 9	[t]
stats pooling (frame7, frame9)	[0, T]
segment1	[0, T]
softmax	[0, T]

3.3. PLDA scoring

As mentioned in Section 2 we tested our system with three different PLDA models; the first one was trained on a mixture of both DIHARD II dev and RTVE 2018 data augmented with MUSAN [13] noises and reverberation. The second PLDA used VoxCeleb1+2 out-of-domain data for training. For our third PLDA model, we followed [18]; this method aims to compute a robust PLDA based on the mixture of in-domain and out-of-domain PLDAs. This PLDA results from a weighted interpolation of the VoxCeleb1+2 out-of-domain data PLDA and the in-domain RTVE 2018+DIHARD II dev mixture PLDA. Both PLDAs were centered, whitened, and length normalized using the RTVE 2018+DIHARD II dev mixture data. Finally, the x-vectors were projected from 512 dim to 220 using Linear Discriminant Analysis (LDA).

3.4. Clustering

Using the similarity scores from one of the PLDAs, an Agglomerative Hierarchical Clustering (AHC) algorithm creates a set of clusters with an overestimation in the number of speakers. The VBx [19] uses the AHC initialization to make a further refinement of the clusters. VBx eliminates redundant speakers across the recording; it can be tuned by modifying the regulation coefficient (aggressiveness of eliminating redundant speakers), the acoustic scaling factor, and the loop-probability (staying in the same state when getting the next observation). The values used for our system are 0.4, 11, 0.80, respectively.

4. Experiments

This section describes some of the experiments that took place during our system's development process. We evaluated our systems using two metrics; the first one was Diarization Error Rate (DER), the most common metric for speaker diarization. DER comprises four types of errors: speaker error, false alarm speech, missed speech, and overlap speaker. Our DER followed the IberSPEECH-RTVE's evaluation plan characteristics, having a forgiveness collar of ±0.25 s before and after each reference boundary; and consecutive segments of the same speaker with a silence of less than 2 s come together as a single segment. The second metric that gave us an idea of the systems' performance was speaker number error; it allowed us to observe how each system estimated the number of speakers for each record-

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

Table 2: *DER (%), speaker error (SE) (%), missed speaker (MS) (%), false alarm (FA) (%) and speaker number error (%) comparison of different setups for the RTVE dev dataset (**post-submission** results are in bold letters). The speaker number error is the mean absolute error of the inferred number of speakers per recording.*

System	alpha, fa, fb, p	DER	SE	MS	FA	Speaker # error
i-vectors + DIHARD/RTVE PLDA + AHC	-	85.55	29.05	48.00	8.50	82.70
x-vectors + DIHARD/RTVE PLDA + AHC	-	80.19	63.29	5.00	11.90	75.98
x-vectors + DIHARD/RTVE PLDA + AHC (oracle # speakers)	-	86.51	69.61	5.00	11.90	0.00
oracle SAD + PLDA mixture + AHC+ VBx	0.55, 0.40, 11, 0.80	15.86	14.56	1.30	0.00	34.89
x-vectors + PLDA mixture + AHC + VBx	0.55, 0.40, 11, 0.80	34.61	17.71	5.00	11.90	37.74
x-vectors + DIHARD/RTVE PLDA + AHC + VBx	0.10, 0.40, 11, 0.80	43.55	26.60	5.00	11.90	51.78
x-vectors + VoxCeleb PLDA + AHC + VBx	0.10, 0.40, 11, 0.80	**22.77**	**5.80**	5.00	11.90	**14.17**
x-vectors + PLDA mixture + AHC	-	32.74	15.80	5.00	11.90	76.20
x-vectors + PLDA mixture + AHC + VBx	0.10, 0.40, 11, 0.80	30.33	13.43	5.00	11.90	28.44

Table 3: *DER (%), speaker error (%), missed speaker (%), false alarm (%) and speaker number error (%) comparison of different setups for the RTVE test dataset (**post-submission** results are in bold letters).*

System	alpha, fa, fb, p	DER	SE	MS	FA	Speaker # error
x-vectors + DIHARD/RTVE PLDA + AHC	-	68.06	57.56	4.40	6.10	80.90
x-vectors + PLDA mixture + AHC + VBx	0.55, 0.40, 11, 0.80	39.48	29.08	4.30	6.10	49.82
x-vectors + DIHARD/RTVE PLDA + AHC + VBx	0.10, 0.40, 11, 0.80	36.03	25.60	4.30	6.10	48.50
x-vectors + VoxCeleb PLDA + AHC + VBx	0.10, 0.40, 11, 0.80	**27.63**	**17.20**	4.30	6.10	**27.07**
x-vectors + PLDA mixture + AHC	-	34.76	24.30	4.30	6.10	58.81
x-vectors + PLDA mixture + AHC + VBx	0.10, 0.40, 11, 0.80	32.69	22.29	4.30	6.10	37.90

ing; this was useful alongside DER during VBx parameter optimization.

The DER and speaker number error results of different setups for RTVE 2020 dev and test datasets are shown in Table 2 and Table 3, respectively. The initial part of our experiments followed a similar strategy to the Kaldi Callhome diarization recipe [20], for i-vectors, we used the RTVE 2018 dataset to train the extractor and PLDA models, and for x-vectors, we added VoxCeleb1+2, as the extractor model requires more data during training; then we tuned the AHC threshold with RTVE 2018 in order to use it to compute performance on the RTVE 2020 dev dataset. As shown in Table 2, the first x-vector-based system outperformed the i-vector-based one, which is expected, so we discarded further experimentation with i-vectors. It should be noted that, in both cases, the estimated number of speakers performed poorly, so we tested the AHC with an oracle number of speakers. Such a test was performed only for reference reasons, as in the final submission, the number of speakers per recording in the test dataset is unknown. We expected that the AHC with an oracle number of speakers would deliver better results, but it was not the case. We believe that unlike traditional speaker diarization datasets, the utterances in the RTVE datasets contained numerous speakers. Since the recordings are from TV broadcast shows, there is an imbalance of speaker corpus (e.j. RTVE 2020 test mean and standard deviation of speaker time: 345s and 838s, respectively). We tested using an energy SAD early in development, but its lousy performance in such conditions directly affected the DER performance; we immediately moved to the pre-trained TDNN SAD model, and it provided better performance by a large margin. Additionally, we provide results with an oracle SAD; despite the oracle segmentation, the speaker number inference error is almost the same, which indicates that it is due to the clustering strategy, as it underestimates how many speakers there are per utterance. We cannot blame the VBx aggressiveness of redundant speakers removal, as the pre- and post-submission experiments without it suffer the same underestimation problem. We believe the AHC may not be the best method in conditions with many speakers; further studies are required.

Table 3 presents the results of our submitted systems; the first one was our contrastive setup; its AHC threshold was calibrated using the RTVE 2020 dev dataset; we can see that it had similar results to its counterpart in Table 2. The second one was our primary system; its VBx parameters were manually calibrated using the RTVE 2020 dev dataset. Post-submission experiments show that our primary system could have performed better with a considerable change in the VBx alpha, obtaining a 6.79% absolute improvement in DER; furthermore, with the PLDA mixture's replacement with the VoxCeleb1+2 one, it obtains an additional 5.06% improvement. It is clear that our DIHARD II dev + RTVE 2018 PLDA hindered our mixture results; the addition of DIHARD II dev and heavy augmentations (reverberation and noises) most probably caused this.

4.1. Development resources

We conducted our experiments on the CLSP Cluster[3] on several Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz nodes using up to 54 threads with 60 GB of RAM, and an NVIDIA GeForce GTX 1080 Ti with 11 GB of VRAM; Our submitted system took 35 hours for training and 15 hours to infer the RTVE test results.

5. Discussion

The x-vector-based system obtained our best results with an out-of-domain PLDA and VBx clustering. However, it falls behind

[3] http://www.statmt.org/jhu/?n=Info.CLSPCluster

our expected performance; we blame our heavy usage of out-of-domain-trained modules. Specifically, the x-vector extractor model, as previously mentioned, the RTVE dataset has specific peculiarities that differentiate it from standard speaker diarization datasets. The same can be said about the used TDNN SAD, as it was an out-of-the-box pre-trained model.

Our system would also be benefited from a better VBx parameter calibration, as shown in the *post-submission* results in Table 2 and Table 3. It should be noted that the usage of variational Bayes clustering greatly improved the system's ability to infer the number of speakers per recording, improving the DER.

6. Future work

Although we used a state-of-the-art approach for speaker diarization, we have room for improvements:

- We have to test domain-specific and hybrid approaches for the TDNN SAD model training, as its quality is directly associated with the diarization performance.
- In-domain data should be used for the x-vector extractor model training.
- Our system cannot handle overlapping speech, as it produces a single label per segment.
- Improve the system's speaker number inference in conditions such as the challenge's, where there are many speakers with imbalanced occurrences.

7. Conclusions

In this paper, we described our submission for the IberSPEECH-RTVE 2020 Speaker Diarization and Identity Assignment Challenge; we tested a state-of-the-art approach for diarization in a challenging Broadcast News scenario. We assessed the effectiveness of variational Bayes clustering as it significantly improved our system's ability to infer the number of speakers.

8. References

[1] D. Karim, C. Adnen, and H. Salah, "A system for speaker detection and tracking in audio broadcast news," in *2017 International Conference on Engineering MIS (ICEMIS)*, 2017, pp. 1–5.

[2] A. Ortega, A. Miguel, E. Lleida, V. Bazán, C. Pérez, M. Gómez, , and A. de Prada, "Albayzin Evaluation IberSPEECH-RTVE 2020 Speaker Diarization and Identity Assignment," 2020. [Online]. Available: http://catedrartve.unizar.es/reto2020/EvalPlan-SD-2020-v1.pdf

[3] E. Lleida, A. Ortega, A. Miguel, V. Bazán-Gil, C. Pérez, M. Gómez, and A. de Prada, "RTVE2020 Database Description," 2020. [Online]. Available: http://catedrartve.unizar.es/reto2020/RTVE2020DB.pdf

[4] J. Patino, H. Delgado, R. Yin, H. Bredin, C. Barras, and N. Evans, "ODESSA at Albayzin Speaker Diarization Challenge 2018," in *Proc. IberSPEECH 2018*, 2018, pp. 211–215. [Online]. Available: http://dx.doi.org/10.21437/IberSPEECH.2018-43

[5] E. Lleida, A. Ortega, A. Miguel, V. Bazán, C. Pérez, M. Gómez, and A. de Prada, "RTVE2018 Database Description," 2018. [Online]. Available: http://catedrartve.unizar.es/reto2018/RTVE2018DB.pdf

[6] Z. Huang, L. P. García-Perera, J. Villalba, D. Povey, and N. Dehak, "JHU Diarization System Description," in *Proc. IberSPEECH 2018*, 2018, pp. 236–239. [Online]. Available: http://dx.doi.org/10.21437/IberSPEECH.2018-49

[7] M. Diez, F. Landini, L. Burget, J. Rohdin, A. Silnova, K. Žmolíková, O. Novotný, K. Veselý, O. Glembek, O. Plchot, L. Mošner, and P. Matějka, "BUT System for DIHARD Speech Diarization Challenge 2018," in *Proc. Interspeech 2018*, 2018, pp. 2798–2802. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1749

[8] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully Supervised Speaker Diarization," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6301–6305.

[9] 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 *Proc. Interspeech 2018*, 2018, pp. 2808–2812. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1893

[10] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A Large-Scale Speaker Identification Dataset," *Interspeech 2017*, Aug 2017. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2017-950

[11] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *Interspeech 2018*, Sep 2018. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1929

[12] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "The Second DIHARD Diarization Challenge: Dataset, task, and baselines," 2019.

[13] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *CoRR*, vol. abs/1510.08484, 2015. [Online]. Available: http://arxiv.org/abs/1510.08484

[14] S. Watanabe, M. Mandel, J. Barker, E. Vincent, A. Arora, X. Chang, S. Khudanpur, V. Manohar, D. Povey, D. Raj, D. Snyder, A. S. Subramanian, J. Trmal, B. B. Yair, C. Boeddeker, Z. Ni, Y. Fujita, S. Horiguchi, N. Kanda, T. Yoshioka, and N. Ryant, "CHiME-6 Challenge:Tackling Multispeaker Speech Recognition for Unsegmented Recordings," 2020.

[15] P. Ghahremani, V. Manohar, D. Povey, and S. Khudanpur, "Acoustic Modelling from the Signal Domain Using CNNs," in *Interspeech 2016*, 2016, pp. 3434–3438. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-1495

[16] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 165–170.

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

[18] F. Landini, S. Wang, M. Diez, L. Burget, P. Matejka, K. Žmolíková, L. Mošner, O. Plchot, O. Novotny, H. Zeinali, and J. Rohdin, "BUT System Description for DIHARD Speech Diarization Challenge 2019," 10 2019.

[19] M. Diez, L. Burget, F. Landini, S. Wang, and H. Černocký, "Optimizing Bayesian Hmm Based X-Vector Clustering for the Second Dihard Speech Diarization Challenge," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6519–6523.

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Vesel, "The Kaldi speech recognition toolkit," *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 01 2011.