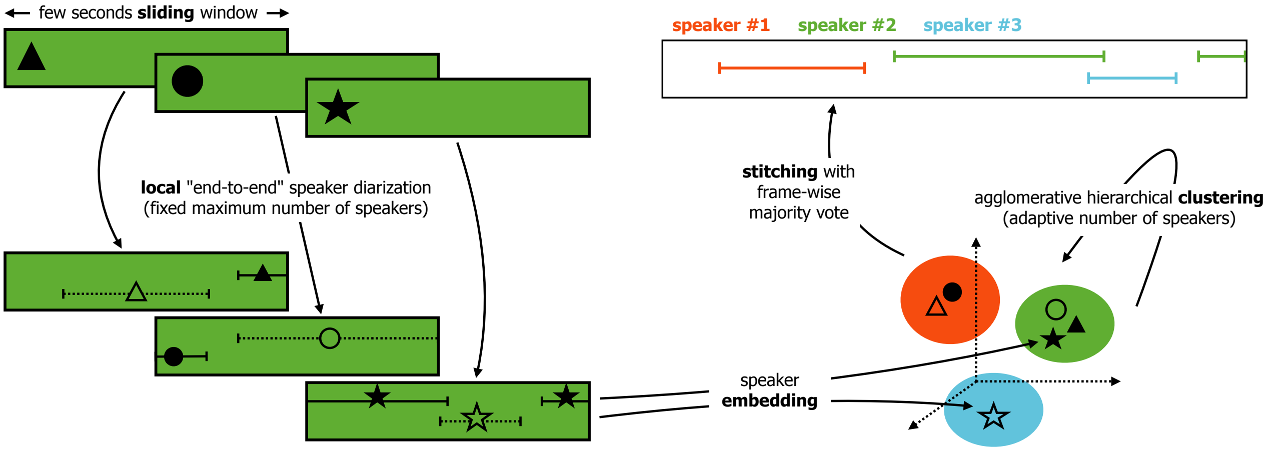
few seconds **sliding** window
local "end-to-end" speaker diarization
(fixed maximum number of speakers)
speaker #1
speaker #2
speaker #3
stitching with
frame-wise
majority vote
agglomerative hierarchical **clustering**
(adaptive number of speakers)
speaker
embedding

pyannote.audio speaker diarization pipeline at *VoxSRC 2023*

Séverin Baroudi, Hervé Bredin, Alexis Plaquet, Thomas Pellegrini

IRIT, Université de Toulouse, CNRS, Toulouse, France

herve.bredin@irit.fr

Abstract

This technical report describes the submission of team *pyannote* to the VoxSRC 2023 speaker diarization challenge. It relies on 3 stages: local end-to-end neural speaker segmentation on a few seconds sliding window, neural speaker embedding of each speaker of each window, and agglomerative hierarchical clustering.

1. Introduction

Every single submission made by the *pyannote* team uses the multi-stage paradigm depicted above and further described in details in [1]. It relies on 3 stages: local end-to-end neural speaker segmentation on a few seconds sliding window, neural speaker embedding of each speaker in each window, and agglomerative hierarchical clustering. In this technical report, we only focus on how our submissions differ from [1]. We therefore recommend reading [1] first to get the full picture.

Table 1 summarizes the changes we brought to our system over the course of the *VoxSRC 2023* challenge, sorted in chronological order of submission. All hyper-parameters were tuned to minimize the diarization error rate (DER) on *VoxConverse 0.3* as the unique criterion to decide which run to submit. We only submitted a run if it was better than the previous one on *VoxConverse 0.3*, hence avoiding inadvertent or unconscious tuning on the *VoxSRC 2023* leaderboard.

2. Local *end-to-end* speaker diarization

2.1. Longer windows

While [1] relies on 5-seconds windows with 500ms stride, all our submissions to the *VoxSRC 2023* challenge rely on 10-seconds windows with 1-second stride. The maximum number of speakers per window (3) remains unchanged. The reasoning behind this change is that speaker embeddings are later extracted from longer audio samples and hence are more robust.

2.2. Powerset multi-class cross entropy loss

While [1, 2] models speaker diarization as a multi-label classification problem (one class per speaker), all our submissions rely on powerset multi-class cross entropy loss (where dedicated classes are assigned to pairs of overlapping speakers) for training the model. Details about this change are available in [3], in which we found that this leads to much better overlapped speech detection.

2.3. Larger training set

While [1, 2] relies on models trained on *AMI* [4], *DIHARD* [5], and *VoxConverse 0.3.0* [6] development set, we use a large training set for all our submissions: *AISHELL* [7], *AliMeeting* [8], *AMI* [4], *AVA-AVD* [9], *DIHARD* [5], *Ego4D* [10], *MSDWild* [11], *REPERE* [12], and *VoxConverse 0.3.0* [6]

Starting at submission #2, an additional finetuning on *VoxConverse 0.3.0* development set is performed systematically (as it brings a 15% relative DER improvement on *VoxConverse 0.3.0* test set).

Starting at submission #7, we used all 232 files of *VoxConverse 0.3.0 test set* for tuning the clustering threshold (instead of a small subset of *VoxConverse 0.3.0 development set*). This led to a 4% relative DER improvement on *VoxSRC2023* test set.

2.4. WavLM feature extraction

While [1, 3] both rely on *SincNet* trainable features [13], we replace them by *WavLM* pretrained features [14].

Starting at submission #3, we used layer 10 of the off-the-shelf *WavLM-large* model pretrained on *Librispeech* [15]. This brings a 7% relative DER improvement on *VoxConverse 0.3.0* test set.

Starting at submission #5, we used layer 8 of *WavLM-base* pretrained from scratch on the training set described in previous section. This brings a 21% relative DER improvement on *VoxConverse 0.3.0* test set.

		VoxSRC 2023		VoxConverse 0.3		
Submission		DER	JER	DER	FA+MD	SC
#1	*pyannote 2.1* + powerset encoding, 10s windows, larger training set	8.3	45.3	7.6	4.3	3.3
#2	#1 + finetuned on VoxConverse 0.3	6.9	27.8	6.5	3.0	3.5
#3	#2 + switched from *SincNet* to *WavLM* (pretrained on *Librispeech*)	6.3	31.6	6.0	3.1	2.9
#4	#3 + switched speaker embedding from ECAPA-TDNN to ResNet34	5.9	25.8	5.6	3.1	2.5
#5	#4 + pretrained *WavLM* from scratch on speaker diarization datasets	5.1	28.6	4.4	2.7	1.7
#6	#5 + switched speaker embedding from ResNet34 to ResNet152	5.1	31.2	4.2	2.7	1.5
#7	#5 + optimized clustering threshold on whole VoxConverse 0.3	4.9	31.3	4.4	2.7	1.7
#8	#7 + switched speaker embedding from ResNet34 to ResNet152	4.8	30.6	4.2	2.7	1.5
#9	#8 + further pretrained *WavLM* on VoxConverse	4.8	28.6	4.0	2.7	1.4

Table 1: *Performance of our submissions on VoxSRC 2023 and VoxConverse 0.3 test sets. DER = diarization error rate. JER = Jaccard error rate. FA = false alarm rate. MD = missed detection rate. SC = speaker confusion error rate. All these metrics are reported with 250ms forgiveness collars, according to the rules of the challenge.*

For our final submission (#9), we further pretrained the *WavLM-base* (introduced in submission #5) on *VoxConverse 0.3.0* development set.

3. Speaker embedding

While [1] relies on *SpeechBrain*'s pretrained *ECAPA-TDNN* speaker embedding [16, 17], we switched to WeSpeaker [18] pretrained ResNet.

Starting at submission #4, we used *ResNet34*, bringing a 6% relative DER improvement on *VoxConverse 0.3.0* test set. Submissions #6 and #8 relied on *ResNet152*, bringing a 4% relative DER improvement on *VoxConverse 0.3.0* test set compared to submissions #5 and #7 respectively.

4. Conclusion

Our final submission reaches a DER of 4.77% on *VoxSRC 2023* test set (with 250ms forgiveness collar). We emphasize that it is **not** the fusion of several systems: it is a single system that follows the paradigm depicted at the top of the first page of this technical report. It takes 2 hours and 18 minutes to process the whole *VoxSRC 2023* test set (using one single V100 GPU): approximately 26 times faster than real time.

5. Acknowledgements

This work was granted access to the HPC resources of IDRIS under the allocations AD011012177R2 and AD011014274 made by GENCI.

6. References

[1] H. Bredin, "pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe," in *Proc. Interspeech 2023*, 2023.

[2] H. Bredin and A. Laurent, "End-to-end speaker segmentation for overlap-aware resegmentation," in *Proc. Interspeech 2021*, Brno, Czech Republic, August 2021.

[3] A. Plaquet and H. Bredin, "Powerset multi-class cross entropy loss for neural speaker diarization," in *Proc. Interspeech 2023*, 2023.

[4] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The ami meetings corpus," in *Proceedings of the Measuring Behavior 2005 symposium on" Annotating and measuring Meeting Behavior*, 2005.

[5] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "The Third DIHARD Diarization Challenge," *arXiv preprint arXiv:2012.01477*, 2020.

[6] J. S. Chung, J. Huh, A. Nagrani, T. Afouras, and A. Zisserman, "Spot the Conversation: Speaker Diarisation in the Wild," in *Proc. Interspeech 2020*, 2020, pp. 299–303. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-2337

[7] Y. Fu, L. Cheng, S. Lv, Y. Jv, Y. Kong, Z. Chen, Y. Hu, L. Xie, J. Wu, H. Bu, X. Xu, J. Du, and J. Chen, "AISHELL-4: An Open Source Dataset for Speech Enhancement, Separation, Recognition and Speaker Diarization in Conference Scenario," in *Proc. Interspeech 2021*, 2021, pp. 3665–3669.

[8] F. Yu, S. Zhang, Y. Fu, L. Xie, S. Zheng, Z. Du, W. Huang, P. Guo, Z. Yan, B. Ma, X. Xu, and H. Bu, "M2MeT: The ICASSP 2022 Multi-Channel Multi-Party Meeting Transcription Challenge," in *Proc. ICASSP 2022*, 2022.

[9] E. Z. Xu, Z. Song, S. Tsutsui, C. Feng, M. Ye, and M. Z. Shou, "Ava-avd: Audio-visual speaker diarization in the wild," ser. MM '22, 2022, p. 3838–3847.

[10] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, M. Martin, T. Nagarajan, I. Radosavovic, S. K. Ramakrishnan, F. Ryan, J. Sharma, M. Wray, M. Xu, E. Z. Xu, C. Zhao, S. Bansal, D. Batra, V. Cartillier, S. Crane, T. Do, M. Doulaty, A. Erapalli, C. Feichtenhofer, A. Fragomeni, Q. Fu, C. Fuegen, A. Gebreselasie, C. Gonzalez, J. Hillis, X. Huang, Y. Huang, W. Jia, W. Khoo, J. Kolar, S. Kottur, A. Kumar, F. Landini, C. Li, Y. Li, Z. Li, K. Mangalam, R. Modhugu, J. Munro, T. Murrell, T. Nishiyasu, W. Price, P. R. Puentes, M. Ramazanova, L. Sari, K. Somasundaram, A. Southerland, Y. Sugano, R. Tao, M. Vo, Y. Wang, X. Wu, T. Yagi, Y. Zhu, P. Arbelaez, D. Crandall, D. Damen, G. M. Farinella, B. Ghanem, V. K. Ithapu, C. V. Jawahar, H. Joo, K. Kitani, H. Li, R. Newcombe, A. Oliva, H. S. Park, J. M. Rehg, Y. Sato, J. Shi, M. Z. Shou, A. Torralba, L. Torresani, M. Yan, and J. Malik, "Ego4D: Around the World in 3,000 Hours of Egocentric Video," in *Proc. CVPR 2022*, 2022.

[11] T. Liu, S. Fan, X. Xiang, H. Song, S. Lin, J. Sun, T. Han, S. Chen, B. Yao, S. Liu, Y. Wu, Y. Qian, and K. Yu, "MSDWild: Multimodal Speaker Diarization Dataset in the Wild," in *Proc. Interspeech 2022*, 2022, pp. 1476–1480.

[12] J. Kahn, O. Galibert, L. Quintard, M. Carré, A. Giraudel, and P. Joly, "A presentation of the repere challenge," in *2012 10th International Workshop on Content-Based Multimedia Indexing (CBMI)*, 2012, pp. 1–6.

[13] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *Proc. SLT 2018*, 2018.

[14] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[15] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[16] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, “SpeechBrain: A general-purpose speech toolkit,” 2021, arXiv:2106.04624.

[17] B. Desplanques, J. Thienpondt, and K. Demuynck, “ECAPA-TDNN: emphasized channel attention, propagation and aggregation in TDNN based speaker verification,” in *Interspeech 2020*, H. Meng, B. Xu, and T. F. Zheng, Eds. ISCA, 2020, pp. 3830–3834.

[18] H. Wang, C. Liang, S. Wang, Z. Chen, B. Zhang, X. Xiang, Y. Deng, and Y. Qian, “Wespeaker: A research and production oriented speaker embedding learning toolkit,” *arXiv preprint arXiv:2210.17016*, 2022.