Interspeech 2024
1-5 September 2024, Kos, Greece

Hybrid-Diarization System with Overlap Post-Processing for the DISPLACE 2024 Challenge

Gabriel Pîrlogeanu[1,2], *Octavian Pascu*[1,2], *Alexandru-Lucian Georgescu*[1,2], *Horia Cucu*[1,2]

[1] Zevo Tech, Romania
[2] POLITEHNICA Bucharest, Romania

gabriel.pirlogeanu@stud.etti.upb.ro

Abstract

This paper describes our team's collaborative efforts in participating in the Track 1 for Speaker Diarization of the Diarization of Speaker and Language in Conversational Environments (DISPLACE) Challenge 2024. Our submission focuses on creating a diarization system that is robust to noisy conditions, as well as high amounts of overlapped speech. We conduct an exhaustive study on each component of a hybrid system using techniques such as semi-supervised learning, ensemble of several systems and experiment with both a neural overlap detection module, as well as a post-processing technique using an external overlap detection system. Our final system achieves a diarization error rate (DER) of 28.04% on Phase 1 Eval set, representing a relative improvement of 19.33% compared to the baseline DER of 34.76%.

Index Terms: speaker diarization, DISPLACE challenge, voice activity detection, overlap detection, DOVER-Lap

1. Introduction

Speaker Diarization represents the process of identifying individual speakers in an audio stream, trying to answer the question ”who spoke when”. Even though great progress was made in recent years, with diarization systems achieving good performances in difficult scenarios with multiple speakers, noisy speech and overlapped speech still pose a great challenge for these systems. Several challenges throughout the years have tackled different aspects of the diarization problem: DiHARD [1], CHIME [2], AVA-AVD [3], Ego4D [4]. One of the challenges from recent years is the DISPLACE 2023 challenge [5], that focuses on speaker diarization in multilingual and multispeaker conversational environments, presenting noisy speech conditions, such as outside corridor noise or background voices, and a high 14% overlapped speech ratio.

The 2023 edition of this challenge highlighted a multitude of approaches across the primary components of a diarization system: (i) speech/voice activity detector, (ii) speaker embeddings extractor and (iii) clustering module. In the domain of Voice Activity Detection (VAD), approaches like the Silero VAD [6], Pyannote's VAD [7], and the Multilingual VAD from NeMo [8] employing MarbleNet [9] were preferred. Speaker embeddings extraction was predominantly explored using x-vectors [10], along with alternatives like the ECAPA-TDNN [11] model, TitaNet-L [12] model, Pyannote's [7] segmentation module, RawNet [13] and ResNet backbones. Clustering methods varied, including Agglomerative Hierarchical Clustering (AHC), Spectral Clustering (SC), as well as the VBx algorithm [14], illustrating the breadth of strategies employed in tackling speaker diarization challenges. The insights drawn from the preceding round have revealed that a notable contributor to the rise in DER was the occurrence of overlap errors.

Our current approach entails an exhaustive study of all components. We integrate both supervised and semi-supervised learning for Speech Activity Detection (SAD) and an ensemble of several systems. Various backbone architectures were explored for multi-scale embedding extraction, as well as investigating a neural method for overlapped speech detection together with a post-processing greedy technique based on overlap detection from an external system. For clustering, we used the Normalized Maximum Eigengap Spectral Clustering technique, given last year's results and the baseline system, as well as the advantage of not requiring parameter tuning on the development set. The systems described in this paper were constructed using various components from the NeMo Conversational Toolkit [8] and the Pyannote.audio Diarization Toolkit [7], as well as the BUT VBx implementation [14].

The paper is organized as follows. Section 2 presents the data resources and the methodology we used. Section 3 summarizes our experimental results, while Section 4 is reserved for conclusions.

2. Methodology

2.1. Data resources

In comparison with the DISPLACE 2023 dataset, we see an increase of over 4 hours of development annotated data through the addition of 8 new annotated audios, totalling 19.75 hours, which we will call `dev`. Furthermore, for this edition, the organizers also provide a substantial amount of unsupervised development data from a similar distribution as the rest of the dataset, more exactly 121.6 hours. Throughout the paper, this subset will be called `dev_unsup`. Besides the DISPLACE development data, we also used the AMI Corpus Microphone Array [15] train and validation split from the setup used in [14]. For evaluation, the organizers provided 17.93 hours of recordings, called `eval`. Lastly, noise and background recordings from the MUSAN [16] and RIR [17] datasets are also used for data augmentation and speech-background dataset balancing.

2.2. Voice Activity Detection

2.2.1. Preliminary Systems and Automatic Annotations

Taking in consideration the analysis of results in the DISPLACE 2023 challenge [5], all teams struggled to alleviate the problem of missed speech, due to noisy conditions and high overlap percentages. Therefore, we explore several Voice Activity Detection (VAD) solutions and evaluate their performance after fine-tuning the post-processing parameters on the development set. The goal of this evaluation is to use an ensemble of these systems in order to automatically annotate the unsuper-

10.21437/Interspeech.2024-1287

Table 1: *Preliminary VAD results reported on the development set. Systems from 1 to 5 were not fine-tuned, they only had their post-processing parameters tuned on the development set. System 6 was used in order to automatically annotate the 121 hours of unsupervised data provided by the challenge's organizers. "Seg" represents the [21] segmentation model and "Seg3.0" represents the [22] segmentation model, both of them used for the VAD task.*

Sys.	**VAD System**	**FA**	**MISS**	**DetER**
1	MarbleNet [19]	9.27	1.38	10.66
2	Silero v4 th=0.25	9.25	**1.29**	10.55
3	ASR combined	11.51	11.33	22.00
4	Seg	6.73	2.18	8.91
5	Seg3.0	5.46	2.80	**8.26**
6	Ensemble Sys. 2 + 4 + 5	**4.72**	3.95	8.67

vised data for later fine-tuning, as explored in several studies of semi-supervised learning [18]. In order to have as few annotation errors as possible in the resulting dataset, the goal of the ensemble system is to reduce the False Alarm Error (FA) through majority voting over the speech segment boundaries.

Table 1 presents the five preliminary systems investigated and the final ensemble system used for automatic annotations, alongside their Detection Error Rates (DetER) on the `dev` subset. (i) The first system reproduces the `Multilingual MarbleNet` network fine-tuned in [19] on the DISPLACE 2023 development subset. (ii) For the second system, we analyzed the VAD implemented by the 1^{st} place team of DISPLACE 2023, respectively the pretrained `Silero V4`, whose threshold is tuned on the development subset to 0.25. (iii) For the third system, we explore the use of Automatic Speech Recognition (ASR) for Voice Activity Detection, using the estimated timestamps of the transcribed words as speech regions. Given the multi-lingual aspect of the dataset, we combine the `Whisper v3 Small Hindi` and `Whisper v3 Small English` [20] models' outputs on `dev` and filter the resulted transcription with a confidence threshold for the predicted words of 0.3. (iv-v) For the last two systems, we explore the pretrained `pyannote-audio/segmentation` [21] and `pyannote-audio/segmentation-3.0` [22] models for the Voice Activity Detection task, as the later one is also used as the baseline system's VAD. Out of these preliminary systems, only the Marblenet model and Pyannote segmentation models are trainable.

Using the above mentioned modules, we explored several ensembles on the `dev` dataset. The sixth system obtained the smallest FA error through majority voting of the speech intervals using the pretrained `Silero V4`, `pyannote-audio/segmentation` and `pyannote-audio/segmentation-3.0` systems. Using this system for automatic annotations, we obtain 116.96 hours of speech from the `dev_unsup` subset.

2.2.2. Multilingual MarbleNet Fine-tuning

In order to leverage the 141 hours of development data, we use the manually and automatically annotated audios, as well as background sounds extracted from the MUSAN dataset, to fine-tune the segment-based Multilingual MarbleNet model [9]. We expect an improvement in results compared to the system fine-tuned in [19], due to the new automatically annotated data provided by the challenge's organizers. In addition, we add RIR online noise with a probability of 0.3 and noise sourced from MUSAN freesound subset. We follow the setting of Jia *et al.* [9], using a window of 0.63s, but we use a 0.3, 0.02 and 0.1 stride for speech, background and noise respectively. We obtain through this setting 979K, 114K and 57K segments for speech, background and noise respectively. In order to balance the speech and non-speech, we use over-sampling on the background segments. We split these final speech-background segments in a 85-10-5 split for train-validation-test.

We fine-tune the weights of the model for 150 epochs, with a configuration similar to the one used in [19], but we also use checkpoint averaging on the best 10 checkpoints saved based on the validation loss. Lastly, we also tune the model's post-processing parameters on the development set, obtaining the lowest DetER for an `onset` of 0.1, `offset` of 0.15, `shift length` of 0.1 and the rest of parameters equal to 0.

Table 2: *VAD systems that will be used on the `eval` dataset. Results reported on `dev` dataset. The systems with ** notation had their weights fine-tuned on `dev` subsets. The rest of the systems are assumed to be pretrained.*

Sys.	**VAD System**	**FA**	**MISS**	**DetER**
1	MarbleNet**	6.23	2.01	8.24
2	Seg3.0**	**4.40**	1.92	**6.32**
3	Seg + Seg3.0 + MarbleNet**	7.33	**0.82**	8.15

2.2.3. Pyannote Powerset Segmentation Fine-tuning

The powerset multi-class segmentation approach presented the best VAD results on the development set, among the preliminary systems, through the `pyannote-audio/segmentation-3.0` model. Therefore, we decide to adapt the above mentioned segmentation module on the 19.75 hours of supervised diarization labels, expecting an improvement in VAD performance too. The dataset is split in 80% training and 20% validation. We train the end-to-end model in the powerset multi-class mode, with a maximum of 2 speakers per frame and 3 speakers per chunk, for 150 epochs with the Adam optimizer, a learning rate of 10^{-3} and cosine annealing scheduler with warm restarts [23]. Lastly, we perform checkpoint averaging on the best 10 validation checkpoints. It is worth mentioning that we will also use this fine-tuned model for overlapping speech detection.

2.2.4. Final VAD systems

In Table 2 we present the results on the 35 supervised development audios. Both fine-tuned VAD systems show enhancements compared to their initial versions from Table 1. Notably, the fine-tuned `pyannote-audio/segmentation-3.0` module exhibits an approximately 2% decrease in DetER. Furthermore, we explore several ensembles with both pretrained and fine-tuned systems, the best one consisting in the ensemble of the pretrained `pyannote-audio/segmentation`, the pretrained `pyannote-audio/segmentation-3.0` and the fine-tuned MarbleNet model. The three systems listed above will undergo evaluation using the `eval` dataset. It is important to note that these systems were trained on various subsets of the development data, possibly biasing the results.

2.3. Embeddings Extraction and Clustering

In our exploration of the speaker embeddings module, we prioritize the multi-scale diarization approach [24], assessing various architectures and scale configurations. This approach, introduced by Park et al., mitigates the trade-off between speaker representation quality and temporal resolution. For clustering speaker features, we use the auto-tuning NME Spectral Clustering algorithm [25]. We investigate three pretrained architectures for the multi-scale extractor: ECAPA-TDNN [26], Titanet-S [27], and Titanet-L. Across each architecture, we experiment with scale numbers ranging from 3 to 8, equal scale weights, 50% scale overlap and scale sizes spanning from 0.5 to 4 seconds. Our top-performing system utilizes the pretrained Titanet-L model from the NeMo Toolkit , employing 6 scales with sizes: `[0.5, 1, 1.5, 2, 2.5, 3]`. Additionally, we explore the VBx diarization system that leverages a pretrained ResNet101 backbone to extract x-vectors, followed by the initial Agglomerative Hierarchical Clustering step and the final variational Bayes HMM clustering.

2.4. Overlap Detection

The second significant challenge that the DISPLACE dataset presents is the high percentage of overlapped speech. There is approximately 16% overlapped speech present in the `dev` dataset. One of the biggest limitations of clustering-based diarization systems is the inability to detect more than one label per timestamp, meaning they are not overlap-aware. In order to reduce this significant error, we explore both a multi-scale neural overlapping speech detection (OSD) system, as well as a greedy post-processing technique using the overlap detection outputs of an external system.

2.4.1. Training the Multi-Scale Diarization Decoder

Following [19], we train from scratch a neural multi-scale diarization decoder (MSDD) [24] for overlap detection. For training, we used both the `dev` dataset, as well as the `AMI Microphone Array` dataset. We performed an 80/20 train-validation split on the `dev` dataset and then combined the resulting subsets with the `AMI` dataset. Furthermore, we process the pair-wise speakers files with a step of 20. Finally, we trained the model in a 6-scale configuration, as mentioned in Section 2.3, with the speaker model frozen, for 30 epochs with the Adam optimizer, a learning rate of 10^{-3} and the cosine annealing scheduler. We will use the MSDD module with a sigmoid threshold of 0.5, tuned on the validation split. A smaller sigmoid threshold results in a higher number of overlap segments, which can be beneficial in systems fusion with Dover-Lap [28], even though it might lead to worse results when the neural diarizer module is evaluated alone. This module also takes in consideration speaker embeddings when assigning overlap speaker labels.

2.4.2. External Overlap Post-Processing

In addition to the MSDD module, we also evaluate the performance of both the pretrained `pyannote-audio/segmentation-3.0`, as well as the fine-tuned powerset multi-class segmentation model in Section 2.2.3, for the OSD task. In Table 3 we present the detection error rate of overlapping speech for the pyannote OSD systems on the validation split used in Section 2.2.3, using a `min_duration_on` and `min_duration_off` equal to 0 for both systems. A substantial amount of overlapping speech is detected by both systems, seeing a slight improvement with the fine-tuned system. However, these overlap detections cannot be integrated directly with the multi-scale approach in order to obtain speaker labels, as they do not use the speaker features extracted by the Titanet-L model.

Table 3: *DetER of overlapping speech on the validation split used in Section 2.2.3. The ** notation denotes the system was fine-tuned.*

Sys.	OVL System	FA	MISS	DetER
1	Seg3.0	13.14	65.26	78.98
2	Seg3.0**	12.82	60.61	73.45

Even though several studies on overlap-aware diarization have been conducted in recent years [29, 30], we opt for a simpler method that still yields a substantial decrease in DER. Following the techniques used in [31] and [32], we utilize a greedy post-processing technique for adding the overlapping speech predictions of an OSD, to a non-overlap aware diarization system. In Fig. 1, we present the pipeline for adding external overlap detections to a non-overlap aware system through the proposed algorithm. Firstly, we extract the overlapped speech segments separately with the OSD system and filter them using the speech segments predicted by the VAD module, in order to avoid an increase in FA error. Secondly, we split the overlap segments using the speaker change frontiers, in case the end of the previous segment is equal to the start of the next one. Lastly, we assign the label to the overlap segment by computing the distance from the center of each overlap segment, to the center of the closest speaker that is different than the current speaker. We call this method "greedy", because we make the assumption that nearby speakers are more probable to speak simultaneously and we do not take in consideration the speaker features in the overlapping regions. Another limitation of this system is that we cannot return more than two overlapping speakers per frame.

Figure 1: *Greedy overlap post-processing algorithm for non-overlap aware diarization systems.*

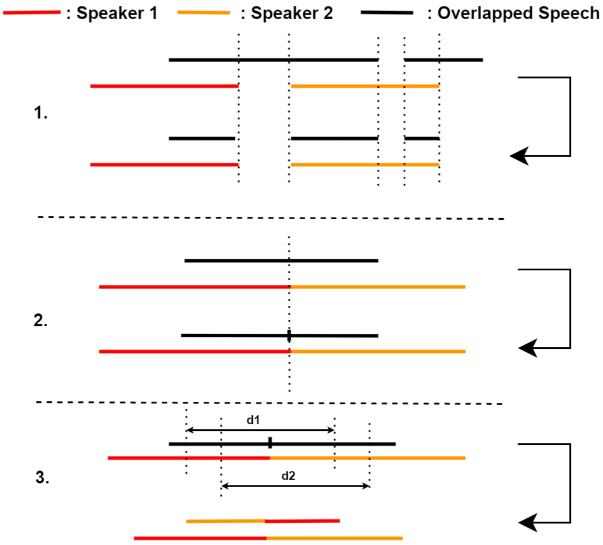

Table 4: *DER results on* `dev` *and* `eval` *subsets, with no collar and overlap error included.* `Seg` *represents the* `pyannote-audio/segmentation` *model.* `Seg3.0` *represents the* `pyannote-audio/segmentation-3.0` *model.* `Multiscale` *and* `VBx` *represent the embeddings extraction and clustering approaches explained in Section 2.3. For the OSD systems, the* `MSDD` *module is the neural diarizer from Section 2.4.1 and the other OSD systems are used in conjunction with the greedy post-processing technique. The last 3 lines represent the DOVER-Lap fusion of several overlap-aware diarization systems. Modules followed by the ** notation are either trained or fine-tuned on the* `dev` *set, while the other modules are assumed to be pretrained.*

System	**VAD**	**Embs. Extractor+Clustering**	**OSD**	**DER**	
				dev	**eval**
Baseline	-	-	-	29.16	34.76
Pyannote Diarization 3.1	Seg3.0	Ecapa-TDNN+AHC	Seg3.0	29.53	34.96
-	MarbleNet [19]	Multiscale	No	32.01	34.62
-	MarbleNet**	Multiscale	No	29.48	33.23
-	Seg	Multiscale	No	-	33.32
-	Seg3.0	Multiscale	No	29.05	32.14
-	Seg3.0**	Multiscale	No	27.10	31.04
-	MarbleNet** + Seg + Seg3.0	VBx	No	29.71	-
-	MarbleNet** + Seg + Seg3.0	Multiscale	No	29.58	31.47
1	MarbleNet** + Seg + Seg3.0	VBx	Seg3.0**	25.51	32.32
2	MarbleNet** + Seg + Seg3.0	Multiscale	Seg3.0	27.66	29.85
3	MarbleNet** + Seg + Seg3.0	Multiscale	Seg3.0**	25.64	29.23
4	Seg3.0**	Multiscale	MSDD** 0.5 th.	29.06	30.90
5	Seg3.0**	Multiscale	Seg3.0**	**22.77**	28.49
2 + 3 + 5	-	-	-	25.99	28.95
1 + 2 + 3 + 5	-	-	-	26.37	28.66
1 + 2 + 3 + 4 + 5	-	-	-	26.37	**28.04**

3. Final Results

In Table 4 we present several configurations that were evaluated on the `dev` and `eval` datasets. We perform an analysis using both non-overlap and overlap aware systems, in order to highlight the impact of both the VAD and OSD modules. In the first section, we present the baseline and 2 pretrained systems that were not adapted on the new development data. In the second section, we observe the impact of the VAD modules through the non-overlap aware systems' results, seeing a major improvement with both the VAD ensemble, as well as the fine-tuned `pyannote-audio/segmentation-3.0` module. For the embeddings extraction and clustering modules, we explore the `VBx` and `Multiscale` approaches.

Systems 1 to 5 are overlap-aware, with systems 3 and 5 achieving the lowest DER, as expected from the non-overlap aware results, with the best single system achieving 28.49% DER on the `eval` dataset. The other overlap-aware systems are mainly used in order to diversify the input systems for the system fusion step. Finally, we use the DOVER-Lap technique [33] with the greedy fusion algorithm and present our final submission using the fusion of 5 overlap-aware systems, achieving the lowest DER on the `eval` dataset of 28.04%.

We can make a few observations from the results presented above: (i) there is a slight mismatch between the results on the `dev` set and `eval` set, that can be caused either by an overfit on the `dev` set or a slight data distribution mismatch between the subsets; (ii) we can observe an improvement between the MarbleNet model trained in Section 2.2.2 and the model trained in [19], concluding that the automatically annotated recordings can also improve speech detection; (iii) the single fine-tuned powerset multi-class segmentation module greatly outperforms any other VAD module, even the ensembles; (iv) our greedy post-processing technique greatly improves the results of non-overlap aware diarization systems; (v) DOVER-Lap fusion benefits from diverse overlap-aware input systems, even if the input systems' performances greatly vary.

4. Conclusions

In this paper we propose a speaker diarization solution for the DISPLACE 2024 challenge. We conduct a comprehensive study for each module of a hybrid-diarization system, exploring semi-supervised learning techniques, ensemble of systems, neural overlap detection, as well as an overlap detection post-processing technique for non-overlap aware systems.

We would like to highlight the impact of the overlap post-processing technique on the final results, and conclude that a technique that would also take in consideration speaker features and more than 2 speakers per frame, when assigning labels, may further improve the results. However, it is also important to note that OSD systems still struggle to accurately detect overlapping speech, as the fine-tuned powerset multi-class segmentation module still had an overlapping speech detection error of over 73% on the `dev` set.

Finally, our best independent system consists in a fine-tuned `pyannote-audio/segmentation-3.0` model used for the VAD and OSD tasks, with a multi-scale embeddings extractor using the Titanet-L model and auto-tuning NME Spectral Clustering. The overlapped speech detections from the OSD are added through a greedy post-processing technique. This system achieves a DER of 28.49% on `eval` dataset, a relative improvement of 18.04% compared to the baseline. The best fusion of systems and our final submission on the Evaluation Phase 1 Track for Speaker Diarization achieves a DER of 28.04% with a relative improvement of 19.33% compared to the baseline.

5. Acknowledgements

This work was partly funded by EU HORIZON project no. 101070190 (AI4TRUST project) and by a grant of the Ministry of Research, Innovation and Digitization, CNCS/CCCDI - UEFISCDI, project number PN-IV-P8-8.1-PRE-HE-ORG-2023-0078, within PNCDI IV.

6. References

[1] 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.

[2] S. Cornell, M. Wiesner, S. Watanabe, D. Raj, X. Chang, P. Garcia, Y. Masuyama, Z.-Q. Wang, S. Squartini, and S. Khudanpur, "The chime-7 dasr challenge: Distant meeting transcription with multiple devices in diverse scenarios," *arXiv preprint arXiv:2306.13734*, 2023.

[3] E. Z. Xu, Z. Song, S. Tsutsui, C. Feng, M. Ye, and M. Z. Shou, "Ava-avd: Audio-visual speaker diarization in the wild," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 3838–3847.

[4] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 995–19 012.

[5] S. Baghel, S. Ramoji, S. Jain, P. R. Chowdhuri, P. Singh, D. Vijayasenan, and S. Ganapathy, "Summary of the displace challenge 2023–diarization of speaker and language in conversational environments," *arXiv preprint arXiv:2311.12564*, 2023.

[6] S. Team, "Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier," https://github.com/snakers4/silero-vad, 2021.

[7] H. Bredin, R. Yin, J. M. Coria, G. Gelly, P. Korshunov, M. Lavechin, D. Fustes, H. Titeux, W. Bouaziz, and M.-P. Gill, "Pyannote. audio: neural building blocks for speaker diarization," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2020, pp. 7124–7128.

[8] O. Kuchaiev, J. Li, H. Nguyen, O. Hrinchuk, R. Leary, B. Ginsburg, S. Kriman, S. Beliaev, V. Lavrukhin, J. Cook, P. Castonguay, M. Popova, J. Huang, and J. M. Cohen, "NeMo: a toolkit for building AI applications using neural modules," *CoRR*, vol. abs/1909.09577, 2019. [Online]. Available: http://arxiv.org/abs/1909.09577

[9] F. Jia, S. Majumdar, and B. Ginsburg, "MarbleNet: Deep 1d time-channel separable convolutional neural network for voice activity detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6818–6822.

[10] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *ICASSP 2019-2019 IEEE International conference on acoustics, speech and signal processing (ICASSP).* IEEE, 2019, pp. 5796–5800.

[11] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[12] N. R. Koluguri, T. Park, and B. Ginsburg, "Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2022, pp. 8102–8106.

[13] J.-w. Jung, Y. J. Kim, H.-S. Heo, B.-J. Lee, Y. Kwon, and J. S. Chung, "Pushing the limits of raw waveform speaker recognition," *arXiv preprint arXiv:2203.08488*, 2022.

[14] F. Landini, J. Profant, M. Diez, and L. Burget, "Bayesian hmm clustering of x-vector sequences (vbx) in speaker diarization: theory, implementation and analysis on standard tasks," *Computer Speech & Language*, vol. 71, p. 101254, 2022.

[15] W. Kraaij, T. Hain, M. Lincoln, and W. Post, "The ami meeting corpus," in *Proc. International Conference on Methods and Techniques in Behavioral Research*, 2005.

[16] 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

[17] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2017, pp. 5220–5224.

[18] S. de Vries and D. Thierens, "A reliable ensemble based approach to semi-supervised learning," *Knowledge-Based Systems*, vol. 215, p. 106738, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705121000010

[19] G. Pîrlogeanu, D. Oneață, A.-L. Georgescu, and H. Cucu, "The SpeeD–ZevoTech Submission at DISPLACE 2023," in *Proc. INTERSPEECH*, 2023, pp. 3572–3576.

[20] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356

[21] H. Bredin and A. Laurent, "End-to-end speaker segmentation for overlap-aware resegmentation," *arXiv preprint arXiv:2104.04045*, 2021.

[22] A. Plaquet and H. Bredin, "Powerset multi-class cross entropy loss for neural speaker diarization," in *Proc. INTERSPEECH 2023*, 2023, pp. 3222–3226.

[23] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," 2017.

[24] T. J. Park, N. R. Koluguri, J. Balam, and B. Ginsburg, "Multi-scale speaker diarization with dynamic scale weighting," in *Interspeech*, 2022.

[25] T. J. Park, K. J. Han, M. Kumar, and S. Narayanan, "Auto-tuning spectral clustering for speaker diarization using normalized maximum eigengap," *IEEE Signal Processing Letters*, vol. 27, p. 381–385, 2020. [Online]. Available: http://dx.doi.org/10.1109/LSP.2019.2961071

[26] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," 10 2020.

[27] N. R. Koluguri, T. Park, and B. Ginsburg, "Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context," 2021.

[28] N. Tawara, M. Delcroix, A. Ando, and A. Ogawa, "Ntt speaker diarization system for chime-7: Multi-domain, multi-microphone end-to-end and vector clustering diarization," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 11 281–11 285.

[29] S. Horiguchi, P. Garcia, Y. Fujita, S. Watanabe, and K. Nagamatsu, "End-to-end speaker diarization as post-processing," 2020.

[30] M. Cheng, W. Wang, Y. Zhang, X. Qin, and M. Li, "Target-speaker voice activity detection via sequence-to-sequence prediction," 2023.

[31] L. Bullock, H. Bredin, and L. P. Garcia-Perera, "Overlap-aware diarization: resegmentation using neural end-to-end overlapped speech detection," 2019.

[32] F. Landini, S. Wang, M. Diez, L. Burget, P. Matějka, K. Žmolíková, L. Mošner, A. Silnova, O. Plchot, O. Novotný, H. Zeinali, and J. Rohdin, "But system for the second dihard speech diarization challenge," 2020.

[33] D. Raj, L. P. Garcia-Perera, Z. Huang, S. Watanabe, D. Povey, A. Stolcke, and S. Khudanpur, "Dover-lap: A method for combining overlap-aware diarization outputs," in *IEEE Spoken Language Technology Workshop*, 2021, pp. 881–888.