IDIAP RESEARCH REPORT

ENHANCING SPEAKER DIARIZATION USING CORRELATION-BASED CLUSTERING INITIALIZATION

Pradeep Rangappa Amrutha Prasad
Srikanth Madikeri Petr Motlicek

Idiap-RR-09-2025

AUGUST 2025

Centre du Parc, Centre du Parc, Rue Marconi 19, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

Enhancing Speaker Diarization using Correlation-Based Clustering Initialization

Pradeep Rangappa[1], Amrutha Prasad[1,2], Srikanth Madikeri[3], Petr Motlicek[1,2]

[1] Idiap Research Institute, Martigny, Switzerland
[2] Brno University of Technology, Czech Republic
[3] University of Zurich, Switzerland

{pradeep.rangappa, amrutha.prasad, petr.motlicek}@idiap.ch, srikanth.madikeri@cl.uzh.ch

***Abstract*—Speaker diarization becomes challenging in multilingual and code-switched speech due to frequent speaker changes and acoustic variability. While PyAnnote achieves state-of-the-art performance on standard benchmarks, its effectiveness drops on complex datasets like DISPLACE-2. To address this issue, we propose to improve the performance of the global agglomerative clustering by improving the input embeddings. Specifically, we enhance the embeddings by analyzing their pairwise correlations and averaging highly correlated embeddings. This approach improves speaker representation for highly correlated embeddings while reducing speaker confusion and improving clustering accuracy. Evaluated on DISPLACE-2 Track-1 (multilingual speaker diarization), our method shows a 3% relative DER improvement over the baseline, and 8% when combined with segmentation fine-tuning. Notably, the approach reduces DER in rapid turn-taking and language transition regions, improving robustness in code-mixed speech.**

***Index Terms*—speaker diarization, ECAPA-TDNN embedding, local speaker segmentation, DISPLACE-2**

I. Introduction

Speaker diarization involves the segmentation of speech into coherent segments based on speaker identities, addressing the question of "who spoke when". Traditional methods [1]–[3] employ a cascade of steps, including voice activity detection, speaker embedding for discriminative representations, and clustering. However, these approaches have drawbacks, such as error propagation and difficulty handling overlapping speech regions without additional post-processing. In response to these limitations, a new family of approaches, known as end-to-end diarization (EEND) [4], has emerged. EEND rethinks speaker diarization entirely, aiming to train a single neural network that directly processes the audio recording and outputs the desired partitions. There are two popular and recent approaches, such as Variational Bayessian Clustering (VBx) [5] and Pyannote [6] that have played crucial roles for speaker diarization. VBx leverages a modular approach by incorporating components like VAD, x-vector extraction using RESNET architecture, and clustering algorithms. This framework employs a variational Bayesian model to handle uncertainty in speaker modeling, enhancing the robustness of diarization results.

Recent advances in speaker embeddings, such as d-vectors [7], i-vectors [8], x-vectors [5], TITANET [9] [10] and ECAPA-TDNN [11], have improved performance on monolingual corpora but struggle in multilingual settings, especially with code-switching. Although, the use of different embedding vectors have shown improvement on the standard monolingual corpus [12] [13] , they do not guarantee the impact on a multi-lingual scenario [14]. However, the embeddings trained for mono-lingual diarization do no perform well in the presence of code-switching. Efforts in ongoing research are consistently aimed at progressing speaker diarization within the context of natural multilingual conversations. There are many challenges [15] [16] organized for mono-lingual corpus. In contrary, [17] aimed to highlight new hurdles and development in speaker and language diarization for multi-lingual conversational speech data. The second DIriazation of SPeaker and LAnguage in Conversational Environments (DISPLACE) challenge [18] emphasizes addressing challenges in speaker and language diarization and automatic speech recognition (ASR) in code-mixed/switched multi-accent conversational scenarios. Track-1 focuses on Speaker Diarization in multilingual scenarios, requiring teams to diarize audio with speakers using multiple code-mixed and/or code-switched languages. Our participation is in track-1 i.e., speaker diarization in multilingual scenarios.

Despite the demonstrated effectiveness of the PyAnnote system in monolingual corpora, it encounters notable challenges when dealing with multilingual code-mixed data, such as DISPLACE-2, leading to a decline in performance. Motivated by these challenges, we address the speaker diarization in multilingual scenario in two phases: (i) improving the ECAPA-TDNN embedding generated by using the average of the highly correlated speaker segments and, (ii) fine-tuning the local speaker segmentation model by generating the pseudo labels on the unsupervised speech data along with the development data.

The paper is organized as follows: Section II briefly describes the proposed work. Section III and IV-A discusses experimental design and results. Section V concludes the paper, suggesting future directions.

II. Two-Stage Clustering

Figure 1 illustrates a high-level overview of the speaker diarization pipeline. Part (A), enclosed in a blue dashed box,

represents the PyAnnote system. It consists of the following sub-processes:

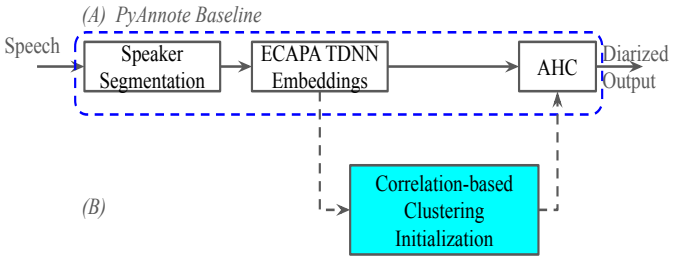

Fig. 1. Functional blocks of the state-of-the-art PyAnnote speaker diarization system (A) and the proposed correlation-based clustering initialization module (B).

- **Speaker segmentation** [19] is applied to create speaker-specific segments for a given audio file. An end-to-end neural model [20] is applied using a 5 s sliding window with a 500 ms step. For each step, the model outputs the probability of a speaker being active every 16 ms for up to $K_{max} = 3$ speakers.
- **Speaker embeddings**: For each window w, K_w speaker embeddings are extracted—one per active speaker. These embeddings are based on the ECAPA-TDNN architecture.
- **Global agglomerative clustering** identifies the embeddings from same speaker and assign them as a single cluster. The clustering uses Agglomerative Hierarchical Clustering (AHC) approach thereby aggregating the clustered local speaker segments into a diarization output.

In the PyAnnote pipeline, relations between local speaker embeddings are considered only during global agglomerative clustering. Part (B) of Figure 1 introduces an enhancement over the baseline by incorporating a correlation-based clustering initialization module. This component leverages the similarity structure among speaker embeddings to provide a more informed starting point for the AHC algorithm. It connects to the pipeline via dashed arrows, indicating it's an optional add-on or enhancement to the baseline.

A. Proposed Clustering Initialization

In Figure 2, the segmentation output is the probability of the local speaker being active every 16 ms as shown in Figure 2 (a). Chunks [C_1, C_2, ...C_n], with a 1-second size and overlap every 16 ms, are processed (Figure 2 (b)). Each chunk j generates one embedding per speaker forming S_{1j}, S_{2j}, S_{3j} vectors of 256 dimensions. For example, $\{S_{11}, S_{21}, S_{31}\}$ and $\{S_{12}, S_{22}, S_{32}\}$ represent embeddings for C_1 and C_2 chunks, respectively.

We modify the embeddings by analyzing the pairwise correlation among the ECAPA-TDNN embeddings. Let $\mathbf{S}_1$ and $\mathbf{S}_2$ denote the sets of ECAPA TDNN vectors for chunks 1 and 2:

$$\mathbf{S}_1 = \{\mathbf{S}_{11}, \mathbf{S}_{21}, \mathbf{S}_{31}\} \tag{1}$$

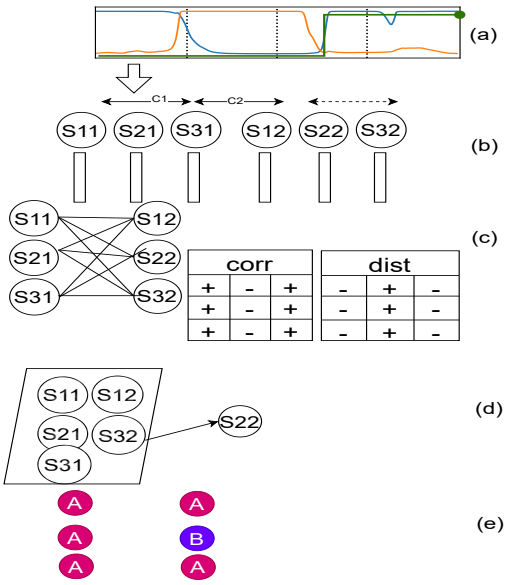

Fig. 2. Functional blocks in the proposed clustering initialization. The details of (a) to (e) are described in section II-A

$$\mathbf{S}_2 = \{\mathbf{S}_{12}, \mathbf{S}_{22}, \mathbf{S}_{32}\} \tag{2}$$

We then proceed as follows:

- *Correlation Matrix*: Calculate the Pearson correlation [21] R_{ij} between vectors from chunk 1 and chunk 2:

$$R_{ij} = \text{Corr}(\mathbf{S}_{1i}, \mathbf{S}_{2j}) \tag{3}$$

- *Distance Matrix*: Compute the Euclidean distance [22] D_{ij} between vectors from chunk 1 and chunk 2:

$$D_{ij} = \|S_{1i} - S_{2j}\| \tag{4}$$

  The matrices are shown in Figure 2 (c), where the '+' sign indicates higher correlation or distance, and '-' represents lower values.
- *Average the Embeddings*: If the correlation R_{ij} is above a threshold T_R and the distance D_{ij} is below T_D, the embeddings are averaged:

$$\mathbf{S}_{\text{avg},i} = \frac{\mathbf{S}_{1i} + \mathbf{S}_{2j}}{2} \tag{5}$$

  Other vectors remain unchanged.

For highly correlated and closely spaced vectors, such as S_{11}, S_{21}, S_{31}, S_{12}, and S_{32}, we represent them as their mean, replicated across local speakers with high correlation ('A' in Figure 2 (e)). In cases like S_{22}, where the correlation is low and the distance is high, the vector is scaled (represented as 'B') to minimize confusion at later stages. The modifications to the ECAPA TDNN are extended to every pair of consecutive chunks until the last chunk, i.e., the end of the utterance.

B. Fine-Tuning Speaker Segmentation Model

Fine-tuning on domain-specific data has been shown to improve the system performance. Therefore, to further improve clustering accuracy, we finetune a part of the PyAnnote

pipeline that is the speaker segmentation model.The speaker segmentation model in the PyAnnote-based system uses a PyanNet architecture with SincNet [23] features. It consists of two stacked bi-directional LSTM layers (128 units each), no temporal pooling, two feedforward layers (128 units, tanh activation), and a final classification layer (2 units, softmax activation). In our work, we finetune the speaker segmentation

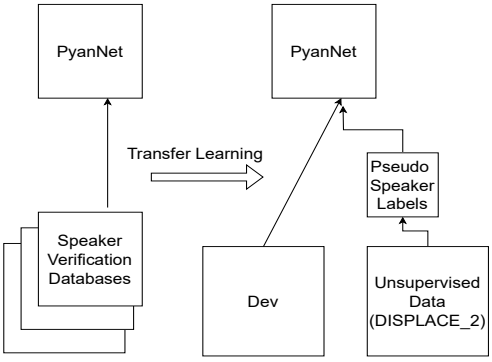

Fig. 3. Fine-tuning of the speaker segmentation model

model by using the annotated development data (Dev) provided by the DISPLACE-2 team as shown in Figure 3. On top of it, we utilize the unsupervised data (≈ 100 hours) provided as a part of the challenge, and generate the pseudo speaker labels from the pyannote system. Fine-tuning involves adapting the pre-trained model using annotated development data (Dev) from DISPLACE-2 with permutation-invariant training. The permutation-invariant loss function L is given by:

$$L(y, \hat{y}) = \min_{\text{perm}(y)} L_{\text{BCE}}(\text{perm}(y), \hat{y}) \tag{6}$$

where $\hat{y} = f(X)$ is the model output and $\text{perm}(y)$ represents all permutations of y. We compute binary cross-entropy losses between all pairs of dimensions and use the Hungarian algorithm to find the permutation that minimizes the loss, ensuring effective handling of speaker label permutations.

III. Experimental Setup

A. Database

For the DISPLACE-2 challenge, the organizers released over 100 hours of data for development and evaluation. This data presents challenges for speech-based systems like speaker and language identification or automatic speech recognition.

TABLE I
DESCRIPTION OF DISPLACE SECOND CHALLENGE DATASET FOR TRACK-1

Dataset	#Files	Duration (hh:mm:ss)
DEV (supervised)	35	19:44:44
DEV (unsupervised - part 01)	104	66:45:11
DEV (unsupervised - part 02)	65	54:51:11
Eval (phase 01)	32	17:56:11
Hinglish Data[1]	6	02:11:09

[1]Special thanks to Muskan Gupta for open sourcing the audio samples and speaker annotations on "Hindi English Conversational Speech". Link: https://github.com/muskang48/Speaker-Diarization?tab=readme-ov-file#dataset

B. Hardware Requirements

We utilised a NVIDIA GeForce RTX 3090 single GPU instance for the PyAnnote baseline (mainly for local speaker segmentation model and ECAPA-TDNN embeddings extraction). The real-time factor is around 2.6% meaning 1 hour of speech was diarized in 1.6 minutes of time. For the experiments related to modifications of embedding vectors, they were saved as numpy n-dim array and loaded on CPU Architecture (x86_64) with 64-bit operating mode.

IV. Results and Discussion

A. Evaluation on DISPLACE 2 Dataset

Table II presents the results of different methods evaluated on the development and test datasets of DISPLACE-2 (Track-1) in phase-01, specifically focusing on Diarization Error Rate (DER).

TABLE II
DER ON DIFFERENT METHODS EVALUATED ON THE DEV AND TEST (PHASE-01) DATASET OF DISPLACE-2 (TRACK-1)

Methods	DER (%)	
	DEV	TEST
Baseline [Spectral Clustering]	30.44	-
Baseline [VB-HMM + Overlap]	29.16	34.76
PyAnnote [3.1]	29.53	34.96
Fine-tuned seg model [on DEV]	25.21	33.99
Fine-tuned seg model [on DEV + Pseudo Labels]	27.37	34.28
Fine-tuned seg model [on DEV + Hinglish data]	25.46	33.34
Proposed clustering initialization	29.08	34.00
Fine-tuned seg model + Proposed clustering initialization	24.93	32.10

It can be observed that the baseline system provided by the DISPLACE_2024 team (xvector from TDNN architecture with VB-HMM clustering) and the PyAnnote (v3.1) systems operate at a DER of 34.76% and 34.96% on test data. The DER on the test data was obtained from the CodaLab portal (an automated system that accepts diarization RTTMs and provides utterance-level DER) shared by the organizers. In the fine-tuning stage, we adapted the SincNet (also called PyAnnote) model on dev data, pseudo labels from unsupervised data, and a sample of 2 hours of code-mixed data (Hindi mixed with English). Fine-tuning on dev data reduces DER by 0.97% absolute (34.96% compared to 33.99%, as shown in Table II). Extending fine-tuning to dev+pseudo labels yields better DER than the baseline and PyAnnote systems. Additionally, fine-tuning on 2 hours of multi-lingual data decreases DER by 1.62% absolute (34.96% compared to 33.34% in the table). The proposed clustering initialization, when evaluated with a pre-trained local speaker segmentation model, achieves a 0.55% absolute improvement (34.96% compared to 33.34% in the table). The effect of the clustering initialization is

most pronounced when applied to fine-tuned local speaker segments. The combination of the fine-tuned segmentation model and the modified ECAPA-TDNN embedding yields a 2.86% absolute improvement (34.96% compared to 32.10% in the table) or 8.2% relative in DER compared to the baseline PyAnnote system. The proposed framework secured 4th position on the leaderboard out of 10 entries, demonstrating the effectiveness of analyzing patterns in the embedding space with less data.

B. Evaluation on AMI Dataset

In order to test the generalization capability of the proposed work, we extended the same framework on the "AMI-SDM-Mini" dataset [24] and observed the improvement in the DER as shown in Table III.

TABLE III
DER ON DIFFERENT METHODS EVALUATED ON THE TEST SET OF AMI-SDM-MINI DATASET

Methods	DER on Test set (%)
PyAnnote (3.1)	25.97
Fine-tuned segmentation model	25.14
Finetune segmentation model + Proposed clustering initialization	24.88

C. Distribution of False Alarms, Missed Detection and Speaker Confusions

Table IV gives the comparison of different systems on missed detection, false alarm, confusion, total, and DER on DISPLACE 2 dataset. The false alarm rate has decreased from 6063.2 seconds (Baseline) to 5175.9 seconds (Proposed clustering initialization) and 5427.7 seconds (finetuned speaker segmentation model + proposed clustering initialization), indicating an improvement in system performance.

D. Impact of Proposed Clustering Initialization on Speaker Transition Regions

Figure 4 illustrates the DER observed during speaker transitions at the utterance level for different systems. The comparison includes the PyAnnote baseline, embeddings modified using the proposed clustering initialization, and a finetuned speaker segmentation model with improved ECAPA-TDNN embeddings. We observe that the majority of utterances in the test set show a reduction in DER (indicated by pink asterisks) when using the modified ECAPA-TDNN embeddings. This suggests that the proposed initialization helps in better distinguishing speakers, particularly during transition regions where speaker overlap or quick turn-taking occurs. The improvement appears to stem from reduced speaker confusions, although a more detailed analysis of the distribution of false alarms, missed detections, and speaker confusions is needed to confirm this trend.

TABLE IV
COMPARISON OF DIFFERENT SYSTEMS ON MISSED DETECTION, FALSE ALARM, CONFUSION, TOTAL, AND DER.

System	Missed Detection (s)	False Alarm (s)	Confusion (s)	Total (s)	DER (%)
Baseline	7557.7	**6063.2**	10846.4	70034.3	34.9
Proposed clustering initialization	7801.4	**5175.9**	10829.9	70034.3	33.9
Finetuned + Proposed clustering initialization	7424.0	**5427.7**	9621.8	70034.3	32.1

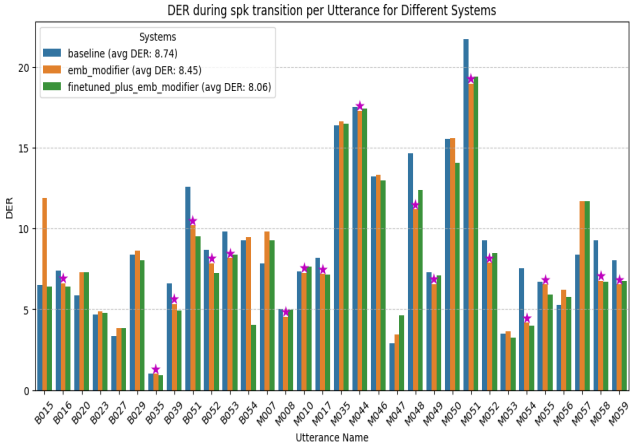

Fig. 4. Utterance-level DER during speaker transition regions for different systems: (i) PyAnnote baseline, (ii) system with modified embeddings using the proposed clustering initialization, and (iii) fine-tuned speaker segmentation model with improved ECAPA-TDNN embeddings. The plot highlights improvements in DER, particularly in transition regions, indicating reduced speaker confusions.

These findings highlight the effectiveness of the proposed clustering initialization and embedding refinements in improving speaker transition accuracy, which is often a challenging aspect of diarization tasks.

E. Impact of Proposed Clustering Initialization on Language Transition Regions

Figure 5 presents the DER measured at utterance-level segments containing language transitions, particularly in code-mixed regions where speakers switch between languages. The comparison spans three systems: the PyAnnote baseline, a system using embeddings adjusted through the proposed clustering initialization, and a fine-tuned speaker segmentation model incorporating enhanced ECAPA-TDNN embeddings.

Quantitatively, the baseline system shows an average DER of 6.05%, while the system with modified embeddings reduces it to 5.77%. Further finetuning on top of the embedding modifications brings the DER down to 5.57%. These gains are particularly evident in utterances highlighted by magenta stars. This suggests that the combination of clustering initialization and embedding refinement enhances the system's ability to handle language transitions, which are often acoustically diverse and challenging for diarization systems.

Overall, the figure indicates consistent performance improvement across a majority of utterances, with the proposed method effectively reducing speaker confusions and detection errors during multilingual transitions.

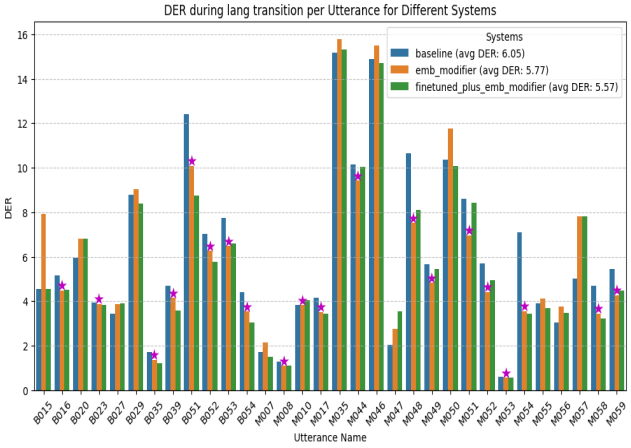

Fig. 5. Utterance-level DER during language transition regions for different systems: (i) PyAnnote baseline, (ii) system with modified embeddings using the proposed clustering initialization, and (iii) fine-tuned speaker segmentation model with improved ECAPA-TDNN embeddings.

F. Illustrative Example

An audio file from the development set (file ID: B034.wav, duration=00:30:20) is chosen to compare the impact of segmentation outputs on the overall DER. Figure 6 shows the details of the segmention outputs and the diarized outputs obtained for different diarization pipelines.

The impact of DER is studied under three below categories of speaker diarization pipeline:

- *Pre-trained PyAnnote pipeline*: This method is designed to identify an active local speaker at nearly every time instance. However, it brings about a increase in false alarms, thereby increasing the DER in a multilingual scenario. The DER obtained for this illustrative example is 27.18% as shown in Figure 6 (i).
- *Finetuned segmentation model*: In this approach, adaptation to speaker changes is achieved based on the multilingual corpus. It exhibits a gradual reduction in false alarms and miss detection over time, showcasing enhanced effectiveness compared to the pre-trained method. The DER associated with this strategy is 24.11% as shown in Figure 6 (ii).
- *Finetuned segmentation model with proposed clustering initialization*: Similar to the finetuned stage, this method preserves the capacity to adjust to speaker changes utilizing the multilingual corpus. Despite retaining the instances of false alarms and miss detections, the proposed clustering initialization enhances the discrimination of ecapa-tdnn embeddings, thereby minimizing speaker confusions. The corresponding DER for this method is 21.96% as shown in Figure 6 (iii).

Out of 35 utterances in the development set, we observed that there are 28 utterances where the DER of the proposed method shows improvement as compared to that of the baseline. However, the remaining 7 utterances served as negative examples. We observed that these utterances had small duration speaker turns and the speakers sounded almost similar in nature in the conversation. Due to this, the embeddings are probably smoothened out in the proposed clustering initialization. As a result, the DER slighlty dropped in some of these utterances. The process of explicitly utilizing the statistical features that can add more discriminative power is considered as one of the future scopes of our work.

V. Conclusion

The Pyannote system is a robust solution for speaker diarization, offering local speaker segmentation, embedding extraction, and clustering. Our work specifically enhances the local speaker segmentation model for multilingual contexts and refines speaker embedding by analyzing pairwise correlations between the ECAPA-TDNN embeddings generated from the speaker embedding model. Improving the speaker embeddings with the average of all the highly correlated embeddings improves the speaker representation which can be seen as a complementary method to the cosine distance used in the Global Agglomerative Clustering. We evaluated our method on the first track of the second DISPLACE challenge, which focuses on multilingual speaker diarization. We observe a relative improvement of 3% with our proposed clustering. By fine-tuning the speaker segmentation model and applying our initialization, we achieved an 8% relative improvement in DER compared to the DISPLACE-2 baseline and the Pyannote system.

Acknowledgment

A part of the research described in this work was performed within the ROXANNE project – Real-time netwOrk, teXt and speaker ANalytics for combating orgaNized crimE. This project received funding from the European Union's Horizon 2020 research and innovation programme (2018–2020) under grant agreement number 833635. Part of this work was also supported by the EU Horizon 2020 project ELOQUENCE5 (grant agreement number 101070558).

References

[1] L. Sun, J. Du, C. Jiang, X. Zhang, S. He, B. Yin, and C.-H. Lee, "Speaker diarization with enhancing speech for the first DIHARD challenge." in *Interspeech*, 2018, pp. 2793–2797.

[2] H. Aronowitz and W. Zhu, "Context and uncertainty modeling for online speaker change detection," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8379–8383.

[3] Z. Bai and X.-L. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.

[4] T.-Y. Leung and L. Samarakoon, "Robust end-to-end speaker diarization with conformer and additive margin penalty." in *Interspeech*, 2021, pp. 3575–3579.

[5] 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.

[6] H. Bredin, "Pyannote audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe," in *24th INTERSPEECH Conference (INTERSPEECH 2023)*. ISCA, 2023, pp. 1983–1987.

[7] W. Kang, B. C. Roy, and W. Chow, "Multimodal speaker diarization of real-world meetings using d-vectors with spatial features," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6509–6513.

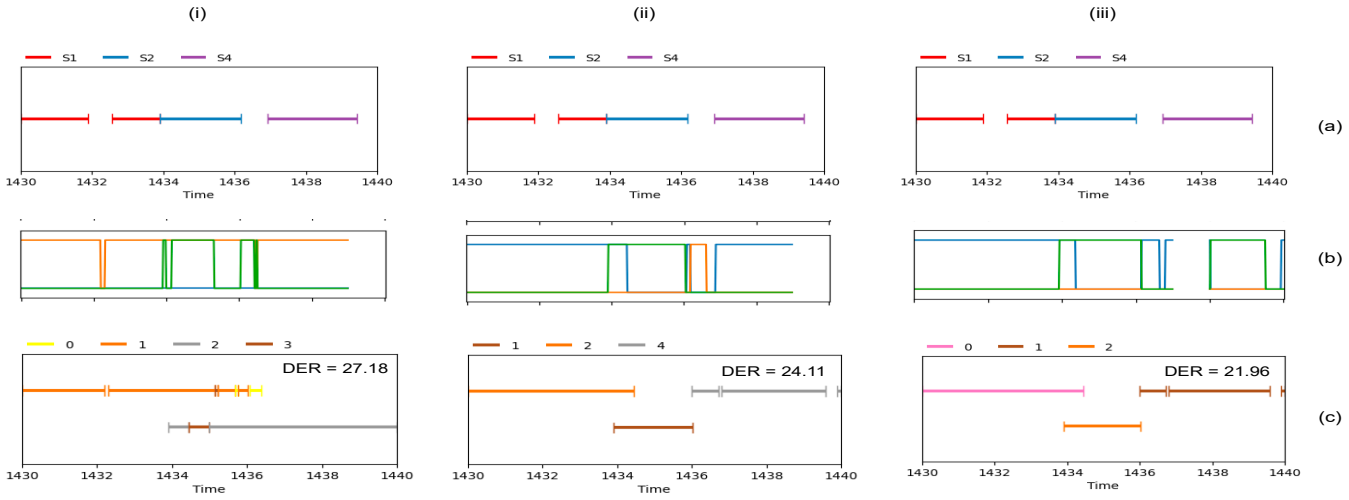

Fig. 6. Illustrative example of the file B034.wav on (i) pre-trained PyAnnote pipeline, (ii) fine-tuned segmentation , (iii) fine-tuned segmenation + proposed clustering initialization between 1430 s and 1440 s. (a) reference rttm, (b) speaker segmentation model output and (c) diarized output along with the total DER at utterance level

[8] G. Sell and D. Garcia-Romero, "Speaker diarization with plda i-vector scoring and unsupervised calibration," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 413–417.

[9] 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.

[10] J. M. Coria, H. Bredin, S. Ghannay, and S. Rosset, "Overlap-aware low-latency online speaker diarization based on end-to-end local segmentation," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 1139–1146.

[11] N. Dawalatabad, M. Ravanelli, F. Grondin, J. Thienpondt, B. Desplanques, and H. Na, "Ecapa-tdnn embeddings for speaker diarization," in *Interspeech*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233024933

[12] S. Novoselov, A. Gusev, A. Ivanov, T. Pekhovsky, A. Shulipa, A. Avdeeva, A. Gorlanov, and A. Kozlov, "Speaker diarization with deep speaker embeddings for DIHARD challenge II." in *INTERSPEECH*, 2019, pp. 1003–1007.

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

[14] O. H. Anidjar, Y. Estève, C. Hajaj, A. Dvir, and I. Lapidot, "Speech and multilingual natural language framework for speaker change detection and diarization," *Expert Systems with Applications*, vol. 213, p. 119238, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417422022564

[15] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. W. Church, C. Cieri, J. Du, S. Ganapathy, and M. Y. Liberman, "The third DIHARD diarization challenge," *ArXiv*, vol. abs/2012.01477, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:227254213

[16] J. Huh, A. Brown, J.-w. Jung, J. S. Chung, A. Nagrani, D. Garcia-Romero, and A. Zisserman, "Voxsrc 2022: The fourth voxceleb speaker recognition challenge," *arXiv preprint arXiv:2302.10248*, 2023.

[17] 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*, vol. abs/2311.12564, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:265308781

[18] S. B. Kalluri, P. Singh, P. Roy Chowdhuri, A. Kulkarni, S. Baghel, P. Hegde, S. Sontakke, D. K T, S. M. Prasanna, D. Vijayasenan, and S. Ganapathy, "The second DISPLACE challenge: Diarization of speaker and language in conversational environments," in *Interspeech 2024*, 2024, pp. 1630–1634.

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

[20] 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.

[21] I. Cohen, Y. Huang, J. Chen, J. Benesty, J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," *Noise reduction in speech processing*, pp. 1–4, 2009.

[22] N. Krislock and H. Wolkowicz, *Euclidean distance matrices and applications*. Springer, 2012.

[23] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *2018 IEEE spoken language technology workshop (SLT)*. IEEE, 2018, pp. 1021–1028.

[24] I. Mccowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska Masson, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus," *Int'l. Conf. on Methods and Techniques in Behavioral Research*, 01 2005.