Mathematical Problems of Computer Science 60, 52–62, 2023.
doi: 10.51408/1963-0109

UDC 004.934

Building a Speaker Diarization System: Lessons from VoxSRC 2023

Davit S. Karamyan[1,2] and Grigor A. Kirakosyan[2,3]

[1]Russian-Armenian University, Yerevan, Armenia
[2]Krisp.ai, Yerevan, Armenia
[3]Institute of Mathematics of NAS RA, Yerevan, Armenia
e-mail: {dkaramyan, gkirakosyan}@krisp.ai

Abstract

Speaker diarization is the process of partitioning an audio recording into segments corresponding to individual speakers. In this paper, we present a robust speaker diarization system and describe its architecture. We focus on discussing the key components necessary for building a strong diarization system, such as voice activity detection (VAD), speaker embedding, and clustering. Our system emerged as the winner in the Voxceleb Speaker Recognition Challenge (VoxSRC) 2023, a widely recognized competition for evaluating speaker diarization systems.

Keywords: Speaker recognition, Speaker diarization, VoxSRC 2023.
Article info: Received 27 September 2023; sent for review 28 September 2022; received in revised form 14 November 2023; accepted 16 November 2023.

1. Introduction and Related Work

Speaker diarization (SD) is the process of dividing audio into segments according to the speaker's identity. It is the process of determining "who spoke when" in a multi-speaker audio signal. A typical SD system usually consists of several steps: (1) segment the input audio into speech segments using a Voice Activity Detector (VAD), (2) generate speaker segments from the speech segments by either using a uniform sliding window segmentation or by detecting speaker turns, (3) extract speaker embeddings for each of the speaker segment, (4) group the resulting speaker embeddings into clusters using clustering algorithms. Commonly used clustering algorithms include Spectral Clustering (SC) [1] and Agglomerative Hierarchical Clustering (AHC) [2].

Despite recent advancements in speaker diarization [3], several factors make solving SD task difficult:

- *Uniform speaker segmentation*: Long segments very likely contain speaker turn boundaries, while short segments carry insufficient speaker information.

- *Unknown number of speakers*: In general, both the identity of the speakers and the number of speakers are unknown beforehand.

- *Speaker talk time*: A speaker needs to talk long enough to be accurately detected.

- *Overlap speech*: Talking over each other or interrupting.

- *Background noise, room acoustics*: Environmental sounds and room conditions can interfere with speaker recognition.

- *Consisting of multiple steps*: The SD system involves several steps, each of which introducing some level of error.

Speaker change detection systems have been proposed to mitigate the uniform segmentation issue [4, 5]. These systems involve a dedicated model trained to detect the exact moment when speakers change. To deal with a trade-off between long and short segment lengths, a group of works employs multi-scale segmentation [6, 7]. They use multiple scales (segment lengths) and fuse the similarity scores between embeddings obtained from the results of each scale.

To address the overlap speech problem, the recently introduced target-speaker voice activity detection (TS-VAD) model [8] has attracted much interest due to its great success in challenging tasks such as VoxSRC [9, 10, 11] and DIHARD-III [12]. Based on the speaker profiles obtained from a clustering-based diarization, the TS-VAD system can estimate each speaker's frame-level voice activities to refine the initial clustering-based results.

A line of research aims to improve the performance of conventional clustering-based methods by enhancing either through methods like embedding refinement [13, 14] or by refining similarity scores among speaker embeddings [1]. In [15], the Teacher-Student approach was employed to increase the robustness of the speaker embedding extractor against different acoustic conditions.

An alternative line of research ([16, 17, 18]) tackles the segmentation and clustering modules jointly. These models are referred to as ”end-to-end”. End-to-end algorithms have demonstrated their effectiveness over traditional modular systems in controlled situations with a limited number of speakers. However, their performance suffers in real-world recordings with a larger number of speakers.

In this paper, we describe our clustering-based SD system[1] for the Diarization Task of the 2023 VoxCeleb Speaker Recognition Challenge (VoxSRC23)[2]. The proposed system consists of several sub-modules, such as voice activity detection, speaker embedding extraction, clustering, and overlap speech detection (OSD). Along with the description, we will outline how to build a strong speaker diarization system and give a detailed analysis of each method.

2. About the VoxSRC 2023 Challenge

The goal of the VoxSRC challenge is to probe how well current methods can recognize speakers from speech obtained 'in the wild'. The Voxconverse dataset [19] was used for the speaker diarization task. The VoxConverse dataset contains 74 hours of human conversation

[1] http:
//mm.kaist.ac.kr/datasets/voxceleb/voxsrc/data_workshop_2023/reports/krisp_report.pdf

[2] http://mm.kaist.ac.kr/datasets/voxceleb/voxsrc/interspeech2023.html

extracted from YouTube videos. The dataset is divided into a development set (20.3h, 216 recordings) and a test set (53.5h, 232 recordings). The number of speakers in each recording has a wide range of variety from 1 speaker to 21 speakers. The audio comprises a variety of noises, such as background music, laughter, and so on. It also contains a significant portion of overlapping speech from 0% to 30.1% depending on the recording. The primary metric for this task is the Diarization Error Rate (DER), which is the sum of three terms: false alarm (FA, incorrectly marking non-speech as speech), missed detection (MS, incorrectly marking speech as non-speech) and speaker confusion error rate (CER, assigning the wrong speaker ID within a speech region). A separate evaluation dataset (VoxSRC-23 Test) was used to establish the rankings on the leaderboard.

3. System Configuration

3.1 Voice Activity Detection

Voice Activity Detection is the process of identifying speech segments within an audio signal, serving as an essential initial phase for speaker diarization. We employ four different VAD models, each designed to capture various facets of the task.

3.1.1 GRU-Based VAD

We use a stack of 4 Gated Recurrent Unit (GRU) layers, incorporating layer normalization between each layer. The final dense layer with sigmoid activation is responsible for calculating the likelihood of speech occurrence. With this setup, we generate a probability score for every 30ms of speech. Values nearing 1, signify the presence of speech, whereas values closer to 0 suggest its absence. We use the Voxconverse dev set for training and the Voxconverse test set for validation.

3.1.2 NC-Based VAD

We adopt the Noise Cancellation (NC) model [20] to detect voice activity. First, we apply the NC model to remove any noise and non-speech signals from the original audio. Subsequently, for each 50ms interval, we calculate the energy of that interval and establish a threshold. If the energy level exceeds the threshold, we label the segment as speech; otherwise, it is categorized as non-speech. Additionally, we apply simple post-processing steps to obtain homogeneous speech activity segments. The architecture of the NC model is the same as the GRU-VAD architecture, with the exception that it generates a mask. This mask is subsequently applied to the input spectrogram and transformed into a waveform using the Inverse Fourier Transform.

3.1.3 ASR-Based VAD

Another approach to detecting voice activity segments involves making use of an Automatic Speech Recognition (ASR) model to generate timestamps at the level of individual words. We derive word-level timestamps by employing the Conformer-Medium checkpoint available in the NeMo[3] package. Similar to NC-based VAD, here we also apply post-processing steps to obtain homogeneous speech segments.

[3] `https://github.com/NVIDIA/NeMo`

3.1.4 Pyannote VAD

We also provide evaluation results for an open-source VAD model available in *pyannote* package [21]. Specifically, we employ the *pyannote.audio 2.1*[4] segmentation pipeline for computing the voice activity regions.

Table 1. Detection Error Rate of the VAD model on Voxconverse test set.

#Model	FA	MISS	Detection Error
GRU-based	2.59%	1.40%	3.99%
NC-based	2.83%	2.09%	4.92%
ASR-based	3.04%	1.74%	4.79%
Pyannote	2.01%	1.19%	3.20%
Fusion	2.02%	0.82%	2.84%

Table 1 shows that *NC-based* and *ASR-based* VAD models have inferior performance compared to systems trained under direct supervision. However, when we fuse these models using a majority vote, we achieve a reduction in detection error rate by 0.36%.

3.2 Speaker Embedding Extraction

Speaker embeddings are fixed-size vector representations from a speech signal that exclusively capture unique characteristics of the speaker's identity. Speaker embeddings are commonly used to classify and discriminate between different speakers.

A few publicly available speaker embedding models listed in Table 2 were compared with the corresponding performance results and the corresponding training datasets. Performance results are reported in equal error rates (EER), which is a standard metric used to evaluate speaker verification.

Table 2. Equal Error Rate values for different embedding extraction models evaluated on the Voxceleb test benchmark.

Embedding	EER	Training Datasets
TitaNet-Large[22]	0.68% Vox1-Clean	Voxceleb1+Voxceleb2, Fisher, Switchboard, Librispeech
TitaNet-Small[22]	1.08% Vox1-Clean	Voxceleb1+Voxceleb2, Fisher, Switchboard, Librispeech
RawNet3[23]	0.89% Vox1-O	Voxceleb1+Voxceleb2
ECAPA-TDNN[24]	0.80% Vox1-Clean	Voxceleb1+Voxceleb2

To increase the accuracy of speaker recognition and speaker diarization for noisy audios, we finetune TitaNet-Small with the Teacher-Student method [15] by adding L_2-regularization term to the AAM loss [25], between embeddings for augmented and non-augmented versions of the same audio utterance. We follow the fine-tuning steps presented in [15]. For

[4] `https://huggingface.co/pyannote/segmentation`

fine-tuning, we use the VoxCeleb1 [26] and VoxCeleb2 [27] datasets. By employing this approach, we achieved EER comparable to the pre-trained TitaNet-Small[5] model under normal conditions. However, the technique demonstrated superior performance in noisy conditions.

3.3 Clustering

Once computed, the speaker embeddings are grouped into clusters. We use two different clustering algorithms for SD. One method relies on spectral clustering and the other is based on agglomerative hierarchical clustering.

3.3.1 Spectral Clustering

Our SC-based diarization is similar to [15]. We perform multi-scale segmentation [7] and extract embeddings with different window and shift sizes. The affinity matrices are constructed using the cosine similarity between segment embeddings and are then fused into a single matrix (see Fig. 1). We further apply the following sequence of refinement operations on the affinity matrix A (see Fig. 2):

- *Row-wise Thresholding*: For each row, keep the *top-p* largest elements and set the rest to 0
- *Symmetrization*: $Y = \frac{1}{2}(A + A^T)$
- *Diffusion*: $Y = AA^T$

Afterwards, we apply the spectral clustering algorithm to obtain speaker IDs. The number of speakers is determined using the maximal eigen-gap approach [1].

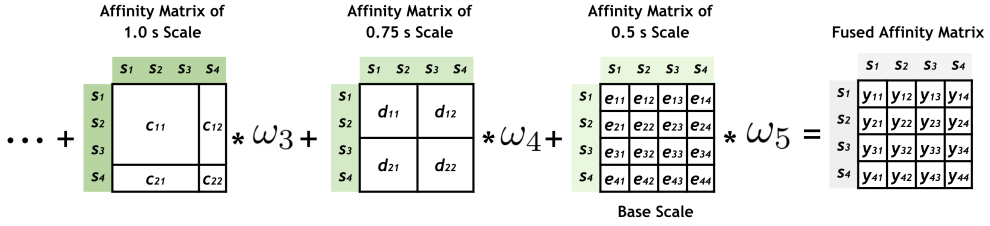
Affinity Matrix of 1.0 s Scale
Affinity Matrix of 0.75 s Scale
Affinity Matrix of 0.5 s Scale
Fused Affinity Matrix
Base Scale
$\cdots + \; * \omega_3 + \; * \omega_4 + \; * \omega_5 =$
S1 S2 S3 S4
c11 c12 c21 c22
d11 d12 d21 d22
e11 e12 e13 e14 e21 e22 e23 e24 e31 e32 e33 e34 e41 e42 e43 e44
y11 y12 y13 y14 y21 y22 y23 y24 y31 y32 y33 y34 y41 y42 y43 y44

Fig. 1. Multi-scale segmentation scheme.

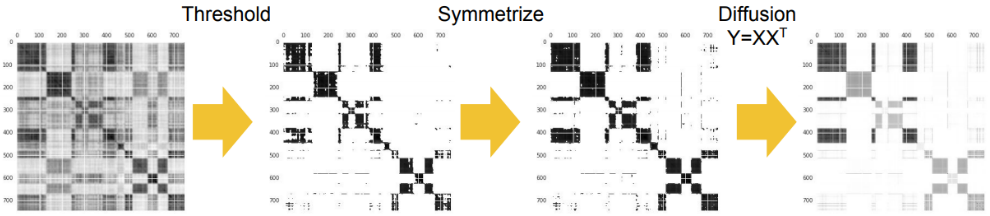

Fig. 2. Refinement operations on the affinity matrix.

[5] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/titanet_small

3.3.2 Agglomerative Hierarchical Clustering

First, we extract speaker embeddings from uniformly segmented speech regions. Then, we refine these embeddings through spectral dimensionality reduction[6] and affinity aggregation (AA) techniques [14]. We merged consecutive segments into a longer one if the distance was greater than the *segment threshold*. Afterwards, we perform a plain agglomerative clustering on the refined embeddings with a relatively high *stop threshold* to obtain the clusters with high confidence. The clusters from AHC were further processed using the short-duration filter [2, 10]. We categorize a cluster as "short" if the combined duration of that cluster is below the specified *duration threshold*. Later, each short cluster is assigned to the nearest long cluster based on the cosine distance of their central embeddings. Finally, if a short cluster significantly differs from all long clusters, which means that the distance between them is lower than a *speaker threshold*, we consider it as a new speaker.

3.4 Overlap Speech Detection

To detect regions where two or more speakers are speaking simultaneously, we use *pyannote overlap speech detection* pipeline[7]. After an overlapped region is detected, we replace the label with the two closest speakers near this region in the time domain.

3.5 Fusion

To improve the diarization accuracy, a series of studies were conducted on the fusion method of multiple diarization results. More recently, the diarization output voting error reduction (DOVER) method [28] was proposed to combine multiple diarization outputs based on the voting scheme. The DOVER method has an implicit assumption that there is no overlapping speech, i.e., at most only one speaker is assigned for each time index. To accommodate diarization outputs with overlapping speakers, the DOVER-LAP [29] method was subsequently introduced.

We combine different diarization systems using the DOVER-Lap[8] fusion method with the Hungarian label mapping algorithm.

4. Experimental Results

Table 3 shows the results on the voxconverse test set and the challenge evaluation test set. The first row of the table displays the baseline result (*VGG baseline*), provided by the challenge organizers. We start with the pyannote VoxSRC22 pipeline (#1) as our initial system and enhance it by applying the affinity aggregation technique (#2) to refine the embeddings. This adjustment results in a reduction of 0.59% in DER on the voxconverse test set.

Next, we designed several diarization systems based on spectral clustering with different embedding extractors (#3 – #9). These systems all rely on uniform speaker segmentation, which leads to speaker errors, mainly around the speaker turns. To mitigate this issue, we use

[6] `https://scikit-learn.org/stable/modules/generated/sklearn.manifold.SpectralEmbedding.html`

[7] `https://huggingface.co/pyannote/overlapped-speech-detection`

[8] `https://github.com/desh2608/dover-lap`

different segmentation setups by changing both the window size and the shift size. Multi-scale segmentation (#5, #8) is also designed to tackle this problem and to remove noisy entries from the affinity matrix. Furthermore, to make the systems more robust, we apply a sequence of refinement operations on the affinity matrix. In single-scale segmented setups, we establish the *top-p* value for row-wise thresholding as 8. In the case of multi-scale segmented setups, this value is adjusted to 30. As one can see from Table 3, multi-scale segmented systems outperform single-scale ones by a margin of 0.3%. Surprisingly, system #9, which was finetuned with the Teacher-Student technique, achieves a similar score (5.23%) on the voxconverse test set without using multi-scale segmentation.

As noted in [10], SC-based and AHC-based clustering methods complement each other. Through our experiments, we also observed similar behaviour. Spectral clustering provides a more precise estimation of the number of speakers, whereas AHC-based clustering tends to consistently overestimate it. Conversely, AHC-based clustering excels at identifying the dominant speakers and demonstrates superior performance on shorter audio files compared to spectral clustering. We conduct a hyperparameter search for AHC-based systems (#10, #11, #12) on the voxconverse test subset to determine the optimal values for *segment threshold*, *stop threshold*, *duration threshold*, and *speaker threshold*. As it is illustrated in Table 3, AHC-based systems show slightly worse DER scores (5.32%-5.41%) compared to SC-based systems.

Table 3. The performance of different speaker diarization systems.

N	System	Window [s]	Shift [s]	Voxconverse Test	VoxSRC-23 Test
				DER[%]	DER[%]
	VGG baseline	-	-	-	8.68
#1	Pyannote VoxSRC22	-	-	5.89	7.33
#2	Pyannote VoxSRC22+AA	-	-	5.30	-
#3	TitaNet-Large-SC	1.0	0.75	6.00	-
#4	TitaNet-Large-SC	2.0	1.0	5.59	-
#5	TitaNet-Large-SC	[2.0, 1.5, 0.75]	[1, 0.5, 0.25]	5.25	-
#6	ECAPA-TDNN-SC	1.0	0.75	6.05	-
#7	ECAPA-TDNN-SC	2.0	1.0	5.71	-
#8	ECAPA-TDNN-SC	[2, 1.5, 0.75]	[1, 0.5, 0.25]	5.38	-
#9	TitaNet-Small-SC	1.5	0.5	5.23	-
#10	TitaNet-Large-AHC	1.5	0.5	5.41	-
#11	ECAPA-TDNN-AHC	1.5	0.5	5.38	-
#12	RawNet3-AHC	1.5	0.75	5.32	-
	Fusion(3+4+5+6+7+8)+OSD	-	-	4.80	6.35
	Fusion(2+3+4+5+6+7+8)+OSD	-	-	4.76	5.98
	Fusion(2+5+8+9+10+11+12)+OSD	-	-	4.39	4.71

Our best system combines 7 different systems fused by DOVER-Lap. Among them, 3 systems are based on spectral clustering, while 4 systems are based on AHC (including pyannote system #2). We first fused the systems and then dealt with the overlap because fusing with overlapping labels did not demonstrate any improvement on the voxconverse test set. This fused system achieves 4.39% DER on the voxconverse test set and 4.71% DER on the challenge evaluation set, which ranks 2nd place in the VoxSRC 2023 challenge.

5. Discussions

Throughout our experiments, we observed that better performance on widely adopted speaker verification evaluation protocols does not lead to better diarization performance. Additionally, the embedding extractors did not encounter situations where multiple speakers were present in audio utterances. Such scenarios are unavoidable in speaker diarization due to factors like overlapping speech and speaker transitions.

In contrast to speaker verification, which uses speaker embeddings to represent an endless number of speakers, speaker diarization only uses embeddings to represent a small number of speakers in a single session. For instance, only a small part of the information included in the embeddings will be used to distinguish between a small number of speakers, even if high-dimensional embeddings are extracted.

Another drawback of conventional clustering-based SD systems is that they do not take into consideration embedding ordering. Conversations involving multiple speakers are highly structured, and turn-taking behaviours are not dispersed randomly throughout time.

In our future work, we plan to investigate speaker verification evaluation protocols that better simulate the diarization scenario. Additionally, we will explore techniques aimed at adapting and contextualizing speaker embeddings for the speaker diarization task, as well as exploring approaches to leverage ordering information of embeddings.

6. Conclusions

In this paper, we described our submitted SD system for the diarization task of the 2023 VoxSRC challenge. We mainly focused on reducing speaker confusion errors. To achieve this goal, we used various methods, such as multi-scale segmentation, affinity refinement operations, and teacher-student techniques to make our SD systems robust to background noise and errors that might arise from uniform speech segmentation. Our final system yielded notable results, reaching a DER of 4.39% on the voxconverse test set and 4.71% on the challenge evaluation set.

References

[1] Q. Wang, C. Downey, L. Wan, P. Mansfield and I.Moreno, “Speaker diarization with LSTM”, *2018 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 5239-5243, 2018.

[2] X. Xiao, N. Kanda, Z. Chen, T. Zhou, T. Yoshioka, S. Chen, Y. Zhao, G. Liu, Y. Wu, J. Wu and et.a, “Microsoft speaker diarization system for the voxceleb speaker recognition challenge 2020”, *IEEE International Conference on Acoustics, Speech And Signal Processing (ICASSP)*, pp. 5824-5828, 2021.

[3] T. Park, N. Kanda, D. Dimitriadis, K. Han, S. Watanabe and S. Narayanan, “A review of speaker diarization: Recent advances with deep learning”, *Computer Speech & Language*, vol. 72, pp. 101317, 2022.

[4] R. Yin, H. Bredin and C. Barras, “Speaker change detection in broadcast TV using bidirectional long short-term memory networks”, *Interspeech 2017*, 2017.

[5] W. Xia, H. Lu, Q. Wang, A. Tripathi, Y. Huang, I. Moreno and H. Sak, "Turn-to-diarize: Online speaker diarization constrained by transformer transducer speaker turn detection", *IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 8077-8081, 2022.

[6] T. Park, M. Kumar and S. Narayanan, "Multi-scale speaker diarization with neural affinity score fusion", *IEEE International Conference on Acoustics, Speech And Signal Processing (ICASSP)*, pp. 7173-7177, 2021.

[7] Y. Kwon, H. Heo, J. Jung, Y. Kim, B. Lee and J. Chung, "Multi-scale speaker embedding-based graph attention networks for speaker diarisation", *IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 8367-8371, 2022.

[8] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny , A. Laptev and A. Pomanenko, "Target-Speaker Voice Activity Detection: a Novel Approach for Multi-Speaker Diarization in a Dinner Party Scenario", arXiv:2005.07272, 2020.

[9] W. Wang, D. Cai, Q. Lin, L. Yang, J. Wang, J. Wang and M. Li, "The dku-dukeece-lenovo system for the diarization task of the 2021 voxceleb speaker recognition challenge", arXiv:2109.02002, 2021.

[10] W. Wang, X. Qin, M. Cheng, Y. Zhang, K. Wang, and M. Li, "The dku-dukeece diarization system for the voxceleb speaker recognition challenge 2022', arXiv:2210.01677, 2022.

[11] M. Cheng, W. Wang, Y. Zhang, X. Qin and M. Li, "Target-speaker voice activity detection via sequence-to-sequence prediction" *IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 1-5, 2023.

[12] Y. Wang, M. He, S. Niu, L. Sun, T. Gao, X. Fang, J. Pan, J. Du and C. Lee, "USTC-NELSLIP system description for DIHARD-III challenge", arXiv:2103.10661, 2021.

[13] J. Wang, X. Xiao, J. Wu, R. Ramamurthy, F. Rudzicz and M. Brudno, "Speaker diarization with session-level speaker embedding refinement using graph neural networks", *IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 7109-7113, 2020.

[14] Y. Kwon, J. Jung, H. Heo, Y. Kim, B. Lee and J. Chung, "Adapting speaker embeddings for speaker diarisation", arXiv:2104.02879, 2021.

[15] D. Karamyan, G. Kirakosyan and S. Harutyunyan, "Making speaker diarization system noise tolerant", *Mathematical Problems of Computer Science.* vol.59, pp. 57-68, 2023.

[16] A. Zhang, Q. Wang, Z. Zhu, J. Paisley and C. Wang, "Fully supervised speaker diarization", *Proceedings of IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pp. 6301-6305, 2019.

[17] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives", *Proc. Interspeech*, pp. 4300-4304, 2019.

[18] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu and S. Watanabe, "End-to-end neural speaker diarization with self-attention", *Proceedings of IEEE Automatic Speech Recognition And Understanding Workshop (ASRU)*, pp. 296-303, 2019.

[19] J. Chung, J. Huh, A. Nagrani, T. Afouras and A. Zisserman, “Spot the conversation: Speaker diarisation in the wild”, *Proc. Interspeech 2020*, pp. 299-303, 2020.

[20] Y. Xu, J. Du, L. Dai, and C. Lee, “A regression approach to speech enhancement based on deep neural networks”, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 7-19, 2014.

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

[22] N. Koluguri, T. Park and B. Ginsburg, “TitaNet: Neural model for speaker representation with 1D Depth-wise separable convolutions and global context”, *Proceedings of IEEE International Conference on Acoustics, Speech And Signal Processing (ICASSP)*, pp. 8102-8106, 2022.

[23] J. Jung, Y. Kim, H. Heo, B. Lee, Y. Kwon and J. Chung, “Pushing the limits of raw waveform speaker recognition”, *23rd Annual Conference of the International Speech Communication Association, INTERSPEECH 2022*, pp. 2228-2232, 2022.

[24] B. Desplanques, J. Thienpondt and K. Demuynck, “ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification”, *Proc. Interspeech 2020*, pp. 3830-3834, 2020.

[25] J. Deng, J. Guo, N. Xue and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition”, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690-4699, 2019.

[26] A. Nagrani, J. Chung and A. Zisserman, “Voxceleb: a large-scale speaker identification dataset”, arXiv:1706.08612, 2017.

[27] J. Chung, A. Nagrani and A. Zisserman, “Voxceleb2: Deep speaker recognition”, arXiv :1806.05622, 2018.

[28] A. Stolcke and T. Yoshioka, “DOVER: A method for combining diarization outputs”, *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 757-763, 2019.

[29] D. Raj, L. Garcia-Perera, Z. Huang, S. Watanabe, D. Povey, A. Stolcke and S. Khudanpur, “Dover-lap: A method for combining overlap-aware diarization outputs”, *IEEE Spoken Language Technology Workshop (SLT)*, pp. 881-888, 2021.

Խոսնակների դիարիզացիայի համակարգի կառուցում. դասեր VoxSRC-ից

Դավիթ Ս. Քարամյան[1,2] և Գրիգոր Ա. Կիրակոսյան[2,3]

[1]Հայ-Ռուսական համալսարան, Երևան, Հայաստան
[2]Krisp.ai, Երևան, Հայաստան
[3]ՀՀ ԳԱԱ մաթեմատիկայի ինստիտուտ,Երևան, Հայաստան
e-mail: {dkaramyan, gkirakosyan }@krisp.ai

Ամփոփում

Խոսնակների դիարիզացիայի նպատակը աուդիո ձայնագրության մեջ տարբեր խոսնակների հայտնաբերելն ու առանձնացնելն է: Այս հոդվածում ներկայացված է դիարիզացման հուսալի համակարգ, ինչպես նաև նկարագրված են այդ համակարգի կառուցվածքն ու հիմնական բաղադրիչները, ինչպիսիք են ձայնի հայտնաբերումը, խոսնակների ձայնային հատկանիշներ դուրս բերող մոդելը և կլաստերացումը, որոնք անհրաժեշտ են դիարիզացման հուսալի համակարգ ստեղծելու համար: Այս համակարգը հաղթող է ճանաչվել Voxceleb Speaker Recognition Challenge (VoxSRC) 2023 մրցույթում, որը լայնորեն ճանաչված է խոսնակների դիարիզացման համակարգերի գնահատման մրցույթում:

Բանալի բառեր՝ Խոսնակների նույնականացում, խոսնակների դիարիզացիա, VoxSRC2023:

Построение системы диаризации дикторов: опыт из VoxSRC 2023

Давид С. Карамян[1,2] и Григор А. Киракосян[2,3]

[1]Российско-Армянский университет, Ереван, Армения
[2]Krisp.ai, Ереван, Армения
[3]Институт математики НАН РА, Ереван, Армения
e-mail: {dkaramyan, sharutyunyan, gkirakosyan}@krisp.ai

Аннотация

Диаризация дикторов - это процесс разделения аудиозаписи на сегменты, которые соответствуют отдельным дикторам. В этой статье представлена надежная система диаризации говорящих и описана архитектура данной системы. Сосредоточено внимание на обсуждении ключевых компонентов, таких как обнаружение речевой активности экстрактор речевых характеристик и кластеризация, которые необходимы для создания надежной системы диаризации. Данная система стала победителем конкурса Voxceleb Speaker Recognition Challenge (VoxSRC) 2023, широко признанного конкурса по оценке систем диаризации дикторов.

Ключевые слова: Распознавание по голосу, диаризация дикторов, VoxSRC 2023.