INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

ECAPA-TDNN Embeddings for Speaker Diarization

Nauman Dawalatabad[1,2]*, Mirco Ravanelli*[2]*, François Grondin*[3]*,*
Jenthe Thienpondt[4]*, Brecht Desplanques*[4]*, Hwidong Na*[5]

[1]Indian Institute of Technology Madras, India
[2]Mila - Quebec Artificial Intelligence Institute, Canada
[3]Université de Sherbrooke, Canada
[4]IDLab, Ghent University - imec, Belgium
[5]Samsung Advanced Institute of Technology, Suwon, South Korea

nauman@cse.iitm.ac.in, mirco.ravanelli@gmail.com

Abstract

Learning robust speaker embeddings is a crucial step in speaker diarization. Deep neural networks can accurately capture speaker discriminative characteristics and popular deep embeddings such as x-vectors are nowadays a fundamental component of modern diarization systems. Recently, some improvements over the standard TDNN architecture used for x-vectors have been proposed. The ECAPA-TDNN model, for instance, has shown impressive performance in the speaker verification domain, thanks to a carefully designed neural model.

In this work, we extend, for the first time, the use of the ECAPA-TDNN model to speaker diarization. Moreover, we improved its robustness with a powerful augmentation scheme that concatenates several contaminated versions of the same signal within the same training batch. The ECAPA-TDNN model turned out to provide robust speaker embeddings under both close-talking and distant-talking conditions. Our results on the popular AMI meeting corpus show that our system significantly outperforms recently proposed approaches.

Index Terms: speaker diarization, speaker embedding, data augmentation, spectral clustering.

1. Introduction

Speaker diarization answers the question of "*who spoke when?*" in a given conversation [1, 2]. Diarization is used in many conversational AI systems and applied in various domains such as telephone conversations, broadcast news, meetings, clinical recordings, and many more [2]. Modern diarization systems rely on neural speaker embeddings coupled with a clustering algorithm.

Despite the recent progress, speaker diarization is still one of the most challenging speech processing tasks [3]. Research in this field is very active, and it is fostered by popular challenges such as DIHARD [4]. As for clustering, various approaches have been proposed in the literature, including top-down and bottom-up agglomerative clustering [1]. Spectral clustering, which is a graph clustering method based on the eigenanalysis of the Laplacian matrix, has recently shown promising performance on speaker diarization [2, 5, 6].

Several research efforts have been devoted to neural speaker embeddings as well. Modern speaker embeddings such as d-vectors [7], c-vectors [8], and x-vectors [9] have shown to capture speaker discriminative characteristics very well. The x-vector model, for instance, is based on a Time Delay Neural Network (TDNN) and is now a fundamental component of the state-of-the-art diarization systems [2].

An enhanced version of the standard TDNN model based on Emphasized Channel Attention, Propagation, and Aggregation (ECAPA-TDNN) [10] employs a channel- and context-dependent attention mechanism, Multilayer Feature Aggregation (MFA), as well as Squeeze-Excitation (SE) and residual blocks. This model has recently shown impressive performance in the speaker verification domain [10]. It has shown the best performance in the text-independent task of the Short-duration Speaker Verification (SdSV) challenge [11, 12]. This makes it a good choice for speaker diarization, as speaker turns in realistic conversations can be of short duration.

The following are the contributions to this work.
(i) *Model*: This is the first time that the ECAPA-TDNN model architecture is used in the context of speaker diarization.
(ii) *Augmentation*: We improve the robustness of ECAPA-TDNN speaker embeddings by training the model with an extensive on-the-fly augmentation scheme such that all the contaminated versions are concatenated within the same training batch. We propose to use combination of different augmentation techniques such as waveform dropout, frequency dropout, speed perturbation, reverberation, and additive noise.

We conducted our experimental studies using the popular AMI [13] meeting dataset considering different types of audio streams. The proposed system shows highly competitive performance and overtakes recent approaches in speaker diarization. To foster replicability, we made the code and the pre-trained models available in the SpeechBrain project[1].

2. ECAPA-TDNN Diarization

In this section, we describe the various modules involved in the proposed ECAPA-TDNN based speaker diarization system.

2.1. Speaker embeddings

Modern speaker embeddings are computed from neural models trained to classify speaker identities from a large pool of speakers [9, 10, 14]. A temporal statistics pooling layer is used to map the variable length input to a fixed-length representation. After training, the fixed-length speaker embeddings are extracted from the activations of the penultimate layer in the network.

As shown in Figure 1, the ECAPA-TDNN [10] model architecture is based on the popular x-vector topology [9] and it

This work is done when N. Dawalatabad was at Mila - Quebec Artificial Intelligence Institute, Canada, and H. Na was at SAIT AI Lab, Montreal.

[1]SpeechBrain: https://speechbrain.github.io/

http://dx.doi.org/10.21437/Interspeech.2021-941

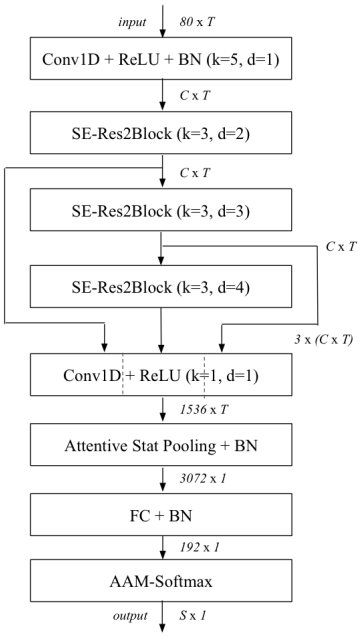

Figure 1: *Block diagram of the ECAPA-TDNN model [10]. The C, and T denotes the channel and the temporal dimension of the feature maps, respectively. k represents the kernel size and d denotes dilation spacing of the Conv1D layers or SE-Res2Blocks.*

introduces several enhancements to create more robust speaker embeddings. The pooling layer uses a channel- and context-dependent attention mechanism, which allows the network to attend different frames per channel. 1-dimensional Squeeze-Excitation (SE) [15] blocks rescale the channels of the intermediate frame-level feature maps to insert global context information in the locally operating convolutional blocks. Next, the integration of 1-dimensional Res2-blocks [16] improves performance while simultaneously reducing the total parameter count by using grouped convolutions in a hierarchical way. Finally, Multi-layer Feature Aggregation (MFA) [17] merges complementary information before the statistics pooling by concatenating the final frame-level feature map with intermediate feature maps of preceding layers.

The network is trained by optimizing the AAM-softmax [18] loss on the speaker identities in the training corpus. The AAM-softmax is a powerful enhancement compared to the regular softmax loss in the context of fine-grained classification and verification problems. It directly optimizes the cosine distance between the speaker embeddings. As a consequence, complex scoring backends such as Probabilistic Linear Discriminant Analysis (PLDA) [19] can be avoided.

2.2. Data augmentation

Data augmentation is a common approach to improve the robustness of a neural model. Speech can be contaminated in different ways. In this study, we train the ECAPA-TDNN model with the following augmentation strategies:

- *Waveform dropout:* replaces some random chunks of the original waveform with zeros [20].
- *Frequency dropout:* filters the original signal with random band-stop filters to add zeros in the frequency spectrum. [20].
- *Speed Perturbation:* resamples the audio signal to a sampling rate that is slightly different from the original one. With this simple trick, we can synthesize a speech signal that sounds a bit faster or slower than the original one. This is useful as the speaking rate may vary within and across speakers [21]. To avoid changing the speaker characteristics significantly, we restrict the speed perturbation to a maximum of $\pm 5\%$.
- *Reverberation:* introduces reverberation by convolving the signal with a randomly selected room impulse response.
- *Additive Noise:* adds a randomly selected noise sequence to the speech signal with a random signal-to-noise ratio.
- *Noise + Reverberation:* combines the noise and reverberation disturbances.

All of these augmentations are applied on-the-fly to every speech sentence processed by the neural network. This way, we generate a different contaminated data at every epoch. In standard augmentation pipelines, the signal is contaminated with one or more augmentation strategies and then used for training a neural network. Instead, in this work, we propose to concatenate the original speech signal with all the contaminated versions produced by the aforementioned contamination techniques [22]. This way, within each training batch, our model observes the same sentence corrupted in different ways. These six different "views" of the same signal force the gradient to point to a direction of the parameter space that is inherently robust against signal variations, thus proving an important regularization effect. As we will show in Section 4.4, the proposed augmentation scheme for the ECAPA-TDNN model outperforms the standard one.

2.3. Spectral clustering

Spectral clustering is a popular clustering approach for speaker diarization that has recently shown highly competitive performance compared to the traditional Agglomerative Hierarchical Clustering (AHC) with PLDA backend [5, 6].

There are multiple methods to perform spectral clustering [23]. We follow the unnormalized spectral clustering approach similar to [2, 5, 23]. The affinity matrix A is calculated using the cosine similarity metric. It is important to prune out the smaller values in A to focus more on prominent values in the matrix. Similar to [23], we use the actual similarity values in affinity matrix while calculating the Laplacian matrix. An unnormalized Laplacian matrix is estimated using the symmetrized A as done in [2,23]. The Laplacian matrix is subjected to eigendecomposition. We estimate the number of speakers k using the maximum eigengap approach [23]. Next, we compute the first k eigenvectors. The rows of the eigenvector matrix are k dimensional spectral embeddings corresponding to each analyzed speech segment. The estimated spectral embeddings that are expected to be more separable than the original speaker embeddings, are clustered using the standard k-means algorithm.

3. Experimental Setup

3.1. Datasets

The ECAPA-TDNN model is trained with VoxCeleb1 and VoxCeleb2 data [24, 25]. The RIRs[2] and MUSAN [26] datasets are used for data-augmentation purposes.

[2]RIRs: `https://www.openslr.org/28/`

For diarization, we use the Augmented Multi-party Interaction (AMI) meeting dataset [13]. We use the official "Full ASR corpus" split with TNO meetings excluded from the Dev and Eval set. Official manual annotations serve as the ground truth for evaluation. The channels in the microphone array are beamformed with the standard BeamformIt toolkit [27]. The same split is used in many other works [6, 8, 28–30].

3.2. Speaker embeddings

The ECAPA-TDNN model is fed with 80-dimensional log Mel filterbank energies that are mean normalized per input segment. The model parameter updates are determined by the Adam [31] optimizer with a Cyclical Learning Rate (CLR) [32] using a triangular policy. Training is done for 10 epochs with batches of 32 segments. The original batch is augmented in six different ways, leading to an equivalent batch size of 192. The contamination with MUSAN additive noise is done with a random Signal-to-Noise Ratio (SNR) ranging from 0 to 10 dB. Reverberation is added by convolving with a random impulse response from the aforementioned RIRs dataset.

We train the model with 3 sec random crops of the speaker utterances. The architectural hyper-parameters of the ECAPA-TDNN model are the same as in [10]. The model achieves a promising EER of 0.69% and a minDCF of 0.0826 on the original VoxCeleb1 (cleaned) verification set. Additional details can be found in the VoxCeleb recipe in SpeechBrain [33].

3.3. Diarization setup

The embeddings of each continuous speech segment are extracted with a sliding window of size 3 sec and a shift of 1.5 sec. The maximum number of estimated speakers is set to 10. The pruning threshold for the affinity matrix is determined on the AMI Dev set.

We use the standard Diarization Error Rate (DER) as evaluation metric [34]. The DER consists of a Speaker Error Rate (SER), False Alarm (FA), and Missed Speech (MS) component. The SER represents the errors introduced due to incorrect labeling of speaker segments. The FA and MS occur due to errors introduced by the Voice Activity Detection (VAD) system. Since this work focuses on improving the clustering module, similar to [6] we use oracle speech/non-speech labels from the ground truth. To enable direct comparison with [6], we do not use any realignment post-processing step. Similar to [6], a forgiveness collar of 0.25 sec is used and the speaker overlap regions are ignored during scoring (as also standard by NIST). We use the standard NIST evaluation tool available in SpeechBrain [33].

4. Results

4.1. Baseline systems

We compare our system with the recently proposed ClusterGAN [6], MCGAN [6] and Variational Bayes [28] based diarization techniques.

The approaches mentioned in [6] are sophisticated systems where ClusterGAN and MCGAN models are trained using x-vectors extracted from TDNN [9]. The embeddings obtained from the models are fused with x-vectors and then clustered using Spectral Clustering (SC) algorithm. The number of speakers is estimated with the Normalized Maximum Eigengap (NME-SC) technique [5]. The NME-SC algorithm is expensive in terms of runtime as it requires iterating through multiple values of the pruning threshold to find the best setting. Therefore, this work estimates the number of speakers with the standard maximum eigengap criterion proposed in [23]. We also compare our system with the Variational Bayes (VBx) approach [28] under the same experimental setup as used in [6]. VBx uses ResNet101-based x-vector embeddings [35, 36] that are clustered using Bayesian Hidden Markov Model (BHMM) [36–38].

All three approaches have shown competitive or state-of-the-art performance (to the best of our knowledge) on the AMI dataset, making them strong baselines for comparison.

Table 1: *Diarization Error Rates (DERs) on AMI dataset using the beamformed array signal on baseline and proposed systems – less is better.*

Embedding	Back-end	Oracle num of speakers		Estimated num of speakers	
		Dev	Eval	Dev	Eval
xvector+ClusterGAN	k-means [6]	6.62	6.46	9.57	8.63
xvector+MCGAN		5.64	5.48	6.47	8.76
xvector+ClusterGAN	SC [6]	3.93	3.60	6.21	**2.87**
xvector+MCGAN		5.49	4.23	5.02	4.92
xvector (ResNet101)	VBx [28]	-	-	4.27	4.58
Proposed Approach					
ECAPA-TDNN	k-means	3.03	3.69	4.65	5.10
	SC	**2.82**	**2.65**	**3.66**	3.01

4.2. Comparison with baseline systems

Table 1 compares the aforementioned baselines with the proposed ECAPA-TDNN diarization system. The DERs reported in the table are estimated on the beamformed audio in two scenarios, i.e., (i) when the number of speakers is known before diarization (oracle number of speakers), and (ii) when the number of speakers is not known apriori and has to be automatically estimated. We estimate the number of speakers with maximum eigengap criterion for both, k-means and Spectral Clustering (SC) backends.

From Table 1 it emerges that the proposed approach significantly outperforms the baselines in most of the cases. There is a significant improvement compared to the baseline systems with SC as a backend. The only exception is the performance of x-vector+ClusterGAN embeddings with an unknown number of speakers, for which the DER of 2.87% on the Eval set is comparable to the DER of 3.01% achieved by our system. In all the other cases, the proposed system significantly outperforms the x-vector+ClusterGAN based system. Our system also outperforms x-vector+MCGAN based systems in all the cases. With an SC backend, relative improvements of 37.4% (from 4.23% to 2.65%) and 38.8% (from 4.92% to 3.01%) are observed on the Eval set for oracle and estimated number of speakers, respectively. This improvement is 34.3% (from 4.58% to 3.01%) with respect to the VBx system for an unknown number of speakers.

Interestingly, the ECAPA-TDNN diarization system achieves a noteworthy performance also with a simple k-mean clustering backend. For instance, comparing with the x-vector+ClusterGAN system, our system shows improvement of 42.9% (from 6.46% to 3.69%) and 40.9% (from 8.63% to 5.10%) on Eval set for oracle and estimated number of speakers, respectively. Compared to the x-vector+MCGAN based system, our system show improvements of 32.7% (from 5.48% to 3.69%) and 41.8% (from 8.76% to 5.10%) on the Eval set

Table 2: *DERs comparison between x-vectors and ECAPA-TDNN with SC backend on beamformed data – less is better.*

Embedding	Oracle num of speakers		Estimated num of speakers	
	Dev	Eval	Dev	Eval
x-vector	6.14	8.57	9.21	11.04
ECAPA-TDNN	**2.82**	**2.65**	**3.66**	**3.01**

Table 3: *DERs achieved when ECAPA-TDNN is trained using different augmentation techniques. Diarization is performed with SC backend on beamformed data – less is better.*

Augmentation	Oracle num of speakers		Estimated num of speakers	
	Dev	Eval	Dev	Eval
Without Aug.	3.95	3.92	5.72	7.45
Standard Aug.	3.04	**2.64**	4.48	4.51
Proposed Aug.	**2.82**	**2.65**	**3.66**	**3.01**

for oracle and estimated number of speakers, respectively. This further confirms the robustness of the proposed embeddings.

4.3. Comparison with x-vector embedding

Table 2 shows the DERs obtained using standard x-vector embeddings on beamformed data with SC as backend. The x-vector embeddings used here are trained with the same augmentation scheme described in Section 2.2.

The diarization performance with ECAPA-TDNN embeddings is far superior to that achieved by standard x-vector embeddings. This is due to various improvements introduced by the ECAPA-TDNN model. The same trend is observed on the other audio streams (individual distant mics, HeadsetMix, LapelMix).

4.4. Ablation study on augmentation

To confirm the effectiveness of the proposed augmentation approach, we conducted experiments by training the ECAPA-TDNN model with no data augmentation and with standard data augmentation techniques. The standard augmentation is performed by applying the aforementioned contamination methods to each training sentence (including a clean version of the signal). The proposed augmentation, instead, uses the batch construction technique described in Section 2.2.

It can be seen from Table 3 that the performance achieved with the proposed augmentation technique is better than that of a standard augmentation. The same trend is observed on all other audio streams. The DER is even worse when no data augmentation is used. The special batch construction technique adopted in the proposed augmentation scheme clearly helps to improve the robustness of the diarization system.

4.5. Distant and close talking microphones

Table 4 reports the performance of the proposed system achieved with the different microphone settings, including distant microphones, HeadsetMix, and LapelMix audio streams. The DERs obtained on Headsetmix and LapelMix audio streams are reported to facilitate the comparison with other works. The signals in the HeadsetMix and LapelMix streams are relatively clean, and hence the results on these streams give an estimate of the lowest DERs that can be achieved by the proposed system. In the case of the oracle number of speakers, the best DERs achieved on the HeadsetMix signal are 2.02% and 1.78% for Dev set and Eval set, respectively. For the LapelMix audio, the best DERs of 2.34% and 2.57% on the Dev and the Eval set are observed for an unknown number of speakers.

Table 4: *DERs achieved by the proposed system on distant and close talking audio streams with SC backend – less is better.*

Audio Streams	Oracle num of speakers		Estimated num of speakers	
	Dev	Eval	Dev	Eval
HeadsetMix	2.02	1.78	2.43	4.03
LapelMix	2.17	2.36	2.34	2.57
Distant-Mic (avg.)	2.81	3.12	3.33	3.75
Beamformed	2.82	2.65	3.66	3.01

The row *Distant-Mic* in Table 4 reports the average DER over the eight distant microphones of the AMI microphone array. It is interesting to note that the average DERs on the distant mics is comparable to the performance obtained on the beamformed audio. It is also worth noticing that the performance degradation observed when switching from close-talking microphones (e.g., HeadsetMix and LapelMix) to distant-talking ones (e.g., beamformed or distant microphone) is not huge. Similar behavior was also observed with the k-means backend. These results can be attributed to the data augmentation scheme. With the proposed augmentation approach, we indeed train the neural network with different environmental contamination effects (noise, reverberation, and noise + reverberation) and we thus implicitly achieve robustness in different acoustic conditions.

5. Conclusion and Future Work

In this paper, we improved speaker diarization performance on AMI meeting data by focusing on two critical components of the speaker diarization pipeline. (i) A better data augmentation technique which includes waveform dropout, frequency dropout, speed perturbation, reverberation, and adding noise disturbances. Crucially, these multiple augmented views of the signal are gathered in the same batch. (ii) The use of the ECAPA-TDNN model for the extraction of more robust speaker embeddings. The proposed approach has surpassed the performance of the most recent techniques such as ClusterGAN, MCGAN, and VBx. Moreover, the diarization performance remains consistent on speech recorded with distant or close-talking microphones.

Further improvements can include the use of a re-segmentation procedure, the adoption of VBx Bayesian HMM, the exploration of iterative clustering approaches with embeddings from longer segments in a second pass, and re-visiting automated ways to estimate the number of speakers in a recording such as NME-SC.

6. Acknowledgements

We would like to thank Yoshua Bengio, Samuele Cornell, and the rest of the SpeechBrain team for the helpful suggestions.

7. References

[1] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[2] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," 2021, arXiv:2101.09624.

[3] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, and S. Khudanpur, "Diarization is hard: Some experiences and lessons learned for the JHU team in the inaugural DIHARD challenge," in *Proc. Interspeech*, 2018, pp. 2808–2812.

[4] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "The second DIHARD diarization challenge: Dataset, task, and baselines," in *Proc. Interspeech*, 2019, pp. 978–982.

[5] 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.

[6] M. Pal, M. Kumar, R. Peri, T. J. Park, S. H. Kim, C. Lord, S. Bishop, and S. Narayanan, "Meta-learning with latent space clustering in generative adversarial network for speaker diarization," 2020, arXiv:2007.09635.

[7] L. Wan, Q. Wang, A. Papir, and I. Lopez-Moreno, "Generalized end-to-end loss for speaker verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4879–4883, 2018.

[8] G. Sun, C. Zhang, and P. C. Woodland, "Speaker diarisation using 2d self-attentive combination of embeddings," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5801–5805, 2019.

[9] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5329–5333, 2018.

[10] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.

[11] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "SdSV challenge 2020: Large-scale evaluation of short-duration speaker verification," in *Proc. Interspeech*, 2020, pp. 731–735.

[12] J. Thienpondt, B. Desplanques, and K. Demuynck, "Cross-lingual speaker verification with domain-balanced hard prototype mining and language-dependent score normalization," in *Proc. Interspeech*, 2020, pp. 756–760.

[13] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus: A pre-announcement," in *Proc. the Second International Conference on Machine Learning for Multimodal Interaction*, ser. MLMI'05, 28–39, 2006.

[14] D. Garcia-Romero, G. Sell, and A. McCree, "Magneto: X-vector magnitude estimation network plus offset for improved speaker recognition," in *Proc. Odyssey*, 2020, pp. 1–8.

[15] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *IEEE/CVF CVPR*, 2018, pp. 7132–7141.

[16] S. Gao, M.-M. Cheng, K. Zhao, X. Zhang, M.-H. Yang, and P. H. S. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE TPAMI*, pp. 652–662, 2019.

[17] Z. Gao, Y. Song, I. McLoughlin, P. Li, Y. Jiang, and L.-R. Dai, "Improving aggregation and loss function for better embedding learning in end-to-end speaker verification system," in *Proc. Interspeech*, 2019, pp. 361–365.

[18] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *IEEE/CVF CVPR*, 2019, pp. 4685–4694.

[19] S. Ioffe, "Probabilistic linear discriminant analysis," in *ECCV*, 2006, pp. 531–542.

[20] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[21] N. Dawalatabad, S. Madikeri, C. C. Sekhar, and H. A. Murthy, "Novel architectures for unsupervised information bottleneck based speaker diarization of meetings," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 14–27, 2021.

[22] E. Hoffer, T. Ben-Nun, I. Hubara, N. Giladi, T. Hoefler, and D. Soudry, "Augment your batch: better training with larger batches," *CoRR*, vol. abs/1901.09335, 2019.

[23] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, 2007.

[24] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, 2017, pp. 2616–2620.

[25] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[26] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015, arXiv:1510.08484v1.

[27] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2011–2022, 2007.

[28] 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," 2020, arXiv:2012.14952.

[29] Q. Li, F. L. Kreyssig, C. Zhang, and P. C. Woodland, "Discriminative neural clustering for speaker diarisation," 2020, arXiv:1910.09703.

[30] M. Pal, M. Kumar, R. Peri, T. J. Park, S. H. Kim, C. Lord, S. Bishop, and S. Narayanan, "Speaker diarization using latent space clustering in generative adversarial network," 2019, arXiv:1910.11398.

[31] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proc. ICLR*, 2014.

[32] L. N. Smith, "Cyclical learning rates for training neural networks," in *IEEE WACV*, 2017, pp. 464–472.

[33] 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.

[34] X. Anguera, "Diarization Error Rate," http://www.xavieranguera.com/phdthesis/node108.html, 2008, [Online; accessed 15-Jun-2021].

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[36] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "BUT system description to voxceleb speaker recognition challenge 2019," 2019, arXiv:1910.12592.

[37] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Černocký, "Bayesian HMM based x-vector clustering for speaker diarization," in *Proc. Interspeech*, 2019, pp. 346–350.

[38] A. Nagrani, J. S. Chung, J. Huh, A. Brown, E. Coto, W. Xie, M. McLaren, D. A. Reynolds, and A. Zisserman, "VoxSRC 2020: The second VoxCeleb speaker recognition challenge," 2020, arXiv:2012.06867.