STUDY ON THE TEMPORAL POOLING USED IN DEEP NEURAL NETWORKS FOR SPEAKER VERIFICATION

Mickael Rouvier
LIA - Avignon University
Avignon, France
mickael.rouvier@univ-avignon.fr

Pierre-Michel Bousquet
LIA - Avignon University
Avignon, France
pierre-michel.bousquet@univ-avignon.fr

Jarod Duret
LIA - Avignon University
Avignon, France
jarod.duret@alumni.univ-avignon.fr

***Abstract*—The x-vector architecture has recently achieved state-of-the-art results on the speaker verification task. This architecture incorporates a central layer, referred to as temporal pooling, which stacks statistical parameters of the acoustic frame distribution. This work proposes to highlight the significant effect of the temporal pooling content on the training dynamics and task performance. An evaluation with different pooling layers is conducted, that is, including different statistical measures of central tendency. Notably, 3rd and 4th moment-based statistics (skewness and kurtosis) are also tested to complete the usual mean and standard-deviation parameters. Our experiments show the influence of the pooling layer content in terms of speaker verification performance, but also for several classification tasks (speaker, channel or text related), and allow to better reveal the presence of external information to the speaker identity depending on the layer content.**

Index Terms: speaker verification, speaker embedding, pooling layer

I. Introduction

Speaker recognition refers to the task of verifying the identity claimed by a speaker from that person's voice [1]. For example, it has been shown useful for speaker diarization [2], forensics [3] or voice dubbing [4].

These last years, Deep Neural Networks (DNN) have allowed to emerge new voice representations, outperforming the state-of-the-art i-vector framework [5]. One of this DNN approach seeks to extract an embedding representation of a speaker directly from its acoustic excerpts. This high-level speaker representation is called x-vector [6]. The DNN models are trained through a speaker identification task, *i.e.* by classifying speech segments into one of n speaker identities. In that context, the different layers of the DNN are trained to extract information for discriminating between different speakers. The idea is to use one of the hidden layer as the speaker representation (the x-vector). One of the main advantage is that x-vectors produced by the DNN can generalize well to speakers beyond those present in the training set. The benefits of x-vectors, in terms of speaker detection accuracy, have been demonstrated during the recent evaluation campaigns: NIST SRE [7]–[9], VoxCeleb 2020 [10]–[12], SdSVC [13]–[15].

In the x-vector framework, the DNN uses a stack of convolution layers followed by a temporal pooling layer that computes the mean and standard deviation of an input sequence, in order to filter and capture the speaker characteristics throughout the recording. The temporal pooling is a very critical part of the DNN as it compacts the information along the full recording into a single vector representation. One of the main goals of temporal pooling is to capture only the salients part of the utterance in a compact representations, while removing irrelevant details. For this reason, the selection of a good temporal pooling in the model is important since it has a significant effect on the task performance.

Moreover, the authors in [16], [17] found that, in addition to the speaker-related information, the extracted x-vector contains meta-information such as session, speaking style, lexical content... The hypothesis in our study is that the types and proportions of such meta-information captured by the x-vector are greatly depending on the statistical parameters picked up to make up the pooling layer. As a consequence, well combining distinct DNN architectures, in terms of pooling content, could play a significant role in filtering out such unwanted information and, thus, better focusing the system on the goal of speaker discrimination.

In this study, we propose to evaluate the performance of various pooling contents, as well as their combination. In addition to traditional pooling, we propose to evaluate two new poolings : *skewness-pooling* and *kurtosis-pooling*. To further validate our hypothesis about pooling and information filtering, we experimentally evaluate information contained inside the x-vectors depending on various pooling contents, through numerous applications: speaker gender, speaker nationality, augmentation type, words recognition....

The papers is organized as follows: Section II summarizes the x-vector approach. Section III defines the different poolings used in our study. Section IV presents the classifiers and probing tasks, and classifiers for the probing tasks. In Section V, we analyze the results of the probing tasks and present results for our new x-vector based system on speaker verification. A conclusion is finally provided in Section VI.

TABLE I
THE PROPOSED RESNET34 ARCHITECTURE. N IN THE LAST ROW IS THE NUMBER OF SPEAKERS. BATCH-NORM AND RELU LAYERS ARE NOT SHOWN. THE DIMENSIONS ARE (FREQUENCYXTIMEXCHANNELS). THE INPUT COMPRISES 60 FILTER BANK FROM SPEECH SEGMENTS. DURING TRAINING WE USE A FIXED SEGMENT LENGTH OF 400.

Layer name	Structure	Output
Input	–	$60 \times 400 \times 1$
Conv2D-1	3×3, Stride 1	$60 \times 400 \times 128$
ResNetBlock-1	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 3$, Stride 1	$60 \times 400 \times 128$
ResNetBlock-2	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$, Stride 2	$30 \times 200 \times 128$
ResNetBlock-3	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 256 \end{bmatrix} \times 6$, Stride 2	$15 \times 100 \times 256$
ResNetBlock-4	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 3$, Stride 2	$8 \times 50 \times 256$
Pooling	–	8×256
Flatten	–	2048
Dense1	–	256
Dense2 (Softmax)	–	N
Total	–	–

II. x-VECTOR BASED ON RESNET

An x-vector is a high-level speaker features extracted from DNN models trained through a speaker identification task. The x-vector extractor proposed in this paper is a variant based on ResNet [18]. The detailed topology of the used ResNet is shown in Table II. The DNN model for extracting x-vectors consists of three modules: a *frame-level* feature extractor, a *statistics-level* layer, and *segment-level* representation layers.

- The *frame-level* component is based on the well-known ResNet34 topology. The component is composed of four residual blocks. This network uses 2-dimensional features as input and process them using 2-dimensional Convolutional Neural Networks (CNN) layers.
- The *statistics-level* component is an essential component that converts from a variable length speech signal into a single fixed-dimensional vector. The statistics-level is composed of one layer: the statistics-pooling, which aggregates over frame-level output vectors of the DNN and computes their mean and standard deviation.
- The *segment-level* component maps the segment-level vector to speaker identities. The mean and standard deviation are concatenated together and forward to additional hidden layers and finally to softmax output layer.

The DNN is trained using ArcFace softmax to classify speakers contained in the training set. The ResNet uses 60-dimensional Filter bank features as input, extracted from 25ms audio signal, mean-normalized over a sliding window of up to 3 seconds. Unvoiced frames are filtered out from the utterances using a Voice Activity Detection (VAD) based on signal energy.

In order to increase the diversity of the acoustic conditions in the training set, a data augmentation strategy is used, which adds four corrupted copies of the original recordings to the training list. The recordings are corrupted by adding noise, music and mixed speech (babble) drawn from the MUSAN database [19] and adding reverberation by using simulated Room Impulse Responses (RIR).

III. STUDIED POOLING STRATEGIES

The main goal of the pooling operation is to aggregate all the outputs given in the *frame-level* into a compact vector. The most commonly used pooling strategies are : *max-pooling*, *mean-pooling* and *standard-deviation-pooling*. We propose to evaluate two new pooling strategies : *skewness-pooling* and *kurtosis-pooling*.

- **max-pooling** : The max function is the most common choice for the pooling. This operation aggregates all vectors present in frame-level component and calculates the maximum, or largest, value. The max pooling is calculated as follows:

$$max = \max_{i=1}^{n} x_i \tag{1}$$

- **mean-pooling** : The mean computes the average of each vector present at the frame-level. The mean pooling is calculated as follows:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{2}$$

- **standard-deviation-pooling** : The standard-deviation is a measure of variance (*i.e.* dispersion) of a series. The standard-deviation pooling is calculated as follows:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2} \tag{3}$$

- **skewness-pooling** : Skewness provides information about the symmetry of a distribution around the mean. In the case of unimodal distribution, negative skew indicates that the main tail is on the left side of the distribution and positive skew indicates that it is on the right side. The skewness pooling is calculated as follows:

$$Skew = \frac{1}{n} \sum_{i=1}^{n} (\frac{x_i - \mu}{\sigma})^3 \tag{4}$$

- **kurtosis-pooling** : the kurtosis is a measure of flattening, providing information about how fat/heavy are the tails of a distribution (and hence, how peaked/flat around its mode), therefore about how frequent extreme deviations (or outliers) are from the average value. The kurtosis pooling is calculated as follows:

$$Kurt = \frac{1}{n} \sum_{i=1}^{n} (\frac{x_i - \mu}{\sigma})^4 \tag{5}$$

It is worth noting that there are two kinds of pooling. The max-pooling provides information on features (if the feature is present) and is a non-parametric measure, unlike the mean, standard-deviation, skewness and kurtosis poolings, which provide information on the statistical distribution. The

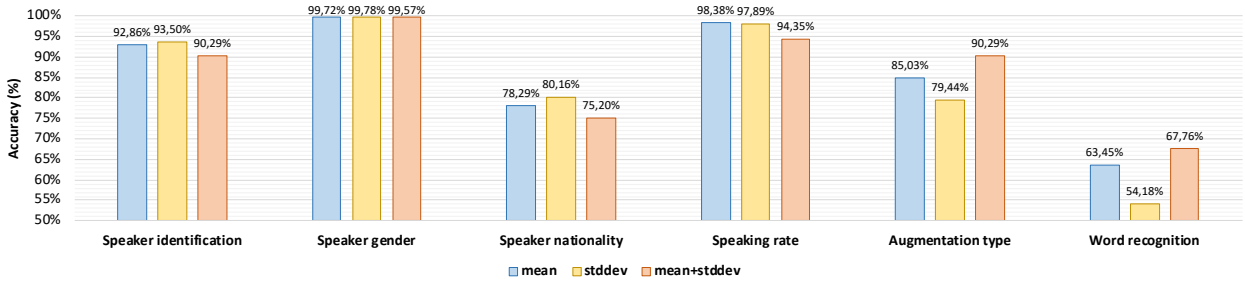

Fig. 1. Results obtained by using different pooling layer on different classification tasks.

latter are respectively coming from the first, second, third and fourth order moment, and are belonging to the set of values referred to as "measures of central tendency" in the field of statistics.

IV. Classification tasks on statistical pooling

To assess the assumption proposed in the introduction, we have to reveal the link between the statistical parameters selected to make up the x-vector and the patterns (*i.e.* the meta-information cited above) contained in the resulting x-vector.

To do that, the following classification tasks are carried out, each time with varying pooling configurations :

- **Speaker identification task** : this task measures to what extent the x-vectors encode the speaker's identity, which is crucial for the speaker recognition task. The evaluation set contains 106 different speakers and we report the recognition accuracy.
- **Speaker gender task** : this task measures whether the x-vectors can distinguish between gender (*i.e.* male or female). We train a two-classes classifier and report the classification accuracy.
- **Speaker nationality task** : this task measures whether the x-vectors can distinguish between speaker's nationality. The evaluation set contains 35 different nationalities and we report the classification accuracy.
- **Speaking rate task** : we augment all utterances by 3-way speed perturbation with rates of 0.9, 1.0 and 1.1. This task measures whether the x-vectors can capture information on speaking rate. We train a three-classes classifier and report the accuracy of recognition.
- **Augmentation type task** : this task measures whether the x-vectors can distinguish the type of data augmentation : noise, music, speech or no augmentation. We train a four-class classifier and provide the accuracy.
- **Word recognition task** : this task measures whether the x-vector can capture information about words in the utterance. We select the 25 most-frequent words and set up a classifier that predicts, for each word, whether the word is present or not. The average accuracy of correctly identified words is reported.

The ability of the x-vector to discriminate between these various tasks, among the different pooling strategies, will provide us information about the filtering-out property of the statistical measures, when used as components of a DNN-pooling layer. In other words, if a pattern of these tasks is present in the x-vector, we can train a classifier to recognize it and the performance of the classifier should depend on how well the pattern is embedded in the speaker representation. Let us note that the first four tasks presented above are about speaker-related information (speaker identification, speaker gender, speaker nationality, speaking rate) while the last two are about text- and channel-related information.

Our hypothesis is that some pooling operations could more easily provide information about features (if the feature is present). Even if the DNN models are trained for the speaker identification task, these kind of poolings would focus more on the biases present in the corpora in order to more easily identify the speakers (*i.e.* speakers who always speak in the same environment or microphone). On the other hand, some pooling strategies, due to their structure, would have more difficulties in indicating the presence or absence of a parameter and therefore would focus more on the speaker's parameters.

V. Experiments and results

This section describes the experimental setup in terms of dataset and evaluation protocol.

A. Experimental Protocol

Concerning the speaker verification task, the x-vector extractors are trained on the VoxCeleb2 dataset [20], only on the development partition, which contains speech excepts from 5,994 speakers with a 16 Khz sampling rate. The trained x-vectors are assessed on the Speakers in the Wild (SITW) core-core task [21] and Voxceleb1-E Cleaned [22] dataset with a 16 KHz sampling rate. Note that the development set of VoxCeleb2 is completely disjoint from the VoxCeleb1 dataset (*i.e.* no speaker in common).

Concerning the classification task, our objective is to evaluate the x-vector obtained by varying pooling configurations on the tasks presented in the previous section. As in [16], [17], for each classification task, we use a MultiLayer Perceptron (MLP) classifier with a single hidden layer and ReLU activations. The hidden layer size is fixed at 500 for all the different tasks. We used Librispeech [23] for word recognition task and Voxceleb1 for all others tasks. For all the tasks we trained on 80% of this data and evaluated on the remaining 20%.

B. Performance criterion used in speaker verification

Equal Error Rate (EER) and Detection Cost Function (DCF) are used as the performance criterion of speaker verification. EER is the threshold value such that false acceptance rate and miss rate are equals.

C. Classification task

Figure 1 reports results of the classification tasks described in Section IV. It can be observed that the x-vectors extracted from models using mean and standard-deviation as pooling layer achieve the best performance on the tasks related to speaker : speaker identification, speaker gender, speaker nationality and speaking rate. However, the x-vectors extracted from model that use max-pooling achieve better accuracy on the tasks : augmentation type and word recognition. These experiments highlight the importance of the choice of the pooling layer in the architecture. They show that the x-vector, which is fitted to focus on the speaker verification task, actually embeds different biases of the corpus. Moreover, it is also shown that these biases are strongly depending on the statistical parameters chosen to make up the pooling layer.

D. Speaker verification

Table II shows results obtained with the x-vectors-based systems using a single statistical measure for pooling. The system that obtains the best results is the one using the standard-deviation pooling (denoted as *std*). This results is very surprising since it means that the automatic speaker verification is not done on speaker-specific traits but on their variations. It can be observed that systems using mean (denoted as *mean*) and standard-deviation pooling better perform than systems using the max pooling (system called *max*). This results tends to show that in the context of speaker verification task, the pooling layer must treat the embeddings given by the frame-level component as a statistical distribution and not as non-parametric measure.

Lastly, the systems using skewness (denoted as *skew*) or kurtosis (denoted as *kurto*) pooling yield very bad results. It can be explained by the fact that the skewness and kurtosis poolings do not provide as much information as other poolings about the speaker signal of voice.

TABLE II
RESULTS OBTAINED BY SYSTEMS USING A SINGLE POOLING. THE SYSTEMS CALLED : MAX, MEAN, STD, SKEW AND KURTO REFER RESPECTIVELY TO MAXIMUM, MEAN, STANDARD-DEVIATION, SKEWNESS AND KURTOSIS POOLINGS.

System	**VoxCeleb1** *-E cleaned*		**VoxCeleb1** *-H cleaned*		**SITW** *core-core*	
	EER	DCF	EER	DCF	EER	DCF
max	1.50	0.155	2.54	0.231	1.78	0.147
mean	1.45	0.161	2.45	0.236	1.72	0.151
std	**1.29**	**0.136**	**2.15**	**0.204**	**1.39**	**0.138**
skew	46.19	0.994	46.33	0.995	31.44	0.951
kurto	49.57	0.994	49.81	0.995	39.18	0.951

Table III shows results obtained by systems using multi-statistical poolings. This operation is done by concatenating, for each system, the outputs of different statistical poolings into a unique layer. Let us note that all the different combinations of poolings have been tested, but only the most interesting results are reported. The best performance is obtained by the system concatenating mean, standard-deviation and skewness (denoted as *mean-std-skew)*. The skewness pooling achieves a very slight improvement compared to the baseline system, which uses the mean and standard-deviation pooling (denoted as *mean-std*). The information about the distribution tails (kurtosis pooling) did not improve performance. Also, it can be observed that, in the context of speaker verification, it is better to use a pooling that brings information on statistical measures of central tendency rather than on non-parametric values (the maximum).

TABLE III
RESULTS OBTAINED BY SYSTEMS USING MULTI-STATISTICAL POOLINGS.

System	**VoxCeleb1** *-E cleaned*		**VoxCeleb1** *-H cleaned*		**SITW** *core-core*	
	EER	DCF	EER	DCF	EER	DCF
max-mean	1.45	0.151	2.46	0.226	2.08	0.188
max-std	1.42	0.152	2.36	0.223	1.97	0.163
mean-skew	1.33	0.148	2.27	0.217	1.72	0.185
std-skew	1.28	0.139	2.18	0.203	1.45	**0.133**
mean-std	1.25	0.141	2.11	0.200	1.42	0.135
mean-std-skew	**1.24**	**0.137**	**2.11**	**0.198**	**1.39**	0.138

Table IV summarizes results of fusion of scores carried out on various pooling-content based systems. The fusion of systems is done at the score level, by simply averaging the scores provided by the systems with equal weights. The goal of these experiments is to assess the contribution of the skewness measure to a fusing approach. Let us note that, for all the systems tested, the mini-batch and the weight initialization of the DNN are the same.

In the upper part of Table IV (rows 1 to 3) we propose to fuse the scores of an initial system to those of its version with the additional skewness statistic. Each time, the fusion leads to a significant gain of performance. Moreover, the fusion reported in row 3 of the Table significantly improves performance compared to the previous single best one of Table III, by a relative gain of around 7%.

The lower part of the table compares the fusions of different systems without (rows 4 to 6) or with (rows 7 to 9) the skewness statistic added to the pooling. The reported results show that fusion of systems using several pooling-layer compositions and, also, inclusion of the skewness statistic, contributes to a significant enhancement of performance.

VI. CONCLUSION

Extracting speaker embeddings for speaker recognition by using deep neural network approaches has achieved remarkable results in recent years, when compared to traditional GMM-based probabilistic supervector or i-vector frameworks. It has been noticed that the resulting fixed-size representation of an utterance (referred to as x-vector) embeds, in addition to speaker-related information, meta-information such as session, speaking style or lexical content. In this study, we show that

TABLE IV

RESULTS OBTAINED BY MERGING SYSTEMS MADE UP OF DIFFERENT POOLING STATISTICS. THE TERMS MEAN, STD AND SKEW REFER RESPECTIVELY TO THE MEAN, STANDARD-DEVIATION AND SKEWNESS STATISTICS USED TO FILL THE POOLING LAYER OF THE DNN. THE $\oplus$ SIGN INDICATES THE FUSION OF SCORES

System	**VoxCeleb1** *-E cleaned*		**VoxCeleb1** *-H cleaned*		**SITW** *core-core*	
	EER	DCF	EER	DCF	EER	DCF
(mean)$\oplus$(mean-skew)	1.25	0.144	2.18	0.209	1.45	0.142
(std)$\oplus$(std-skew)	1.18	0.130	2.01	0.191	1.39	0.131
(mean-std)$\oplus$(mean-std-skew)	**1.15**	0.131	**1.97**	**0.190**	**1.31**	**0.129**
(mean)$\oplus$(std)	1.21	0.136	2.11	0.203	1.45	0.137
(mean)$\oplus$(mean-std)	1.24	0.140	2.11	0.205	1.45	0.139
(std)$\oplus$(mean-std)	1.16	0.130	1.97	0.192	1.42	0.132
(mean-skew)$\oplus$(std-skew)	1.15	0.132	2.00	0.191	1.37	0.134
(mean-skew)$\oplus$(mean-std-skew)	1.16	0.130	2.00	0.192	1.29	0.136
(std-skew)$\oplus$(mean-std-skew)	1.15	**0.128**	1.98	0.187	1.39	0.131

the parts of desired information (the one used to discriminate between speakers) and of additional meta-information captured into the utterance representation are significantly dependent on the parameters of the frame distribution that are selected to make up the DNN pooling layer. This central layer, between the acoustic parameters and the speaker identity, usually stacks the 1st and centered-2nd order statistics of the frames (mean and standard deviation). Here, the normalized 3rd and 4th moments (respectively skewness, which measures the asymmetry of the probability distribution, and kurtosis, for the flattening level) are also foreseen and implemented, as well as the non-parametric maximum value. Moreover, several combinations of these five measures are studied. Experiments carried out for our analysis show non-negligible relations between these measures, when used as pooling layer components, and the ability of the resulting x-vectors to detect some meta-information such as gender, nationality, speaking rate, augmentation type and word recognition.

On the other hand, the fusion of speaker recognition systems based on various DNN architectures has proven to be beneficial in terms of speaker detection accuracy. This study shows that, given an architecture (here ResNet with angular margin), varying the only content of the statistic-level components provides a set of subsystems which are able, by a simple fusion (i.e. with equal weights in our experiments, to avoid weak conclusions in terms of robustness), to greatly improve performance of the speaker detection task.

ACKNOWLEDGEMENT

This research was supported by the ANR agency (Agence Nationale de la Recherche), RoboVox project (ANR-18-CE33-0014).

REFERENCES

[1] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP Journal on Advances in Signal Processing*, vol. 2004, no. 4, p. 101962, 2004.

[2] M. Rouvier and S. Meignier, "A global optimization framework for speaker diarization," in *IEEE Odyssey - The Speaker and Language Recognition Workshop*, 2012.

[3] J. P. Campbell, W. Shen, W. M. Campbell, R. Schwartz, J.-F. Bonastre, and D. Matrouf, "Forensic speaker recognition," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 95–103, 2009.

[4] A. Gresse, M. Rouvier, R. Dufour, V. Labatut, and J.-F. Bonastre, "Acoustic pairing of original and dubbed voices in the context of video game localization," in *Interspeech*, 2017.

[5] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)*, vol. 19, no. 4, pp. 788–798, 2010.

[6] 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)*. IEEE, 2018, pp. 5329–5333.

[7] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, J. Borgstrom, F. Richardson, S. Shon, F. Grondin *et al.*, "The jhu-mit system description for nist sre18," 2018.

[8] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, T. Koshinaka, J. Zhang, and K. Shinoda, "The nec-tt 2018 speaker verification system." in *Interspeech*, 2019, pp. 4355–4359.

[9] P.-M. B. Mickael Rouvier, "The lia system description for nist sre 2019," 2019.

[10] J. Thienpondt, B. Desplanques, and K. Demuynck, "The idlab voxceleb speaker recognition challenge 2020 system description," 2020.

[11] N. Brummer, L. Burget, O. Glembek, P. Matejka, L. Mošner, O. Novotnỳ, O. Plchot, J. Rohdin, A. Silnova, T. Stafylakis *et al.*, "But+ omilia system description voxceleb speaker recognition challenge 2020."

[12] N. Torgashov, "Id r&d system description to voxceleb speaker recognition challenge 2020," 2020.

[13] J. Villalba and N. Dehak, "The jhu system description for sdsv2020 challenge," 2019.

[14] M. R. Pierre-Michel Bousquet, "The lia system description for sdsv challenge task 2," 2019.

[15] S. P. Guillermo Barbadillo, "Veridas solution for sdsv challenge technical report," 2019.

[16] D. Raj, D. Snyder, D. Povey, and S. Khudanpur, "Probing the information encoded in x-vectors," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 726–733.

[17] S. Wang, Y. Qian, and K. Yu, "What does the speaker embedding encode?" in *Interspeech*, 2017, pp. 1497–1501.

[18] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "But system description to voxceleb speaker recognition challenge 2019," 2019.

[19] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," 2015.

[20] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," 2018.

[21] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (sitw) speaker recognition database." in *Interspeech*, 2016, pp. 818–822.

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

[23] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.