mathematics

Article

Residual Information in Deep Speaker Embedding Architectures

Adriana Stan

Department of Communications, Technical University of Cluj-Napoca, 400114 Cluj-Napoca, Romania; adriana.stan@com.utcluj.ro

Abstract: Speaker embeddings represent a means to extract representative vectorial representations from a speech signal such that the representation pertains to the speaker identity alone. The embeddings are commonly used to classify and discriminate between different speakers. However, there is no objective measure to evaluate the ability of a speaker embedding to disentangle the speaker identity from the other speech characteristics. This means that the embeddings are far from ideal, highly dependent on the training corpus and still include a degree of residual information pertaining to factors such as linguistic content, recording conditions or speaking style of the utterance. This paper introduces an analysis over six sets of speaker embeddings extracted with some of the most recent and high-performing deep neural network (DNN) architectures, and in particular, the degree to which they are able to truly disentangle the speaker identity from the speech signal. To correctly evaluate the architectures, a large multi-speaker parallel speech dataset is used. The dataset includes 46 speakers uttering the same set of prompts, recorded in either a professional studio or their home environments. The analysis looks into the intra- and inter-speaker similarity measures computed over the different embedding sets, as well as if simple classification and regression methods are able to extract several residual information factors from the speaker embeddings. The results show that the discriminative power of the analyzed embeddings is very high, yet across all the analyzed architectures, residual information is still present in the representations in the form of a high correlation to the recording conditions, linguistic contents and utterance duration. However, we show that this correlation, although not ideal, could still be useful in downstream tasks. The low-dimensional projections of the speaker embeddings show similar behavior patterns across the embedding sets with respect to intra-speaker data clustering and utterance outlier detection.

Keywords: speaker embeddings; x-vectors; deep representations; deep embeddings; speaker disentanglement; speaker recognition; residual information; neural architectures; deep learning; artificial intelligence

MSC: 68T01

check for updates

Citation: Stan, A. Residual Information in Deep Speaker Embedding Architectures. *Mathematics* **2022**, *10*, 3927. https://doi.org/10.3390/math10213927

Academic Editor: Bo-Hao Chen

Received: 16 September 2022
Accepted: 19 October 2022
Published: 23 October 2022

1. Introduction

Recorded speech is an inherently complex signal including information related to the linguistic contents, prosodic or style factors (such as rhythm and intonation), as well as speaker characteristics (such as physiological traits, gender, ethnicity or social background). Humans have the ability to disentangle almost all of these factors and extract their abstractions, being able to reproduce and recognize similar patterns across spoken data from different sources. An essential part of the speech signal with a multitude of downstream applications is related to the disentanglement of the speaker identity. Such an accurate representation of the speaker identity would make it extremely useful in tasks such as speaker recognition and verification applications, text-to-speech synthesis and voice cloning [1], anonymization or generating new, unseen speaker identities [2]. There is already a large number of published works which focus on speaker discrimination, meaning that their task is to estimate if two or more acoustic signals pertain to the same speaker identity. However, in terms of accurately representing the speaker for generative processes (such as text-to-speech or voice cloning), to date, there are no published methods which can accurately solely represent the speaker identity and disregard other factors related to the acoustic signal, such as recording conditions or linguistic contents, although many of these applications use the derived representations as input or conditioning. It is therefore essential to perform an analysis of how well the current speaker representations model the speaker identity.

As a result, in this paper, the focus is on finding out how much *residual information* (i.e., not pertaining to the speaker identity) is present in various deep speaker embeddings. Six open source, easy-to-use, readily available implementations were selected. The implementations report the state-of-the-art results for speaker classification and diarization tasks. A first evaluation carried out in this work aimed at directly comparing the architectures' performances with respect to their intended use, i.e., speaker discrimination. The equal error rates (EER) and inter- and intra-speaker similarity measures were computed over a large multi-speaker parallel dataset. In the second step of the evaluation, the architectures' derived speaker embeddings were analyzed in terms of the amount of residual information present within them, such as the utterance duration, signal-to-noise ratio, linguistic contents and recording conditions. Simple classification and regression algorithms were employed in an attempt to extract this information from the representations. The results show that, to a large extent, the embeddings exhibit a high dependency on these factors, and as such, the speaker identity is not truly disentangled. Based on these initial results, we then attempted to explore if this residual information could still be useful in downstream tasks. The derived speaker embeddings were plotted in a low-dimensional representation to verify if they exhibit similar patterns with respect to clustering different recording sessions and background conditions, as well as to separate utterance outliers and ill-behaved speakers. Such information could be exploited for selecting the appropriate speakers and a set of samples for a text-to-speech synthesis system or data augmentation process in multi-speaker systems.

The **contributions** of this paper can be summarized as follows:

- Six of the most recent speaker embedding deep neural networks are directly compared with respect to their discriminative and generative characteristics;
- The analysis is carried out over a parallel dataset consisting of 46 speakers uttering the same prompts;
- The equal error rates (EER) and inter- and intra-speaker similarity measures for the six architectures are evaluated;
- Decision trees and light gradient boosting machine algorithms are employed to evaluate the amount of residual information present in the embeddings;
- Low-dimensional tSNE-based representations of the embedding space for the six architectures are evaluated in terms of outlier detection and intra-speaker data clustering.

The paper is organized as follows: Section 2 presents some of the previous studies which address the development of accurate speaker embeddings, as well as their use in voice cloning and text-to-speech synthesis systems. Section 3 describes the audio data and speaker embedding architectures adopted in this work. The results of the evaluation are shown in Section 4, while the conclusion and discussions are introduced in Section 5.

2. Related Work

Speaker recognition has been the focus of the research community for quite a long time now, as it is essential for identification, verification and diarization tasks. Speaker identification refers to determining the identity of a spoken utterance from a set of predefined speakers. Speaker verification aims to predict if two utterances pertain to the same speaker or not, while speaker diarization is targeted at separating the different speaker identities present in a larger audio clip and assigning each audio segment to the corresponding speaker. Although these three sub-tasks of speaker recognition seem different in principle, they all share the common component of extracting the numeric representations able to accurately depict individual speaker identities.

Some of the first methods for speaker recognition were based on spectral and template matching, commonly using Mel-Frequency Cepstral Coefficients [3,4], Linear Prediction

Coefficients (LPC) [5,6] or Perceptual Linear Prediction (PLP) Coefficients [7,8]. Starting with the 1990s, Gaussian Mixture Models (GMM) [9,10] became more prevalent. The models estimated the statistics of various speech signal representations for each individual speaker within the dataset. The verification or recognition was performed by computing the distance between the target speaker and each of the probability distribution functions within the GMMs set. With the addition of the Universal Background Model (UBM) [11] and Support Vector Machines (SVM) [12], the performances of speaker recognition methods kept improving. Yet, the evaluations were performed on small, curated, clean datasets and more than often in text-dependent scenarios. Dimensionality reduction techniques were subsequently applied so as to extract the axes of maximum variation among the speakers of interest. Within this area, Principal Component Analysis (PCA) [13] and i-vectors [14] rapidly gained popularity.

The major improvements in speaker recognition, as in many other application fields, came from the introduction of deep learning architectures able to abstractize the information present in the speech signal benefiting from a large amount of spoken data. The first step toward the DNN-based representations was simply to use the deep architecture's posterior probabilities instead of the GMM-based ones [15]. Similar to the PCA technique in traditional speaker recognition models, DNN-based bottleneck features became popular [16], being called d-vectors and extracted at the frame level. D-vectors are part of a larger category of DNN-based representations, called embeddings. X-vectors are also embeddings extracted with Time-Delay Neural Networks (TDNN) [17–19] at the segment level, and they became the standard method for speaker recognition applications. Some other deep architectures employed in speaker recognition are RawNet [20,21] and ResNet [22–24]. These types of embeddings are extracted in an end-to-end manner, meaning that the network is in charge of both finding adequate representations as well as determining the final decision related to the speaker-related task. Previous methods used either the Probabilistic Linear Discriminant Analysis (PLDA) or cosine similarity to estimate the similarities or dissimilarities between the output representations. Some recent studies even attempted to adapt other speech-based neural representations for speaker recognition [25]. An extended overview of the deep learning-based speaker embedding representations can be found in [26,27].

As more and more methods were published, a common evaluation benchmark was required to correctly compare their individual results. Several speaker recognition workshops and challenges have been organized, such as the NIST Speaker Recognition Evaluation (https://www.nist.gov/itl/iad/mig/speaker-recognition, accessed on 18 October 2022), Odyssey (http://www.odyssey2022.org/, accessed on 18 October 2022) or VoxSRC (https://www.robots.ox.ac.uk/~vgg/data/voxceleb/ interspeech2022.html, accessed on 18 October 2022). Within the 2022 VoxSRC challenge, there were two main tracks related to speaker verification (open and closed sets) and speaker diarization. The best performing systems included ResNet and ECAPA-TDNN architectures augmented with self-supervised learned (SSL) representations of the audio signal [28–31].

Although the methods described above are aimed solely at speaker recognition, verification and diarization applications, their findings can be applied to other speech-related tasks. One of the most important and widely used is that of speech synthesis and voice cloning. Speaker embeddings extracted from networks trained on a large number of speakers can be used to condition multi-speaker synthesis models. Using externally learned embeddings enables the models to perform a fast or zero-shot adaptation for unseen speakers [32–36]. And it is for these tasks that a more elaborate analysis of the embeddings' accuracy is extremely important.

3. Experimental Setup

3.1. Speech Data

A problematic part of the speaker embeddings' evaluation is the fact that the spoken data across the speakers may vary. This means that there is a possibility that the number and linguistic content of each speaker's utterance subset may influence the results. Therefore, in this study, we used one of the largest parallel spoken datasets available.

The dataset is the extended version of the SWARA corpus [37]. The initial version of the corpus—which will be referred to as SWARA1.0—includes 18 speakers recorded in a professional studio. Each speaker read aloud between 921 and 1493 utterances. This dataset was recently extended with an additional 28 speakers—we will refer to this subset as SWARA2.0. However, due to the COVID-19 pandemic, the recordings were performed in the speakers' home environments with semi-professional equipment. The speakers read between 1597 and 1797 utterances. As the SWARA2.0 was recorded in home conditions, we expect the background noise and reverberation to affect the performance of the embeddings extracted from this dataset.

In both SWARA1.0 and SWARA2.0 subsets, the speakers were provided with the same text prompts to be read aloud. However, due to the lack of control, especially in the SWARA2.0 scenario, only 712 utterances are truly parallel across all 46 speakers. This means that for the rest of the utterances, the speakers either made deletions, insertions or substitutions with respect to the prompt or did not record some of the prompts at all. There are 24 female speakers and 22 male speakers in the combined datasets, which amount to 32.752 utterances with a total duration of 38 h and 29 min. All data were resampled at 16 kHz and start and end silence segments were trimmed.

3.2. Speaker Embedding Networks

Numerous studies focused on extracting deep learning-based representations for speaker characteristics. Most of these studies are, of course, aimed at discriminating between speakers and performing accurate recognition and diarization tasks. For our evaluation, we targeted the DNN-based architectures which are open source, easily accessible and usable and also provide good pre-trained models. The following architectures were selected:

(1) **Pyannote** (https://github.com/pyannote/pyannote-audio, accessed on 18 October 2022) is an open-source toolkit written in Python for speaker diarization [38,39]. It uses a SincNet [40] architecture, followed by a series of TDNN layers and an average pooling layer. It also includes the implementations for Speech Brain and NeMo Titanet architectures, but we did not use them in this study.

(2) **Speech Brain's** [41] speaker verification network (https://github.com/speechbrain/speechbrain, accessed on 18 October 2022) is based on the ECAPA-TDNN [42] model. It includes a sequence of convolutional and residual blocks, using the additive margin softmax loss as training objective. The speaker embeddings are formed using attentive statistical pooling.

(3) **Clova AI** [22,23] uses a ResNet-like architecture, (https://github.com/clovaai/voxceleb_trainer, accessed on 18 October 2022) and similarly averages the frame-level representations in the final embeddings. The difference is in the change in objective function and the use of training data augmentation.

(4) **NeMo Titanet** (repository for all NeMo architectures is available here: https://github.com/NVIDIA/NeMo/, accessed on 18 October 2022) uses the Titanet architecture [43] which is based on ContextNet [44]. The model uses 1D depth-wise separable convolutions with squeeze-and-excitation layers. The output embeddings are obtained by averaging the statistics of the intermediate variable-length representations.

(5) **NeMo SpeakerNet** uses an ASR architecture's encoder, namely QuartzNet [45], as a high-level feature extractor and averages these features within a pooling layer so as to capture the time-independent speaker features.

(6) **NeMo ECAPA-TDNN** is similar to the Speech Brain architecture and uses the ECAPA-TDNN structure [46]. The difference is that, instead of the residual blocks, the NeMo implementation uses group convolution blocks of single dilation.

Because we aim to use the embeddings in speech synthesis systems, no fine-tuning was performed over the available pre-trained models. Fine-tuning would imply that any time a new dataset is used for synthesis, the embedding networks would need to be re-trained, which would not be feasible. Therefore, we explore the architecture's behavior as is and as their authors made them available for the wide research community.

4. Evaluation

We base our evaluation on a set of analyses which we consider relevant for the use of the speaker embeddings in downstream tasks, outside the speaker recognition or discrimination applications. The following subsections introduce the results of the evaluation scenarios.

4.1. EER, Intra- and Inter-Speaker Similarity

Being derived from neural architectures aimed at speaker recognition, thus in a discriminative-oriented task, the speaker embeddings' performance is commonly evaluated in terms of the equal error rate (EER). The EER is defined as the point on the ROC curve where the false acceptance rate (FAR) is equal to the false rejection rate (FRR). The threshold to compute the EER is generally based on the cosine similarity between pairs of speaker embeddings, defined as:

$$\cos(\mathbf{e}_1, \mathbf{e}_2) = \frac{\mathbf{e}_1 \mathbf{e}_2}{\|\mathbf{e}_1\| \|\mathbf{e}_2\|} = \frac{\sum_{i=1}^{n} \mathbf{e}_{1,i} \mathbf{e}_{2,i}}{\sqrt{\sum_{i=1}^{n} (\mathbf{e}_{1,i})^2} \sqrt{\sum_{i=1}^{n} (\mathbf{e}_{2,i})^2}} \tag{1}$$

Because we are using a new speech dataset, it is essential to first evaluate the targeted performance of the selected architectures. Therefore, the first step of our analysis involved the computation of the EER results across the architectures, speakers and sets of speakers. A set of 46,000 random pairs of utterances were selected. The random pairs were chosen such that each speaker is present in 1000 pairs, of which 200 are same-speaker pairs. The EER results are shown in Table 1. We evaluate the results over the entire set of 46,000 pairs and separately for pairs containing only samples for the female speakers, male speakers, speakers from the SWARA1.0 subset and speakers from the SWARA2.0 subset. It can be noticed that all architectures achieve an EER below 1% and that female speakers are better discriminated than male speakers. The same is true for the SWARA1.0 speakers versus the SWARA2.0 speakers. The best results are obtained by the NeMo Titanet architecture.

Table 1. EER values for the different speaker embedding architectures and speaker subsets. Arrows mark the direction of best performance. Best results are highlighted in boldface.

Architecture	**All↓**	**Female↓**	**Male↓**	**SWARA1.0↓**	**SWARA2.0↓**
Pyannote	0.040	0.055	0.039	0.024	0.047
Speech Brain	0.025	0.027	0.031	0.011	0.031
Clova AI	0.055	0.060	0.081	0.031	0.073
NeMo Titanet	**0.018**	**0.014**	**0.027**	**0.005**	**0.024**
NeMo SpeakerNet	0.039	0.045	0.051	0.024	0.048
NeMo ECAPA-TDNN	0.032	0.035	0.041	0.023	0.038

NeMo Titanet is also the best performing architecture for each individual speaker (see Figure 1). However, speakers `bvl`, `mgl` and `pbl` have significantly higher EER values than the rest of the speakers. When analyzing these speakers' recordings, we noticed that indeed the background conditions are considerably poorer than for the other speakers and that, in some cases, the segmentation of the waveform is performed after or before the end of the utterance.

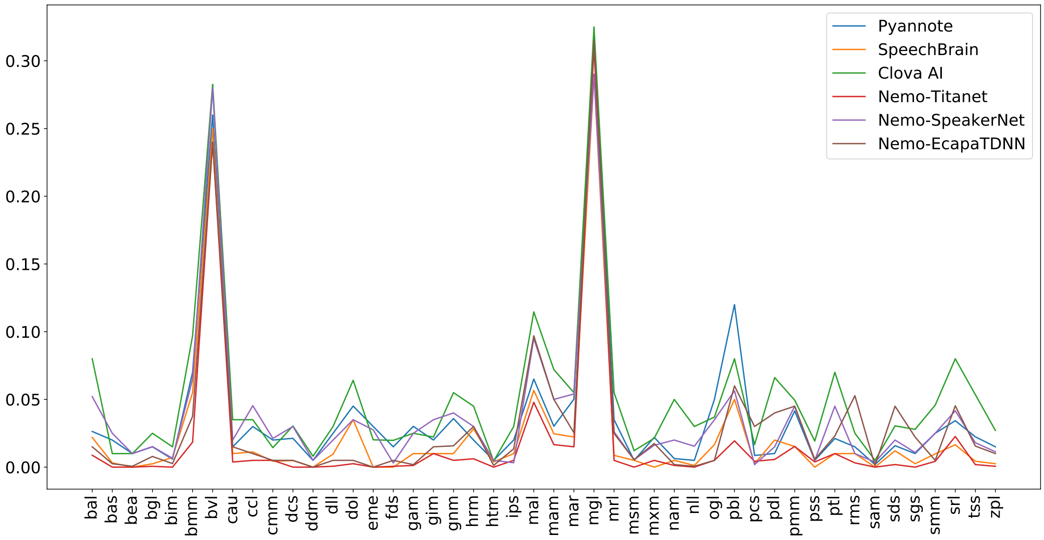

Figure 1. EER values for the individual speakers across the embedding architectures.

The EER gives the optimum threshold for which the *FAR* is equal to the *FRR*, but it does not detail the accuracy of the representation. For downstream tasks, the inter- and intra-speaker similarity measures would be more informative. This means that the discrimination between speakers can be translated into high intra-speaker similarity and low inter-speaker similarity, while any intermediate values should ideally represent perceptually similar speakers. The intra-speaker similarity values are presented in Figure 2. The similarity was computed over all pairs of utterances from the same speaker averaged by their number. It can be noticed that the Clova AI architecture exhibits the highest intra-speaker similarity measure, and that again, speakers `bvl, mgl` and `pbl` have the lowest intra-speaker similarity. The least performing architecture is Pyannote. In Table 2, we average the above scores at the system-level and also across the different speaker subsets. The Clova AI, Female and SWARA1.0 speakers exhibit the highest intra-speaker similarity measures.

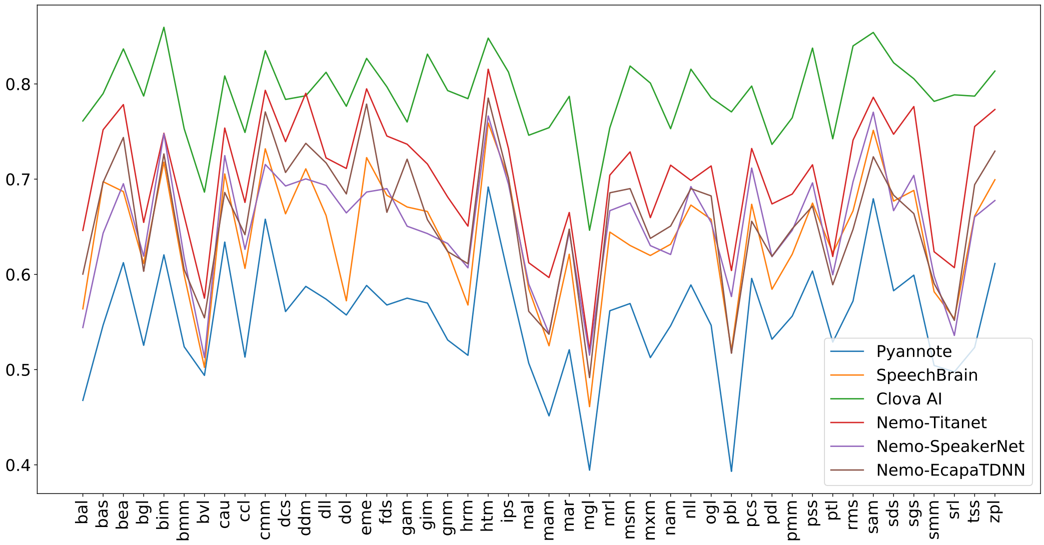

Figure 2. Intra-speaker cosine similarity for each speaker and embedding architecture.

Table 2. Average **intra-speaker** cosine similarity across the different speaker embedding architectures and different speaker subsets. Arrows mark the direction of best performance. Best results are highlighted in boldface.

Architecture	**All↑**	**Female↑**	**Male↑**	**Swara1.0↑**	**Swara2.0↑**
Pyannote	0.554	0.557	0.550	0.589	0.531
Speech Brain	0.640	0.651	0.629	0.686	0.610
Clova AI	**0.788**	**0.790**	**0.786**	**0.810**	**0.774**
NeMo Titanet	0.702	0.711	0.695	0.750	0.672
NeMo SpeakerNet	0.651	0.658	0.644	0.693	0.623
NeMo ECAPA-TDNN	0.658	0.670	0.647	0.696	0.635

For the inter-speaker similarity, we use the 46,000 random pairs of utterances used to evaluate the EER and select only those which pertain to different speaker identities. We then compute the average cosine similarities between each pair of speakers. In this scenario, we would expect the architectures to exhibit very low values so as to maximize the discriminative characteristics of the representation. In Table 3, we introduce these results averaged across the architectures and speaker subsets. Although NeMo Titanet seemed to show the best performance in the previous tasks, in terms of discriminative power, NeMo ECAPA-TDNN is the most efficient (with an exception for the inter-male speakers where Pyannote is best). We show the inter-speaker similarity matrix for the NeMo ECAPA-TDNN architecture in Figure 3. The closest speakers based on these scores are `htm` and `mar`, with a similarity of 0.42, followed by `cmm` and `cau` at a 0.40 similarity. These pairs are female speakers and do indeed have perceptually similar voices.

In this section, we looked at common measures to evaluate speaker embedding architectures while aiming to extract additional information that may be useful for downstream tasks. For example, inter-speaker similarity could be used to train speech synthesis systems in limited data scenarios, where data augmentation can be performed by using speech from a different, yet similar sounding speaker.

Table 3. Average **inter-speaker** cosine similarity across the different speaker embedding architectures and different speaker subsets. Arrows mark the direction of best performance. Best results are highlighted in boldface.

Architectures	**All↓**	**Female↓**	**Male↓**	**Swara1.0↓**	**Swara2.0↓**
Pyannote	0.127	0.195	**0.129**	0.139	0.127
Speech Brain	0.122	0.188	0.143	0.139	0.118
Clova AI	0.133	0.188	0.157	0.142	0.135
NeMo Titanet	0.132	0.187	0.156	0.143	0.135
NeMo SpeakerNet	0.195	0.272	0.226	0.199	0.198
NeMo ECAPA-TDNN	**0.107**	**0.151**	0.140	**0.136**	**0.110**

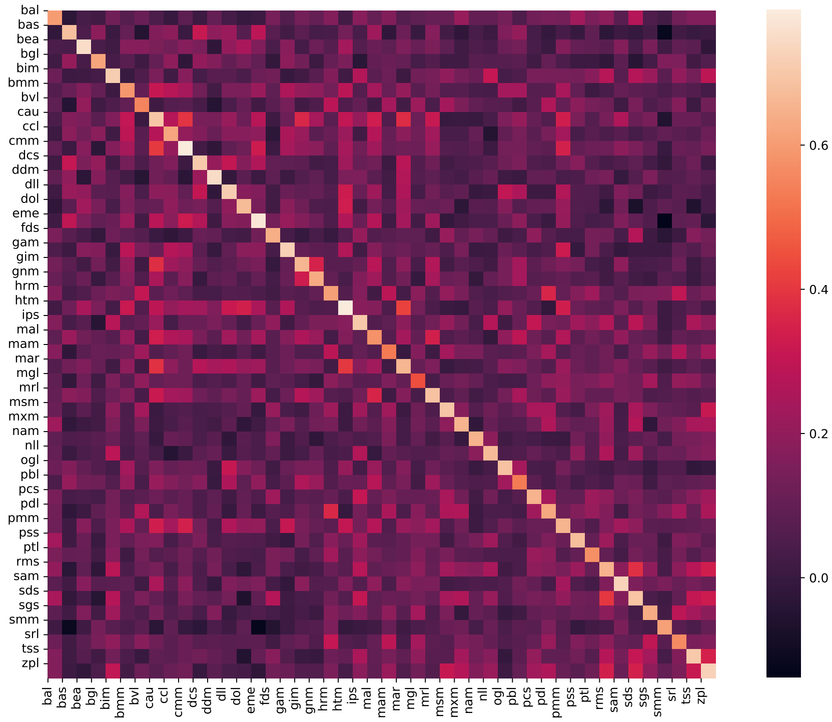

Figure 3. Inter-speaker similarity matrix for the NeMo ECAPA-TDNN architecture.

4.2. Speaker Identity Disentanglement

The previous set of results are definitely aimed at providing the best representation for speaker classification tasks. However, in many downstream applications of the speaker embeddings, this is not enough. And this is true especially in speech synthesis systems, where the embeddings are used as additional input, while information related to the linguistic content and prosodic patterns are the main inputs. Therefore, if information pertaining to other aspects of the speech is present in the embeddings, this can lead to unwanted effects and bias within the training procedure. Starting from the above statement, in this section, we want to explore how much residual information is present within the speaker embeddings, and thus if the speaker identity is truly disentangled from all other speech factors.

A simple method to determine the presence of residual information is to see if simple machine learning algorithms are able to extract this information from the embeddings. We adopt two separate algorithms, depending on the task: a decision tree for the classification tasks and a LightGBM [47] for the regression tasks. The two algorithms were chosen as they are some of the simplest, yet most powerful traditional machine learning methods, and their results are comparable across tasks and datasets. The same parameters were used across the tasks, and the speech dataset was randomly split into 80% training and 20% test sets.

Table 4 shows the results expressed in F1-scores for the classification of speaker gender and speaker identity. For these two targets, the results are supposed to be high, as the embeddings should incorporate this information. Two other targets shown in the table are the text length—expressed in number of characters—and the recording conditions. The recording condition is encoded as a binary classification for the two subsets, SWARA1.0 and SWARA2.0. For these two columns, the accuracy of the predictions should be limited, as information about these two targets should not be easily extracted from the embeddings. For the text length, the results are as expected, and the number of characters present in the utterance cannot be extracted from the embeddings. Yet, an interesting result occurs in the recording conditions column, where the prediction accuracy is rather high. This means that the additional spectral artefacts present in the home recordings are indeed influential for the final embeddings across all embedding architectures.

Table 4. F1-scores for various classification tasks. Arrows mark the direction of best performance. Best results are highlighted in boldface.

Architecture	Speaker ID↑	Gender↑	Utterance Duration↓	Recording Condition↓
Pyannote	0.76	0.94	0.016	**0.87**
Speech Brain	0.84	0.96	0.011	0.90
Clova AI	0.85	0.98	0.015	0.92
NeMo Titanet	**0.90**	**0.97**	**0.010**	0.95
NeMo SpeakerNet	0.85	0.96	0.014	0.91
NeMo ECAPA-TDNN	0.87	0.96	0.015	0.95

For the regression tasks, we look at the utterance duration (measured in seconds), the signal-to-noise ratio (SNR) and the linguistic contents. The SNR was computed with the WADA [48] algorithm. For the text contents, we are only looking at the utterance id, assuming that similar characteristics would be present across speaker embeddings for the same linguistic contents. The accuracy of the LightGBM is measured in terms of the Spearman Rank Correlation Coefficient (SRCC) of the predicted values versus the target values. Table 5 shows the results, and they are not encouraging. All the architectures show a high correlation factor (>0.7) with respect to the evaluated targets. This means that this type of information is not efficiently disentangled from the resulting speaker embedding.

A separate regression task looked into the prediction of the average F0 value at the utterance level (the last column in Table 5). The correlations are rather high and the average mean squared error for the six architectures is 12 Hz.

Table 5. LightGBM-based SRCC results for various regression tasks. Arrows mark the direction of best performance. Best results are highlighted in boldface.

Architecture	Utterance Duration↓	SNR↓	Linguistic Contents↓	F0↑
Pyannote	0.830	0.771	0.723	0.958
Speech Brain	0.734	0.749	0.742	0.959
Clova AI	0.750	0.791	0.734	**0.976**
NeMo Titanet	**0.704**	**0.747**	0.798	0.964
NeMo SpeakerNet	0.796	0.758	**0.709**	0.958
NeMo ECAPA-TDNN	0.862	0.787	0.775	0.962

The results in this section showed that although the task of the embedding architectures is to represent the speaker identity as accurately as possible, residual information is still present within them and that more work should be performed to find a more suitable representation for the downstream tasks. In the following section, we would like to explore how we can use this residual information to the benefit of other tasks by using visual representations of the embeddings.

4.3. Visual Representations

Visually examining high-dimensional data is not feasible. However, in most cases, visual representations of the data are more informative to the developer than just percentages and numbers. As such, we employed a t-SNE dimensionality reduction technique [49] and plotted the speaker embeddings obtained from the six architectures into a two-dimensional space. The algorithm was applied over the entire set of embeddings from each individual architecture, ran over 1000 steps and a perplexity of 30 was used. Figure 4 shows these t-SNE plots for each architecture. It can be noticed that, with some minor exceptions, the speakers are clustered nicely. Moreover, it does not seem that any of the speaker embedding architectures shows a better performance in terms of grouping the speakers. It appears that for some of the speakers, sub-clusters of the embeddings are formed, and in a few cases, outliers are to be observed. By examining the individual speech samples, we noticed that the outliers commonly pertain to the short utterances, reaffirming the previous result of the correlation between the embeddings and the duration of the audio. For the sub-clusters, most speakers recorded the entire set of prompts in at least two recording sessions. Between the sessions, there were differences in the background noise, distance from the microphone, speaking rhythm and vocal effort. The formed t-SNE clusters are indeed consistent with the different recording sessions. It can be noticed that these clusters are more common within the SWARA2.0 subset containing the home recordings. Among the different embedding architectures, it appears that all of them are able to detect these sub-clusters, while NeMo SpeakerNet and Pyannote seem to be more affected by the duration of the audio and present more outliers than the other systems. This means that these two architectures would be more suitable in pre-processing a speaker dataset for downstream applications.

However, when we zoom in on these low-dimension visualizations, we notice some interesting patterns. Figures 5 and 6 show the speaker-level t-SNE representations for the two data subsets, i.e., SWARA1.0 and SWARA2.0, respectively.

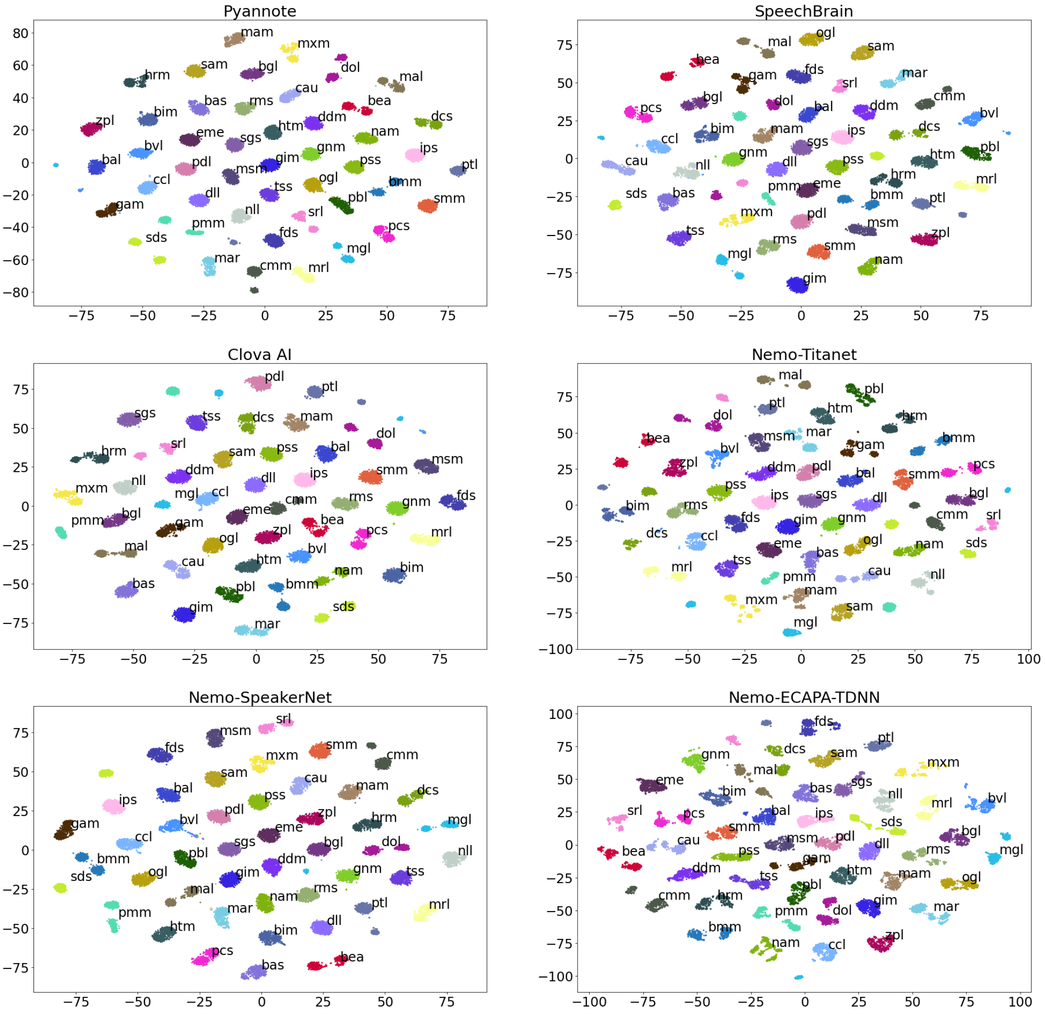

Figure 4. t-SNE representations of the speaker embeddings extracted from the different architectures.

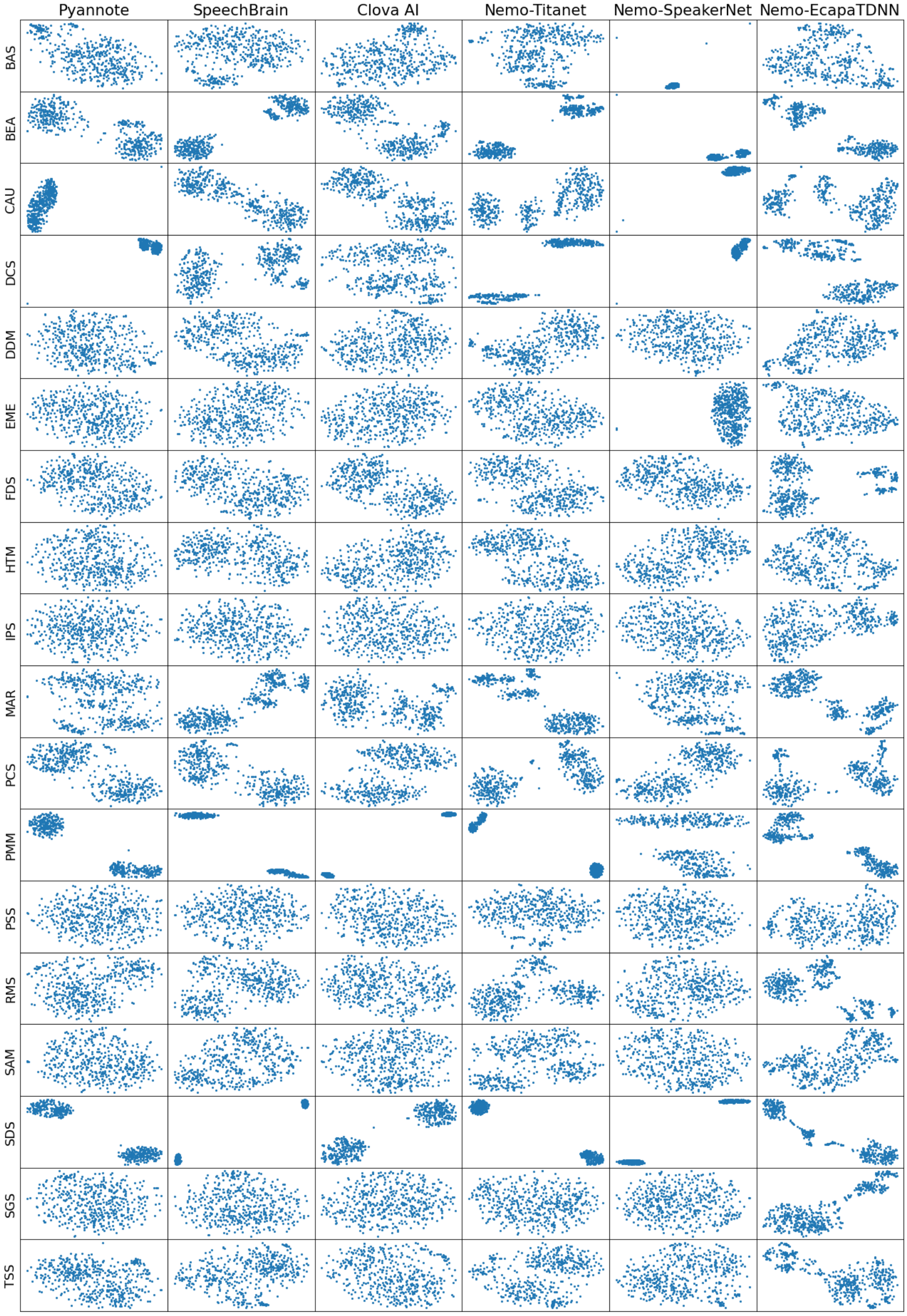

Figure 5. Speaker-level t-SNE plots of the different embedding architectures—**SWARA1.0** subset.

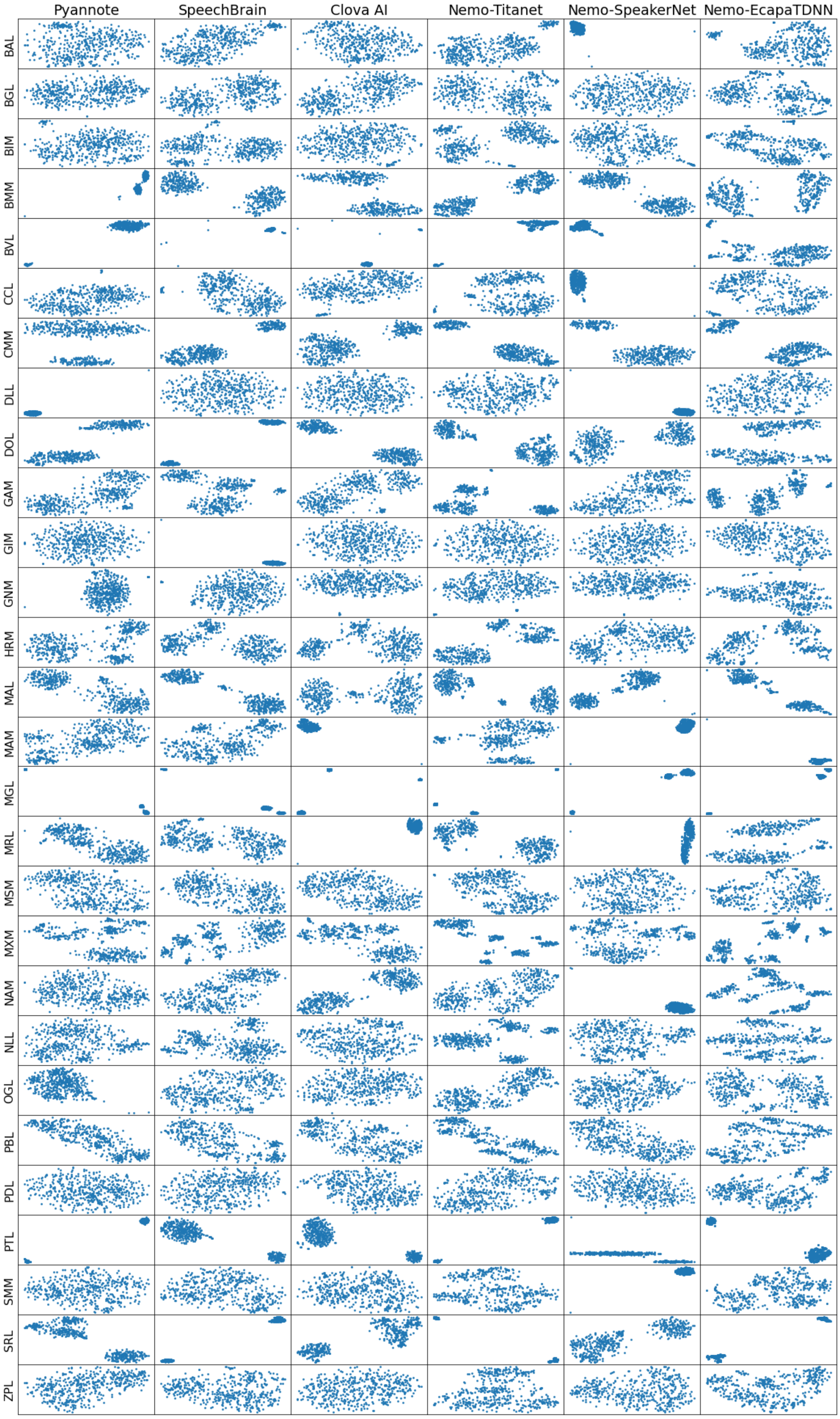

Figure 6. Speaker-level t-SNE plots of the different embedding architectures—**SWARA2.0** subset.

5. Conclusions and Future Work

In this paper, we attempted to evaluate some of the most recent and high-performing deep speaker embeddings with respect to their intended use, i.e., speaker discrimination, as well as with respect to their drawbacks in terms of residual information present in the embeddings. The selected architectures are Pyannote, Speech Brain, Clova AI, NeMo Titanet, NeMo SpeakerNet and NeMo ECAPA-TDNN, and they were evaluated over a large multi-speaker parallel dataset containing over 38 h of spoken data.

In a first set of experiments, the architectures were evaluated in terms of the EER and inter- and intra-speaker similarity measures. With respect to the EER, the best discrimination was obtained by NeMo Titanet. However, in terms of the intra-speaker similarity, the architecture which was able to better cluster the speakers was Clova AI, while NeMo ECAPA-TDNN performed best at maximizing the distance between the speakers' representations. This set of evaluations also looked into the different subsets of the audio data, i.e., the male vs. female and studio recordings vs. home recording subsets. The results showed that the female and studio-recorded speakers achieve lower EER and higher intra-speaker cosine similarity measures. In addition, the male and home-recorded speakers exhibit larger inter-speaker cluster distances.

A second set of experiments measured the amount of residual information present in the six sets of speaker embeddings. Simple classification and regression algorithms were employed. These algorithms were supposed to achieve high accuracy measures when different speech factors were present in the embeddings. The examined factors were: the utterance duration in terms of the number of characters and signal duration, recording conditions, signal-to-noise ratio and linguistic contents. All six architectures showed high correlations to the length of the signal and recording conditions, including the SNR. The least amount of residual information pertaining to the recording conditions was present in the Pyannote architecture. With respect to the utterance duration and SNR, NeMo Titanet-based embeddings were less correlated to these factors, and NeMo SpeakerNet embeddings had the smallest correlation factor with the linguistic contents of the utterance. However, the differences between the six architectures are minimal, and we posit that, to this point, none of them have truly obtained a disentangled speaker representation.

Given the results of the residual information's presence in the embeddings, a third set of experiments looked into how these residual factors could be exploited in further downstream applications of the speaker embeddings. Low-dimensional t-SNE-based representations of the six sets of embeddings were plotted. With respect to global speaker representations, all the architectures showed a similar performance with well-behaved clusters, with the exception of NeMo ECAPA-TDNN for which the clusters had larger distributions. When zooming in on these t-SNE representations at the speaker level, all deep representations exhibited sub-clusters pertaining to different recording sessions, as well as outlier utterances correlated to short utterances. This means that this information present in the embeddings' projections could in fact be used, for example, in the data selection process for text-to-speech or voice cloning applications. Outlier utterances could be removed from the training set, and ill-behaved speaker datasets could be further curated or removed altogether. This is in fact one of the next steps to extend the work presented in this paper. We plan to examine how different text-to-speech architectures are affected by the variability of certain speakers and if removing utterance outliers enhances the performance of the output synthesized speech. Another important result of this work pertains to the use of embedding-based similar speakers for data augmentation in TTS systems, meaning that using the most similar speaker with respect to the target speaker will indeed improve the naturalness and speaker similarity of the resulting system.

Moreover, given the availability of the speaker embedding networks, we are planning to use the findings of this study in a task of multi-speaker text-to-speech synthesis system training and determine the most efficient manner to input these embeddings into the synthesis networks, as well as to verify how the embeddings are affected by the synthetic output and how they can be adjusted to better represent the various speaker identities.

Funding: The work carried out in this paper was partially funded by Zevo Technology under grant agreement no. 21439. The APC was funded by the Technical University of Cluj-Napoca.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable.

Data Availability Statement: The processed datasets and experiment flows used in this paper can be obtained from the author.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Jia, Y.; Zhang, Y.; Weiss, R.J.; Wang, Q.; Shen, J.; Ren, F.; Chen, Z.; Nguyen, P.; Pang, R.; Lopez-Moreno, I.; et al. Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis. *CoRR* **2018**, Available online: http://xxx.lanl.gov/abs/1806.04558 (accessed on 18 October 2022).
2. Stanton, D.; Shannon, M.; Mariooryad, S.; Skerry-Ryan, R.; Battenberg, E.; Bagby, T.; Kao, D. Speaker Generation. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7897–7901. [CrossRef]
3. Luck, J.E. Automatic speaker verification using cepstral measurements. *J. Acoust. Soc. Am.* **1969**, *46*, 1026–1032. [CrossRef] [PubMed]
4. Furui, S. Cepstral analysis technique for automatic speaker verification. *IEEE Trans. Acoust. Speech Signal Process.* **1981**, *29*, 254–272. [CrossRef]
5. Ye, W.; Wu, D.; Nucci, A. Experimental Study on GMM-Based Speaker Recognition. In Proceedings of the SPIE—The International Society for Optical Engineering, Orlando, FL, USA, 28 April 2010. [CrossRef]
6. Chandra, M.; Nandi, P.; Kumari, A.; Mishra, S. Spectral-Subtraction Based Features for Speaker Identification. *Adv. Intell. Syst. Comput.* **2015**, *328*, 529–536._58. [CrossRef]
7. McLaren, M.; Scheffer, N.; Graciarena, M.; Ferrer, L.; Lei, Y. Improving speaker identification robustness to highly channel-degraded speech through multiple system fusion. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6773–6777. [CrossRef]
8. Plchot, O.; Matsoukas, S.; Matejka, P.; Dehak, N.; Ma, J.; Cumani, S.; Glembek, O.; Hermansky, H.; Mallidi, S.; Mesgarani, N.; et al. Developing a speaker identification system for the DARPA RATS project. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6768–6772. [CrossRef]
9. Reynolds, D.A. A Gaussian Mixture Modeling Approach to Text-Independent Speaker Identification. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 1992.
10. Reynolds, D.; Rose, R. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Trans. Speech Audio Process.* **1995**, *3*, 72–83. [CrossRef]
11. Reynolds, D.A.; Quatieri, T.F.; Dunn, R.B. Speaker Verification Using Adapted Gaussian Mixture Models. *Digit. Signal Process.* **2000**, *10*, 19–41. [CrossRef]
12. Campbell, W.; Sturim, D.; Reynolds, D. Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Process. Lett.* **2006**, *13*, 308–311. [CrossRef]
13. Zhang, W.; Yang, Y.; Wu, Z. Exploiting PCA classifiers to speaker recognition. In Proceedings of the International Joint Conference on Neural Networks, Portland, OR, USA, 20–24 July 2003; Volume 1, pp. 820–823. [CrossRef]
14. Dehak, N.; Kenny, P.J.; Dehak, R.; Dumouchel, P.; Ouellet, P. Front-End Factor Analysis for Speaker Verification. *IEEE Trans. Audio Speech Lang. Process.* **2011**, *19*, 788–798. [CrossRef]
15. Chen, K.; Salman, A. Learning Speaker-Specific Characteristics with a Deep Neural Architecture. *IEEE Trans. Neural Netw.* **2011**, *22*, 1744–1756. [CrossRef]
16. Hinton, G.E.; Salakhutdinov, R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [CrossRef]
17. Snyder, D.; Garcia-Romero, D.; Sell, G.; Povey, D.; Khudanpur, S. X-Vectors: Robust DNN Embeddings for Speaker Recognition. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5329–5333. [CrossRef]
18. Garcia-Romero, D.; Snyder, D.; Sell, G.; McCree, A.; Povey, D.; Khudanpur, S. x-Vector DNN Refinement with Full-Length Recordings for Speaker Recognition. In Proceedings of the INTERSPEECH 2019, Graz, Austria, 15–19 September 2019.
19. You, L.; Guo, W.; Dai, L.; Du, J. Multi-Task Learning with High-Order Statistics for x-Vector Based Text-Independent Speaker Verification. *arXiv* **2019**, arXiv:1903.12058.
20. Jung, J.W.; Heo, H.S.; Kim, J.H.; Shim, H.J.; Yu, H.J. RawNet: Advanced End-to-End Deep Neural Network Using Raw Waveforms for Text-Independent Speaker Verification. *arXiv* **2019**, arXiv:1904.08104.
21. Jung, J.W.; Kim, S.B.; Shim, H.J.; Kim, J.H.; Yu, H.J. Improved RawNet with Feature Map Scaling for Text-Independent Speaker Verification Using Raw Waveforms. *arXiv* **2020**, arXiv:2004.00526.
22. Chung, J.S.; Huh, J.; Mun, S.; Lee, M.; Heo, H.S.; Choe, S.; Ham, C.; Jung, S.; Lee, B.J.; Han, I. In defence of metric learning for speaker recognition. *arXiv* **2020**, arXiv:2003.11982.

23. Kwon, Y.; Heo, H.S.; Lee, B.J.; Chung, J.S. The ins and outs of speaker recognition: Lessons from VoxSRC 2020. In Proceedings of the ICASSP 2021—2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021.
24. Thienpondt, J.; Desplanques, B.; Demuynck, K. The IDLAB VoxCeleb Speaker Recognition Challenge 2020 System Description. *arXiv* **2020**, arXiv:2010.12468.
25. Vaessen, N.; Van Leeuwen, D.A. Fine-Tuning Wav2Vec2 for Speaker Recognition. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7967–7971. [CrossRef]
26. Bai, Z.; Zhang, X.L. Speaker recognition based on deep learning: An overview. *Neural Netw.* **2021**, *140*, 65–99. [CrossRef] [PubMed]
27. Ohi, A.Q.; Mridha, M.F.; Hamid, M.A.; Monowar, M.M. Deep Speaker Recognition: Process, Progress, and Challenges. *IEEE Access* **2021**, *9*, 89619–89643. [CrossRef]
28. Makarov, R.; Torgashov, N.; Alenin, A.; Yakovlev, I.; Okhotnikov, A. *ID R&D System Description to VoxCeleb Speaker Recognition Challenge 2022*; ID R&D Inc.: New York, NY, USA, 2022.
29. Cai, Q.; Hong, G.; Ye, Z.; Li, X.; Li, H. The Kriston AI System for the VoxCeleb Speaker Recognition Challenge 2022. *arXiv* **2022**, arXiv:2209.11433.
30. Suh, S.; Park, S. The ReturnZero System for VoxCeleb Speaker Recognition Challenge 2022. *arXiv* **2022**, arXiv:2209.10147.
31. Zhao, Z.; Li, Z.; Wang, W.; Zhang, P. The HCCL system for VoxCeleb Speaker Recognition Challenge 2022. 2022. Available online: https://www.robots.ox.ac.uk/~vgg/data/voxceleb/data_workshop_2022/reports/zzdddz_report.pdf (accessed on 18 October 2022).
32. Casanova, E.; Shulby, C.; Gölge, E.; Müller, N.M.; de Oliveira, F.S.; Junior, A.C.; Soares, A.d.S.; Aluisio, S.M.; Ponti, M.A. SC-GlowTTS: An Efficient Zero-Shot Multi-Speaker Text-To-Speech Model. *arXiv* **2021**, arXiv:2104.05557.
33. Cooper, E.; Lai, C.I.; Yasuda, Y.; Fang, F.; Wang, X.; Chen, N.; Yamagishi, J. Zero-Shot Multi-Speaker Text-To-Speech with State-Of-The-Art Neural Speaker Embeddings. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6184–6188. [CrossRef]
34. Zhou, Y.; Tian, X.; Li, H. Language Agnostic Speaker Embedding for Cross-Lingual Personalized Speech Generation. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 3427–3439. [CrossRef]
35. Hu, Q.; Bleisch, T.; Petkov, P.; Raitio, T.; Marchi, E.; Lakshminarasimhan, V. Whispered and Lombard Neural Speech Synthesis. In Proceedings of the 2021 IEEE Spoken Language Technology Workshop (SLT), Shenzhen, China, 19–22 January 2021; pp. 454–461. [CrossRef]
36. Zhang, M.; Zhou, Y.; Zhao, L.; Li, H. Transfer Learning From Speech Synthesis to Voice Conversion with Non-Parallel Training Data. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 1290–1302. [CrossRef]
37. Stan, A.; Dinescu, F.; Tiple, C.; Meza, S.; Orza, B.; Chirila, M.; Giurgiu, M. The SWARA Speech Corpus: A Large Parallel Romanian Read Speech Dataset. In Proceedings of the 9th Conference on Speech Technology and Human-Computer Dialogue (SpeD), Bucharest, Romania, 6–9 July 2017.
38. Bredin, H.; Yin, R.; Coria, J.M.; Gelly, G.; Korshunov, P.; Lavechin, M.; Fustes, D.; Titeux, H.; Bouaziz, W.; Gill, M.P. Pyannote.audio: Neural building blocks for speaker diarization. In Proceedings of the ICASSP 2020, IEEE International Conference on Acoustics, Speech, and Signal Processing, Barcelona, Spain, 4–8 May 2020.
39. Bredin, H.; Laurent, A. End-to-end speaker segmentation for overlap-aware resegmentation. *arXiv* **2021**, arXiv:2104.04045.
40. Ravanelli, M.; Bengio, Y. Speaker Recognition from Raw Waveform with SincNet. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018. [CrossRef]
41. Ravanelli, M.; Parcollet, T.; Plantinga, P.; Rouhe, A.; Cornell, S.; Lugosch, L.; Subakan, C.; Dawalatabad, N.; Heba, A.; Zhong, J.; et al. SpeechBrain: A General-Purpose Speech Toolkit. *arXiv* **2021**, arXiv:2106.04624.
42. Desplanques, B.; Thienpondt, J.; Demuynck, K. ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. In Proceedings of the Interspeech 2020, Shanghai, China, 25–29 October 2020; pp. 3830–3834. [CrossRef]
43. Koluguri, N.R.; Park, T.; Ginsburg, B. TitaNet: Neural Model for speaker representation with 1D Depth-wise separable convolutions and global context. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022.
44. Han, W.; Zhang, Z.; Zhang, Y.; Yu, J.; Chiu, C.C.; Qin, J.; Gulati, A.; Pang, R.; Wu, Y. ContextNet: Improving Convolutional Neural Networks for Automatic Speech Recognition with Global Context. *arXiv* **2020**, arXiv:2005.03191.
45. Koluguri, N.R.; Li, J.; Lavrukhin, V.; Ginsburg, B. SpeakerNet: 1D Depth-wise Separable Convolutional Network for Text-Independent Speaker Recognition and Verification. *arXiv* **2020**, arXiv:2010.12653.
46. Dawalatabad, N.; Ravanelli, M.; Grondin, F.; Thienpondt, J.; Desplanques, B.; Na, H. ECAPA-TDNN Embeddings for Speaker Diarization. *arXiv* **2021**, arXiv:2104.01466.
47. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Long Beach, CA, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 3149–3157.
48. Kim, C.; Stern, R. Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis. In Proceedings of the Interspeech 2008 Incorporating SST 08, Brisbane Australia, 22–26 September 2008; pp. 2598–2601. [CrossRef]
49. van der Maaten, L.; Hinton, G.E. Visualizing High-Dimensional Data Using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.