applied sciences

MDPI

Article

Robust Deep Speaker Recognition: Learning Latent Representation with Joint Angular Margin Loss

Labib Chowdhury [1], Hasib Zunair [2] and Nabeel Mohammed [1,*]

[1] Department of Electrical & Computer Engineering, North South University, Bashundhara, Dhaka-1229, Bangladesh; labib.chowdhury@northsouth.edu

[2] Gina Cody School of Engineering and Computer Science, Concordia University, Montreal, QC H3G, Canada; h_zunair@encs.concordia.ca

* Correspondence: nabeel.mohammed@northsouth.edu

Received: 5 October 2020; Accepted: 21 October 2020; Published: 26 October 2020

Abstract: Speaker identification is gaining popularity, with notable applications in security, automation, and authentication. For speaker identification, deep-convolutional-network-based approaches, such as SincNet, are used as an alternative to i-vectors. Convolution performed by parameterized sinc functions in SincNet demonstrated superior results in this area. This system optimizes softmax loss, which is integrated in the classification layer that is responsible for making predictions. Since the nature of this loss is only to increase interclass distance, it is not always an optimal design choice for biometric-authentication tasks such as face and speaker recognition. To overcome the aforementioned issues, this study proposes a family of models that improve upon the state-of-the-art SincNet model. Proposed models *AF-SincNet*, *Ensemble-SincNet*, and *ALL-SincNet* serve as a potential successor to the successful SincNet model. The proposed models are compared on a number of speaker-recognition datasets, such as TIMIT and LibriSpeech, with their own unique challenges. Performance improvements are demonstrated compared to competitive baselines. In interdataset evaluation, the best reported model not only consistently outperformed the baselines and current prior models, but also generalized well on unseen and diverse tasks such as Bengali speaker recognition.

Keywords: speaker recognition; speaker identification; margin loss; SincNet; inter dataset testing; biometric authentication; feature embedding

1. Introduction

Speaker recognition is of interest in biometric authentication and security, and consists of two subtasks, speaker verification and identification. The process of verifying the claimed identity of a speaker on the basis of speech signals from a person is known as speaker verification. Speaker identification is the task in which a speaker's signal is compared with a set of known speaker signals. Previously, the i-vector method [1] was widely used as a speaker-recognition technique, where handcrafted features performed classification using methods such as probabilistic linear discriminant analysis (PLDA) [2] and heavy-tailed PLDA [3]. Handcrafted features are mostly FBANK and MFCC coefficients [4–6]. Since these handcrafted features are designed from perceptual evidence, they are lacking in many aspects and are unable to attain optimal performance for a variety of tasks in the speech domain.

Though speaker recognition is still a daunting task, remarkable performance improvements in speech domain [7–11] have been achieved in recent years by deep learning. In fact, when the i-vector framework was used in conjunction with deep neural networks (DNNs), there was performance improvement [12,13]. Convolutional neural networks (CNNs) proved to be effective in image-based

tasks such as image classification and recognition, and object detection due to their ability to automatically extract meaningful features. With few studies done in the direction of speaker verification using CNNs [14,15], current waveform-based DCNNs suffer in the first convolutional layer because of high-dimensional inputs [16]. They also suffer from the vanishing-gradient problem when the architecture is very deep [16].

In this regard, in speaker-recognition tasks, promising results were demonstrated by SincNet [16], which is a CNN-based architecture. The convolutional layer of SincNet consists of parameterized sinc functions to implement band-pass filters, and it is responsible for extracting low-level features from the waveform of an audio signal. This is followed by processing the extracted features by the deeper layers of the network. The low and high cutoff frequencies are the only parameters of the filters learned from data [16] at this stage. SincNet, due to its significantly smaller number of learnable parameters, converges much faster compared to conventional CNNs. This model uses a softmax function in its last layer to generate probability distribution over the training classes. However, softmax creates a decision boundary to distinguish samples from different classes, but it does not minimize intraclass distance [17,18].

Both speaker recognition (SR) and facial recognition (FR) are viewed an open-set problems because of their nature. An open-set problem is where there is no boundary point or limited class. Speaker and facial recognition does not have a limited number of class boundaries. Therefore, there is demand for specialized loss functions tailored to enforce this criterion. Classical softmax loss works well in optimizing a selection boundary that can distinguish classes. On the other hand, margin-based loss maximizes interclass distance and decreases intraclass variation, which is significant. This has motivated more studies on designing loss functions for FR tasks [17–20]. On the other hand, limited work has been done for SR tasks. AM-SincNet [21] was proposed, where the authors adopted additive margin softmax loss [17] from FR and integrated it with SincNet, showing 40% improvement in frame-error rate (FER) compared to the SincNet model trained with softmax loss. Margin-based loss has not only shown class variations, but has also proven to be robust.

In this paper, a family of models is proposed to serve as a potential successor to the SincNet model. The proposed models consist of angular-margin-based losses integrated with the original SincNet model, which showed performance improvements when compared to competitive baselines on multiple datasets. Interdataset evaluation was conducted, which showed that among all the proposed models, ALL-SincNet consistently outperformed the baselines and prior models. The contributions of this study can be summarized as follows:

- A family of models is proposed that utilizes angular-margin-based losses to improve the original SincNet architecture.
- Experimentally significant performance improvements are demonstrated in comparison to the performance of competitive baselines over a number of speaker-recognition datasets.
- Interdataset evaluation was performed, which demonstrated that one of the proposed models, ALL-SincNet, consistently outperformed the baselines and prior models.
- Cross-domain evaluation was performed on Bengali speaker recognition, which is considered a more diverse domain task, and it showed that ALL-SincNet could generalize reasonably well compared to the other baselines.

The remainder of the paper is organized as follows. The next two sections discuss related work and previously designed loss functions for SR and FR tasks. Sections 4 and 6 discuss the methodology of the proposed work and demonstrate the experiment results. We end with Sections 7 and 8, which summarize our study and outline future work directions.

2. Related Work

2.1. Speaker Recognition

For SR tasks, the i-vector [1] is used as a de facto feature-extraction method. Extracted features are then classified using PLDA [2] and heavy-tailed PLDA [3]. Though these methods achieve considerable results, there is still scope for improvement [22]. Recently, the use of low-level speech was investigated by researchers [23]. They considered under degraded conditions for speaker recognition, feature-extraction methods, and short-term features. Deep-learning methods demonstrated major advances in feature extraction and pattern recognition [24–31]. For example, in [25], the authors proposed a CNN-based method where the recording of a speaker was treated as an image. In [26], the authors proposed a CNN-based deep neural network for the speaker's short speech embedding. They used 5 s of speech for both sides of verification. A DNN was directly applied to time-frequency speech representation. In [29], the authors proposed to add a filter-bank layer within a CNN as an extra layer that learned jointly with the rest of the network to optimize cross-entropy loss. A filter-bank layer with multiple hidden layers has been proposed by the authors in [30] for spoofing detection. They showed that the filter bank produced cepstral coefficient features that distinguished between natural and synthetic speech more precisely than naive DNN features can, and manually designed cepstral coefficients. Before the output layer of the DNN, the authors of [32] proposed to add an L2 normalization layer followed by a scale layer, which normalizes the learned embeddings in an end-to-end fashion. The authors showed from the experiment that performance could be significantly improved by setting a proper value of scale parameter α. In [33], the authors proposed robust speaker embedding where embeddings are extracted without a nonlinear activation function. Towards deep speaker embedding, the authors from [34] proposed attentive statistical pooling for extracting features. The pooling layer calculated weighted standard deviation and weighted means over frame-level features that only focused on important frames. A recent trend is to learn directly from raw waveforms and completely avoid the feature-extraction step; this showed good results, including in speech-emotion recognition [35], speaker verification [36], and in spoofing detection with raw waveforms [37].

2.2. Loss

For some time, margin-based loss has been a popular research field in FR tasks. Previous works in open-set biometric-authentication tasks were done in the direction of FR [18–20]. Although SR tasks are similar to FR tasks, they have not received enough attention [21,38,39]. Additive margin softmax loss incorporated with a SincNet architecture was proposed in [21], and it demonstrated significantly improved performance compared to that of SincNet with softmax loss. On the other hand, a new ensemble additive margin softmax for speaker-verification motivation inspired by the work from [40] by the authors in [38] , where [38] the ensemble was the Hilbert–Schmidt independence criterion [41] with additive margin softmax loss. In [39], the authors introduce central loss and A-softmax loss for open-set speaker verification towards the more discriminative speaker embeddings where the frame-level feature extracted from a deep convolutional neural network.

To the best of our knowledge, this is the first study to show comprehensive analysis of different angular-margin-based loss functions. This is also the first study that performed interdataset and interlanguage evaluation in speaker recognition.

3. Loss Function

This section includes an indepth discussion of different loss functions used in our experiments.

3.1. Softmax Loss

Softmax loss is formulated by

$$L_{softmax} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{\phi}{\sum_{c=1}^{C} e^{\mathbf{W}_c^T \mathbf{f}_i}} \tag{1}$$

where

$$\phi = e^{\mathbf{W}_{y_i}^T f_i},$$

where $\mathbf{W}_c$ (c = 1,....,C; C is the number of classes) represents the weight vector of last fully connected layer, $\mathbf{f}_i$ denotes the feature input vector of the last fully connected layer corresponding to the original input x_i with the label y_i. N represents the number of training samples in a minibatch, and bias was set to 0. This can only penalize the classification error by increasing interclass discrepancy [17].

3.2. A-Softmax Loss

The inner product of Equation (1) can be factorized into $\|\mathbf{W}_c\| \, \|\mathbf{f}_i\| \cos(\theta_c)$. Equation (1) can be rewritten as

$$L_{softmax_{revised}} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{\|\mathbf{W}_{yi}\| \|\mathbf{f}_i\| \cos(\theta_{y_i})}}{\sum_{c=1}^{C} e^{\|\mathbf{W}_c\| \|\mathbf{f}_i\| \cos(\theta_c)}} \tag{2}$$

A-Softmax loss [18] was proposed, which is derived from Equation (2), and imposed to normalize the weight vector ($\|\mathbf{W}_c\| = 1$), modifying softmax loss to angular softmax loss by restoring $\|\mathbf{f}_i\| \cos(\theta_{y_i})$ with $\|\mathbf{f}_i\| \quad \phi(\theta_{y_i})$

$$L_{A-softmax} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{\phi}{\phi + \sum_{c=1, c\neq y_i}^{C} e^{\|\mathbf{f}_c\| \cos(\theta_c)}} \tag{3}$$

$$where \;\; \phi = e^{\|\mathbf{f}_i\| \Phi(\theta_{y_i})}$$

The authors of [18] proposed to define $\phi(\theta) = (-1)^k \cos(m\theta) - 2k, \; \theta \in [\frac{k\pi}{m}, \frac{(k+1)\pi}{m}]$ and $k \in [0, m-1]$ for removing the restriction, in which θ must be in the range of $[0, \frac{\pi}{m}]$.

3.3. AM-Softmax Loss

Additive margin softmax loss works as a better class separator than the original softmax and A-Softmax [17] do. Here, the authors proposed to introduce an additive margin to the original softmax loss's decision boundary. It can be derived from Equation (1) by adding margin to Equation (1). Both deep feature vectors $\mathbf{f}_i$ and weight vectors $\mathbf{W}_c$ are normalized in the implementation settings, and s is a hyperparameter for scaling the cosine values.

$$\begin{aligned} L_{AM-Softmax} &= -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{\alpha}}{e^{\alpha} + \sum_{c=1, c\neq y_i}^{C} e^{s\mathbf{W}_c^T \mathbf{f}_i}} \\ &= -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{\beta}}{e^{\beta} + \sum_{c=1, c\neq y_i}^{C} e^{s\cdot\cos(\theta_c)}} \end{aligned} \tag{4}$$

$$where \;\; \alpha = s \cdot (\mathbf{W}_{y_i}^T \mathbf{f}_i - m)$$

$$\beta = s \cdot (\cos\theta_{y_i} - m)$$

3.4. CosFace Loss

Equation (1) can be reformulated in a way in which the posterior probability only relies on the cosine of the angle between weights and input vectors by normalizing input vectors x_i and weights w_i. Normalized softmax (NS) loss is derived as

$$L_{NS} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{\cos(\theta_{y_i},i)}}{\sum_c e^{\cos(\theta_c,i)}} \tag{5}$$

where $\cos(\theta_{y_i}, i)$ is the output of normalised dot product of w_i and x_i. However, this normalized form of softmax is not sufficiently discriminative because NS loss only penalizes classification error [19]. To address this problem, cosine margin m in Equation (5) is introduced by the authors in [19], where θ_j is the angle between W_j and x_i. Equation (6) is the proposed large-margin cosine loss (LMCL) in [19]:

$$L_{CosFace} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{s(\cos(\theta_{y_i},i)-m)}}{e^{s(\cos(\theta_{y_i},i)-m)} + \sum_{c\neq y_i} e^{s\cos(\theta_c,i)}} \tag{6}$$

subject to

$$W = \frac{W^*}{\|W^*\|},$$
$$x = \frac{x^*}{\|x^*\|},$$
$$\cos(\theta_j, i) = {W_j}^T x_i$$

4. Method

4.1. SincNet

On the basis of CNNs, SincNet showed superior results and consistently performed better than conventional CNNs MFCC and FBANK [21] on SR tasks. The initial convolutional layer in SincNet is a set of parameterized sinc functions that implement band-pass filters, which are responsible of convolving the audio signal to vital low-level features. The convolutional operation is performed by a predefined function g that depends on learnable parameters θ. The only learned parameters of the filter are high and low cutoff frequencies [16]. Later, these features are processed by deeper layers of the architecture. SincNet enforces itself to emphasize only the filter parameters with prime effect on performance [16]. The convolution formula is shown as Equation (7):

$$y[n] = x[n] * g[n, \theta] \tag{7}$$

where $x[n]$ is a chunk of a speaker signal, $y[n]$ is the filtered output, and band-pass filter g is defined as

$$g[n, a_1, a_2] = 2a_2\frac{\sin(2\pi a_2 n)}{2\pi a_2 n} - 2a_1\frac{\sin(2\pi a_1 n)}{2\pi a_1 n} \tag{8}$$

where a_1 and a_2 are the learned low and high cutoff frequencies. Due to this filtering procedure with a sinc function, SincNet accurately reduces the number of parameters in the first convolutional layer compared to conventional CNNs [16]. Sinc functions were designed to handle digital signals such as audio and electroencephalograms (EEGs). So, using the sinc function helps the network to extract more meaningful features. If a standard CNN has F filters of L length, and if $F = 100$ and $L = 100$, then the CNN employs $F \cdot L = 10k$ parameters. SincNet employs $2 \cdot F = 0.2k$, which is significantly less than in a standard CNN. Moreover, sinc functions are symmetrical, so computational cost can be decreased by simply calculating half of the filters and turning them to the other side [21].

4.2. Proposed Architecture

Though the original SincNet showed promising results in SR tasks, it applies softmax loss to compute posterior probabilities over the selected speaker. Despite being a rational choice, it is not specifically capable of producing sharp divergence between classes in the final classification layer. To address that, this study proposes three different methods that can differentiate between classes more than traditional softmax loss can. Figure 1 illustrates the overall network architecture, which is originally from [16], where the model was modified by replacing softmax loss with our set of loss functions. This led to three models, namely, AF-SincNet, Ensemble-SincNet, and ALL-SincNet. The following sections discuss these different configurations.

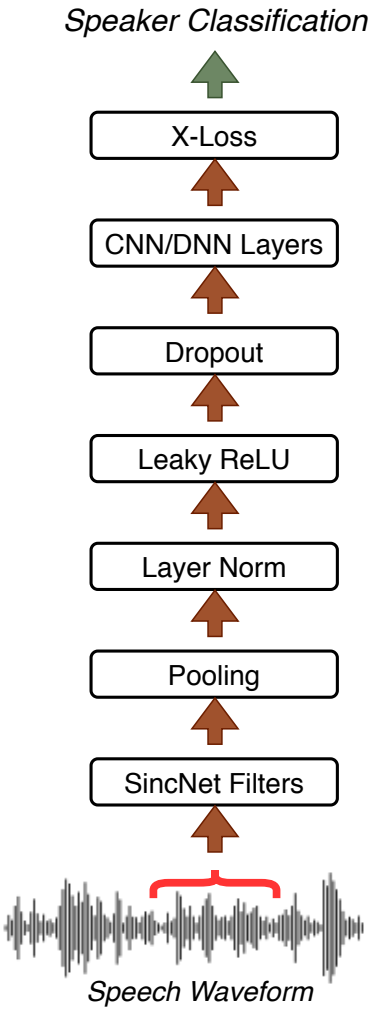

Figure 1. Visual representation of our model architecture. X-Loss illustrates loss functions used in experiments (e.g., AF-SincNet).

4.2.1. ArcFace Loss

The difference between ArcFace loss [20] and CosFace loss [19] is that CosFace loss uses the cosine margin, and ArcFace loss utilizes the arc-cosine function to calculate the angle between weights and vectors and add the additive margin to the target angle. ArcFace loss is currently the state-of-the-art

technique in FR-based systems [20]. The geodesic distance margin penalty is equal to the additive margin penalty in the normalized hypersphere [20].

$$L_{ArcFace} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{\alpha}{\alpha + \sum_{c=1,c\neq y_i}^{C} e^{s\cos(\theta_c,i)}} \quad (9)$$

$$where \ \ \alpha = e^{s(\cos(\theta_{y_i},i+m)}$$

This study uses ArcFace [20] as classification loss and integrates it with SincNet in the AF-SincNet model.

4.2.2. Ensemble Loss

Inspired from [20], this study also performed experiments where three separate loss terms were incorporated in a single framework. Loss terms ArcFace, CosFace, and A-Softmax were used and integrated with the SincNet classification layer. The formula was as in Equation (10):

$$L_{PS1} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{\alpha}{\alpha + \sum_{c=1,c\neq y_i}^{C} e^{s\cos(\theta_c)}} \quad (10)$$

where

$$\alpha = e^{s(\cos(m_1\theta_{y_i}+m_2)-m_3)}$$

Here, $m_1 = m$ from A-Softmax, $m_2 = m$ from Equation (9), and $m_3 = m$ from Equation (6).

4.2.3. Combination of Margin-Based Toss

An open-set task has no limitations of class label, which means that the open-set model has to be more robust in learning feature embedding in a latent space. Experience from FR tasks showed that embedding vectors found from training using an angular-margin-based loss function can become useful in open-set tasks. Utilizing this understanding, this study proposes an objective function that is a linear combination of the three loss functions described above, as shown in Equation (11):

$$L_{PS2} = L_{ArcFace} + L_{CosFace} + L_{A-softmax} \quad (11)$$

In essence, the proposed loss function is the joint optimization of the three different angular-margin-based loss functions. Through trial and error, we found that ascribing equal weights to each of the loss components performed well for speaker-recognition tasks.

5. Experiments

This section discusses the considered datasets, training and testing procedure, and employed metrics in this study.

5.1. Datasets

This study considered multiple datasets for training and evaluation purposes. The TIMIT [42] and LibriSpeeh [43] datasets were considered in this study. These two datasets are widely used in SR-related tasks, such as [16,21,44,45]. For the interlanguage test, the large Bengali ASR dataset [46] was used for evaluation only.

5.1.1. TIMIT

The TIMIT dataset has 462 classes, and the sample rate of each audio sampleis 16 kHz. There are eight samples for each class and the total number of sample is 3686. For training, five samples of each class were used, and the rest were used for testing. The zero signal (silence) was removed at the

beginning and end of each sentence as a preprocessing step by following the same procedures as those in [16]. The utterances for all speakers with the same text were removed.

5.1.2. LibriSpeech

This study considered exactly the same LibriSpeech data distribution as that in [16]. There are 2484 classes in total in LibriSpeech. The training and test materials were randomly sampled; 12–15 s of material was used for training, and 2–6 s in testing. The total number of samples was 21,933, where 14,481 were used as training data and the rest for testing. For both datasets, starting and end silence was removed.

5.1.3. Large Bengali ASR Dataset

Part of the large Bengali ASR Dataset [46] was used for interdataset testing of our model. The large Bengali ASR Dataset can be obtained from the OpenSLR site [47]. The dataset is mainly used for Bengali speech-recognition tasks, and, to the best of our knowledge, this study is the first to use it as a test set for a speaker-recognition task. The dataset is available in 16 segments, and this study only used a single segment (asr_bengali_0) for testing. Furthermore, this dataset was used only to test the proposed models in an interlanguage test setting to explore how the models performed in a true open-set biometric-recognition setting, so no data samples were involved in the training phase. For this experiment, the dataset was rearranged following the TIMIT test set, meaning that, from each class, only three samples were considered. A single sample from each class was registered, and the two other samples were used for testing. A total of 269 classes are available in asr_bengali_0.

5.2. Baselines

The proposed models were compared with several baselines. First, the recent SincNet [16] model was considered. This network uses classical softmax loss in training procedures. This was followed by AM-SincNet, which uses additive margin softmax loss [21]. For easy reference, Table 1 illustrates the model configurations and their references.

Table 1. Model-configuration details with reference to configuration names.

Model Name	Configuration
SincNet (baseline)	SincNet + softmax loss [16]
AM-SincNet (baseline)	SincNet + AM-Softmax loss [21]
AF-SincNet	SincNet + ArcFace loss (Section 4.2.1)
Ensemble-SincNet	SincNet + ensemble loss (Section 4.2.2)
ALL-SincNet	SincNet + combination of margin-based loss (Section 4.2.3)

5.3. Training and Testing Procedure

The raw waveforms were split into chunks of 200 ms with an overlap of 10 ms, similarly to previous work [16]. In SincNet, the first 3 layers are convolutional layers, where the first is a sinc-based convolutional layer, and the two following layers are typical convolutional layers. Sinc-based convolution used 80 filters of length $L = 251$. The later layers had 60 filters of length $L = 5$. Input samples and all convolutional layers were normalized [48]. Three fully connected layers composed of 2048 neurons were applied with batch normalization [49]. Leaky_ReLU [50] was used as the activation function of all hidden layers. This study followed the same protocol as that of [16] to initialize the model parameters. In order to train the models, RMSprop was used as the optimization algorithm. Minibatches of 128 were used with a learning rate of 10^{-2}, alpha $\alpha = 0.95$, and epsilon $\epsilon = 10^{-7}$.

Each configuration was trained on each dataset. During training, hyperparameters m and s were set to $m = 0.5$ and $s = 30$, similarly to [21]. For all our training configurations, $s = 30$ was considered,

but hyperparameter m was set by trying out different m values in the range of $0.3 \leq m \leq 0.75$ and using the optimal value. For Equation (9), $m = 0.5$. In Equation (10), $m_1 = 4$, $m_2 = 0.5$, and $m_3 = 0.35$. The same setting was used for Equation (11). For Equations (9)–(11), s was set to 30.

Two distinct kinds of evaluations were performed in this study. First, for speaker identification, the model was tested on the same distribution as that on which it was trained (e.g., test set of TIMIT, while the model was trained on the remaining set). For speaker verification, interdataset evaluation was performed. In this case, the model was tested on a different dataset/domain than the one on which it was trained (e.g., test set of TIMIT, while the model was trained on the training set of LibriSpeech). This was to demonstrate the ability to generalize on unseen data distributions. To further test the generalizability of the proposed models, interlanguage evaluation was performed using Bengali speech recognition, a more diverse task. As previously stated in Section 5.1, interlanguage evaluation was performed by using the large Bengali ASR dataset.

For this study, Ubuntu 18.04 with 16 GB RAM and RTX 2060 SUPER with 8 GB RAM were used. All codes were implemented using the PyTorch [51] framework. All the codes are available at github (https://github.com/jongli747/robust-dsr).

5.4. Metrics

For evaluation, frame-error rate (FER) was used in percentage; it is widely used in SR-based tasks [16,21]. Frame-level error was calculated at each 200 ms frame. This study also used classification-error rate (CER) in percentage [16]. CER is sentence-level classification error. Sentence-level error rate is computed by averaging posterior probabilities computed at each frame composing the sentence and voting for the speaker with the highest average probability. For interdataset and interlanguage evaluation, as mentioned in Section 5.3, the CER metric was used for evaluation.

6. Results

In this section, the proposed models' performance is discussed in comparison with the baselines. For TIMIT, speaker identification was evaluated, with 1386 test samples and 462 total classes; in the case of the LibriSpeech dataset, there were 7452 test samples of 2484 classes. First, the proposed models were compared with other baseline settings with two datasets on SR. Then, interdataset comparison for all models is performed. Lastly, to check the proposed model's language independence, the TIMIT trained model was tested with the Bengali dataset.

6.1. Intradataset Evaluation

FER and CER in percentage were used to compare our proposed models with the baselines. Table 2 shows that AF-SincNet outperformed all baselines on both datasets. FER represents the frame-level error rate, where frame size was 200 ms. In Table 2, the performance gap between AF-SincNet and baseline settings was particularly large in TIMIT. Section 5.1 showed that the number of training samples in the TIMIT dataset was much less than that in LibriSpeech, which showed the effectiveness of our proposed AF-SincNet model when the dataset is small. Ensemble-SincNet and ALL-SincNet only provided comparable results with those of the original SincNet on the TIMIT dataset.

Table 2. Frame-error rate (FER%) of speaker-identification systems. In both datasets, AF-SincNet outperformed the baseline settings.

Configuration	TIMIT ↓	LibriSpeech ↓
SincNet [16]	47.38	45.23
AM-SincNet [21]	28.09	44.73
AF-SincNet	**26.90**	**44.65**
Ensemble-SincNet	35.98	45.97
ALL-SincNet	36.08	45.92

Table 3 reports the achieved classification-error rate (CER%) on TIMIT and LibriSpeech datasets. AF-SincNet outperformed all baseline settings on the TIMIT dataset. In the LibriSpeech dataset, AF-SincNet produced comparable results. The two other proposed methods performed poorly in LibriSpeech, but in TIMIT, all proposed methods outperformed Sincnet with the softmax approach.

Table 3. Classification-error rate (CER%) of speaker-identification systems.

Configuration	CER on TIMIT ↓	CER on LibriSpeech ↓
SincNet [16]	1.08	**3.2**
AM-SincNet [21]	0.36	6.1
AF-SincNet	**0.28**	5.7
Ensemble-SincNet	0.79	7.2
ALL-SincNet	0.72	6.4

6.2. Interdataset Evaluation

To test the general effectiveness of the proposed model on different data distribution, the models underwent interdataset evaluation. Here, the model was tested on the LibriSpeech dataset, where the model was originally trained on the TIMIT dataset and vice versa. This study followed this experimental protocol for all configurations. The interdataset evaluation procedures were as follows.

- A single speaker's single speech was registered in our system.
- Cosine similarity was performed using Equation (12) with rest of the test set and identified with the highest similar score.

$$Similarity(A, B) = \frac{A \cdot B}{\|A\| \, \|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{12}$$

For instance, consider ALL-SincNet models trained on the TIMIT dataset. LibriSpeech's 2484 speaker's single speech (2484 samples in total) was registered on that model. Then, cosine similarity was performed with the rest of the LibriSpeech test set. For each registered sample, the rest of the LibriSpeech test set (4968 samples) was compared using Equation (12). In the above equation, the two vectors were A, B, and A_i, B_i, representing the feature-vector components of A and B. Target class was identified via the highest cosine-similarity score. The equation returned values in the range of 0 to 1, where 0 was totally dissimilar and 1 was exactly similar. Interdataset evaluation was performed for both datasets, such that, for TIMIT test data, the LibriSpeech dataset trained model was taken and registered using 462 speeches (TIMIT has 462 classes), and the rest of the procedure was the same as the previous one.

Table 4 reports the comparison of interdataset evaluation between proposed models and baselines on both datasets.

6.2.1. Trained on TIMIT and Tested on LibriSpeech

Table 4 shows that proposed method ALL-SincNet outperformed all other configurations. It also shows that, by introducing our proposed methods, performance gradually increased and CER was decreased. As stated above, 33% of the LibriSpeech test data were registered, and 66% data were used for the test. TIMIT's training set was much smaller than the LibriSpeech training set, so the training time of TIMIT was much less than that of LibriSpeech. So, less than 10% error was achieved with less time, though this came with a performance trade-off. Moreover, Table 3 shows that SincNet achieved 3.2% error on intradataset evaluation, and, shown in Table 4, SincNet achieved 10.09%, so there was performance inconsistency. Our proposed ALL-SincNet was much more consistent in terms of performance.

Table 4. Comparison of interdataset evaluation for both TIMIT and LibriSpeech.

TIMIT Trained LibriSpeech Test		**LibriSpeech Trained TIMIT Test**	
Configuration	**CER (%)**	**Configuration**	**CER (%)**
SincNet [16]	10.09%	SincNet	10.94%
AM-SincNet [21]	9.39%	AM-SincNet	13.10%
AF-SincNet	9.14%	AF-SincNet	10.83%
Ensemble-SincNet	8.10%	Ensemble-SincNet	12.87%
ALL-SincNet	**7.15**%	ALL-SincNet	**10.72**%

6.2.2. Trained on LibriSpeech and Tested on TIMIT

Table 4 shows that a model trained on Librispeech and tested on TIMIT, representing the proposed method, outperformed the baseline settings. ALL-SincNet achieved a smaller error rate of 10.72%. The registered data and test data distribution were the same as before. The main motivation of this study was to propose a generalized method that could distinguish different feature vectors in latent space. Our proposed ALL-SincNet, where we calculated different margin-based losses and optimized them, jointly achieved the best performance on the LibriSpeech-trained TIMIT test.

In both cases, our proposed methods outperformed the other systems. By introducing our proposed methods, our model generalized better than with the baseline settings.

6.2.3. Interdataset Test on Bengali ASR Dataset (Interlanguage Test)

The TIMIT-trained models were tested on the Bengali ASR dataset, as discussed in Section 5.1. For our experiments, a subset of the dataset [46] was used only for testing purposes to show our proposed model's generalizability. There were 269 unique speakers (or classes), from which three samples from each class were taken in a total of 807 samples. The testing procedure was the same as that in Section 6.2.

Table 5 shows the comparison of the proposed models and baseline settings, where the models were trained on the TIMIT dataset, which is an English speech dataset. Interdataset evaluation was performed with a more diverse dataset, Bengali ASR [46]. Similar patterns were found in which ALL-SincNet outperformed all configurations. AF-SincNet and Ensemble-SincNet performed better than the baselines. Our findings suggest that our proposed model performed reasonably well compared tothe baselines for interdataset testing of speaker-recognition tasks. By testing the Bengali dataset on an English-trained model, the effectiveness of our proposed model was found towards open-set biometric-recognition tasks because, in open-set recognition tasks such as speaker recognition, models are not tested with the same dataset distribution. Moreover, in open-set biometric-recognition tasks such as facial recognition [17–19], it is more important to distinguish feature embedding from different classes in latent space rather than the classification layer, and our proposed model did the exact same thing for speaker recognition. So, our proposed ALL-SincNet model is more robust in learning to discriminate high-dimensional features in latent space than in the baseline settings.

Table 5. Evaluation of Bengali ASR with TIMIT-trained (English) model.

Configuration	**CER (%)**
SincNet [16]	31.98%
AM-SincNet [21]	29.19%
AF-SincNet	28.07%
Ensemble-SincNet	28.44%
ALL-SincNet	**27.51%**

7. Discussion

In this paper, three different methods, namely, AF-SincNet, Ensemble-SincNet, and ALL-SincNet, were proposed. The proposed methods were compared with previous state-of-the-art methods SincNet [16] and AM-SincNet [21]. Tables 2 and 3 show the proposed method's performance on the TIMIT and LibriSpeech datasets. Tables 2 and 3 show that the performance of AF-SincNet fell within a single phase, which was CER in LibriSpeech test data. Figure 2 and 3 shows the comparison of FER over training epochs for both the datasets. Figure 2 shows that only the SincNet curve stopped converging at its optimal point. All curves were still converging towards the optimal point. So, the proposed methods can still be improved by training more epochs (we trained 1600 epochs). Table 4 shows that ALL-SincNet outperformed all other configurations, whereas Tables 2 and 3 show that it did not perform well, which implies that our proposed ALL-SincNet is more robust than the traditional SincNet and other settings are, and this has actual implications in open-set biometric-authentication tasks, e.g., speaker or facial recognition. Table 5 (pretrained model that trained with the TIMIT dataset and was tested on the Bengali ASR dataset) shows that ALL-SincNet also performed better than with other settings, which implies that our proposed ALL-SincNet is robust and helps to generalize the SincNet architecture towards interlanguage speaker-recognition tasks.

During our training, the original SincNet with softmax loss suffered from overfitting problems in small datasets, e.g., TIMIT. Our proposed model also mitigated that issue. Figure 4 shows that our proposed angular-margin-loss-based methods were less prone to overfitting than the original SincNet architecture.

Figure 4 shows that our proposed margin-loss-based models were less prone to overfitting than the original SincNet architecture. This also supports the findings of the interdataset evaluations in Table 4.

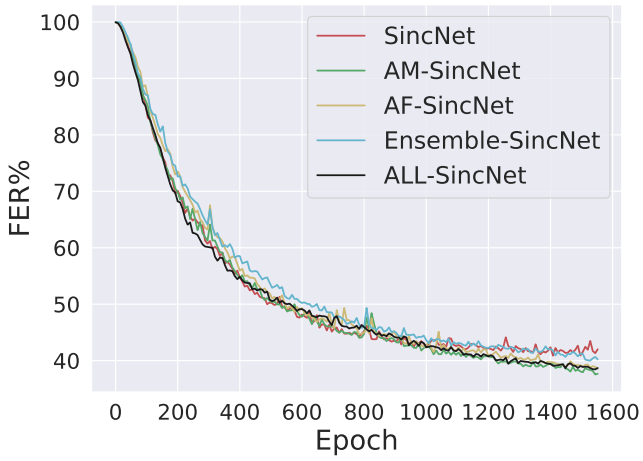

Figure 2. FER (%) comparison over training epochs on LibriSpeech dataset.

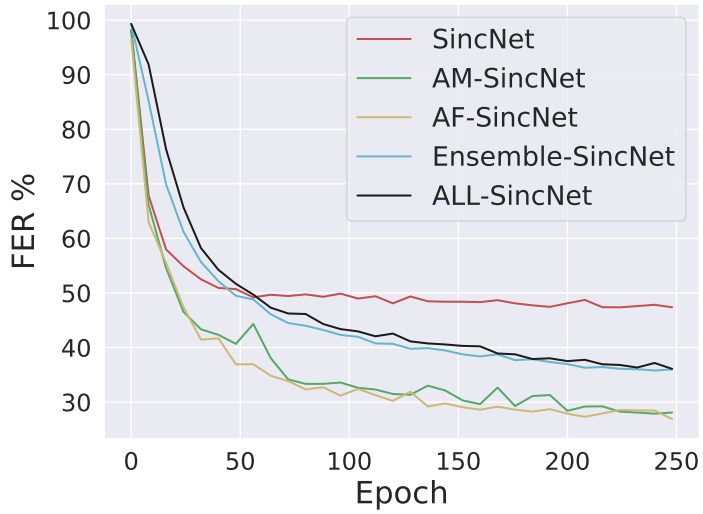

Figure 3. FER (%) comparison over training epochs on TIMIT dataset.

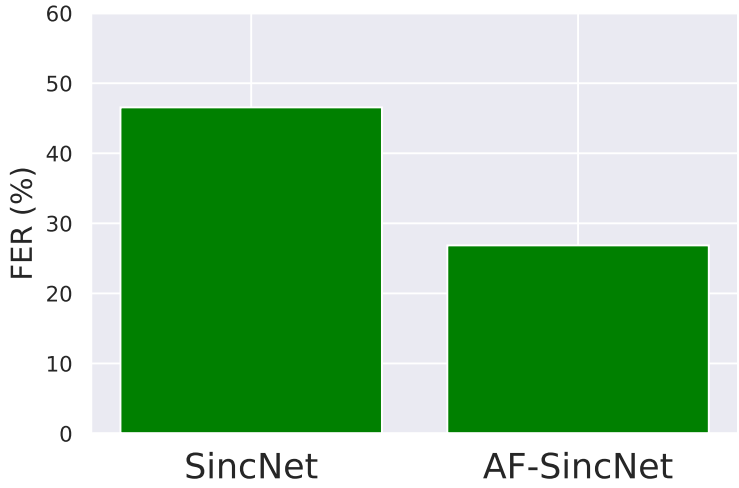

Figure 4. Comparison of overfitting problem between SincNet and proposed AF-SincNet.

8. Conclusions

There is some work in the field of speaker recognition using traditional handcrafted features and, more recently, deep-learning. SincNet, a deep-learning-based model, achieved state-of-the-art results in speaker-recognition tasks. It is a CNN-based architecture where the initial layer consists of a band-pass-filter-based convolutional layer. Similar to many models predicting probability distribution, the last SincNet layer uses a softmax function of which the output is optimized using cross-entropy loss. From other biometric systems, particularly facial recognition, it can be observed that loss functions leveraging angular margins have been successful. Inspired by such results, this study employed angular-margin-based loss functions, singly and jointly, on the SincNet architecture in a systematic manner.

This study proposed three different SincNet-based models: AF-SincNet, Ensemble-SincNet, and ALL-SincNet. The proposed models were evaluated with competitive baselines such as SincNet and AM-SincNet. All configurations were tested with different evaluation protocols, namely, intradataset, interdataset, and interlanguage evaluations. In the intradataset experiments, the AF-SincNet model performed better than with the other settings in terms of FER and CER metrics; in the interdataset and interlanguage settings, ALL-SincNet, which employed the proposed joint optimization of three angular-margin losses, performed the best overall. In fact, ALL-SincNet

outperformed other models for both interdataset evaluations performed on TIMIT and LibriSpeech. ALL-SincNet also outperformed other settings when trained on the English TIMIT dataset and evaluated on Bangla speaker-recognition data. Even though ALL-SincNet performed worse than the other configurations in intradataset evaluation, it consistently outperformed the other settings in interdataset and interlanguage evaluations. This suggests that the model was not overfitting the intradataset samples and was more robust than the baselines and other configurations, thus indicating the efficacy of jointly optimizing for the three angular margin-based loss functions for this type of task.

In future work, the proposed models will be tested in a noisy environment setting. We also plan to evaluate larger datasets such as VoxCeleb [26] and VoxCeleb2 [24], which contain millions of samples from more than 6000 speakers.

Author Contributions: conceptualization, L.C. and N.M.; methodology, L.C. and N.M.; software, L.C. and H.Z.; validation, L.C., H.Z.; formal analysis, L.C. and H.Z.; investigation, L.C. and N.M.; resources, L.C. and H.Z.; data curation, L.C.; writing—original-draft preparation, L.C., H.Z.; writing—review and editing, L.C., H.Z. and N.M.; visualization, L.C.; supervision, N.M.; project administration, N.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dehak, N.; Kenny, P.J.; Dehak, R.; Dumouchel, P.; Ouellet, P. Front-End Factor Analysis for Speaker Verification. *Trans. Audio Speech Lang. Proc.* **2011**, *19*, 788–798. [CrossRef]
2. Prince, S.J.D.; Elder, J.H. Probabilistic Linear Discriminant Analysis for Inferences About Identity. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–20 October 2007; pp. 1–8.
3. Matějka, P.; Glembek, O.; Castaldo, F.; Alam, M.J.; Plchot, O.; Kenny, P.; Burget, L.; Černocky, J. Full-covariance UBM and heavy-tailed PLDA in i-vector speaker verification. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 4828–4831.
4. Variani, E.; Lei, X.; McDermott, E.; Moreno, I.L.; Gonzalez-Dominguez, J. Deep neural networks for small footprint text-dependent speaker verification. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 4052–4056.
5. Richardson, F.; Reynolds, D.A.; Dehak, N. A Unified Deep Neural Network for Speaker and Language Recognition. *CoRR* **2015**, *abs/1504.00923*. Available online: http://xxx.lanl.gov/abs/1504.00923 (accessed on 20 April 2020).
6. Snyder, D.; Garcia-Romero, D.; Povey, D.; Khudanpur, S. Deep Neural Network Embeddings for Text-Independent Speaker Verification. *Proc. Interspeech* **2017**, 999–1003. [CrossRef]
7. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 20 March 2020).
8. Yu, D.; Deng, L. *Automatic Speech Recognition: A Deep Learning Approach*; Springer Publishing Company: New York City, NY, USA, 2014.
9. Dahl, G.E.; Yu, D.; Deng, L.; Acero, A. Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition. *IEEE Trans. Audio Speech Lang. Process.* **2012**, *20*, 30–42. [CrossRef]
10. Ravanelli, M. Deep learning for distant speech recognition. *arXiv* **2017**, arXiv:1712.06086.
11. Ravanelli, M.; Brakel, P.; Omologo, M.; Bengio, Y. A network of deep neural networks for distant speech recognition. In *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017*; IEEE: New York City, NY, USA, 2017; pp. 4880–4884.
12. Kenny, P.; Stafylakis, T.; Ouellet, P.; Gupta, V.; Alam, M.J. Deep Neural Networks for extracting Baum-Welch statistics for Speaker Recognition. In Proceedings of the Odyssey 2014, Joensuu, Finland, 16–19 June 2014; pp. 293–298.
13. Yaman, S.; Pelecanos, J.; Sarikaya, R. Bottleneck Features for Speaker Recognition. In Proceedings of the Odyssey 2012—The Speaker and Language Recognition Workshop, Singapore, 25–28 June 2012; Volume 12.

14. Salehghaffari, H. Speaker verification using convolutional neural networks. *arXiv* **2018**, arXiv:1803.05427.
15. Lukic, Y.; Vogt, C.; Dürr, O.; Stadelmann, T. Speaker identification and clustering using convolutional neural networks. In Proceedings of the 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP), Salerno, Italy, 13–16 September 2016; pp. 1–6.
16. 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; IEEE: New York City, NY, USA, 2018; pp. 1021–1028.
17. Wang, F.; Cheng, J.; Liu, W.; Liu, H. Additive margin softmax for face verification. *IEEE Signal Process. Lett.* **2018**, *25*, 926–930. [CrossRef]
18. Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; Song, L. Sphereface: Deep hypersphere embedding for face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 212–220.
19. Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J.; Li, Z.; Liu, W. Cosface: Large margin cosine loss for deep face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5265–5274.
20. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, NY, USA, 15–21 June 2019; pp. 4690–4699.
21. Nunes, J.A.C.; Macêdo, D.; Zanchettin, C. Additive margin sincnet for speaker recognition. In Proceedings of the 2019 IEEE International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–5.
22. Hansen, J.H.L.; Hasan, T. Speaker Recognition by Machines and Humans: A tutorial review. *IEEE Signal Process. Mag.* **2015**, *32*, 74–99. [CrossRef]
23. Dişken, G.; Tüfekçi, Z.; Saribulut, L.; Çevik, U. A Review on Feature Extraction for Speaker Recognition under Degraded Conditions. *IETE Tech. Rev.* **2017**, *34*, 321–332. [CrossRef]
24. Chung, J.S.; Nagrani, A.; Zisserman, A. VoxCeleb2: Deep Speaker Recognition. *arXiv* **2018**, arXiv:abs/1806.05622.
25. Bhattacharya, G.; Alam, M.J.; Kenny, P. Deep Speaker Embeddings for Short-Duration Speaker Verification. In Proceedings of the Interspeech 2017, Stockholm, Sweden, 20–24 August 2017.
26. Nagrani, A.; Chung, J.S.; Zisserman, A. VoxCeleb: A Large-Scale Speaker Identification Dataset. *arXiv* **2017**, arXiv:1706.08612.
27. Palaz, D.; Magimai-Doss, M.; Collobert, R. *Analysis of CNN-Based Speech Recognition System Using Raw Speech as Input*; Idiap Research Institute: Martigny, Switzerland, 2015.
28. Sainath, T.N.; Kingsbury, B.; Mohamed, A.; Ramabhadran, B. Learning filter banks within a deep neural network framework. In Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, 8–13 December 2013; pp. 297–302.
29. Yu, H.; Tan, Z.; Zhang, Y.; Ma, Z.; Guo, J. DNN Filter Bank Cepstral Coefficients for Spoofing Detection. *IEEE Access* **2017**, *5*, 4779–4787. [CrossRef]
30. Seki, H.; Yamamoto, K.; Nakagawa, S. A deep neural network integrated with filterbank learning for speech recognition. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 5480–5484.
31. 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.
32. Cai, W.; Chen, J.; Li, M. Analysis of Length Normalization in End-to-End Speaker Verification System. *arXiv* **2018**, arXiv:1806.03209.
33. Shon, S.; Tang, H.; Glass, J.R. Frame-Level Speaker Embeddings for Text-Independent Speaker Recognition and Analysis of End-to-End Model. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018; pp. 1007–1013.
34. Okabe, K.; Koshinaka, T.; Shinoda, K. Attentive Statistics Pooling for Deep Speaker Embedding. *arXiv* **2018**, arXiv:abs/1803.10963.

35. Trigeorgis, G.; Ringeval, F.; Brueckner, R.; Marchi, E.; Nicolaou, M.A.; Schuller, B.; Zafeiriou, S. Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 5200–5204.
36. Muckenhirn, H.; Magimai-Doss, M.; Marcel, S. Towards Directly Modeling Raw Speech Signal for Speaker Verification Using CNNS. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 4884–4888.
37. Dinkel, H.; Chen, N.; Qian, Y.; Yu, K. End-to-end spoofing detection with raw waveform CLDNNS. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 4860–4864.
38. Yu, Y.; Fan, L.; Li, W. Ensemble Additive Margin Softmax for Speaker Verification. In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 6046–6050.
39. Cai, W.; Chen, J.; Li, M. Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System. *arXiv* **2018**, arXiv:1804.05160.
40. Wang, X.; Zhang, S.; Lei, Z.; Liu, S.; Guo, X.; Li, S.Z. Ensemble Soft-Margin Softmax Loss for Image Classification. *arXiv* **2018**, arXiv:1805.03922.
41. Gretton, A.; Fukumizu, K.; Teo, C.-H.; Song, L.; Schölkopf, B.; Smola, A. A kernel statistical test of independence. In *Proceedings of the Advances in Neural Information Processing Systems 20—2007 Conference, Vancouver, BC, Canada, 3–6 December 2007*; Curran Associates Inc: New York City, NY, USA, 2009; pp. 1–8.
42. Garofolo, J.S.; Lamel, L.; Fisher, W.M.; Fiscus, J.G.; Pallett, D.S.; Dahlgren, N.L. DARPA TIMIT: Acoustic-phonetic continuous speech corpus CD-ROM, NIST speech disc 1-1.1. *STIN* **1993**, *93*, 27403.
43. Panayotov, V.; Chen, G.; Povey, D.; Khudanpur, S. Librispeech: An ASR corpus based on public domain audio books. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, Australia, 19–24 April 2015; pp. 5206–5210.
44. Li, J.; Zhang, X.; Jia, C.; Xu, J.; Zhang, L.; Wang, Y.; Ma, S.; Gao, W. Universal Adversarial Perturbations Generative Network For Speaker Recognition. In Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME), London, UK, 6–10 July 2020; pp. 1–6.
45. Tawara, N.; Ogawa, A.; Iwata, T.; Delcroix, M.; Ogawa, T. Frame-Level Phoneme-Invariant Speaker Embedding for Text-Independent Speaker Recognition on Extremely Short Utterances. In Proceedings of the ICASSP 2020—IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–9 May 2020; pp. 6799–6803.
46. Kjartansson, O.; Sarin, S.; Pipatsrisawat, K.; Jansche, M.; Ha, L. Crowd-Sourced Speech Corpora for Javanese, Sundanese, Sinhala, Nepali, and Bangladeshi Bengali. In Proceedings of the 6th International Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU), Gurugram, India, 29–31 August 2018; pp. 52–55.
47. Kjartansson, O.; Sarin, S.; Pipatsrisawat, K.; Jansche, M.; Ha, L. Large Bengali ASR Training Data Set. 2018. Available online: https://www.openslr.org/53/ (accessed on 15 January, 2020).
48. Ba, J.; Kiros, J.R.; Hinton, G.E. Layer Normalization. *arXiv* **2016**, arXiv:abs/1607.06450.
49. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:abs/1502.03167.
50. Maas, A.L. Rectifier Nonlinearities Improve Neural Network Acoustic Models. In Proceedings of the 30th International Conference on Machine Learning (ICML 2013), Atlanta, GA, USA, 16–21 June 2013.

51. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: New York City, NY, USA, 2019; pp. 8024–8035.