Max Margin Cosine Loss for Speaker Identification on Short Utterances

Ruifang Ji[1,2], *Junhua Cao*[3], *Xinyuan Cai*[1], *Bo Xu*[1]

[1]Institute of Automation, Chinese Academy of Sciences, China
[2]University of Chinese Academy of Sciences, China
[3]Chongqing Public Security Bureau

jiruifang2016@ia.ac.cn, cjh815@126.com, xinyuan.cai@ia.ac.cn, xubo@ia.ac.cn

Abstract

Speaker identification has made extraordinary progress owing to the advancement of deep neural networks. Speaker feature discrimination is a vital term in speaker recognition. However, the traditional softmax loss usually lacks the power of discrimination. To address this problem, this paper explores a novel loss function, namely max margin cosine loss (MMCL). To be specific, we realize the function by L2 normalizing both features and weight vectors in the softmax loss, together with a cosine margin term to maximize the decision margin in the angular space. In addition, max margin constraint, as one regularization term, is incorporated into the proposed loss function. Experimental results demonstrate the effectiveness of our proposed max margin cosine loss and superiority over pervious losses. For example, on 2s condition, MMCL reduces the equal error rate by 10.63% relatively compared to additive angular margin cosine loss (AMCL), while AMCL has already obtained 6.37% relative reduction than softmax loss.[1]

Index Terms: speaker identification, short utterances, softmax loss, max margin cosine loss

1. Introduction

Speaker identification aims to classify the identity of an unknown voice, based on the speakers‘ known utterances. When the content of utterances is lexically fixed, the task is considered as text-dependent speaker identification (TD-SI), otherwise it is text-independent speaker identification (TI-SI).

I-vectors systems has been dominant for years in the speaker recognition field[1, 2]. The framework consists of three stages [3]: Baum-Welch statistics collection, i-vector extraction, and classifying backend. For that the three subtasks in the system are loosely connected, many attempts have been focused on using an end-to-end architecture, and draw much attention [4, 5, 6]. In [7], d-vector system constructed with a DNN model was first proposed, and showed robustness over the i-vector system. In [5], long short-term memory (LSTM) [8] model obtained 30% more gain than the d-vector system on the “Ok, Google” benchmark. In [6], residual convolution neural network (ResCNN) model [9, 10] and stacked gated recurrent unit (GRU) architecture both reduced the verification equal error rate(EER) by over 50% (relatively) than i-vector system when training with 50k speakers. It seems that, the deep neural networks are promising.

The multilayer networks are finally followed by a classifying backend to produce similarity scores of different utterances, usually softmax loss. However, studies [5, 11, 6] found that this traditional loss was insufficient to acquire the discriminating power for classification. Thus, a number of loss functions, such as PLDA-based loss [3], contrast loss [11] and triplet loss [12], have been introduced to optimize models and achieved some improvements. All these losses share the same idea that: maximizing inter-class variance and minimizing intra-class variance.

In this paper, we present a simple but effective GRU structure that involves a convolutional layer and a GRU layer to learn speaker embedding. Then, the whole system is optimized by using the proposed loss function, called max margin cosine loss (MMCL). Max margin cosine loss takes ideas from additive angular margin cosine loss (AMCL) in the face recognition field [13] and further optimized. As the cosine of the angular has intrinsic consistency with softmax [14, 15], we L2 normalize both features and weight vectors in the softmax loss, and introduce a cosine margin term to maximize the decision margin. Specifically, max margin constraint, as one regularization term, is incorporated into the proposed loss function. The motivation behind is that the posterior probability of the true class should be larger than a threshold, and the posterior probabilities of all the false classes should be smaller than the threshold. In the training process, batch normalization and network reduction [16] are adopted to handle the training difficulty. We test our proposed system on speaker identification task with a Short Duration Corpus. Experiments show that GRU architecture that involves a convolutional layer learns more speaker information, and the max margin cosine loss significantly improves the discriminative ability of our system.

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 presents the GRU structure and max margin cosine loss. The performance of our proposed system is evaluated, and the results are discussed in Section 4. Section 5 gives a conclusion of this paper.

2. Related work

Recurrent neural network (RNNs) and CNNs have been applied to speech recognition with good performance [17, 18, 19]. CNNs are effective in reducing spectral variations and modeling spectral correlations in acoustic features [17], while RNNs performs well in capturing the sequential nature of audio signals[6]. In order to make full use of both models advantages, we employ a GRU structure that consists of a convolutional layer and a gru layer to learn speaker embeddings. In addition, instead of averaging the multiple outputs, we only select the last output of the GRU as the utterance-level feature, to maintain one‘s speaking style.

Many classifying backend loss functions, such as center loss [15], contrast loss [11] and triplet loss [12], have been proposed to optimize models. Among them, center loss adds extra penalty to shrink intra-variance, but it has to be combined with softmax in the training process. Contrast loss is composed of

[1] This work is supported by National Key Research and Development Program of China under Grant No.2016YFB1001404.

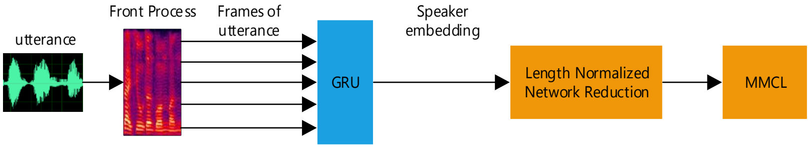

Figure 1: *Architecture of Our Proposed Speaker Identification System*

positive pairs and negative pairs, and focuses on difficult pairs of embeddings and negative centroid. Triplet loss manages to minimize the distance between an anchor and the positive sample, while maximizing the distance from the anchor and a negative one. Nevertheless, contrast loss and triplet loss are not easily to train, due to the selection of effective training samples. In our work, we learn from AMCL that puts an angular margin penalty on the classification boundary, and further extend with a max margin constraint. Its effectiveness is demonstrated in the experimental results.

3. Model and Approach

Fig.1 presents the architecture of our proposed system. Raw utterances are first processed to frame-level features with the detail steps in Section 4.2. Then, features are input to the GRU model to learn robust speaker embedding, as described in Section 3.1. Finally, the max margin cosine loss layer estimates the identity of the speaker embeddings, as explained in Section 3.2.

3.1. GRU Network

We carry out experiments with recurrent networks on speaker embeddings learning, because they have worked well for speech recognition [20]. As a variant of Long Short-Term Memory (LSTM), GRU has simpler structure and less parameters [19]. Previous experiments showed that GRU and LSTM achieved similar accuracy with the same number of parameters, but GRU was faster to train and more likely to diverge [19].

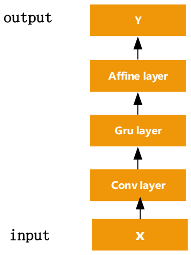

Figure 2: *Structure of GRU Network*

The structure of GRU network is showed in Figure 2, and Table 1 presents the network hyperparameters. To model spectral correlations in acoustic features, we add a 5×5 filter size, 2×2 stride convolutional layer in front of the gru layer. However, this layer can also make training faster, for it helps reducing the dimensionality of features in both time and frequency domains, which is verified in our experiment. Then, the features are fed into a unidirectional GRU layer with 1024 cells. Since that audios temporality contains one's speaking style information, we only pick the last output of the GRU to be the utterance-level feature, rather than averaging the outputs. Finally, the utterance-level features are reshaped by an affine layer of 512-dimension to the low-demensional speaker embeddings. Besides, to reduce internal covariate shift, we adopt sequence-wise batch normalization (BN) and clipped-ReLU activation [19] in the model.

Table 1: *Network Hyperparameters of GRU Network. bs denotes batch size and fn denotes frame number.*

layer name	struct	stride	dim
input	-	-	bs × fn × 64 × 3
conv16-s	5 × 5,16	2 × 2	2048
GRU	1024 cells	1	1024
affine	1024 × 3072	-	3072
ln	-	-	512
total	-	-	-

3.2. Max Margin Cosine Loss

3.2.1. Softmax

Softmax loss is the most widely used classifying backend loss function. It separates features into different classes by maximizing the posterior possibility of its belonged class. Given an input feature vector x_i and its label y_i, the loss is defined as::

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{f_{y_i}}}{\sum_{j=1}^{C} e^{f_j}} \tag{1}$$

where N is the batch size, and C is the class number; f_{y_i} is the output of the last fully-connected layer for x_i , which is formulated as:

$$f_{y_i} = W_{y_i}^T x_i + b_{y_i} = ||W_{y_i}|| \cdot ||x_i|| \cos\theta_{y_i} + b_{y_i} \tag{2}$$

where W_{y_i} and b_{y_i} is the weight vector and bias of the fully-connected layer, and θ_{y_i} is the angle between W_{y_i} and b_{y_i}.

3.2.2. Normalisation

For simplicity, the bias b_{y_i} is fixed to zero[21]. Then, the prediction f_{y_i} is denoted as:

$$f_{y_i} = W_{y_i}^T x_i + b_{y_i} = ||W_{y_i}|| \cdot ||x_i|| \cos\theta_{y_i} \tag{3}$$

To make feature learning more effectively, $||W_{y_i}||$ is set to 1 by L2 normalisation[22, 23]. Then, only the norm of the input vector x_i and the angle θ_{y_i} contributes to the prediction vector f_{y_i}:

$$f_{y_i} = ||x_i|| \cos\theta_{y_i} \tag{4}$$

In the field of face recognition, L2 normalisation on feature is informative of the quality of face, and significantly boosts the performance of face verification [24, 22]. The assumption behind is to remove the radial variations on a hypersphere mainfold. In real scenarios, ones utterances may also captured in various conditions, like various emotional states and diverse channels. Thus, the norms of his utterances feature embeddings vary largely. However, The task of speaker identification is to minimize intra-class variance, so L2 normalisation on feature can also be adopted. Fixing $||x_i||$ to s, the loss function is formulated as:

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot\cos\theta_{y_i}}}{\sum_{j=1}^{C}e^{s\cdot\cos\theta_j}} \quad (5)$$

3.2.3. Additive Angular Margin Loss

For that softmsx loss emphasizes on correct classification, thinking little of the penalty on misclassification. To handle this problem, additive angular margin loss [13] adds an angular margin within θ_{y_i} to put margin constraint on the classification boundary. The function is defined as:

$$L_3 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot(\cos(\theta_{y_i}+m))}}{e^{s\cdot(\cos(\theta_{y_i}+m))}+\sum_{j=1,j\neq y_i}^{C}e^{s\cdot\cos\theta_j}} \quad (6)$$

When θ_{y_i} is in the scope of $[0, \pi - m]$, $\cos(\theta_{y_i} + m)$ is smaller than $\cos\theta_{y_i}$. So the inter-class variance is enlarged while the intra-class variance lessens.

3.2.4. Max Margin Constraint

Faced with the identification task, our goal is to decide which identity the unknown utterances belong to. Commonly, for the feature vector x_i, we calculate its posterior possibility on all classes f_j. We assume that the posterior probability f_{y_i} of the true class should be larger than a threshold t, and the posterior probabilities $f_{j(j\neq y_i)}$ of all the false classes should be smaller than the threshold t. This can be described as:

$$f_{y_i} \geq t \geq \max(f_{j(j\neq y_i)}) \quad (7)$$

In this constraint, when the posterior possibility f_{y_i} of x_i belonged class is larger than the threshold t, this classification task perfectly completes. However, when f_{y_i} is smaller than t, misclassification occurs, and we define the loss to be their difference value. We represent the loss L_+ in this condition as:

$$L_+ = \begin{cases} 0 & , \; if \; f_{y_i} \geq t \\ t - f_{y_i} & , \; else \end{cases} \quad (8)$$

For brevity's sake, the function of L_+ is simplified as:

$$L_+ = \max(t - f_{y_i}, 0) \quad (9)$$

Similarly, when the posterior possibility $f_{j(j\neq y_i)}$ of x_i not belonged class is smaller than the threshold t, this sample is classified correctly. when $f_{j(j\neq y_i)}$ is lager than t , loss arises, and we also set it to be their difference value. L_- is defined as:

$$L_- = \max(f_{y_i} - t, 0) \quad (10)$$

To sum up L_+ and L_- , we introduce a function δ_y:

$$\delta_y = \begin{cases} 1 & , \; if \; y = f_{y_i} \\ -1 & , \; else \; y = f_{j(j\neq y_i)} \end{cases} \quad (11)$$

Thus, for the the feature vector x_i, its constraint loss C_{x_i} is:

$$C_{x_i} = \max(\delta_y \cdot (t - f_y), 0) \quad (12)$$

For the whole samples, the constraint loss is:

$$C_{\max\_mar} = \frac{1}{N}\sum_{y=1}^{C}\max(\delta_y \cdot (t - f_y), 0) \quad (13)$$

In summary, the max margin cosine loss is defined as follows:

$$\begin{aligned} & s, t, (W_{y_i})_{i=1}^{N} = \arg\max L \\ & \text{subject to} \quad ||W_{y_i}||_2^2 = 1, i = 1, ..., N. \end{aligned} \quad (14)$$

where L is the weighted sum of L_3 and $C_{\max\_mar}$, and formulated as:

$$L = L_3 + \lambda C_{\max\_marg} \quad (15)$$

4. Experiments

In this section, we first describe the database used in the experiments, then explain the front process, and finally report the results. All the experiments are conducted with the Keras toolkit [25].

4.1. Data Sets

The corpus used for the network training and evaluation, is mainly collected from three different channels, i.e., Interview, Android mobilephone, and Apple mobilephone. It comprises 968 speakers, 35,983 utterances, about 27 hours utterances in Mandarin. Each person has about 37 utterances, the durations of which are mostly around 2 to 5 seconds. The corpus is split into training and evaluation by randomly selecting 100 speakers to be the evaluating set. The details of the dataset are shown in Table 3.

Table 2: *Corpus statistics*

	#spk	#utt	#utt/spk	dur/utt
Training	868	32,322	37.2	2.74s
Eva	100	3,661	36.61	2.72s
Total	968	35,983	37.17	2.74s

4.2. Basic Setup

For our proposed system, raw audio is first converted to 64-dimensional log mel-filter bank (Fbank) coefficients [2] with a frame-length of 25ms. This raw feature is augmented by its first and second order derivatives, and further made frame-level energy-based Voice Activity Detector (VAD) selection. Then, the feature goes through the GRU model described in Section 3.1, with the output input to the max margin cosine loss described in Section 3.2. To observe the performance of the convolutional layer in the GRU network, we build a structure without convolutional layer for comparison, called $\mathrm{GRU\_no\_conv}$ system. In addition, we estimate the system performance with MMCL, AACL, and softmax loss, respectively.

In the training stage, we use ADAM [26], with a linear decreasing learning rate from 0.05 to 0.001. Besides, to relieve the training difficulty, embeddings are normalized before sending to the loss function, and fifty percent of network reduction is adopted.

4.3. $\mathrm{GRU\_conv}$ vs. $\mathrm{GRU\_no\_conv}$

$\mathrm{GRU\_conv}$ is the GRU structure that consists a convolutional layer and a gru layer, while $\mathrm{GRU\_no\_conv}$ is the GRU structure without convolutional layer. To investigate the impact of the convolutional layer, we carry out experiments with the two different structures with softmax loss on 2s condition, and the results are showed in Fig.3. It is shown that $\mathrm{GRU\_conv}$ reaches convergence at around 25 epochs with the EER of 6.43%, while $\mathrm{GRU\_no\_conv}$ achieves convergence until 50 epochs with the

EER of 7.56%. It indicates that GRU_conv structure goes faster in learning feature with better performance. This verifies the assumption that the convolutional layer contributes to faster network computation, as well as capturing more acoustic information. In the following experiments, we employ the GRU_conv structure.

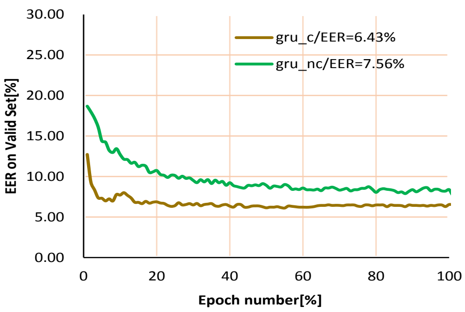

Figure 3: *The speaker identification performance across epochs on 2s condition.* gru_c *denotes the GRU structure with convolutional layer,* gru_nc *denotes the GRU structure without convolutional layer.*

4.4. MMCL vs. AMCL vs. sof

In Fig.4 we report the primary results of our GRU structure with three different classifying backends on 2s condition. AMCL reduces the equal error rate by 6.37% relatively compared to the softmax loss baseline, and MMCL further brings 10.63% relative reduction. This indicates that the additive angular margin helps enlarge the distance between embeddings from different speakers while shrinking the intra-class variance. The max margin constraint pays attention to reducing the general classification error, which makes further performance promotion. In addition, the value of s and m is a term we can not ignore. In our work, when the value of s , m, t and λ are set to 1, 0.5, 0.4, 10, respectively, the system with MMCL achieves its best performance.

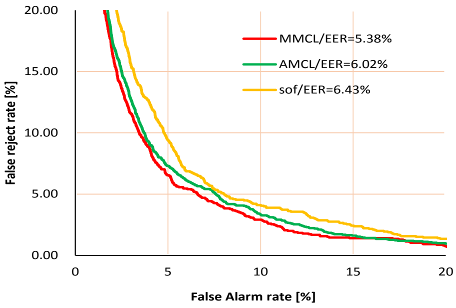

Figure 4: *DET plots on different classifying backends on 2s condition.* sof *denotes softmax loss.*

4.5. Performance by Number of Enrolled Utterances

In the speaker identification, there is always a vital step that builds the speaker model with enrolled utterances. In our work, we average embeddings across one's enrollment utterances to make his final representation. Take into account the effect of enrolled utterances number, we made experiments and the results are present in Fig.5. As can be seen, a performance degradation is observed when we reduce the enrolled number from 5 to 2. In terms of EER, the amounts of relative degradation are 12.45% and 19.34% when the number of enrolled utterances decreases from 5 to 3, and 3 to 2. The observation reminds us that the number of enrolled utterances is an essential item in designing speaker verification systems, and 3 to 5 utterances may be appropriate.

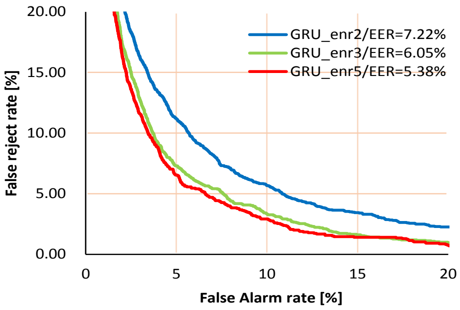

Figure 5: *DET plots on different enrolled utterances number. Models are trained with the max margin cosine loss. $enr*$ denotes that the model is trained with $*$ utterances.*

4.6. Performance against Different Duration

The system is further applied to different duration conditions, and the performances are indicated in Table 5. More specifically, 2s, 3s, 5s, and 8s conditions are tested with our GRU framework. Sharp increasements are witnessed when the duration grows from 2s to 3s, 3s to 5s, and 5s to 8s, whose relative improvements are 28.44%, 29.35%, and 20.96%. Mainly because that longer utterances contain more speaking style information and spectral correlations, which benefits the discriminative speaker embeddings learning. The observation shows guidance to us that 5s is enough in training short-duration speaker identification system. Plus, the longer duration may bring further improvement as well as redundancy. It is also found that, network training with MMCL works better than AAML, and much greater than softmax loss in the four duration periods, further confirming the experment in Section 4.4.

Table 3: *System performance on different duration conditions*

loss	2s	3s	5s	8s
sof	0.0643	0.0437	0.0363	0.0301
AMCL	0.0602	0.0410	0.0307	0.0254
MMCL	0.0538	0.0385	0.0272	**0.0215**

5. Conclusions

In this study, we present a simple but effective text-independent speaker identification system with a novel classifying backend. GRU allows us to learn the sequential nature of audio signals, and the front convolutional layer captures more acoustic informations as well as accelerating computation. Max margin cosine loss makes it possible to optimize the entire system. Experiments show that our GRU structure achieves consistently better performance than the model without the front convolutional layer, and the max margin cosine loss makes the embeddings more discriminative.

It is believed that our system can be extended to related areas, such as speaker verification, diarization and clustering. Of course, there are still some improvements to make through different loss functions, front feature processing, neural networks, etc. All in all, this system shows both decent advancements and a direction where our further research goes forward.

6. References

[1] P. Kenny, "Bayesian speaker verification with heavy-tailed priors." in *Odyssey*, 2010, p. 14.

[2] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[3] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Proc. Interspeech 2017*, pp. 999–1003, 2017.

[4] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 165–170.

[5] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," *CoRR*, vol. abs/1509.08062, 2015. [Online]. Available: http://arxiv.org/abs/1509.08062

[6] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint arXiv:1705.02304*, 2017.

[7] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4052–4056.

[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[9] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," *arXiv preprint arXiv:1710.10467*, 2017.

[12] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *Proc. of Interspeech*, 2017.

[13] H. Wang, Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," *arXiv preprint arXiv:1801.09414*, 2018.

[14] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 539–546.

[15] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European Conference on Computer Vision*. Springer, 2016, pp. 499–515.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, 2015, pp. 448–456.

[17] Y. Zhang, M. Pezeshki, P. Brakel, S. Zhang, C. L. Y. Bengio, and A. Courville, "Towards end-to-end speech recognition with deep convolutional neural networks," *arXiv preprint arXiv:1701.02720*, 2017.

[18] Y. Wang, X. Deng, S. Pu, and Z. Huang, "Residual convolutional ctc networks for automatic speech recognition," *arXiv preprint arXiv:1702.07793*, 2017.

[19] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[20] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform cldnns," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[21] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2017.

[22] J. Deng, J. Guo, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," *arXiv preprint arXiv:1801.07698*, 2018.

[23] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "Normface: l_2 hypersphere embedding for face verification," *arXiv preprint arXiv:1704.06369*, 2017.

[24] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," *arXiv preprint arXiv:1703.09507*, 2017.

[25] F. Chollet *et al.*, "Keras," 2015.

[26] D. R. Wilson and T. R. Martinez, "The general inefficiency of batch training for gradient descent learning," *Neural Networks*, vol. 16, no. 10, pp. 1429–1451, 2003.