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

Adaptive Margin Circle Loss for Speaker Verification

Runqiu Xiao[1,2], *Xiaoxiao Miao*[1,2], *Wenchao Wang*[1,2], *Pengyuan Zhang*[1,2], *Bin Cai*[3], *Liuping Luo*[3]

[1] Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Guangdong Provincial Public Security Department, China

{xiaorunqiu,miaoxiaoxiao,wangwenchao,zhangpengyuan}@hccl.ioa.ac.cn
jiexiren@gmail.com,jclp1122@163.com

Abstract

Deep-Neural-Network (DNN) based speaker verification systems use the angular softmax loss with margin penalties to enhance the intra-class compactness of speaker embeddings, which achieved remarkable performance. In this paper, we propose a novel angular loss function called adaptive margin circle loss for speaker verification. The stage-based margin and chunk-based margin are applied to improve the angular discrimination of circle loss on the training set. The analysis on gradients shows that, compared with the previous angular loss like Additive Margin Softmax(Am-Softmax), circle loss has flexible optimization and definite convergence status. Experiments are carried out on the Voxceleb and SITW. By applying adaptive margin circle loss, our best system achieves 1.31%EER on Voxceleb1 and 2.13% on SITW core-core.

Index Terms: speaker verification, speaker embedding, circle loss, adaptive margin

1. Introduction

The modern Deep Neural Network (DNN) based speaker verification systems have achieved remarkable performances than the traditional i-vector with Probabilistic Linear Discriminant Analysis(PLDA) systems [1, 2, 3]. It generally consists of three parts during the training phase, the DNN model for segment-level speaker feature extraction, the pooling layer for statistics extraction, and the loss function for classifying [3, 4]. In recent years, more and more advanced model architectures are proposed for the improvement in ASV performance. Such as the extensions of the basic TDNN [5] structure like TDNN-F [6], ECAPA-TDNN [7], e.g., the primary Residual Network (ResNet) [8], and their subsequent like Res2Net, ResNeXt [9], e.g.

In speaker verification, the cross-entropy loss function with softmax is most widely used for training the speaker embedding model. However, the previous work shows that it is more suitable for classification because it only learns features that are not discriminative enough [10], which causes larger generalization errors for unseen speakers. To address this issue, Contrastive Loss [11] and Triplet Loss [12, 13] were first presented to directly optimize the similarity between the speaker embeddings so that the distance of the intra-class embeddings is smaller than the inter-class embeddings over a threshold. Though they performed well by selecting appropriate training samples, the number of pairs or triplets increases explosively with the number of training samples, and the performance depends strongly on the strategy to search effective pairs or triplets.

On the other hand, some angular-based losses are proposed to boost the discriminative power of face representations in the face recognition field, including SphereFace [10], Am-Softmax [14], and Arc-Softmax [15], which are also resultful for ASV task [16, 17, 18, 19]. To ensure that the embeddings are more distinguishable in the angular direction, they proposed to normalize the weights and features of the classifier [20] and add a margin to tighten the decision boundary. However, the angular-based losses have two drawbacks:(1) after normalization, the network will pay more attention to the low-quality samples [14] and may amplify the impact of noisy samples. (2) The performance relys on the super-parameter, which needed to be obtained through brute force search. To address this, [18, 19] propose to set the super-parameter according to the cosine angle dynamically. [21] suggest the Sub-center ArcFace for decreasing the influence from noise.

In this paper, we introduce a novel angular-based losses called adaptive margin circle loss [22] for speaker verification. Analysis on gradients shows that circle loss has flexible optimization and definite convergence status when compared with the other angular-based losses. Then we investigate stage-based and chunk-based strategies to generate adaptive margin, which can enhance the intra-class compactness of speaker embeddings. Experiments on VoxCeleb and SITW show that circle loss achieves better performance than the original softmax loss and the common angular-based losses, Am-Softmax and Arc-Softmax.

2. From original softmax to angular softmax loss

2.1. Softmax Loss

First, we give a brief review of the original softmax loss. The widely used softmax loss is defined as:

$$\begin{aligned} L_s &= -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\mathbf{w}_{y_i}^T\mathbf{x}_i+b_{y_i}}}{\sum_{j=1}^{C}e^{\mathbf{w}_j^T\mathbf{x}_i+b_j}} \\ &= -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\|\mathbf{w}_{y_i}\|\cdot\|\mathbf{x}_i\|\cdot cos\theta_{y_i}}}{\sum_{j=1}^{C}e^{\|\mathbf{w}_j\|\cdot\|\mathbf{x}_i\|\cdot cos\theta_j}} \end{aligned} \tag{1}$$

where C and N is the number of speakers and samples in the mini batch, respectively. $\mathbf{x}_i$ is the input of the last classify layer and $\mathbf{w}_j$ is the j-th column of the weights in the classify layer, $\mathbf{y}_i$ is the ground truth label for the i-th sample. For convenience, we omit the bias b_j and the logit $\mathbf{w}_{y_i}^T\mathbf{x}_i$ is equivalent to $\|\mathbf{w}_{y_i}\|\,\|\mathbf{x}_i\|\,cos\theta_{y_i}$, where θ_{y_i} is the angle between $\mathbf{w}_{y_i}$ and $\mathbf{x}_i$. In order to reduce L_s, the network tends to:

- Increase the weight norm $\|\mathbf{w}_{y_i}\|$. So the more training samples in the i-th class, the larger the corresponding weight norm tends to be [10].

http://dx.doi.org/10.21437/Interspeech.2021-1043

- Increase the feature norm $\|\mathbf{x}_i\|$, making the simple samples have greater feature norm [20].
- Decrease θ_{y_i}. Assuming that θ_{y_i} belongs to $[0, \frac{\pi}{2}]$.

However, the weight norm $\|\mathbf{w}_{y_i}\|$ and the feature norm $\|\mathbf{x}_i\|$ are generally useless in the open-set recognition problem. So the authors [14, 15] proposed to normalize the weight and feature vectors(making $\|\mathbf{w}_{y_i}\|$=$\|\mathbf{x}_i\|$=1), ensuring that the embeddings $\mathbf{x}_i$ are more distinguishable in the angular direction.

2.2. Angular Softmax Loss

The general formula of the Angular Softmax Loss function can be summarized as:

$$L_{as} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s\cdot\psi(\theta_{y_i})}}{e^{s\cdot\psi(\theta_{y_i})}+\sum_{j=1,j\neq i}^{C}e^{s\cdot cos(\theta_j)}} \quad (2)$$
$$\psi(\theta_{y_i}) = cos(m_1\theta_{y_i}+m_2)-m_3$$

where s is the scale factor that makes the lower bound of L_{as} close to 0. The $m1$, $m2$ and $m3$ are used to tighten the decision boundary. When $m1$, $m2$, and $m3$ are used individually, the losses are denoted as angular softmax (A-Softmax), additive angular margin softmax (Arc-Softmax), and additive margin softmax loss (Am-Softmax), respectively.

Without loss of generality, we analyze the gradients of Am-Softmax under the toy scenario where there are only a single s_p and s_n:

$$L_{am-s} = -log\frac{e^{s\cdot(s_p-m)}}{e^{s\cdot(s_p-m)}+(C-1)e^{s\cdot s_n}} \quad (3)$$

where $s_p = cos\theta_{y_i}$ and $(C-1)e^{s\cdot s_n} = \sum_{j=1,j\neq i}^{C} e^{s\cdot cos(\theta_j)}$. s_p and s_n refer to the positive pairs similarity and negative pairs similarity, notice that generally, both s_p and s_n belong to $[0, 1]$. The gradients of L_{am-s} with respect to s_p and s_n are derived as follows:

$$\frac{\partial L}{\partial s_p} = (1-\frac{e^{s\cdot(s_p-s_n-m)}}{e^{s\cdot(s_p-s_n-m)}+(C-1)})\cdot s$$
$$\frac{\partial L}{\partial s_n} = (\frac{(C-1)}{e^{s\cdot(s_p-s_n-m)}+(C-1)})\cdot s \quad (4)$$

As shown in Figure 1(a), the gradients with both s_p, s_n are the same to each other and only depend on $(s_p - s_n)$. For some noisy training samples, if s_p is small and s_n already close to 0(such as A(0.2, 0.2)), both s_p and s_n still get a large gradient, which means that the loss function keeps on penalizing s_n with a large gradient though s_n has reach optimum. So angular softmax loss will amplify the impact of noise samples.

3. Adaptive Margin Circle Loss

3.1. Circle loss

In [22], the author provided a self-paced weighting strategy to enhance the optimization flexibility. The proposed Circle loss is defined as

$$L_{circle} = -\log\frac{e^{s\cdot\alpha_p\cdot(s_p-\Delta_p)}}{e^{s\cdot\alpha_p\cdot(s_p-\Delta_p)}+\sum_{j=1,j\neq i}^{C}e^{s\cdot\alpha_n^j\cdot(s_n^j-\Delta_n)}}$$
$$\alpha_p = O_p - s_p, \quad \alpha_n^j = s_n^j + O_n \quad (5)$$

where α_p, α_n are the self-paced weight, and O_p, O_n are the optimum for s_p, s_n, respectively. Δ_p and Δ_n are the between-class and within-class margins. When s_p deviates far from O_p or s_n deviates far from O_n, s_p or s_n will get effective update with large gradient. If s_p or s_n has reached its optimum, they will get no gradient update. Then the authors proposed to reduce the hyper-parameters by setting $O_p = 1 + m$, O_n=-m, Δ_p=1-m, Δ_n=m. Consequently, the circle loss finally becomes:

$$L_{circle} = -\log\frac{e^{s\cdot(m^2-(1-s_p)^2)}}{e^{s\cdot(m^2-(1-s_p)^2)}+\sum_{j=1,j\neq i}^{C}e^{s\cdot((s_n^j)^2-m^2)}} \quad (6)$$

where the decision boundary is $(1-s_p)^2 + s_n^2 = 2m^2$, the arc of a circle so that the loss function is referred as circle loss. It aims to optimize s_p to 1 and s_n to 0. Similar to the assumption of equation(3), the gradients of circle loss under the toy scenario can be denoted as:

$$\frac{\partial L}{\partial s_p} = (1-\frac{e^{s\cdot(2m^2-(1-s_p)^2-s_n^2)}}{e^{s\cdot(2m^2-(1-s_p)^2-s_n^2)}+(C-1)})\cdot 2s\cdot(1-s_p)$$
$$\frac{\partial L}{\partial s_n} = (\frac{(C-1)}{e^{s\cdot(2m^2-(1-s_p)^2-s_n^2))}+(C-1)})\cdot 2s\cdot s_n \quad (7)$$

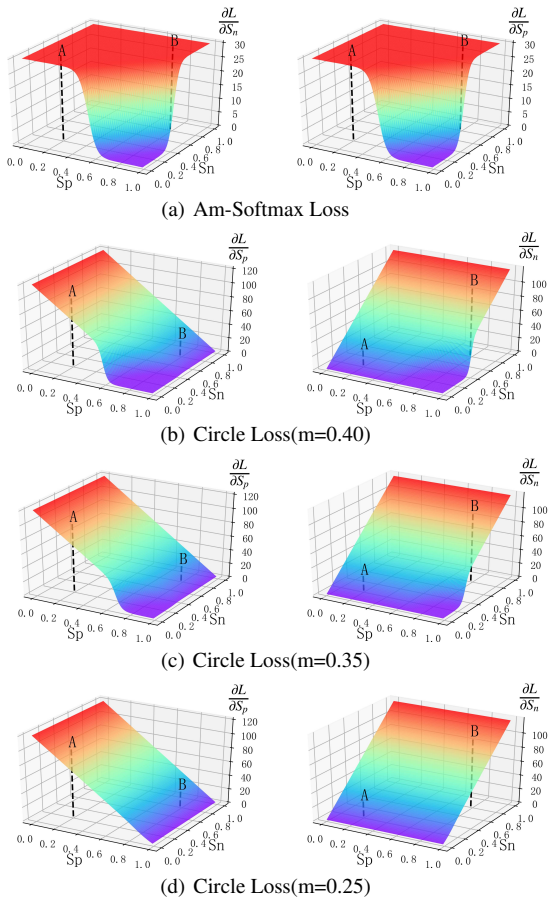


Figure 1: *The gradients of the loss functions. (a)Am-Softmax. (b,c,d) Circle loss with different margin.*

We visualize the gradients of circle loss with different margin in Figure 1, from which we obtain the following observations:

- Compared with Am-Softmax, circle loss gives gradually attenuated gradients on s_p and s_n. As they gradually

reach their optimal, the gradients correspondingly decay, reducing the influence of noisy samples. For instance, point A(0.2, 0.2) gets larger gradients on s_p and smaller gradients on s_n, on the contrary point B(0.8, 0.8) gets larger gradients on s_n and smaller gradients on s_p.

- In Figure 1(d), if the margin is too small, the gradient will degenerate into a linear function, and the loss function keeps on penalizing both s_p and s_n. In Figure 1(b), a larger margin allows s_p and s_n to converge easily, but the gradient quickly approaches 0 if s_p and s_n cross the decision boundary, which means the loss function will not optimize s_p and s_n.

Based on the above analysis, we investigate two strategies to generate an appropriate margin that balances convergence speed and discrimination of s_p and s_n.

3.2. Stage-based margin

After training with fixed margin circle loss, we randomly sample 10% training samples and calculate the mean of s_{p-mean} and s_{n-mean} of them. The mean radius for each epoch is defined as: $r_{mean} = \sqrt{(1 - s_{p-mean})^2 + (s_{n-mean})^2}$. As shown in Figure 2 and Table1, with a lower margin m, such as $m = 0.25$, it has a lower mean radius and better angle discrimination in the training set. However, its performance is much worse than the system with $m = 0.40$. With the unreachable decision boundary, the circle loss keeps on optimizing s_p and s_n and eventually causes the model to overfit the noisy samples. So that we propose the stage-based margin for circle loss, which initializes a larger margin for m and decreases it in the different training stages. The model can converge quickly and reasonably in the first training stage under loose constraints, and the constraints on the model become stricter when the model has learned identification information, making the model have better angular discrimination when converging.

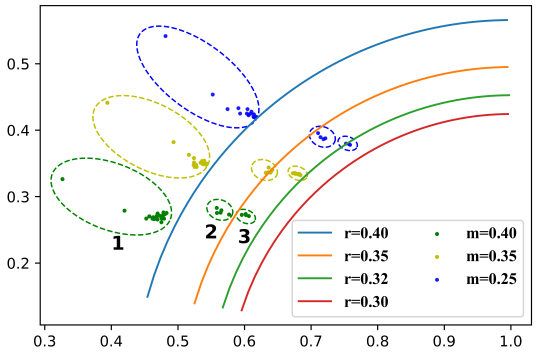


Figure 2: *The change of mean radius during different training stage with fixed margin circle loss. The number 1, 2 and 3 represent three stages, from left to right.*

3.3. Chunk-based margin

Inspired by [23], the authors propose an adaptive mechanism based on the magnitude which can measure the quality of the given sample. This prevents models from overfitting on noisy low-quality samples. In ASV tasks, we generally randomly cropped or extended the training sample to L frames, while L is randomly sampled from the interval $[L_{min}, L_{max}]$. The training sample is harder for the model when L is smaller, so we propose an adaptive margin based on the chunk width. The formulation is:

$$m = (1 - \lambda \frac{L - L_{min}}{L_{max} - L_{min}}) m_0 \tag{8}$$

where λ is hyper-parameters and m_0 is the original margin. Parameter λ controls the effect of chunk width on the margin. When L is close to L_{max}, the margin is smaller and the constraints on the model become stricter.

4. Experimental setup

4.1. Dataset

We run experiments on the VoxCeleb dataset [24, 25] and SITW [26]. The development set of VoxCeleb2 (5994 speakers) is used for training. The whole VoxCeleb1 and SITW are used as the evaluation set with four publicly available test trails: VoxCeleb1-O-clean, VoxCeleb1-E-clean, VoxCeleb1-H-clean, and core-core from SITW.

The equal error rate (EER) and minimum detection cost function with P_{target} equal to 0.01 are presented to demonstrate the performance.

4.2. Training details

In our experiments, the input features are the 64-dimensional log Mel-filterbank energies with cepstral mean and variance normalization. No voice active detection (VAD) or data augmentation is applied to the training data.

Following the previous work in [8, 27], we use the standard ResNet-34 architecture to extract speaker embeddings. The initial number of channels is set to 32, and only the mean of the frame-level features is used as statistics. Besides, no dropout is applied in our networks.

All models are trained using the stochastic gradient descent (SGD) optimizer with momentum 0.9 and weight decay 1e-3. The learning rate is started with 0.1 and is reduced by 10x when the training loss reaches stability. Mini-batch size is set to 64.

For every training step, a chunk-width L is randomly sampled from the interval $[\mathrm{L}_1, \mathrm{L}_2]$ and each training sample in the mini-batch is randomly cropped or extended to L frames. We allow L_1 and L_2 to increase when the learning rate drops. According to [28], with a large initial learning rate, the model learns hard-to-generalize, easily fit patterns. So we give the model fewer features by applying a smaller chunk-width L to prevent overfitting in the first training stage. The finally used interval is set to [200,400], [300,500], and [400,600] in the three training stages.

When training with the angular sofmtax loss, we follow the best set in [17], $m = 0.2$ and $s = 30$ for Am-Softmax, $m = 0.25$ and $s = 30$ for Arc-Softmax. As to circle loss, s is set to 60. The cosine similarity is used as the back-end scoring method.

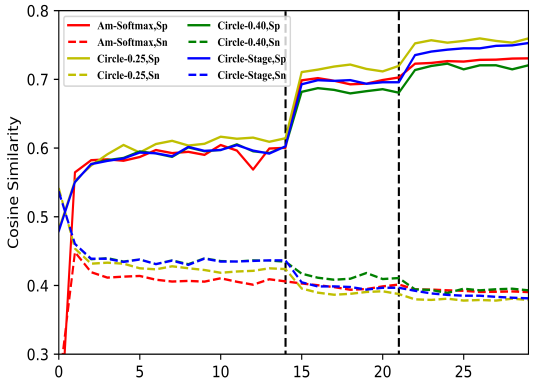
5. Results

Table 1 summarizes the results with different loss functions. The first row shows the performance of our baseline system, which is consistent with the results reported in [27]. Compared with softmax loss, the second and third row shows that Arc-Softmax and Am-Softmax achieve a similar improvement in all test sets. The performance of the circle loss is exhibited in the following sections of Table 1.

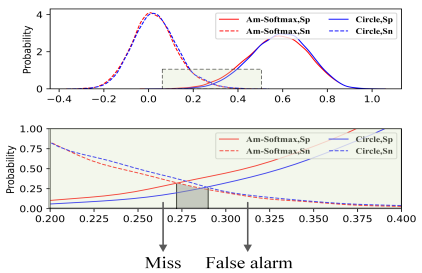
Table 1: *The results on the VoxCeleb1 test set, the extended and hard test sets (VoxCeleb1-E and VoxCeleb1-H, repsectively), and the evaluation set of SITW Core. The cleaned trial lists are used for Voxceleb.*

Loss		VoxCeleb1-O		VoxCeleb1-E		VoxCeleb1-H		SITW Core
		EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF	EER(%)
Softmax		1.77	0.192	1.79	0.202	3.23	0.306	3.25
Arc-Softmax	s=30, m=0.25	1.64	0.170	1.63	0.177	2.91	0.273	2.52
Am-Softmax	s=30, m=0.20	1.71	0.161	1.65	0.183	2.83	0.263	2.46
Circle	s=60, m=0.25	1.68	0.200	1.74	0.194	2.96	0.275	2.46
	s=60, m=0.30	1.75	0.160	1.74	0.189	3.02	0.286	2.52
	s=60, m=0.35	1.44	0.161	1.58	0.170	2.72	0.258	2.36
	s=60, m=0.40	1.45	**0.133**	1.56	0.166	2.64	**0.240**	2.27
Circle-Chunk	m: 0.40	1.41	0.145	1.55	0.162	2.67	0.253	2.19
Circle-Stage	m: 0.40, 0.35, 0.30	1.35	0.144	1.54	0.165	2.65	0.251	2.24
	m: 0.40, 0.35, 0.32	**1.31**	0.135	**1.51**	**0.163**	**2.61**	0.250	**2.13**

We investigate the influence of different margins for circle loss in the fourth row. It is clear that the larger the margin m, the better the performance of circle loss. And it achieves the best result when $m = 0.40$, which reduces 11.6% and 21.8% in terms of EER and MinDCF in Voxceleb1-O compared with Arc-Softmax. Besides, there is a performance gap between $m = 0.40$ and $m = 0.25$. As shown in Figure 2, it is helpful for circle loss to get better performance by training with the reachable decision boundary.



(a) The change of s_p and s_n values during the three training stage.



(b) The cosine similarity distributions for the same and different speaker spaces in test set.

Figure 3: *(a)The similarity distributions during training. (b)The similarity distributions in test set.*

We visualized the positive and negative similarity distributions of all training epochs in Figure 3(a) and the similarity distributions of the test set in Figure 3(b). In Figure 3(a), the circle loss with $m = 0.40$ (the green one) has a lower positive similarity but higher performance than Am-Softmax (the black one), from which we can infer that circle loss with a suitable margin achieves a more reasonable and favorable convergence state. Though the performance of circle loss with $m = 0.25$ (the yellow on) is not goodd as $m = 0.40$, it achieves the best angle discrimination of training samples, which is better than Am-Softmax. Besides, Circle-Stage (the blue one) gets similar best angle discrimination with $m = 0.25$ in the last training stage and maintains good performance in the test set. The performance of Circle-Chunk is slightly worse than Circle-Stage in our experiments.

As shown in Figure 3(b), the negative similarity distributions of circle loss and Am-Softmax loss are similar, but circle loss has better positive similarity distributions than Am-Softmax loss, especially in the area framed by the dotted line. The area is enlarged in the figure below, from which we can find that the false alarm of circle loss is smaller than Am-Softmax, and the value is the area of the gray area. So that the EER of circle loss is much lower than Am-Softmax.

6. Conclusions

In this paper, we propose a novel angular-based loss called adaptive margin circle loss for speaker verification. Circle loss has flexible optimization and definite convergence status than the other angular-based losses like Am-Softmax. By selecting a fixed appropriate margin, circle loss can achieve promising results. In addition, we explore two strategies, stage-based and chunk-based, to generate a more suitable margin, which can enhance the intra-class compactness of speaker embeddings. With adaptive margin circle loss, our best system achieves 1.31%EER on Voxceleb1 and 2.13% on SITW, core-core, which is competitive among the existing reported results.

7. References

[1] 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, 2010.

[2] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

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

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

[5] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5796–5800.

[6] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks." in *Interspeech*, 2018, pp. 3743–3747.

[7] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[8] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," *arXiv preprint arXiv:1804.05160*, 2018.

[9] T. Zhou, Y. Zhao, and J. Wu, "Resnext and res2net structures for speaker verification," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 301–307.

[10] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 212–220.

[11] 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*, vol. 650, 2017.

[12] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances." in *Interspeech*, 2017, pp. 1487–1491.

[13] H. Bredin, "Tristounet: triplet loss for speaker turn embedding," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 5430–5434.

[14] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[15] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[16] W. Cai, J. Chen, and M. Li, "Analysis of length normalization in end-to-end speaker verification system," *arXiv preprint arXiv:1806.03209*, 2018.

[17] Y. Liu, L. He, and J. Liu, "Large margin softmax loss for speaker verification," *arXiv preprint arXiv:1904.03479*, 2019.

[18] D. Zhou, L. Wang, K. A. Lee, Y. Wu, M. Liu, J. Dang, and J. Wei, "Dynamic margin softmax loss for speaker verification," *Proc. Interspeech 2020*, pp. 3800–3804, 2020.

[19] M. Rybicka and K. Kowalczyk, "On parameter adaptation in softmax-based cross-entropy loss for improved convergence speed and accuracy in dnn-based speaker recognition," *Proc. Interspeech 2020*, pp. 3805–3809, 2020.

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

[21] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, "Sub-center arcface: Boosting face recognition by large-scale noisy web faces," in *European Conference on Computer Vision*. Springer, 2020, pp. 741–757.

[22] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6398–6407.

[23] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," *arXiv preprint arXiv:2103.06627*, 2021.

[24] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[25] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

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

[27] S. Wang, J. Rohdin, O. Plchot, L. Burget, K. Yu, and J. Černockỳ, "Investigation of specaugment for deep speaker embedding learning," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7139–7143.

[28] Y. Li, C. Wei, and T. Ma, "Towards explaining the regularization effect of initial large learning rate in training neural networks," *arXiv preprint arXiv:1907.04595*, 2019.