BOUNDARY DISCRIMINATIVE LARGE MARGIN COSINE LOSS FOR TEXT-INDEPENDENT SPEAKER VERIFICATION

Rongjin Li[1,2*], *Na Li*[1], *Deyi Tuo*[1], *Meng Yu*[1], *Dan Su*[1], *Dong Yu*[1]

[1]Tencent AI Lab
[2]College of Electronic Science and Technology, Xiamen University, China

ABSTRACT

Deep neural network based speaker embeddings have attracted much attention in text-independent speaker verification task. In addition to the network architecture, an appropriate design of the loss function is crucial for the deep discriminative embedding extractor. Inspired by the success of Large Margin Cosine Loss (LMCL) in face recognition, we propose an enhanced LMCL named boundary discriminative LMCL (BD-LMCL) to emphasize the discriminative information inherited in the speaker boundaries. Unlike LMCL, where all training samples contribute equally for the objective function, only the samples around the speaker boundaries are considered during the network training with BD-LMCL. Specifically, those samples close to the boundaries are dynamically selected using top-k zero-one loss. Experimental results on a short duration corpus Android Cellphone and NIST SRE 2012 demonstrate better performance compared to LMCL and other popular loss functions.

Index Terms— speaker embedding, loss function, boundary, top-k loss, speaker verification

1. INTRODUCTION

Text-independent speaker verification is a challenging task in the presence of a large number of ambiguous speakers. The speaker information should be extracted from the complex speech data, especially for an unseen speaker. I-vector/PLDA framework [1, 2, 3] is a well known state-of-the-art solution to this task. Although this framework performs well when long utterances are available, it suffers performance degradation in presence of short utterances.

Recently, speaker embedding based systems have shown significant performance improvement for short-duration speaker verification compared to the i-vector system. In [4], the speaker embeddings referred to as *d*-vector were created by averaging bottle-neck layer activations of a feed-forward DNN which was trained to classify speakers at the frame-level. Work in [5, 6] also exhibits better performance compared to i-vector/PLDA framework for short utterances by training a deep CNN similar to VGG net with softmax loss. To handle utterances with arbitrary duration, [7, 8] proposed to extract utterance-level speaker embeddings using a statistics pooling layer to aggregate the frame-level inputs for a time-delay neural network (TDNN) [9] based architecture. The resulting speaker embeddings called x-vectors followed by probability linear discriminant analysis (PLDA) [3] backend demonstrate promising performance for utterances with variant duration.

The discriminating power of feature representation is crucial for speaker verification system, and a robust embedding extractor should be effective in minimizing intra-speaker variation and maximizing inter-speaker discrepancy. However, the embedding extractor with traditional softmax loss usually overfits to corpus-specific speech, resulting in speaker embeddings with insufficient discriminating power for verification. To address this issue, triplet loss was introduced to speaker verification in [10, 11]. However, such loss function requires thoroughly scheming the mining of triplet samples, which is an extremely time-consuming procedure. The work in [12] employed center loss to reduce the intra-speaker variation in the Euclidean space. The center loss, however, ignores the inter-speaker variances, which may result in suboptimal solutions. More particularly, angular softmax (A-softmax) loss [13, 14, 15] and LMCL [16] project the features from Euclidean space to an angular space, and introduce an angular margin or a cosine margin term to maximize the decision margin. Since all the training samples share the same margin in A-softmax and LMCL, both of the two methods are sensitive to the initial margin, if the initial margin is too large, the network will be hard to converge and tend to be instable, otherwise, a small margin will limited the performance improvement compared to the softmax loss.

To address this issue, we propose an enhanced LMCL referred as boundary discriminative LMCL (BD-LMCL) for text-independent speaker verification in this paper. Our motivation is based on the assumption that the training samples far away from the corresponding speaker boundaries have little contribution to the decision boundary, while those samples around the speaker boundaries are crucial for modelling the classification boundary. We therefore pay more attention to the training samples near the speaker boundaries in the angular space. Specifically, for each training speaker in a mini-batch, we use the top-k zero-one loss [17] to dynamically sample some training samples around the corresponding speaker boundary in the angular space for computing the standard LMCL. In other words, for the training samples which are easy to classify, we will not further introduce an extra margin to the classification boundary in BD-LMCL, but a cosine margin will be introduced for the hard samples to maximize the decision margin in the cosine space. Compared to LMCL, BD-LMCL can accelerate the convergence in the training stage and better model the discriminative information inherited in the speaker boundaries. Experiments on a short duration corpus Android Cellphone and NIST SRE 2012 show that Inception-ResNet architecture [18] with the proposed BD-LMCL outperforms other baseline methods.

The rest of this paper is organized as follows: Section 2 describes the large margin cosine loss. Section 3 presents the proposed boundary discriminative large margin cosine loss in detail. Section 4 presents the experimental setup and the analysis of the results. The conclusions are finally made in Section 5.

*This work was done while R.Li was an intern at Tencent AI Lab, Shenzhen, China

2. RELATED WORK

Speaker verification shares many properties with the face recognition task. In the face recognition community, a deep neural network classifier trained with a regularized strategy is also usually used as a deep discriminative embedding extractor. However, researchers found that the traditional softmax loss lacked the power of discrimination. Recently, several loss functions such as triplet loss, center loss, and angular softmax loss have been proposed to address this problem. All these improved losses share a common idea for improving discrimination capability: maximizing inter-class variance and minimizing intra-class variance. More recently, Wang et al.[16] proposed the LMCL to further maximize the decision margin in the angular space. Specifically, the softmax loss was reformulated as a cosine loss by L2 normalizing of both the features and the weight vectors to remove radial variations, then a cosine margin term m was introduced between different classes to improve the cosine-related discriminative information. The motivation is that the posterior probability of the ground-truth class should be larger than a decision term. After normalizing the weight vectors W and the feature vector x to remove radial variations, the posterior probability merely relies on $cos(\theta_i)$, where θ_i is the angle between the feature vector and the weight vector related to class i. And by fixing the L2 normalization of $\|x\|$ to a constant value, the $cos(\theta_i) - m$ is then fed to the softmax layer with cross entropy loss function. Finally, the LMCL loss function becomes:

$$L_{lmc} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{e^{s(cos(\theta_{y_i,i})-m)}}{e^{s(cos(\theta_{y_i,i})-m)} + \sum_{r\neq y_i} e^{s(cos(\theta_{r,i}))}} \quad (1)$$

subject to

$$cos(\theta_{y_i,i}) = \frac{W_{y_i}^T x_i}{\|W_{y_i}\| \, \|x_i\|} \quad (2)$$

where N is the number of training samples, x_i denotes the i-th embedding from the ground-truth class y_i. The $m \geqslant 0$ is a fixed margin that controls the magnitude of the cosine margin. The W_{y_i} is the weight vector of class y_i, and $\theta_{y_i,i}$ is the angle between x_i and the y_i-th column of the weights $\mathbf{W}$. The constant s is a scaling hyper-parameter.

3. BOUNDARY DISCRIMINATIVE LARGE MARGIN COSINE LOSS

When the network is trained using LMCL, all training samples share a common margin and contribute equally to the loss function. However, the fact may be that when the feature vectors extracted from the corresponding training samples are far away from the speaker boundaries in the angular space, they will have little contribution to the decision boundary. While the feature vectors around the speaker boundaries are significant for modeling the discriminative information inherited in classification boundaries. Hence, it is reasonable to pay more attention to the features vectors near the speaker boundaries in the angular space. We introduce a weight function ω to each training sample. In term of formula, the proposed boundary discriminative large margin cosine loss can be defined as:

$$L_{bd-lmc} = -\frac{1}{N}\sum_{p=1}^{P}\sum_{j=1}^{n} log\frac{e^{s(cos\theta_{p,j}-\omega(p,j)m)}}{Z} \quad (3)$$

$$Z = e^{s(cos\theta_{p,j}-\omega(p,j)m)} + \sum_{r\neq p} e^{s(cos\theta_{r,j})} \quad (4)$$

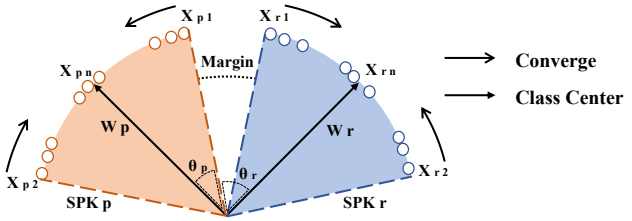

Fig. 1. Decision boundaries learned by BD-LMCL. Distinct color areas represent feature space from distinct speakers. Note that the metrics in the figure do not represent real distances. The real distance is $cos(\theta_i)$, and we use the angle as the distance for a better geometrical interpretation. The analysis is detailed in Section 3.

subject to

$$cos\theta_{p,j} = \frac{W_p^T x_j}{\|W_p\| \, \|x_j\|} \quad (5)$$

where P is the number of training speakers in a mini-batch, each speaker has n training utterances, x_j is the j-th feature vector from speaker p, and r represents some other speaker.

When the weight function $\omega(p, j)$ is constantly equal to 1, the Eq.(3) becomes the standard LMCL. In order to select the feature vectors away from the speaker boundaries dynamically, the values of $\{cos\theta_{p,j}, j = 1, ..., n\}$ should be used as an important reference for selecting feature vectors which are hard to classify. A larger $\theta_{p,j}$ indicates that the x_j is closer to the boundaries corresponding to its ground-truth speaker p. Specifically, we sort a set of $\{cos\theta_{p,j}, j = 1, ..., n\}$ in descend order.

$$cos\theta_{p,n} \geqslant ... \geqslant cos\theta_{p,j} \geqslant ... \geqslant cos\theta_{p,1} \quad (6)$$

And then we introduce a hyper-parameter k to denote how many feature vectors from each speaker are considered to be far from the speaker boundaries. Other $(n - k)$ feature vectors are considered to be around the speaker boundaries. We use $[.]$ as a logical operator. For an expression E, if E is true, then $[E] = 1$, otherwise $[E] = 0$. Given a fixed k and speaker p, the top-k zero-one loss is defined as:

$$err_k(p, j) = [cos\theta_{p,n-k} \geqslant cos\theta_{p,j}], j = 1, ..., n \quad (7)$$

the range of integer k is $0 \leqslant k < n$. Note that for $k = 0$, the $err_k(p, j)$ will become 1 for all feature vectors from the p-th speaker. Finally, in each training iteration, a dynamic margin is introduced by defining the weight function as:

$$\omega(p, j) = err_k(p, j) \quad (8)$$

Fig. 1 illustrates the process of learning decision boundary. As shown in the figure, the embeddings near the speaker boundaries are emphasized during the training stage.

4. EXPERIMENTS AND RESULTS

The experiments were conducted on two corpora: Android cellphone and NIST SRE 2012 [19]. We used equal error rate (EER) and minimum decision cost function (minDCF) defined in NIST 2012 SRE to evaluate the performance of different systems on NIST SRE 2012. For the experiments on Android Cellphone, only EER was used as the evaluation metric.

Table 1. Architecture of the modified *Inception-ResNet-v1*. The output size of each module is the input size of the next one. V denotes 'Valid' padding.

type	patch size/stride or remarks	input size
Conv_1	$3\times1/1\times1$ V	$150\times40\times1$
Conv_2	$3\times3/1\times1$ V	$148\times40\times32$
Conv_3	$3\times3/1\times1$	$146\times38\times32$
MaxPool	$3\times3/2\times2$ V	$146\times38\times64$
Conv_4	$1\times1/1\times1$ V	$72\times18\times64$
Conv_5	$3\times3/1\times1$ V	$72\times16\times80$
5×Inception-ResNet-A	-	$70\times16\times192$
Reduction-A	-	$70\times16\times192$
10×Inception-ResNet-B	-	$34\times7\times832$
Reduction-B	-	$34\times7\times832$
5×Inception-ResNet-C	-	$16\times3\times1728$
Average Pooling	$16\times3\times1$	$16\times3\times1728$
Fully Connected	1728×500	1×1728
Dropout (keep 0.5)	-	1×500
Loss	-	1×500

4.1. Speech Data and Front-end Processing

- *Android Cellphone*: Collected by Android Cellphones, most utterances in this corpus are short with average duration of 2.6s. The training set contains 2,000 Chinese speakers, and each speaker has 300 utterances. In the evaluation set, all utterances come from 500 speakers. For each speaker, 3 utterances were sampled as the enrollment data. Except the enrollment data, we sampled 25 utterances from each speaker and 800 utterances from other speakers, which resulted in 12,500 target trials and 400,000 imposter trials in total.

- *NIST SRE 2012*: The training set includes the Switchboard (SWBD) and NIST Speaker Recognition Evaluations (SREs) corpus. The SWBD corpus consists of SWBD 2 Phase 1, 2 and 3, and SWBD Cellular 1 and 2. The SREs consists of 2004, 2005, 2006, 2008 and 2010. We excluded speakers with fewer than 8 utterances and discarded the utterances less than 5 seconds. To keep the gender balance, 2,000 male speakers and 2000 female speakers were sampled randomly. We also made the data augmentation for the selected training speakers with RIRS_NOISES and MUSAN corpora [20, 21]. The augmentation data together with the original clean training data–including 4,000 speakers and 193,665 utterances in total–were used for training universal background model (UBM), total variability matrix, deep neural networks and PLDA model. We evaluated the system performance on the core condition 4 of NIST 2012 SRE (core set), male speaker.

For each speech utterance, a voice activity detection (VAD) algorithm was applied to detect the silence frames. Then the voice-active regions were extracted and segmented into 25ms Hamming windowed frames with 10ms frame-shift. For i-vector baseline system, the first 19 Mel frequency cepstral coefficients (MFCC) with log energy were calculated with their first and second derivatives to form a 60-dimensional acoustic vector, followed by cepstral mean normalization (CMN). For the neural network systems, the input features were 40-dimensional log mel-filter bank features.

Table 2. Performance of i-vector based system, different loss function based systems on *Android cellphones*, in terms of EER(%).

Method	Cosine Similarity			PLDA		
	1.5s	3s	full	1.5s	3s	full
i-vector	9.10	8.25	7.95	7.43	6.54	5.38
Softmax Loss	4.46	3.96	4.02	4.87	4.44	4.42
Triplet Loss	3.61	2.86	2.86	3.25	2.47	2.56
Center Loss	2.18	1.45	1.44	2.77	1.95	1.87
A-Softmax	1.89	1.29	1.27	2.16	1.62	1.59
LMCL	1.82	1.23	1.17	1.97	1.45	1.37
Proposed	**1.60**	**1.09**	**1.07**	**1.82**	**1.34**	**1.29**

Table 3. Performance of the proposed method with varying ratio of k/n in terms of EER(%) on *Android cellphone*, cosine similarity was used as the scoring back-end, the duration of the test utterances was fixed to 1.5s.

k/n	10%	30%	50%	70%	90%
EER	1.83	1.67	**1.60**	1.68	1.88

4.2. I-vector Baseline

The i-vectors were extracted based on a gender-independent UBM with 1,024 Gaussian components and a total variability matrix with 500 total factors. We applied within-class covariance normalization (WCCN) and length normalization (LN) to the 500-dimensional i-vectors. Then linear discriminant analysis (LDA) was used to reduce the dimension of i-vectors to 200. And the PLDA models with 150 latent identity factors were trained.

4.3. Deep Speaker Embedding Systems

According to the characteristics of input speech, a modified version of *Inception-ResNet-v1* [12, 18] was employed as our deep embedding extractor in our experiments. The architecture of the network is shown in Table 1. As the utterances of Android Cellphone are short, we extracted a segment of 150 frames from each training utterance after VAD as the input of the network. For NIST SRE corpus, all long training utterances were divided into multiple 150-frame segments by employing a sliding-window without overlap. RMSProp optimizer with an initial learning rate of 0.1 was employed for training all the networks with different loss functions. The learning rate was decayed based on the validation set performance. We employed dropout and L2 regularization to reduce overfitting and used batch normalization to accelerate the training process. The batch size was set to 128. Given the trained network, the 500-dimensional embeddings of the enrollment and test utterances were extracted from the fully-connected layer. Then the utterance-level speaker embeddings were obtained by performing average pooling along the time axis.

4.4. Results and Analysis

In this section, we compared the performance of the proposed BD-LMCL with the i-vector system and five other popular loss functions, including the softmax loss, the triplet loss, the center loss, the A-softmax and the standard LMCL. All of the loss functions were evaluated based on the same modified *Inception-ResNet-v1* architecture. Cosine similarity and PLDA were used as the back-ends for all systems.

Table 4. Performance of i-vector based system, different loss function based systems in terms of EER (%) and minDCF on CC4 of NIST 2012 SRE (core set), male speaker.

Method	EER						minDCF					
	Cosine Similarity			PLDA			Cosine Similarity			PLDA		
	6s	30s	full	6s	30s	full	6s	30s	full	6s	30s	full
i-vector	19.08	14.29	12.28	12.06	6.13	3.36	0.935	0.868	0.839	**0.777**	0.421	**0.280**
Softmax Loss	16.28	9.20	4.79	12.78	6.30	3.52	0.967	0.656	0.463	0.949	0.521	0.337
Triplet Loss	15.40	8.61	4.39	12.24	5.99	3.31	0.955	0.625	0.420	0.928	0.503	0.320
Center Loss	14.87	8.15	3.97	11.80	5.77	3.18	0.934	0.590	0.376	0.907	0.491	0.312
A-Softmax	14.24	7.83	3.56	11.61	5.12	3.05	**0.917**	0.569	0.338	0.820	0.426	0.296
LMCL	13.97	7.65	3.49	11.50	5.05	3.01	0.927	0.566	0.328	0.817	0.422	0.296
Proposed	**13.65**	**7.37**	**3.41**	**11.38**	**4.90**	**2.74**	0.923	**0.556**	**0.324**	0.822	**0.420**	0.290

4.4.1. Performance of Different Systems on Android Cellphone

Table 2 presents the performance of different systems on different duration conditions for the test utterances, i.e., 1.5s, 3s and full. For enrollment, 1.5s segments were extracted from the enrollment speech after VAD in all experiments. The margin and scale parameters in LMCL were set to 0.35 and 30 respectively, the angular margin term in A-Softmax was set to 4, and α and λ in center loss were set to 0.2 and 0.001 respectively in the following experiments. Hard trial selection strategy was adopted in triplet loss.

Results in Table 2 show that the deep speaker embedding based systems outperform the i-vector system for short utterances. The proposed BD-LMCL demonstrates better performance compared to other loss functions on different duration conditions. Specially, for the duration condition of 1.5s, the EER of the BD-LMCL achieves 82.42% and 12.09% relative improvement over the i-vector system and standard LMCL respectively when cosine similarity is used as the back-end. Compared to the softmax loss, we can see that the performance of the other loss functions is much better. We also find that except for the triplet loss, the performance of other loss functions degrades when PLDA back-end is employed.

In BD-LMCL function, the value of k/n denotes the ratio between the amount of the discarded training samples and the total amount of training data. If k/n is 0, the proposed BD-LMCL will be equal to the standard LMCL. To further investigate the performance of the proposed BD-LMCL, we changed the ratio from 10% to 90%. The corresponding results are presented in Table 3. When the ratio is set to 10%, the performance of BD-LMCL is close to the standard LMCL. The reason is that almost all training samples are subtracted by the margin, leading to a sub-optimal decision boundary. The best result is obtained when half of training samples are used to compute the loss.

4.4.2. Performance of Different Systems on NIST SRE 2012

For the experiments on NIST SRE 2012, we set the ratio k/n to 50% for our proposed BD-LMCL, the parameters of other loss functions were kept the same as for those experiments on *Android cellphones*. The full-length utterances were used for enrollment, and the performance of different systems were evaluated on different duration conditions for the test utterances, including 6s, 30s and full length.

From Table 4, we can see that the proposed method achieves the best performance compared to other systems under almost all conditions. Compared to the above experiments, the deep embedding systems achieve performance improvement when PLDA is employed as the back-end. A possible reason may be that there are sufficient training utterances for training the PLDA model in the experiments on NIST SRE 2012. Results also indicate that our proposed method can achieve better performance compared to i-vector/PLDA system when long utterances are available.

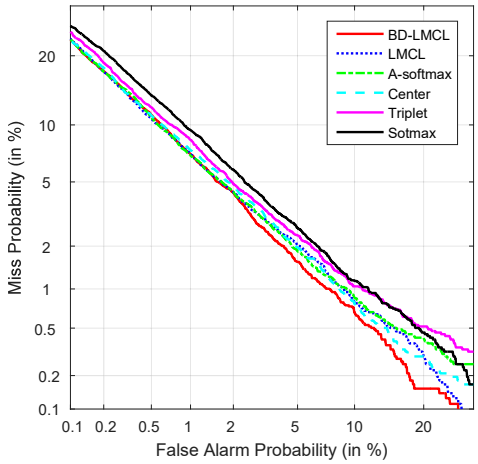

Fig. 2. The DET curves of the proposed BD-LMCL and other loss functions based systems for male speakers on CC4 of NIST SRE 2012 (core set).

Fig. 2 shows the DET curves of different loss function based embedding systems. Again, the proposed BD-LMCL based system performs the best at most of the operating points.

5. CONCLUSION

In this paper, we proposed an enhanced LMCL named boundary discriminative LMCL (BD-LMCL) to emphasize the discriminative information inherited in the speaker boundaries. BD-LMCL was used to guide deep Inception-ResNet CNN to learn highly discriminative speaker embeddings. To demonstrate the effectiveness of the proposed method, extensive experiments were conducted on two corpora. Compared to i-vector/PLDA system and other popular loss function based systems, the proposed BD-LMCL method achieves the best performance. We wish that our explorations on learning deep discriminative speaker embeddings with BD-LMCL will benefit the text-independent speaker verification task.

6. REFERENCES

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

[2] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre 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] Simon JD Prince and James H Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

[4] Ehsan Variani, Xin Lei, Erik Mcdermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 4052–4056.

[5] Shi Xiong Zhang, Chen Zhuo, Zhao Yong, Jinyu Li, and Yifan Gong, "End-to-end attention based text-dependent speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE.*, 2016, pp. 171–178.

[6] Gautam Bhattacharya, Jahangir Alam, and Patrick Kenny, "Deep speaker embeddings for short-duration speaker verification," in *Proc. Interspeech*, 2017, pp. 1517–1521.

[7] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. Interspeech 2017*, 2017, pp. 999–1003.

[8] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5329–5333.

[9] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Interspeech*, 2015, pp. 3214–3218.

[10] Chunlei Zhang, Kazuhito Koishida, and John HL Hansen, "Text-independent speaker verification based on triplet convolutional neural network embeddings," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 26, no. 9, pp. 1633–1644, 2018.

[11] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu, "Deep speaker: an end-to-end neural speaker embedding system," 2017.

[12] Na Li, Deyi Tuo, Dan Su, Zhifeng Li, Dong Yu, and AI Tencent, "Deep discriminative embeddings for duration robust speaker verification," *Proc. Interspeech 2018*, pp. 2262–2266, 2018.

[13] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song, "Sphereface: Deep hypersphere embedding for face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, vol. 1, p. 1.

[14] Sergey Novoselov, Andrey Shulipa, Ivan Kremnev, Alexandr Kozlov, and Vadim Shchemelinin, "On deep speaker embeddings for text-independent speaker recognition," *arXiv preprint arXiv:1804.10080*, 2018.

[15] Zili Huang, Shuai Wang, and Kai Yu, "Angular softmax for short-duration text-independent speaker verification," *Proc. Interspeech 2018*, pp. 3623–3627, 2018.

[16] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Zhifeng Li, Dihong Gong, Jingchao Zhou, and Wei Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5265–5274.

[17] Maksim Lapin, Matthias Hein, and Bernt Schiele, "Loss functions for top-k error: Analysis and insights," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1468–1477.

[18] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[19] NIST, "The NIST year 2012 speaker recognition evaluation plan," *http://www.nist.gov/itl/iad/mig/sre12.cfm*, 2012.

[20] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5220–5224.

[21] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.