Interspeech 2018
2-6 September 2018, Hyderabad

Robust and Discriminative Speaker Embedding via Intra-Class Distance Variance Regularization

Nam Le[1,2], *Jean-Marc Odobez*[1,2]

[1]Idiap Research Institute, Martigny, Switzerland
[2]École Polytechnique Fédéral de Lausanne, Switzerland

nle@idiap.ch, odobez@idiap.ch

Abstract

Learning a good speaker embedding is critical for many speech processing tasks, including recognition, verification, and diarization. To this end, we propose a complementary optimizing goal called intra-class loss to improve deep speaker embeddings learned with triplet loss. This loss function is formulated as a soft constraint on the averaged pair-wise distance between samples from the same class. Its goal is to prevent the scattering of these samples within the embedding space to increase the intra-class compactness.When intra-class loss is jointly optimized with triplet loss, we can observe 2 major improvements: the deep embedding network can achieve a more robust and discriminative representation and the training process is more stable with a faster convergence rate. We conduct experiments on 2 large public benchmarking datasets for speaker verification, VoxCeleb and VoxForge. The results show that intra-class loss helps accelerating the convergence of deep network training and significantly improves the overall performance of the resulted embeddings.

Index Terms: speaker verification, deep neural networks, embedding learning, triplet loss

1. Introduction

Learning speaker representations that can enable comparing speech utterances directly is a crucial for multiple speaker related tasks in speech processing, including diarization, recognition, and verification. Recently, deep learning systems have achieved better benchmarking results than i-vectors in these speaker related tasks [1, 2, 3, 4, 5]. In these systems, a speaker embedding can be learned in two main ways. First, it can be extracted as the derivatives of the speaker recognition task by using the activation of the last layer before classification [4, 6, 7]. Second, it can be learned directly by optimizing the loss functions constraining the distances between same-speaker and different-speaker utterance pairs [3, 8, 9]. Among the distance-based losses, triplet loss has become more and more widely used in deep embedding networks [2, 3, 8].

The main idea of triplet loss is that for the distance between a given pair of same-speaker utterances should be smaller than the distance from each of these utterances to any different-speaker utterance by a constant margin [10]. While this idea is attractive, learning with triplet loss can result in suboptimal performance in practice, especially in text-independent verification. The label information is not explicitly used in this loss function. Therefore, the model has to figure out the identity related factors that differentiate a utterance pairs besides the variation in content, accents, etc. This wide range of variation can lead to a dispersion of intra-class samples, thus rendering the embedding sensitive to noise. Furthermore, the number of triplets increases exponentially with the number of samples, which makes it hard to extract meaningful triplets to learn. Therefore learning with triplet loss can be slow to converge and result in suboptimal performance. To overcome these challenges, one can employ effective sampling strategies [10, 11] or training embedding networks on top of pretrained classification models [5, 9].

In this paper, we address the problem of training embedding networks with triplet loss by proposing a complementary loss function called intra-class loss. This loss acts as a regularizer that reduces the averaged intra-class distance variance of the final embedded features. The effects of this loss is twofold. First, by reducing intra-class distance variance, the embedded features for each class are more compact and less sensitive to noise. Second, by minimizing the variation in utterances due to content or recording condition, the model can subsequently focus on differentiating utterances based on identities. Hence, using intra-class loss can help stabilize training and result in performance improvement. In practice, we optimize an equivalence of intra-class distance variance, which is the averaged pair-wise distance of same-speaker utterances. This upperbound can be efficiently estimated without parametrized means as in [12] and can be combined with triplet loss without expensive overhead cost.

To validate our contribution, experiments are conducted on two benchmark datasets for speaker verification: VoxCeleb and VoxForge. In both datasets, our propose method improves the overall accuracy and accelerates the training of embedding learning with triplet loss. Our results are also competitive with state-of-the-art systems. Our code and pretrained models will be made available publicly.

2. Related Work

Below we discuss prior works on speaker embedding for recognition and verification as well as related work in computer vision which share similarities with our proposed method.

Conventionally, speaker representations are based on i-vectors [13]. To extract i-vectors, Baum-Welch statistics are computed from a Gaussian Mixture Model-Universal Background Model (GMM-UBM), which is learned using a sequence of feature vectors. I-vectors then can be used to compare utterances directly using cosine similarity or probabilistic linear discriminant (PLDA) [14, 15, 16]. To improve upon i-vectors, deep neural networks (DNNs) have been first applied to gradually replace each step in computing i-vectors traditional speaker recognition systems [17, 18].

With the recent advances in deep learning, research effort has been devoted to learn end-to-end DNNs for speaker classification and verification. One common task is to learn a good speaker embedding to compare utterances, which can be ad-

10.21437/Interspeech.2018-1685

dressed by two main types of approaches: learning a representation as a byproduct of classification or directly learning an embedding using distance based losses.

In the first approach, a DNN is trained to classify speakers and the activations of the final hidden layer are averaged over the utterance to create a "d-vector" [6]. D-vectors can be enhanced by concatenating multiple levels of representation [4], PLDA scoring [1], and data augmentation [7]. The speaker embeddings extracted in this manner are not discriminatively trained and therefore often require an classifier such as PLDA or another DNN.

In the second approach, the scoring scheme is fixed as the distance between embedded features, thus the DNNs are optimized with distance-based loss to directly extract the embeddings. The distance-based loss can be contrastive loss [9] or triplet loss [10]. Especially, triplet loss has shown improvements in speaker turn detection [8], speaker diarization [19], and text-independent verification [2, 3]. The main idea is that the distance between same-speaker utterances should be smaller than the distance between different-speaker utterances. The challenge of this approach is the wide range of variation of text-independent utterances. It is hard for a network to distinguish the speaker related factors from other factors, which can lead to suboptimal results. Therefore, the network is often pretrained for classification task in advance to achieve good performance [3, 9]. Pretraining with classification uses the explicit identity labels to the network into speaker discriminative features, thus filtering other sources of variation.

In this paper, we are interested the problem of large variation in text-independent utterances. In deep face recognition, increasing intra-class compactness has been shown to improve the discrimination power of the activation features of the last hidden layer [12]. We follow the same idea but in the embedding space. Regularzing same-class neighbors has also been applied in [20]. In our work, instead of minimizing the distances to means [12] or the empirical positive pair-wise distances [20], we regularize the soft upperbound derived from the intra-class variance, which is the averaged intra-class distance.

3. Proposed learning approach

In this section, we first present the general framework to learn an embedding space with triplet loss and discuss its pros and cons to motivate our new loss function, which is described subsequently.

3.1. Embedding Learning with Triplet Loss

Given a labeled training set $\{(x_i, y_i)\}$, in which $x_i \in \mathbb{R}^D, y_i \in \{1, 2, .., K\}$, embedding learning is a class of algorithms which learn a function f which maps an instance x into $f(x) \in \mathbb{R}^h$, i.e. an element of a h-dimensional space. In this new embedding space, we want the intra-class distances $d(f(x_i), f(x_j))/y_i = y_j$ to be minimized and the inter-class distances $d(f(x_i), f(x_j))/y_i \neq y_j$ to be maximized. To achieve such embedding, one method is to learn the projection that optimizes the triplet loss in the embedding space. A triplet consists of 3 data points: an anchor point x_a, a positive point x_p, and a negative point x_n such that $y_a = y_p$ and $y_a \neq y_n$. Following the embedding goal, we would like the 2 points (x_a, x_p) to be close together and the 2 points (x_a, x_n) to be further away by a margin α in the embedding space. Formally, we define the triplet loss to be minimized as:

$$L_t = \frac{1}{|T|}\sum [d(f(x_a), f(x_p)) - d(f(x_a), f(x_n)) + \alpha]_+ \tag{1}$$

where T is the set of all possible triplets of the training set and d is the Euclidean distance in the embedding space.

We can observe in Eq. 1 that the parameters of f are updated based on the relative distance difference between the positive and negative pairs. Embedded features can be spread out to achieve the margin, thus making the representation sensitive to noise. On the other hand, two speech segments can be differentiated by not only the speaker identities but also by the content of speech, accents, etc. This large intra-class variation can make triplet loss result in low accuracy, especially when trained from scratch. Our intra-class loss is proposed in the next section to address these problems.

3.2. Reducing intra-class variance in the embedding space

Let $S_c = \{(x_i, y_i)\}$ be the set of samples from the class c. We want to minimize the intra-class distance variance of c:

$$\min_f \sum_{x_i/y_i=c} \frac{d(f(x_i), \mu_c)^2}{n_c} \tag{2}$$

in which $n_c = |S_c|$ and the mean of class c features is $\mu_c = \sum_{x_i/y_i=c} \frac{f(x_i)}{n_c}$. Eq. 2 requires estimating the mean μ_c, which changes with each update. To address this problem, a possibility is to compute a moving average of μ_c, but this can be unreliable during early training stage and requires a hyperparameter to tune. To circumvent this issue, we instead minimize an upperbound of the variance, which uses the pair-wise squared distances within the class. This upperbound can be derived as follows:

$$\sum_{x_i/y_i=c} \frac{d(f(x_i), \mu_c)^2}{n_c} = \sum_{x_i/y_i=c} \frac{||f(x_i) - \sum_{x_j} \frac{f(x_j)}{n_c}||_2^2}{n_c}$$
$$= \sum_{x_i} \frac{|| \sum_{x_j} (f(x_i) - f(x_j))||_2^2}{n_c^3} \leq \sum_{x_i, x_j} \frac{||f(x_i) - f(x_j)||_2^2}{n_c^3} \tag{3}$$

One can observe that minimizing Eq. 3 can lead to a trivial solution when all samples are projected to a single point. This can encourage model collapse when training with triplet loss [10]. Hence, we optimize the squared root of Eq. 3 and devise a second upperbound:

$$\sqrt{\sum_{x_i, x_j} \frac{||f(x_i) - f(x_j)||_2^2}{n_c^3}} \leq \sum_{x_i, x_j} \frac{\sqrt{||f(x_i) - f(x_j)||_2^2}}{n_c \sqrt{n_c}}$$
$$= \sum_{x_i, x_j/y_i=y_j=c} \frac{d(f(x_i), f(x_j))}{n_c \sqrt{n_c}} \tag{4}$$

In Eq. 4, the objective is based on the true distance instead of the squared distance, which makes the loss more stable to model collapse [11]. Also, we propose a soft constraint that only requires each pair-wise distance to be smaller than a threshold β. In practice, because n_c is constant across minibatches, we choose the denominator to be n_c^2, thus formulating

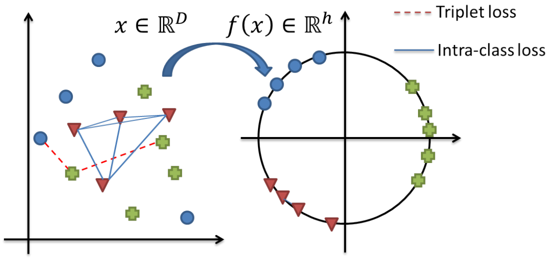

Figure 1: *Illustration of triplet loss and intra-class loss.*

the loss as the soft averaged pair-wise distance. Concretely, our intra-class loss function becomes:

$$\mathcal{L}_c(c) = \sum_{x_i, x_j / y_i = y_j = c} \frac{[d(f(x_i), f(x_j)) - \beta]_+}{n_c^2} \quad (5)$$

This new intra-class loss can be weighted by λ to be combined with the triplet loss in Eq. 1 (as illutrasted in Fig. 1) to form the final loss function:

$$\mathcal{L} = \mathcal{L}_t + \frac{\lambda}{K} \sum_c \mathcal{L}_c(c) \quad (6)$$

Using this intra-class loss as a regularizer has 2 main effects. Firstly, it prevents features to disperse in the embedding space, thus making the representation more robust to noise. Secondly, minimizing variance can reduce the influence of other factors such as speech content or record condition. Therefore, the learned model is more discriminative with respect to speaker identities. We also note that the distances calculated in intra-class loss can be effectively reused from triplet loss, thus reducing the overhead of adding a new loss function.

4. Experiments

We first describe the datasets and implementation details before discussing the experiments and the results. Our codes and models will be publicly available.

4.1. Data and metrics

VoxCeleb. This dataset contains videos of celebrities collected from Youtube [9]. There are more than 140K utterances of 1251 speakers in a free context. 40 speakers are reserved as test data for the verification protocol. We report Equal Error Rate (EER) computed using the provided trial pairs.

VoxForge. This is an open source speech database, where speakers voluntarily contribute speech data for development of open resource speech recognition systems [1]. The utterances have lower variability as the text is read and the data is collected in a clean environment. We follow the same protocol as in [5]. From 300 chosen speakers, three subsets of 100 speakers are constructed for training, development, and evaluation. The training set is used to train / finetune embedding networks. The development set is used to choose a threshold based on EER, and the threshold is applied on the evaluation set to report Half Total Error Rate (HTER).

4.2. Implementation Details

CNN architecture. Our model is built using the ResNet architecture[21]. There are 31 layers configured as in Tab. 1. The key modification is the statistical pooling layer, which concatenates both mean and standard deviation of the previous layer across the whole sequence in time. We also change the configuration of the first max pooling layer to work only on the time domain.

Table 1: *ResNet architecture used in the experiments. Residual block follows the same definition in [21]. Each convolution layer is followed by ReLU and batch normalization.*

Layer	# filt.	Stride
Conv 5×5	64	2×2
Max Pool 3×1	-	2×1
Res. block	64	2×2
Res. block	128	2×2
Res. block	256	2×2
Conv 1×9	256	1×1
Conv 1×9	512	1×1
Stats Pool $n \times 1$	-	1×1
L_2 norm	-	-

Feature extraction. For each utterance, a spectrogram is computed using 512-point FFT, a temporal window of 25ms, and a window shift of 10ms. Mean and variance normalization on each frequency bin is performed as in [9].

Training details. All networks are trained using RMSProp optimizer [22] with a 10^{-3} learning rate. Each minibatch contains 120 samples, and negative triplets are sampled using distance-based sampling method [11]. We train with truncated utterances of 2 seconds or 3 seconds as input. For hyperparameters, we choose $\alpha = 0.2$, $\beta = 0.2$, and $\lambda = 0.001$.

4.3. Experimental Results

Training from scratch. In this setting, a ResNet is initialized randomly and then learned on the VoxCeleb training set using either triplet loss alone or in combination with intra-class loss.

In Fig. 2, we visualize the EER on the VoxCeleb validation set as the model training progress. One can observe that intra-loss accelerates the training speed. The model not only converges faster but also to a lower EER. In Tab. 2, EERs on the validation and test sets with different utterance input lengths are shown. Intra-class loss substantially improves the overall performance of the deep model. The EER is reduced relatively by 14% for 2s-segment input and 7% for 3s-segment input. Overall, 2s-segment input yields worse EER in comparison to using 3s. However, it is important to note that when intra-loss is added, the model learned with 2s-segment input can still reach the same performance as in using 3s-segment. This shows intra-class loss can enhance the embedding space even when the input signals contain less information.

Embedding learning from a pretrained model. In this experiment, a ResNet for speaker recognition is first trained with softmax loss using the speakers in the VoxCeleb training set. Then the convolutional weights are frozen and the last embedding layer is trained with the embedding losses.

When using the activation of the last hidden layer of the pretrained models, one can achieve 14.43% and 11.96% EER on the test set using input of 2s or 3s respectively. As the models were pretrained to predict explicitly the identities, they can focus more on the discriminative features for classification. Therefore, training an embedding layer on top of these models can significantly enhance the results. As the initial model is al-

[1] http://www.voxforge.org/

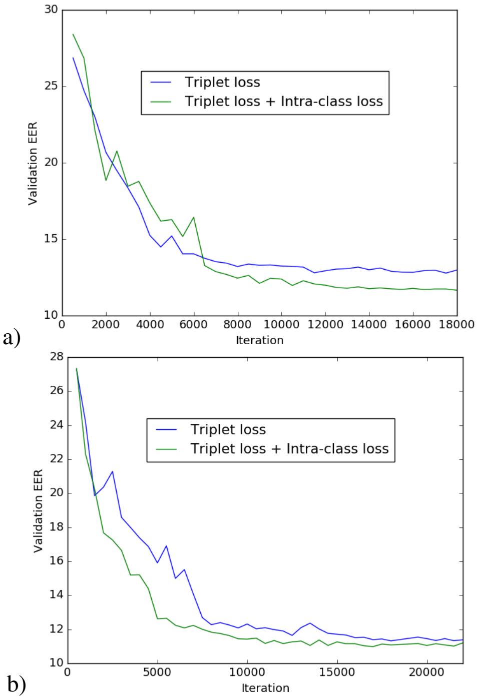

Figure 2: *EER on the validation set of VoxCeleb during training with training samples of different lengths: (a) 2s or (b) 3s.*

Table 2: *Ablation study of how using intra-class loss effect the EER on the validation and test set of VoxCeleb. We also compare how results differ when the training utterances are truncated to 2s or 3s.*

Setting	In. len.	Loss	Val. EER	Test EER
Scratch	2s	Trip.	12.73	12.44
		Trip. + Intra.	11.71	10.74
	3s	Trip.	11.17	10.68
		Trip. + Intra.	10.31	9.93
Pretrained	2s	Softmax	-	14.43
		Trip.	7.21	8.31
		Trip. + Intra.	6.30	7.97
	3s	Softmax	-	11.96
		Trip.	6.84	8.20
		Trip. + Intra.	6.03	8.12

ready well-trained, both cases of with and without intra-class loss yield statistically similar EERs.

In Tab. 3, we compare our method with state-of-the-art systems. Our embedding network with intra-class loss outperforms traditional methods using factor analysis with GMM-UBM. When comparing with other embedding methods, one can see that bidirectional LSTM trained with triplet loss [8] cannot capture the discriminative variation of the data well. Meanwhile, our systems perform on par with [9], which uses pretrained classification model and contrastive loss for embedding learning. This agrees with the conclusion from [11] that shows similar performance between contrastive loss and triplet loss.

Verification on VoxForge. In this experiment, we use the pretrained classification network from VoxCeleb and the embedding layer is learned using either the VoxCeleb or the VoxForge training sets and report test results on the VoxForge evaluation set. In evaluation stage, all distances from a probe utterance to every enrollment utterance is computed and the identity is simply decided based on a threshold. The development set is used to set the threshold with lowest EER. HTER is reported on the evaluation set using this threshold.

Table 3: *Comparison of our embedding method to other state-of-the-arts on VoxCeleb dataset. (*are reported in [9])*

GMM-UBM*	15.0
i-vector + PLDA*	8.8
Bi-LSTM Embedding [8]	14.1
CNN Embedding [9]	7.8
Ours (Pretrained + Intra.)	7.97

Table 4: *Comparison of our embedding method to other state-of-the-arts on VoxForge dataset. (*are reported in [5])*

VoxCeleb	Triplet loss	2.09
	Triplet + Intra-class loss	1.50
VoxForge	Triplet loss	1.69
	Triplet + Intra-class loss	1.16
	GMM-UBM*	3.05
	i-vector + PLDA*	5.87
	ISV*	2.40
	CNN Clas. [5]	1.20

Tab. 4 shows our ablation results together with other methods. Comparing our models when using intra-class loss against using triplet loss only, we can observe a significant relative reduction of 30% in EER in both cases of training sets. Interestingly, the model trained with intra-class loss on out-domain data (VoxCeleb) can still perform better than the model trained with only triplet loss on in-domain data (VoxForge). The improvement shows that intra-class loss can help adapting models to new datasets. This can be explained as the variance of each class is regularized, the learned embedded features are less sensitive to noises which are not presented in the original dataset.

When comparing to other methods on this dataset, our deep embedding models are better than traditional factor analysis systems. Using intra-class loss, our model can work slightly better than the deep method that uses classification CNNs to model specific speakers [5]. It is important to note that in our system, we do not build a specific model for each speaker using their enrollment data. Only the distances from a probe utterance to all enrollment data are used to verify directly. This advantage allows our system to be used when there is no enrollment phase, for example in the setting of speaker verification in the wild.

5. Conclusion

We have presented a novel loss function as an supportive learning goal to improve the speaker embedding spaces learned by deep neural networks. By reducing the averaged intra-class pair-wise distances, our loss aims to increase the robustness of learned features. The results of speaker verification task on two public datasets, VoxCeleb and VoxForge, validate the improvement of our approach. Models learned with intra-class loss not only converge faster but also achieve better accuracy. In the future, we plan to conduct more experiments different strategies for reducing intra-class variance such as using moving averaged class means [12] or using embedding margin based loss [11].

6. Acknowledgement

This work was supported by the EU projects EUMSSI (FP7-611057) and MuMMER (H2020-688147).

7. References

[1] 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.

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

[3] 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.

[4] 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.

[5] H. Muckenhirn, S. Marcel *et al.*, “Towards directly modeling raw speech signal for speaker verification using cnns,” in *ICASSP*. IEEE, 2017.

[6] 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.

[7] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” *ICASSP, Calgary*, 2018.

[8] H. Bredin, “TristouNet: Triplet Loss for Speaker Turn Embedding,” in *ICASSP*. IEEE, 2017.

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

[10] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: a Unified Embedding for Face Recognition and Clustering,” in *CVPR*, 2015.

[11] R. Manmatha, C.-Y. Wu, A. J. Smola, and P. Krähenbühl, “Sampling matters in deep embedding learning,” in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2859–2867.

[12] 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.

[13] 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.

[14] S. J. Prince and J. 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.

[15] S. Cumani, O. Plchot, and P. Laface, “Probabilistic linear discriminant analysis of i-vector posterior distributions,” in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7644–7648.

[16] S. Madikeri, M. Ferras, P. Motlicek, and S. Dey, “Intra-class covariance adaptation in plda back-ends for speaker verification,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5365–5369.

[17] S. H. Ghalehjegh and R. C. Rose, “Deep bottleneck features for i-vector based text-independent speaker verification,” in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 555–560.

[18] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically-aware deep neural network,” in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1695–1699.

[19] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, “Speaker diarization using deep neural network embeddings,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4930–4934.

[20] K. Q. Weinberger, J. Blitzer, and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” in *Advances in neural information processing systems*, 2006, pp. 1473–1480.

[21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *CVPR*. IEEE, 2016.

[22] T. Tieleman and G. Hinton, “Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude,” *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, 2012.