Deep multi-metric learning for text-independent speaker verification

Jiwei Xu\textsuperscript{a}, Xinggang Wang\textsuperscript{a,∗}, Bin Feng\textsuperscript{a}, Wenyu Liu\textsuperscript{a}

\textsuperscript{a}School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract

Text-independent speaker verification is an important artificial intelligence problem that has a wide spectrum of applications, such as criminal investigation, payment certification, and interest-based customer services. The purpose of text-independent speaker verification is to determine whether two given uncontrolled utterances originate from the same speaker or not. Extracting speech features for each speaker using deep neural networks is a promising direction to explore and a straightforward solution is to train the discriminative feature extraction network by using a metric learning loss function. However, a single loss function often has certain limitations. Thus, we use deep multi-metric learning to address the problem and introduce three different losses for this problem, i.e., triplet loss, n-pair loss and angular loss. The three loss functions work in a cooperative way to train a feature extraction network equipped with Residual connections and squeeze-and-excitation attention. We conduct experiments on the large-scale VoxCeleb2 dataset, which contains over a million utterances from over 6,000 speakers, and the proposed deep neural network obtains an equal error rate of 3.48%, which is a very competitive result. Codes for both training and testing and pretrained models are available at https://github.com/GreatJiweix/DmmlTiSV, which is the first publicly available code repository for large-scale text-independent speaker verification with performance on par with the state-of-the-art systems.

Keywords: Speaker verification, n-pair loss, angular loss, triplet loss, SENet

1. Introduction

SV (speaker verification) is a key technology for intelligent interaction. It can be widely used in financial payment, criminal investigation, national defense and other fields. It is one application in speech recognition that aims to verify a claimed identity based on his/her utterance [1]. This task is a 1 : 1 match where one speaker’s voice is matched to a particular template. SV can be categorized into text-dependent and text-independent [2, 3]. The text-dependent SV system requires the speech to be produced from a fixed or prompted text phrase, while the text-independent SV system operates on unconstrained speech. Therefore, text-independent SV is a more challenging problem, but it is more useful in practical applications.

Generally, a deep learning-based SV system contains the training step and testing step [3]. In the training step, we use a large collection of utterances to train an SV neural network. The learned deep neural network model is used as a universal feature extractor for any testing speaker. Then in the testing step, two different utterances are separately sent to the learned deep model for feature extraction, and we compute the similarity based on the two feature vectors to perform SV. In the test phase, the false acceptance/rejection rates depend on the predefined threshold [4]. The equal error rate (EER) metric projects the error when the two aforementioned rates are equal. The basic training verification system is shown in Figure 1.

Before the era of deep learning, traditional SV models made remarkable achievements. For example, the Gaussian mixture model with a universal background model (GMM-UBM) [5] uses a sufficiently large speech dataset of several hours from multiple sources. For a D-dimensional feature vector \(x\), the mixture density used for the likelihood function is defined as

\[
P(x|\lambda) = \sum_{i=1}^{M} w_i p_i(x).
\]

The density is a weighted linear combination of \(M\) unimodal Gaussian densities, \(p_i(x)\), each of which is parameterized by a mean \(D\times 1\) vector, \(\mu_i\), and a \(D\times D\) covariance matrix \(\Sigma_i\) [6]:

\[
P_i(x) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_i)^T (\Sigma_i^{-1})(x - \mu_i) \right\}.
\]

Furthermore, the mixture weights \(w_i\) satisfy the constraint

* Corresponding author

Email address: xgwang@hust.edu.cn (Xinggang Wang)

Preprint submitted to Elsevier

July 22, 2020
\( \Sigma_{i=1}^{M} w_i = 1 \). The GMM-UBM system is a straightforward generative approach for an SV task, using a sufficiently large speech data sample of several hours from multiple sources. The UBM is represented as follows:

\[ \lambda = (w, \mu, \Sigma), \]

where \( w \) is the weight of the \( i \)-th Gaussian component, \( \mu \) represents the mean and \( \Sigma \) is the covariance matrix of the \( i \)-th Gaussian component. Each speaker is represented as a GMM derived by maximum-a-posteriori (MAP) adaptation from UBM.

Apart from GMM-UBM, i-vector \([7]\) is another state-of-the-art SV framework. It models the speaker factors and channel factors and converts the utterance of the speaker identity to a low-dimensional embedding representation. The i-vector representation, whose role is to represent an utterance of arbitrary duration by a vector of fixed dimension which we denote by \( d \) (in the range of 400 to 600), originates from the joint-factor analysis \([8]\) method. The SV systems based on i-vector represent the high-dimensional GMM supervector in a transform vector (TV) space which reduces the supervector into low-dimensional factors. In TV space, the GMM supervector, is projected as

\[ M = m + Ti, \]

where \( T \) is a low-rank factor loading matrix, \( m \) is channel and \( i \) is the speaker-independent supervector whose prior distribution is assumed to be standard normal:

\[ i \sim N(0, I). \]

2. Related Work

The traditional SV models, such as GMM-UBM \([5]\) and i-vector \([7]\), have been the state-of-the-art approaches for a long time. All the above-mentioned methods rely on low-dimensional input features extracted by using mel-frequency cepstrum coefficients (MFCC), however, MFCC is known to suffer from performance degradation under real-world noise conditions as demonstrated by \([22] [23]\). Deep Convolutional Neural Networks (DCNN) have proven to be effective to extract intrinsic features from noisy data, thus various speech applications \([24] [25] [26]\) have been proposed based on DCNN. Why is convolution useful for text-independent SV? First, text-independent SV is based on the intrinsic feature of a human utterance, which can be extracted from small fragments in speech. Convolution allows spatial translation invariance and operates on local features. Thus, convolution is more suitable to extract speech features. Second, the convolutional neural network benefits from data augmentation. Thirdly, it is computationally efficient. Ultimately, similar to image classification and face recognition, SV is suitably solved using DCNN.

In recent years, deep learning, especially deep metric learning, has achieved outstanding results in face verification and re-identification problems. The most commonly used metric learning loss function is the triplet loss \([13]\). The goal of triplet loss is to minimize the distance of the same class pairs and maximize the distance of different class pairs. Although the triplet loss has achieved great results in many tasks, it is restricted to the class imbalance problem and needs a long training time to converge. N-pair loss \([27]\) pays more attention to the information of one negative sample in each optimization. To reduce the training burden, each mini-batch selects \( N \) pair of examples from \( N \) different classes and builds \( N \) triplet to accelerate the model convergence. Different from triplet loss and N-pair loss using the euclidean distance, angular softmax loss \([19]\) modifies the softmax loss function to learn angularly discriminative embeddings and use a controllable parameter to constrain the intra-speaker variation of the learned embeddings. Based on the above analysis, we believe that the three loss functions exhibit a certain complementarity, and combine them to further optimize the network. Inspired by the recent SENet (squeeze-and-excitation networks) method \([23] [24]\), our study on SE block illustrates that different channels of a feature map play different roles in specific objects. The SE block discards the pooling layer and uses \( 1 \times 1 \) convolutional layer to replace the fully-connected layer for learning spatial information. Meanwhile, the SE block is also computationally inexpensive and imposes only a slight increase in model complexity and computational burden.

At present, there are many methods of metric learning that have achieved good results in SV. \([11]\) introduced triplet loss \([29]\) into SV and achieved very competitive results. \([30]\) used angular softmax \([31]\) achieved an obvious performance improvement compared with other methods in the SV task. The combination of center loss \([32]\) and softmax loss has also been shown to provide good results in the SV task \([33] [24]\). Regarding the effectiveness of angular softmax loss, center loss and triplet loss
for face verification, it is worth exploring their power in the task of SV, which has never been studied before.

3. Method

In this section, we first describe the architecture of our network in Figure 2. As the combination of ResNet [21] and SENet [28, 34] has been proven to achieve a good performance on the person re-identification (re-ID) task [35], we will embed the SE block in the ResNet to explore the channel-wise relationship for the SV task. Finally, we will analyze the previous metric losses and combine them to complement each other.

3.1. Training Architecture

As shown in Figure 2, the training architecture of our method can be divided into two components: the ResNet-50 backbone and the loss functions. Here, the ResNet-50 backbone serves as a multi-scale feature extractor. In the second component, we separately calculate the triplet loss, N-pair loss, angular loss and softmax loss, and then devise a combination of those losses to optimize our network.

3.2. SENet Block

In previous studies [36, 37], the importance of attention has been proven in SV. The network will pay more attention to the discriminative local regions for SV. In recent works, the squeeze-and-excitation network (SENet) and Manes [35] illustrate that different channels of a feature map play different roles in specifying objects/parts. Taking those into consideration, we will introduce the SE block to the network in order to improve the performance of the network.

As illustrated in Figure 3, the proposed SE block discards the pooling layer and replaces fully-connected layers with 1 convolutional layers to regain the spatial information [35].

Given the input feature map $F_i$ of SE block, the output attention map $M$ can be computed as follow:

$$M = \text{Sigmoid} (\text{Conv(ReLU(Conv(F_i))))},$$

where the two Conv operators are $1 \times 1$ convolution. The roles of these two $1 \times 1$ convolutional layers are various. The inner one is used for squeeze and the outer one is used for excitation. The $1 \times 1$ convolution kernel can greatly increase the nonlinear characteristics (using the nonlinear activation function followed by the non-loss resolution) while keeping the feature map scale constant (i.e., without loss of resolution), making the network very deep. Via the SE block, we can obtain the feature map $F_o$ with attention information as

$$F_o = F_i \times M + F_i.$$  

3.3. Triplet loss

At each training iteration, we sample a mini-batch of triplets, for each of which $T = (X_a, X_p, X_n)$ consists of an anchor point $X_a$, associated with a pair composed of a positive sample $X_p$ and a negative sample $X_n$. The goal of triplet loss is to push away the negative point $X_n$ from the anchor $X_a$ by a distance margin $m > 0$ compared to the positive $X_p$. Triplet loss is usually defined as follows:

$$L_{tri} = \left( ||X_a - X_p||^2 + m - ||X_a - X_n||^2 \right).$$  

Although triplet loss has achieved good results in many tasks, it has strict requirements on sampling strategies and takes a long training time to converge. Therefore, this paper introduces n-pair loss [38] and angular loss [39].

3.4. N-pair loss

The traditional triplet loss only pays attention to the information of one negative sample in each optimization. Our assessment is that the training is slow and the information concerned is not comprehensive enough, so we introduce $(N + 1)$-triplet loss that optimizes the identification of a positive example from $N - 1$ negative examples. By comparing with Figure 3, we can observe that the traditional triplet loss is only a special case of $(N + 1)$-triplet loss, where $N = 2$.

$$L_{(N+1)-\text{tuplet}} = \log(1 + \sum_{i=1}^{N-1} \exp(f^T f_i - f^T f)), $$

where $f$ is an embedding kernel defined by the deep neural network. To reduce the training burden while making full use of each batch of training samples, we propose a new training strategy. The corresponding $(N+1)$-tuplet loss, which we refer to as the n-pair loss, can be formulated as follows:

$$L_{n\text{-pair}} = \sum_{i=1}^{N} \log(1 + \sum_{j \neq i} \exp(f^T f_i - f^T f_j)).$$

Both the triplet loss and the n-pair loss only consider the distance between the anchor and the positive example and the anchor and the negative example; however, they do not consider the distance between the positive and the negative example, so the information is not comprehensive. Therefore, this paper introduces a more easily trained loss function, i.e., angular loss [39].

3.5. Angular loss

Let us first imagine such an example, assuming three points $X_a$, $X_p$, and $X_n$, and these three points form a triangle $\triangle apn$, whose edges are denoted as $dp = X_a - X_p$, $dn = X_a - X_n$, $dp = X_a - X_p$, the traditional triplet loss can be seen as $d_{ap} + m < d_{an}$. We consider that the anchor and the positive example share the same label, so we can also optimize $d_{ap} + m < d_{pn}$. Within the triangle $\triangle apm$, our goal is to find a solution that satisfies $d_{ap} < d_{pm}$ and $d_{ap} < d_{an}$, taking into account the fact that we can set a threshold that guarantees $\angle n \leq \angle a$, in which $\alpha$ is our pre-set threshold.
However, in the actual optimization process, it is not very stable to consider only \( \angle n \). Figure 5 (a) is a special example, in which \( \angle apm, \angle a > 90^\circ \), and \( d_{apm} \) does not decrease when \( X_m \) is transformed to the position of \( X_m' \).

To fix this issue, we re-construct a triplet triangle for more stable optimization. Let us set point \( X_c \) to the center point of \( X_a \) and \( X_p \), \( X_c = (X_a + X_p) / 2 \). \( X_m \) is a point on the circle centered at \( X_c \) and satisfies that \( X_mX_a \) is vertical to \( X_cX_m \). The triangle we re-construct is \( \triangle mcn \); at this point, the angle \( \angle n' \) is the angle we want to optimize. In the triangle \( \triangle mcn \), we can see that this formula is satisfied:

\[
\sin \angle n' = \frac{\|X_m - X_c\|}{\|X_a - X_c\|} = \frac{\|X_a - X_p\|}{\|X_m - X_c\|} \leq \tan \alpha. \tag{11}
\]

Considering that \( X_c \) is the center point of \( X_p \) and \( X_a \), therefore \( \|X_m - X_c\| = \|X_p - X_c\| = 0.5 \times \|X_p - X_a\| \). According to Eq. (11), we obtain the following equation:

\[
\|X_p - X_a\|^2 \leq 4\|X_c - X_a\|^2 \tan^2 \alpha. \tag{12}
\]

Inspired by the triplet loss our angular loss consists of minimizing the following hinge loss:

\[
L_{ang} = \max(4\|X_a - X_c\|^2 \tan^2 \alpha - \|X_p - X_a\|^2, 0). \tag{13}
\]

### 3.6. Multitask learning

Multi-task learning has achieved good results in many areas, such as face verification [40], person re-ID [41] [35], SV [42], metric learning [43] [44], etc. Considering that these tasks and the SV task have certain similarities, we can also use this method for training. The loss functions learned for metric learning in this paper have their advantages and disadvantages. The benefits of the combination of n-pair loss and angular loss have been verified in [39]. However, to the best of our knowledge, no studies exist that have used triplet loss with them. Our approach is to disclose some complementarity between them, which has been verified in our experiments. In addition, we use softmax loss with them to form a multi-task learning to further improve performance. As shown in Figure 2, the two tasks share the same backbone network. In training, the corresponding three loss functions are optimized jointly. The overall loss is defined as follows:

\[
L = \lambda_{n-pair} L_{n-pair} + \lambda_{tri} L_{tri} + \lambda_{ang} L_{ang} + \lambda_{soft} L_{soft}, \tag{14}
\]

where \( \lambda_{n-pair}, \lambda_{tri}, \lambda_{ang} \) and \( \lambda_{soft} \) are weight factors for the loss functions.

### 4. Experiments

#### 4.1. Dataset

We perform experiments on the VoxCeleb [45] and VoxCeleb2 [46] datasets. We train our model on the training set of VoxCeleb2, which contains 1,128,246 utterances from 5,994 speakers. All models, including our model and the compared models, are tested on the testing set of VoxCeleb, which contains 37,720 utterance pairs from 40 speakers. The average duration of training and testing data are 8.24s and 8.28s, respectively.

The utterances are extracted from videos on YouTube. Since these utterances originate from natural scenes, the signal quality is not very good and the background is noisy. Therefore, we firmly believe that if we can obtain good experimental results on this dataset, our method can be extended to more datasets and applied in the wild.

#### 4.2. Data representation

We first use traditional digital signal processing methods to characterize speech signals. The libRosa package [47] is used for speech feature extraction. Spectrograms are generated in a sliding window fashion using a Hamming window with a width 20ms and a step of 10ms, in exactly the same manner as that of [48]. Then we can obtain a vector whose dimension is the number of frames \( \times \) 161. Without loss of generality, we randomly intercept the three-second speech utterance, convert it to a 300\( \times \) 161 vector, and then copy it three times, constructing a 3\( \times \)300\( \times \)161 vector similar to an image. If the duration of an utterance is less than 3 seconds, we will copy the utterance to 3 seconds.
4.3. Training configurations

We implement our network based on PyTorch [49]. ResNet has been proven to be very effective in many tasks, so we also chose ResNet as the backbone. Considering that [46] chose ResNet-50 as the backbone, for fair comparison, we also chose the network as our backbone. We take the ResNet-50 model pretrained on ImageNet as the backbone. We extract the conv-2, conv-3 and conv-4 feature maps to generate attention masks by SE blocks, and add them back into the mainstream. The last conv-5 feature map is used for generating the final utterance identity feature. We adopt the PK sampling strategy to form every mini-batch. The values of both P and K are set to 64 and 2, respectively. This is to consider the set requirements of n-pair loss and angular loss and to ensure the difficulty of hard negative samples when training triple loss using a batch-hard sampling strategy. The activation function of the last convolutional layer is changed from ReLU to PReLU [50], which can effectively improve the over-fitting problem of the model. λ_{n-pair}, λ_{soft}, λ_{tri}, and λ_{ang} are set to 0.5, 0.1, 1.0 and 1.0, respectively. The margin α in Eq. (8) is set to 45°. We adopt the Adam optimizer with an initial learning rate of 3 × 10^{-4} in our experiments to minimize the three losses.

4.4. Comparisons with the state-of-the-art methods

We evaluated our proposed method against 13 existing methods on VoxCeleb. As shown in Table 1 our model achieves the best result among the compared methods. As shown in the table there are currently two ways [46, 51] that can obtain an EER lower than 4%, and our method exceeds them.

4.5. Ablation study

We further perform several ablative experiments to verify the effectiveness of each individual loss function of our proposed model. The results are shown in Table 2. We always apply softmax classification loss, which is simple and can stabilize the training process. The EER of triplet loss, N-pair loss and angular loss are 5.00%, 4.90% and 5.72%, respectively. It is found that angular loss can not achieve better results than triplet loss, which occurs triplet loss employs a semi-hard sampling strategy while the angular loss function does not. Integrating triplet loss, N-pair loss and angular loss can obtain the best results, which is 3.48% EER.

Through the experiments, we can see that the best results are obtained by training multiple losses together. We analyze this combination approach because there is some complementarity between the three losses. First, if a strict sampling strategy and a large amount of training time are adopted, triplet loss can achieve good results, but the selection of margin is difficult during training. The sampling strategy we use in this paper is the semi-hard example mining method which was presented in [29]. In addition, triplet loss compares one positive sample with one negative sample during training, and we contend that the compared examples in each batch are insufficient. Second, the n-pair loss can make full use of the training data in a batch by considering the pairwise information between negative samples, which speeds up the training process and achieves good results but it does not require a good sampling strategy during training. Third, compared with the traditional triple loss, the n-pair loss is defined on the absolute distance between points. The proposed angular constraint offers three advantages: 1) Angle is a similarity-transform invariant metric, which is insensitive to the magnitude of features. 2) The original triple loss only considers the two sides of the triangle but \( \angle n \) takes into account the three sides of the triangle, so the information considered is more comprehensive. 3) In the original triplet loss, it is difficult to set a standard threshold \( \alpha \). However, in our angular loss, setting the threshold \( \alpha \) is relatively easy because it has a clear
Table 1: Speaker verification results on the standard VoxCeleb benchmark. Results of the compared methods are quoted from their original papers.

| Method          | Dataset(s) | Architecture          | Pooling             | EER (%) |
|-----------------|------------|-----------------------|---------------------|---------|
| VoxCeleb        | VoxCeleb1  | PLDA+SVM              | Variable Length     | 8.80    |
| VoxCeleb        | VoxCeleb1  | VGG-M                 | Variable Length     | 10.2    |
| VoxCeleb        | VoxCeleb1  | VGG-M                 | Variable Length     | 7.80    |
| CNN-TAP         | VoxCeleb1  | Thin ResNet-34        | Multi-Crop          | 5.27    |
| i-vector        | VoxCeleb1+PRISM | PLDA              | Multi-Crop          | 5.39    |
| TDNN            | VoxCeleb1+PRISM | TDNN           | Multi-Crop          | 4.70    |
| LM              | VoxCeleb1+PRISM | TDNN           | Multi-Crop          | 4.69    |
| LDE-ASoftmax    | VoxCeleb1  | Thin ResNet-34        | LDE                 | 4.41    |
| TDNN            | VoxCeleb1  | TDNN                 | Attentive Stat.     | 3.85    |
| VoxCeleb2       | VoxCeleb1+PRISM | ResNet-50       | Variable Length     | 4.19    |
| Ours            | VoxCeleb2  | ResNet-50             | Multi-Crop          | 4.43    |
|                 | VoxCeleb2  | ResNet-50             | Average Dist.       | 3.95    |
|                 | VoxCeleb2  | ResNet-50             | Average Dist.       | 3.48    |

Table 2: Ablation studies of the loss functions.

| Loss function      | EER (%) |
|--------------------|---------|
| Triplet loss       | 5.00    |
| N-pair loss        | 4.90    |
| Angular loss       | 5.72    |
| Softmax loss       | 3.48    |

5. Conclusion

In this paper, we investigate the application of triplet loss, n-pair loss and angular loss in the task of text-independent speaker verification (SV) based on deep learning. A single-metric loss function often has certain limitations and we found that multiple metric losses often have certain complementarity. Therefore, better results can be achieved by combining multiple metric learning losses. To the best of our knowledge, this is the first time that multiple metric learning loss functions have been applied to the field of text-independent SV. Inspired by the fact that attention can localize the most discriminative local regions for SV, we introduce the SE block into the network to further improve performance. Large-scale experiments show that our method has achieved very competitive results on the VoxCeleb1 test set and ablative studies confirm the effectiveness of the proposed deep multi-metric learning. To facilitate the research of text-independent SV, we have release all the training and testing source codes as well as pretrained models. We believe that our method can also be applied to tasks such as image retrieval, face recognition, and person re-identification, which will be explored in our future work.

Acknowledgments

This work was supported by NSFC (No. 61876212, No. 61733007, No. 61773176 and No. 61572207) and National Key R&D Program of China (No. 2018YFB1402600). We sincerely thank the anonymous reviewers for their helpful reviews.

References

To explore our future work.

References

[1] C. Zhang, K. Koishida, J. H. Hansen, Text-independent speaker verification based on triplet convolutional neural network embeddings, IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP) 26 (9) (2018) 1633–1644.
[48] CoinCheung, Spherereid, https://github.com/CoinCheung/Spherereid

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, 2019, pp. 8024–8035.

[50] F. Zuo, X. Liu, Dpgan: Prelu used in deep convolutional generative adversarial networks, in: Proceedings of the 2019 International Conference on Robotics Systems and Vehicle Technology, 2019, pp. 56–61.

[51] K. Okabe, T. Koshinaka, K. Shinoda, Attentive statistics pooling for deep speaker embedding, Proc. Interspeech 2018 (2018) 2252–2256.

[52] W. Cai, J. Chen, M. Li, Exploring the encoding layer and loss function in end-to-end speaker and language recognition system, in: Proc. Odyssey 2018 The Speaker and Language Recognition Workshop, 2018, pp. 74–81.

[53] X. Li, X. Shen, Y. Zhou, X. Wang, T.-Q. Li, Classification of breast cancer histopathological images using interleaved densenet with senet (idsnet), Plos one 15 (5) (2020) e0232127.

[54] W. Yan, Y. Hua, Deep residual senet for foliage recognition, in: Transactions on Edutainment XVI, Springer, 2020, pp. 92–104.