

The following publication Y. Tu, M. -W. Mak and J. -T. Chien, "Contrastive Self-Supervised Speaker Embedding With Sequential Disentanglement," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 32, pp. 2704-2715, 2024 is available at https://doi.org/10.1109/TASLP.2024.3402077.

Contrastive Self-Supervised Speaker Embedding With Sequential Disentanglement

Youzhi Tu, Man-Wai Mak, *Senior Member, IEEE,* and Jen-Tzung Chien, *Senior Member, IEEE*

***Abstract*—Contrastive self-supervised learning has been widely used in speaker embedding to address the labeling challenge. Contrastive speaker embedding assumes that the contrast between the positive and negative pairs of speech segments is attributed to speaker identity only. However, this assumption is incorrect because speech signals contain not only speaker identity but also linguistic content. In this paper, we propose a contrastive learning framework with sequential disentanglement to remove linguistic content by incorporating a disentangled sequential variational autoencoder (DSVAE) into the conventional contrastive learning framework. The DSVAE aims to disentangle speaker factors from content factors in an embedding space so that the speaker factors become the main contributor to the contrastive loss. Because content factors have been removed from contrastive learning, the resulting speaker embeddings will be content-invariant. The learned embeddings are also robust to language mismatch. It is shown that the proposed method consistently outperforms the conventional contrastive speaker embedding on the VoxCeleb1 and CN-Celeb datasets. This finding suggests that applying sequential disentanglement is beneficial to learning speaker-discriminative embeddings.**

***Index Terms*—Speaker verification, speaker embedding, contrastive learning, disentangled representation learning, variational autoencoder.**

I. Introduction

SPEAKER verification (SV) systems aim to authenticate registered speakers and reject non-registered speakers. Modern SV systems mostly adopt a speaker embedding network (front-end) for extracting speaker representations and a scoring back-end for computing the similarity between the enrollment and test utterances. Classical speaker embedding usually uses convolutional neural networks or their variants to process the frame-level acoustic features [1]–[3]. To summarize the frame-level representations into an utterance-level embedding, various aggregation strategies have been employed [2]–[6]. Also, the margin-based classification losses, such as AMSoftmax [7] and AAMSoftmax [8], are often applied to achieve state-of-the-art performance.

The speaker embedding networks mentioned earlier require a large amount of speech data with speaker labels for training. The requirement of speaker labels poses a challenge to system development because manually labeling massive amount of data is expensive and time-consuming. Self-supervised learning has recently emerged as a viable alternative for training the speaker embedding networks to circumvent the labeling challenge [9]–[17].

Self-supervised speaker embedding generally uses contrastive learning [9]–[14] or non-contrastive learning [15]–[17]. Contrastive learning requires negative samples from a class different from that of the positive ones to create contrast. Conversely, non-contrastive learning maximizes the similarity between the speaker representations of different views or augmentations of the same utterance, thereby not requiring negative samples. Contrastive speaker embedding inevitably faces the class collision issue [18], i.e., the negative samples may come from the same speaker as the positive samples in a mini-batch. This problem could drive away the embeddings belonging to the same speaker, resulting in non-discriminative embeddings. Nevertheless, the study in [12] has demonstrated that the probability of a mini-batch containing repeated speakers is remarkably low when an appropriate batch size (e.g., 256) is used on a medium-size dataset such as VoxCeleb [19]. This paper focuses on contrastive embedding for text-independent SV.

Contrastive speaker embedding assumes that the contrast between the positive and negative pairs is due to speaker identity rather than other explanatory factors of variation [20], [21] such as linguistic contents and languages. However, speaker embeddings contain a variety of information besides speaker identity [22], [23], and non-speaker factors can also contribute to the contrast between the positive and negative pairs. This erroneous contrast can introduce nuisance information to the embeddings, causing performance degradation. Therefore, it is essential to disentangle speaker factors from the other factors of variation and only use the speaker factors for contrastive learning to ensure that the learned embeddings are *speaker-discriminative*.

Disentangled representation learning aims to learn independent factors of variation in an embedding space that are responsible for generating the data [20], [21]. It encourages well separation of the underlying generative factors and allows easy removal of nuisance factors, producing the nuisance-invariant representations. This property makes it amenable to domain-invariant speaker modeling [24], given that the speaker factors can be disentangled from the domain factors. However, according to [25], unsupervised disentangled representation learning is basically impossible without inductive biases on both models and data. Because speech signals contain both time-variant and time-invariant information, we can use this inductive bias to factorize the speech representations into static speaker factors and dynamic content factors. For example, in [26], [27], disentangled sequential variational autoencoders

This work was supported by the RGC of Hong Kong SAR, Grant No. PolyU 15210122 and the NSTC of Taiwan, Grant No. 112-2634-F-A49-006.

Y. Z. Tu and M. W. Mak are with the Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong SAR (email: 918tyz@gmail.com; enmwmak@polyu.edu.hk). J. T. Chien is with the Institute of Electrical and Computer Engineering, National Yang Ming Chiao Tung University, Taiwan (email: jtchien@nycu.edu.tw).

(DSVAEs) were introduced to separate the static and dynamic factors in the embeddings of sequential data, facilitating video generation and voice conversion.

In text-independent SV, text content is nuisance and we expect to produce content-invariant speaker embeddings. To this end, we propose eliminating the influence of content information on speaker embeddings by incorporating the DSVAE objective into the contrastive learning of speaker representations. In particular, a specialized DSVAE is proposed to disentangle the speaker factors from the content factors in the latent space so that only the speaker factors will contribute to the contrastive loss. In this way, we attribute the contrast between the positive and negative pairs to the speakers' identities, resulting in content-invariant speaker embeddings.

A recent study with similar motivation as ours is [28]. The authors proposed corrupting the content for each positive pair by shuffling the acoustic frames of one augmented segment. But our method removes content information not only between positive pairs but also between negative pairs. This forces the similarity between the negative pairs to be ascribed to speaker factors, which is amenable to learning speaker-discriminative embeddings. Moreover, instead of shuffling the acoustic frames, our approach removes the dynamic content information in speech frames by separately modeling the content and speaker information using VAEs.

On the other hand, the removal of content factors also corrupts the phonotactic patterns in an utterance. Because phonotactic features significantly contribute to the performance of spoken language recognition [29], our method can also reduce the effect of language on the contrast between the positive and negative pairs. This property benefits our method in learning language-invariant speaker embeddings, making them robust to language mismatch. Our results on using English for training a speaker embedding network and Mandarin for SV evaluation demonstrate this benefit.

This paper is an extension to our recent work [30] on disentangled contrastive speaker embedding. However, different from [30] where the idea was experimented on SimCLR [31] only, this paper verifies the effectiveness of the proposed method on both SimCLR and MoCo [32]. This difference suggests that contrastive learning with sequential disentanglement provides a general strategy for contrastive speaker embedding, which does not depend on a specific contrastive learning framework. Moreover, this paper explores in detail the robustness of the proposed method to language mismatch, which has not been investigated in [30].

The contributions of this paper are summarized as follows:

1) Compared with conventional contrastive speaker embedding, the proposed method disentangles speaker factors from content factors and uses the speaker factors only for contrastive learning. This strategy attributes the contrast between positive and negative pairs to speaker identity and endows the learned embeddings content-invariant and robust to language mismatch.
2) Compared with the frame shuffling [28], which corrupts the content information between positive pairs only, our method separately characterizes the speaker and content factors and removes content information not only between positive pairs but also between negative pairs, facilitating learning speaker-discriminative embeddings.
3) This paper verifies that removing content information for contrastive learning can make the embedding vectors robust to language mismatch.

This paper is organized as follows. In Section II, we briefly overview the related works. Section III presents the principle of the proposed contrastive speaker embedding with sequential disentanglement. The experimental settings and results are detailed in Section IV and Section V, respectively. We then give conclusions in Section VI.

II. Related Work

In this section, we briefly overview self-supervised speaker embedding and disentangled speaker representation learning.

A. Self-Supervised Speaker Embedding

Self-supervised speaker embedding generally uses contrastive and non-contrastive methods. Contrastive methods aim to discriminate between the positive and negative pairs of speech segments. Since the emergence of SimCLR [31] and MoCo [32], contrastive speaker embedding has witnessed fast development. For instance, the authors of [9] applied MoCo to pre-train an embedding network and then used it to generate pseudo labels for iterative refinement. In [10], augmentation adversarial training was incorporated into contrastive learning to learn channel-independent embeddings. The authors of [11], [14] used SimCLR to train embedding networks for pseudo-label generation. To address channel variations, the authors of [12] proposed channel-invariant training by enforcing the similarity between the clean and augmented embeddings. In [13], to alleviate the class collision problem, the authors proposed clustering the whole dataset before negative-segment sampling so that the negative samples in the prototypical memory bank were more likely from different speakers than the positive sample.

Recently, non-contrastive speaker embedding, especially those based on DINO [33], has attracted wide attention. In [15]–[17], the speaker embeddings learned from DINO performed better than the contrastive counterpart. Nevertheless, it has been shown that contrastive and non-contrastive learning objectives are closely related and their optimization is equivalent up to row and column normalization of the embedding matrix [34]. This discovery suggests that contrastive and non-contrastive methods could perform similarly when the model architectures, loss objectives, and hyperparameters have been sufficiently searched through. Note that our method is also applicable to non-contrastive speaker embedding so that the similarity between two augmentations of an utterance is attributed to speaker factors. This paper focuses on contrastive learning and leaves non-contrastive learning to future work.

B. Disentangled Speaker Representation

There have been various works on learning disentangled speaker representations. In [35], the authors adopted the unsupervised adversarial invariance [36] framework to disentangle

speaker factors from noise factors and improved SV performance on VOiCES19 data [37]. In [38], multi-task learning and adversarial training were respectively applied to disentangle the speaker identities from other speaker attributes—gender, age, and nationality—to inspect the contribution of each attribute to speaker discrimination. However, [35] and [38] require speaker labels and/or attribute labels to achieve satisfactory disentanglement, which will face a problem when neither speaker labels nor attribute labels are available.

Unsupervised disentangled representation learning can be a solution when no labeled data are available. In [26], the authors proposed a disentangled sequential variational autoencoder called DSVAE to explicitly generate a speech sequence from a combination of a static speaker factor and a sequence of dynamic content factors in a latent space. The authors of [27] improved the vanilla DSVAE by introducing a mutual information term in the loss function to encourage the independence between the static and dynamic latent variables. In [39], the authors combined factor analysis and masked prediction to jointly train a HuBERT model [40] and successfully disentangled the utterance-level representations from the content variations. On the contrary, the authors of [41] proposed a disentangled framework to remove speaker factors and preserve content information in speech signals. This paper aims to learn disentangled speaker embeddings under the contrastive learning framework by using DSVAEs.

III. Methodology

This section first introduces background knowledge on two commonly used contrastive learning frameworks: SimCLR [31] and MoCo [32]. We then explain the principle of DSVAE [26], [27]. Finally, the proposed contrastive learning with sequential disentanglement is detailed.

A. Contrastive Learning Frameworks

SimCLR [31] and MoCo [32] are two popular contrastive learning frameworks for speaker embedding. They both use a Siamese architecture [42], [43], a weight-sharing network receiving two augmented versions of a speech segment. These two frameworks share similar components such as the augmentation operations, the speaker encoder, and the loss objective. However, there are distinct elements that are unique to each framework.

1) SimCLR: Given a mini-batch of N speech segments $\{\mathbf{x}_n\}_{n=1}^N$, we obtain $2N$ samples $\{\tilde{\mathbf{x}}_{n,0}, \tilde{\mathbf{x}}_{n,1}\}_{n=1}^N$ after data augmentation. $\tilde{\mathbf{x}}_{n,i}$ can be an augmented version of the whole or part of the speech segment $\mathbf{x}_n$, where $i \in \{0, 1\}$ indexes the augmentations. Because $\tilde{\mathbf{x}}_{n,0}$ and $\tilde{\mathbf{x}}_{n,1}$ correspond to the same utterance, they form a positive pair. Note that we do not explicitly sample negative segments. Instead, for each positive pair $\{\tilde{\mathbf{x}}_{n,0}, \tilde{\mathbf{x}}_{n,1}\}$, the other $2(N-1)$ augmented segments in the mini-batch are considered negative samples.

Consider a speaker encoder f, we obtain speaker embeddings $\mathbf{e}_{n,i} = f(\tilde{\mathbf{x}}_{n,i})$. For a positive pair of speaker embeddings $\{\mathbf{e}_{n,i}, \mathbf{e}_{n,|1-i|}\}$, the loss function is defined as

$$\ell_{n,i} = -\log \frac{\exp\left(\cos\left(\mathbf{e}_{n,i}, \mathbf{e}_{n,|1-i|}\right)/\tau\right)}{\sum_{k=1}^{N}\sum_{j=0}^{1} \mathbb{1}_{[k\neq n, j\neq i]} \exp\left(\cos\left(\mathbf{e}_{n,i}, \mathbf{e}_{k,j}\right)/\tau\right)}, \tag{1}$$

where $\mathbb{1}_{[k\neq n, j\neq i]}$ denotes an indicator function which is equal to 1 only when $k \neq n$ and $j \neq i$, and τ is a temperature hyperparameter. The network is trained by minimizing the *NT-Xent* loss [31]:

$$\mathcal{L}_{\text{SimCLR}} = \frac{1}{2N}\sum_{n=1}^{N}\left(\ell_{n,0} + \ell_{n,1}\right). \tag{2}$$

2) MoCo: As mentioned in Section III-A1, the number of negative samples in SimCLR is determined by the mini-batch size. Because SimCLR requires a large number of negative samples to achieve competitive performance, a large batch size is required during training, leading to considerable GPU memory consumption. To decouple the number of negative samples from the mini-batch size, MoCo maintains a queue to dynamically store the negative embeddings encoded from the preceding mini-batches.

Different from SimCLR that uses the same encoder f to encode both augmentations $\{\tilde{\mathbf{x}}_{n,0}, \tilde{\mathbf{x}}_{n,1}\}$ of a speech segment $\mathbf{x}_n$, MoCo employs a query encoder f^{q} to encode one version $\tilde{\mathbf{x}}_{n,0}$ of the augmentations and a momentum encoder f^{k} to encode the other version $\tilde{\mathbf{x}}_{n,1}$. The resulting query embeddings and key embeddings are denoted as $\mathbf{e}_{n,0}^{\text{q}} = f^{\text{q}}(\tilde{\mathbf{x}}_{n,0})$ and $\mathbf{e}_{n,1}^{\text{k}} = f^{\text{k}}(\tilde{\mathbf{x}}_{n,1})$, respectively. Note that f^{q} and f^{k} share the same structure but have different parameters θ^{q} and θ^{k}. The key embeddings from the previous mini-batches are dynamically maintained to construct a queue of length K, and such embeddings are denoted as $\{\mathbf{e}_{j,1}^{\text{queue}}\}_{j=1}^{K}$. The parameters θ^{q} are updated by back-propagating the gradients of the InfoNCE loss [44]:

$$\mathcal{L}_{\text{MoCo}} = -\frac{1}{N}\sum_{n=1}^{N}\log \frac{\exp\left(\cos\left(\mathbf{e}_{n,0}^{\text{q}}, \mathbf{e}_{n,1}^{\text{k}}\right)/\tau\right)}{\left[\exp\left(\cos\left(\mathbf{e}_{n,0}^{\text{q}}, \mathbf{e}_{n,1}^{\text{k}}\right)/\tau\right) + \sum_{j=1}^{K}\exp\left(\cos\left(\mathbf{e}_{n,0}^{\text{q}}, \mathbf{e}_{j,1}^{\text{queue}}\right)/\tau\right)\right]}. \tag{3}$$

The parameters θ^{k} are updated by an exponential moving average of θ^{q} via a momentum hyperparameter $m \in [0, 1)$:

$$\theta^{\text{k}} \leftarrow m\theta^{\text{k}} + (1-m)\theta^{\text{q}}. \tag{4}$$

A large m (e.g., 0.999) ensures a slow update of θ^{k}, making the key embeddings in the queue consistent across multiple mini-batches.

B. Disentangled Sequential Variational Autoencoder

DSVAE [26], [27] is a popular framework for disentangled representation learning on sequential data. It aims to disentangle the time-invariant (static) factors from the time-variant (dynamic) factors in the latent space based on an VAE [45]. For a speech sequence, we use a DSVAE to disentangle the static speaker factors from the dynamic content factors.

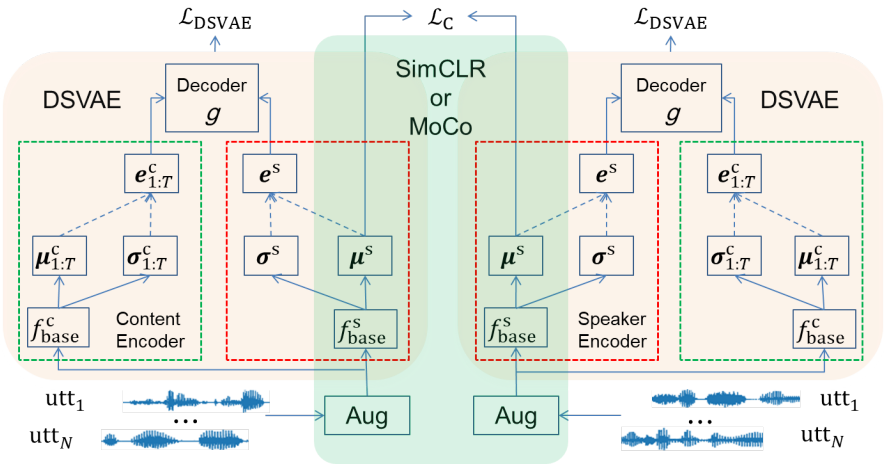


Fig. 1. Schematic of the proposed contrastive speaker embedding with sequential disentanglement. This method incorporates a DSVAE (orange blocks) into a contrastive learning framework (e.g., SimCLR in the green block). Note that the schematic can be easily adapted by replacing SimCLR with MoCo. "Aug" represents the augmentation operation. The dashed arrows within the speaker encoder and the content encoder denote the Gaussian sampling, which can be performed through the reparameterization trick [45]. After training, the vectors produced by the $\boldsymbol{\mu}^{\text{s}}$ node are used as speaker embeddings.

As shown in the orange blocks of Fig. 1, a DSVAE consists of a speaker encoder f^{s}, a content encoder f^{c}, and a decoder g.[1] Consider a sequence of filter-bank features with T frames $\mathbf{x}_{1:T} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$, two groups of latent variables (embeddings) $\mathbf{e}^{\text{s}}$ and $\mathbf{e}^{\text{c}}_{1:T}$ are encoded in the latent space. The sequence generation process can be formulated as

$$\begin{aligned} p\left(\mathbf{x}_{1:T}, \mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T}\right) &= p(\mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T}) p\left(\mathbf{x}_{1:T} | \mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T}\right) \\ &= p(\mathbf{e}^{\text{s}}) p\left(\mathbf{e}^{\text{c}}_{1:T}\right) p\left(\mathbf{x}_{1:T} | \mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T}\right) \\ &= p(\mathbf{e}^{\text{s}}) \prod_{t=1}^{T} \Big[p\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}\right) p\left(\mathbf{x}_t | \mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_t\right) \Big], \end{aligned} \tag{5}$$

where $\mathbf{e}^{\text{c}}_{<t} \equiv (\mathbf{e}^{\text{c}}_0, \ldots, \mathbf{e}^{\text{c}}_{t-1})$ and $\mathbf{e}^{\text{c}}_0 = \mathbf{0}$. We use a standard normal distribution for the speaker prior $p(\mathbf{e}^{\text{s}}) = \mathcal{N}(\mathbf{e}^{\text{s}}; \mathbf{0}, \mathbf{I})$. We assume that $p\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}\right)$ follows a Gaussian distribution

$$p\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}\right) = \mathcal{N}\left(\mathbf{e}^{\text{c}}_t; \boldsymbol{\mu}_t(\mathbf{e}^{\text{c}}_{<t}), \text{diag}\{(\boldsymbol{\sigma}_t(\mathbf{e}^{\text{c}}_{<t}))^2\}\right), \tag{6}$$

where $\boldsymbol{\mu}_t(\cdot)$ and $\boldsymbol{\sigma}_t(\cdot)$ can be modeled by a recurrent neural network (RNN) with long short-term memories (LSTMs) [46] followed by two linear heads. Because $\boldsymbol{\mu}_t(\cdot)$ and $\boldsymbol{\sigma}_t(\cdot)$ depend on $\mathbf{e}^{\text{c}}_{<t}$, the history of $\mathbf{e}^{\text{c}}_t$ prior to time step t is required when computing $\boldsymbol{\mu}_t(\cdot)$ and $\boldsymbol{\sigma}_t(\cdot)$. To sample $\mathbf{e}^{\text{c}}_t$ from $p\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}\right)$, $\mathbf{e}^{\text{c}}_{t-1}$ is first fed into the LSTM cells to forward one step and produce $\boldsymbol{\mu}_t(\cdot)$ and $\boldsymbol{\sigma}_t(\cdot)$ through the following linear heads. The reparameterization trick [45] is then used to draw a sample from the current distribution.

To approximate the posterior of the latent variables, we use a variational inference model

$$\begin{aligned} q\left(\mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T} | \mathbf{x}_{1:T}\right) &= q\left(\mathbf{e}^{\text{s}} | \mathbf{x}_{1:T}\right) q\left(\mathbf{e}^{\text{c}}_{1:T} | \mathbf{x}_{1:T}\right) \\ &= q\left(\mathbf{e}^{\text{s}} | \mathbf{x}_{1:T}\right) \prod_{t=1}^{T} q\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}, \mathbf{x}_{1:T}\right). \end{aligned} \tag{7}$$

[1]To better fit into the context of speaker verification, our terminologies differ from those in [14]. Specifically, our content means linguistic content, whereas the content in [14] means static content, e.g., object identity.

The speaker latent posterior follows a Gaussian distribution

$$q\left(\mathbf{e}^{\text{s}} | \mathbf{x}_{1:T}\right) = \mathcal{N}\left(\mathbf{e}^{\text{s}}; \boldsymbol{\mu}^{\text{s}}(\mathbf{x}_{1:T}), \text{diag}\{(\boldsymbol{\sigma}^{\text{s}}(\mathbf{x}_{1:T}))^2\}\right), \tag{8}$$

where the mean and standard deviation vectors $\boldsymbol{\mu}^{\text{s}}(\cdot)$ and $\boldsymbol{\sigma}^{\text{s}}(\cdot)$ are modeled by a speaker embedding network with two linear heads, respectively. Similarly, we have

$$q\left(\mathbf{e}^{\text{c}}_t | \mathbf{e}^{\text{c}}_{<t}, \mathbf{x}_{1:T}\right) = \mathcal{N}\left(\mathbf{e}^{\text{c}}_t; \boldsymbol{\mu}^{\text{c}}_t, \text{diag}\{(\boldsymbol{\sigma}^{\text{c}}_t)^2\}\right), \tag{9}$$

where $\boldsymbol{\mu}^{\text{c}}_t$ and $\boldsymbol{\sigma}^{\text{c}}_t$ can be implemented by feeding the inputs $\mathbf{e}^{\text{c}}_{<t}$ and $\mathbf{x}_{1:T}$ into bidirectional LSTMs followed by an RNN and two linear layers. To sample $\mathbf{e}^{\text{s}}$ and $\{\mathbf{e}^{\text{c}}_t\}_{t=1}^{T}$, the reparameterization trick is used.

We define the DSVAE loss as the negative of the evidence lower bound (ELBO) [27] of log-likelihood:

$$\begin{aligned} \mathcal{L}_{\text{DSVAE}} = &- \mathbb{E}_{p_{\text{D}}(\mathbf{x}_{1:T})} \mathbb{E}_{q\left(\mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T} | \mathbf{x}_{1:T}\right)} \Big[\log p\left(\mathbf{x}_{1:T} | \mathbf{e}^{\text{s}}, \mathbf{e}^{\text{c}}_{1:T}\right) \Big] \\ &+ \text{KL}\Big[q\left(\mathbf{e}^{\text{s}} | \mathbf{x}_{1:T}\right) \| p(\mathbf{e}^{\text{s}}) \Big] \\ &+ \text{KL}\Big[q\left(\mathbf{e}^{\text{c}}_{1:T} | \mathbf{x}_{1:T}\right) \| p\left(\mathbf{e}^{\text{c}}_{1:T}\right) \Big] \\ &- \left[I\left(\mathbf{e}^{\text{s}}; \mathbf{x}_{1:T}\right) + I\left(\mathbf{e}^{\text{c}}_{1:T}; \mathbf{x}_{1:T}\right) \right] \\ &+ I\left(\mathbf{e}^{\text{s}}; \mathbf{e}^{\text{c}}_{1:T}\right), \end{aligned} \tag{10}$$

where $p_{\text{D}}(\mathbf{x}_{1:T})$ is the empirical distribution of the sequence, $\text{KL}[\cdot\|\cdot]$ denotes Kullback-Leibler (KL) divergence, and $I(\cdot;\cdot)$ is mutual information (MI). The first term of (10) represents the reconstruction loss and the subsequent two terms are KL divergence between the posteriors and the priors w.r.t. time-invariant speaker embeddings $\mathbf{e}^{\text{s}}$ and time-variant content embeddings $\mathbf{e}^{\text{c}}_{1:T}$, respectively. The maximization of the MI between the latent variables and the inputs preserves respective information in the embeddings, whereas minimizing the MI between $\mathbf{e}^{\text{s}}$ and $\mathbf{e}^{\text{c}}_{1:T}$ encourages their independence, resulting in disentangled embeddings or bottleneck features $\boldsymbol{\mu}^{\text{s}}$.

In practice, the reconstruction loss term can be implemented by the mean squared errors between the decoder outputs and the inputs. There are various variational lower bounds to the MI estimator [47], and the InfoNCE lower bound [44] can be

used to approximate the true MI. The KL divergence terms can be analytically computed because both the priors of $\mathbf{e}^{\mathrm{s}}$ and $\{\mathbf{e}_t^{\mathrm{c}}\}_{t=1}^T$ and their posteriors are assumed Gaussian [48].

C. Contrastive Speaker Embedding With Sequential Disentanglement

In conventional contrastive speaker embedding, the contrast between positive and negative pairs can be attributed to speaker identity and content. To learn content-invariant speaker embeddings, we propose using a DSVAE to disentangle the speaker factors from the content factors and remove the content factors from contrastive learning.

The schematic of the proposed method is shown in Fig. 1, where SimCLR is used as an example for contrastive learning. However, the SimCLR can be replaced by MoCo easily. The base speaker encoder $f^{\mathrm{s}}_{\mathrm{base}}$, the speaker mean head $\boldsymbol{\mu}^{\mathrm{s}}$, the standard deviation head $\boldsymbol{\sigma}^{\mathrm{s}}$, and the Gaussian sample head $\mathbf{e}^{\mathrm{s}}$ constitute the speaker encoder f^{s}. The same applies to the content encoder f^{c}. Note that $f^{\mathrm{s}}_{\mathrm{base}}$ and $f^{\mathrm{c}}_{\mathrm{base}}$ can share the lower frame-level layers.

To train the network, we define the total loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{C}} + \lambda \mathcal{L}_{\mathrm{DSVAE}}, \tag{11}$$

where the contrastive loss $\mathcal{L}_{\mathrm{C}}$ can be $\mathcal{L}_{\mathrm{SimCLR}}$ as in (2) or $\mathcal{L}_{\mathrm{MoCo}}$ as in (3), depending on the contrastive learning framework used. $\mathcal{L}_{\mathrm{DSVAE}}$ follows the formulation in (10) and λ is a hyperparameter controlling the contribution of sequential disentanglement. When using $\mathcal{L}_{\mathrm{SimCLR}}$ as $\mathcal{L}_{\mathrm{C}}$, the embedding variable $\mathbf{e}_{n,i}$ in (1) should be changed to $\boldsymbol{\mu}^{\mathrm{s}}_{n,i}$ because the content variables have been discarded from contrastive learning. Similarly, $\mathbf{e}^{\mathrm{q}}_{n,0}$ and $\mathbf{e}^{\mathrm{k}}_{n,1}$ in (3) should be respectively replaced by $\boldsymbol{\mu}^{\mathrm{sq}}_{n,0}$ and $\boldsymbol{\mu}^{\mathrm{sk}}_{n,1}$ when $\mathcal{L}_{\mathrm{MoCo}}$ is used as $\mathcal{L}_{\mathrm{C}}$. After successful training by minimizing $\mathcal{L}$, we extract the vectors produced from the node $\boldsymbol{\mu}^{\mathrm{s}}$ in Fig. 1 as speaker embeddings.

IV. Experimental Setup

In the experiments, three contrastive speaker embedding, i.e., SimCLR-based embedding, MoCo-based embedding, and the proposed disentangled embedding, were compared on the VoxCeleb1 test sets [19] and the CN-Celeb evaluation set [49].

A. Datasets

VoxCeleb: The VoxCeleb corpus [19] was collected from the YouTube videos of over 7,000 celebrities. It contains two releases: VoxCeleb1 and VoxCeleb2. These datasets are multilingual, but the majority of utterances were spoken in English. The audios were recorded in the wild, covering a variety of real-world noise. In this paper, the VoxCeleb2 development (Vox2-dev) subset was used for training the SimCLR, MoCo, and the proposed framework. This dataset contains 1,092,009 utterances from 5,994 speakers. The original, extended, and hard VoxCeleb1 test sets (Vox1-O, Vox1-E, and Vox1-H, respectively) were used for evaluating the SV performance. Vox1-O contains 37,611 enrollment-test pairs from 40 speakers, whereas Vox1-E covers 579,818 pairs created from all of the 1,251 speakers in VoxCeleb1 test. Vox1-H is a more challenging test set than Vox1-O and Vox1-E, in which 550,894 enrollment-test pairs were built within the same nationality and gender. The number of speakers in Vox1-H is 1,190.

CN-Celeb: CN-Celeb [49] is a multi-genre Mandarin corpus, which was collected from Chinese open media. It contains around 3,000 speakers in 11 genres, e.g., interview, singing, live broadcast, etc. The data were recorded in real-world environments. Two subsets, CN-Celeb1 and CN-Celeb2, have been released. In this paper, the CN-Celeb evaluation (CN-eval) subset was used in the comparison between the proposed method and MoCo to investigate the robustness to language mismatch (see Section V-C). This dataset consists of around 18,000 utterances from 196 speakers, covering about 3.6 million enrollment-test pairs. We used the CN-Celeb2 development data (CN2-dev) as the target-domain training set. This dataset contains 2,000 speakers with 529,485 utterances.

VoxSRC 2021: VoxSRC 2021 added a new focus on multilingual evaluation to VoxCeleb [50]. A large number of evaluation pairs in the validation and test sets come from utterances in different languages. The validation set (VoxSRC21-val) comprises 60,000 evaluation trials created from the celebrities in VoxCeleb1. This dataset was used to analyze the robustness of the proposed method to cross-lingual challenges (see Section V-C). The VoxSRC 2021 test data are new and contain speakers not in either VoxCeleb1 or VoxCeleb2. However, we did not experiment on the test set because the trial keys of this set were not publicly available.

Vox1-cus: A customized set was created to verify the robustness of the proposed method to language variation. The customized trials were curated in the same way as that of VoxSRC21-val [50], which contain a large number of cross-lingual enrollment-test pairs. All speakers in the customized set are from VoxCeleb1 and the language labels were predicted by a model trained on VoxLingua107 [51].[2] Compared with VoxSRC21-val, Vox1-cus has more within-language trials in some common languages, e.g., German, Italian, Spanish, etc. The total number of trials in the customized set is 100,000. The trial file will be available on Github soon.[3]

B. Acoustic Feature Extraction

We extracted 80-dimensional filter-bank features from each utterance with a 25-ms window and a 10-ms frame shift. Cepstral mean normalization was applied to the extracted features. Before acoustic feature extraction, data augmentation was performed by adding reverberation, noise, music, and babble to the original speech signals. The MUSAN [52] corpus was used as the additive noise sources. The signal-to-noise ratios (SNRs) of ambient noise, music, and babble were set to 0–15 dB, 5–15 dB, and 13–20 dB, respectively. For creating reverberated speech signals, we convolved the original speech signals with the simulated room impulse responses generated from small- and medium-sized rooms.[4]

[2]https://huggingface.co/TalTechNLP/voxlingua107-epaca-tdnn.
[3]https://github.com/youzhitu/contrastive_disentanglement.
[4]https://www.openslr.org/28/rirs_noises.zip.

C. Network Structure

As shown in Fig. 1, the SimCLR-based network comprises a speaker encoder, a content encoder, and a decoder. For the speaker encoder, we used a 512-channel ECAPA-TDNN [3] as the base encoder. On top of the pooling layer, two linear heads were added for $\boldsymbol{\mu}^{\mathrm{s}}$ and $\boldsymbol{\sigma}^{\mathrm{s}}$, respectively. The number of nodes in these heads is 192. The lowest four layers of the content encoder and the speaker encoder were shared. On top of the shared layers, we sequentially added a bidirectional LSTM layer of 512 hidden nodes and a 512-node RNN layer to the content encoder. Two linear layers of 32 hidden nodes were added to represent $\boldsymbol{\mu}_t^{\mathrm{c}}$ and $\boldsymbol{\sigma}_t^{\mathrm{c}}$, respectively. The decoder g comprises two convolutional layers with 512 and 80 channels, respectively. The dilation rates of these two layers were set to 2 and 1, respectively. The MI estimator in (10) followed the squeeze-DIM configuration in [53] with an InfoNCE lower bound. The *critic* of this estimator was parameterized by two networks with each comprising two 64-node fully-connected layers.

For MoCo-based contrastive learning, we used two speaker encoders, i.e., a query encoder and a key encoder. These two encoders have the same structure as the ECAPA-TDNN in SimCLR. The queue size in MoCo was set to 65,536.

D. Training Configurations

We used Adam [54] for optimization. A linear learning rate warm-up was employed during the first 10 epochs, increasing the learning rate from 1e-4 to 1e-3. After that, it was decayed to 1e-5 with a cosine scheduler. Totally, the networks were trained for 50 epochs. The mini-batch size was set to 256. Each mini-batch was created by randomly selecting speech segments of 3.5 seconds from the training set and the two segments of each positive pair can overlap. The temperature in (1) and (3) was set to 0.05, and we used 0.01 for the hyperparameter λ in (11). For MoCo, the momentum hyperparameter m in (4) was set to 0.999.

V. Results and Discussions

Cosine similarity was used in all experiments. The performance was evaluated in terms of equal error rate (EER) and minimum detection cost function (minDCF) at $C_{\mathrm{miss}} = 1$, $C_{\mathrm{fa}} = 1$, and $P_{\mathrm{target}} = 0.01$.

A. Performance on VoxCeleb1 Test Sets

The results of SimCLR, MoCo, and the proposed disentangled contrastive speaker embedding are shown in Table I. Rows 7–12 show the results of existing contrastive learning frameworks without iterative model refinement. The best performance (Row 7) was achieved by MoCo using 1024-channel ECAPA-TDNNs as speaker encoders. This result is only slightly better than our MoCo baseline (Row 4), which used smaller ECAPA-TDNNs with 512 channels only. The framework in [14] adopted a similar encoder as in our SimCLR baseline (Row 1) and performed similarly as Row 1. However, this method employed an additional augmentation adversarial training strategy [10] to improve performance. These comparisons show that our baseline systems are very competitive.

From Rows 1 and 3 of Table I, we observe that the proposed method consistently outperforms the SimCLR baseline across three evaluation sets. This result indicates that it is effective to apply a DSVAE to disentangle the speaker factors from the content factors and only use speaker factors for contrastive learning. The same conclusion can be obtained from MoCo by comparing Row 4 and Row 6.

We also employed frame shuffling [28] for comparison. As shown in Rows 1–2 of Table I, shuffling the frames in SimCLR degrades the performance in all tasks. A similar trend can be seen between Row 4 and Row 5 when MoCo was used. These observations contradict with that in [28]. One reason of performance degradation could be that shuffling the frames corrupts the underlying acoustic dynamics that are useful to both speech and speaker recognition. Although the content dependency is destroyed as expected, the information related to speaker discrimination is also undermined. An empirical evidence can be found in [55], where the authors observed that shuffling the acoustic sequence at the frame level degrades SV performance. This is in accordance with the results in Table I.

B. Speaker Information in Speaker and Content Embeddings

To further verify the benefit of sequential disentanglement, we investigated the speaker information in the speaker embeddings ($\boldsymbol{\mu}^{\mathrm{s}}$ in Fig. 1) and content embeddings ($\boldsymbol{\mu}_{1:T}^{\mathrm{c}}$ in Fig. 1) on Vox1-O. MoCo was used as the contrastive learning framework in the experiments because MoCo slightly outperforms SimCLR in Table I. But a similar trend can be observed when SimCLR was used.

Table II shows the performance of different representations extracted from MoCo, DSVAE, and their combinations. From Row 2, we observe that the performance of the speaker embeddings extracted from DSVAE is much worse than that of the MoCo baseline. This suggests that using DSVAE alone is not sufficient to separate the speaker and content information in the embedding space. Rows 4 and 5 show the performance of incorporating DSVAE into MoCo when the averages of the content embeddings instead of the speaker embeddings were used for SV. The embedding network in Row 4 was subjected to contrastive training, whereas the network in Row 5 contains random weights. We see that there is almost no speaker information in the temporally-averaged content embeddings after random initialization and also that disentangled representation learning can further reduce the speaker information in the content embeddings. In short, combining DSVAE and MoCo is effective to learn discriminative speaker embeddings.

C. Robustness to Language Mismatch

The proposed method aims to eliminate the effect of content on contrastive learning. However, the removal of content also undermines the phonotactic information in a speech sequence, which could affect the performance of spoken language recognition. This means that the language information, a nuisance source for text-independent SV, can be removed to a certain extent by disentangled contrastive learning. In other words,

TABLE I
PERFORMANCE OF SIMCLR, MOCO, AND THE PROPOSED DSVAE-BASED DISENTANGLED CONTRASTIVE LEARNING ON VOXCELEB1 TEST SETS. THE RESULTS IN ROWS 7–12 WERE OBTAINED WITHOUT ITERATIVE MODEL REFINEMENT. THE MINDCF WITH A SUPERSCRIPT $*$ IN ROWS 8–9 WAS CALCULATED USING $P_{\text{target}} = 0.05$ INSTEAD OF $P_{\text{target}} = 0.01$.

Row	Contrastive Learning Framework	Vox1-O		Vox1-E		Vox1-H	
		EER (%)	minDCF	EER (%)	minDCF	EER (%)	minDCF
1	SimCLR	7.13	0.571	7.89	0.596	12.26	0.692
2	SimCLR + Frame Shuffle [28]	7.90	0.570	8.48	0.600	12.86	0.737
3	SimCLR + DSVAE (Proposed)	**6.37**	**0.533**	**7.36**	**0.574**	**11.72**	**0.677**
4	MoCo	7.38	0.561	7.97	0.584	12.22	0.681
5	MoCo + Frame Shuffle [28]	7.81	0.568	8.43	0.594	12.85	0.737
6	MoCo + DSVAE (Proposed)	**6.29**	**0.534**	**7.17**	**0.567**	**11.42**	**0.668**
7	[9], MoCo using 1024-channel ECAPA-TDNNs as speaker encoders	**7.30**	–	–	–	–	–
8	[10], adding augmentation adversarial training to SimCLR	8.65	0.454*	–	–	–	–
9	[11], using SimCLR with adapted contrastive loss	8.86	0.508*	10.15	0.570*	16.20	0.710*
10	[12], adding channel-invariant training to SimCLR	8.28	0.610	–	–	–	–
11	[13], using MoCo with a prototypical memory bank	8.23	0.590	–	–	–	–
12	[14], adding augmentation adversarial training to SimCLR	7.36	–	7.90	–	12.32	–

TABLE II
PERFORMANCE OF DIFFERENT REPRESENTATIONS ON VOX1-O. "SPK_EMB" DENOTES SPEAKER EMBEDDINGS ($\boldsymbol{\mu}^{\text{s}}$ IN FIG. 1). "AVG_CON_EMB" MEANS TEMPORALLY-AVERAGED CONTENT EMBEDDINGS ($\boldsymbol{\mu}^{\text{c}}_{1:T}$ IN FIG. 1). "INIT" IN ROW 5 MEANS THAT THE MODEL IS RANDOMLY INITIALIZED WITHOUT TRAINING.

Row	Framework	Representation	Vox1-O	
			EER (%)	minDCF
1	MoCo	spk_emb	7.38	0.561
2	DSVAE	spk_emb	22.87	0.915
3	MoCo + DSVAE	spk_emb	**6.29**	**0.534**
4	MoCo + DSVAE	avg_con_emb	47.40	0.999
5	MoCo + DSVAE (Init)	avg_con_emb	42.06	0.994

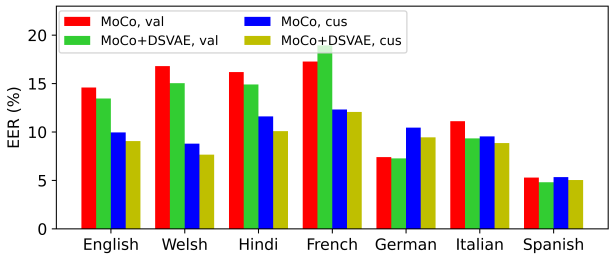


Fig. 2. EERs of MoCo and the proposed method on within-language evaluation trials in VoxSRC21-val ("val" in the legend) and Vox1-cus ("cus" in the legend). Both the enrollment and test utterances of each trial are from the same language.

TABLE III
PERFORMANCE OF MOCO AND THE PROPOSED METHOD ON CN-CELEB EVALUATION SET.

Row	Framework	Training Set	CN-eval	
			EER (%)	minDCF
1	MoCo	Vox2-dev	18.73	0.713
2	MoCo + DSVAE	Vox2-dev	16.21	0.659
3	MoCo	CN2-dev	16.10	0.684
4	MoCo + DSVAE	CN2-dev	**14.52**	**0.653**

TABLE IV
PERFORMANCE OF MOCO AND THE PROPOSED METHOD ON VOXSRC21-VAL AND VOX1-CUS EVALUATION SETS.

Framework	VoxSRC21-val		Vox1-cus	
	EER (%)	minDCF	EER (%)	minDCF
MoCo	15.36	0.709	11.10	0.570
MoCo + DSVAE	16.59	0.743	10.32	0.542

the proposed method is amenable to alleviating the effect of languages and produce the language-invariant speaker embeddings. To investigate the robustness of the proposed method to language mismatch, we adopted the CN-Celeb evaluation set and the VoxSRC 2021 validation set for performance evaluation.

Table III shows the performance of contrastive learning models that were trained on Vox2-dev and CN2-dev, respectively. As mentioned in Section IV-A, most of the utterances in Vox2-dev are spoken in English, whereas CN2-dev is a Mandarin corpus. Therefore, there exists language mismatch between these two datasets. The Vox2-dev dataset was used to train source-domain models and their performance is shown in Rows 1–2. We also used CN2-dev as the target-domain training data and Rows 3–4 show the corresponding performance. Comparing Row 1 with Row 3, we see that the performance of the source-domain model (Row 1) is much worse than that of the target-domain model (Row 3) due to the language mismatch. After incorporating a DSVAE into MoCo as shown in Row 2, a remarkable performance improvement is observed over the MoCo baseline. Despite being trained on Vox2-dev, the proposed model filled the gap between Row 1 and Row 3 and achieved similar performance as the target-domain model (the EER of Row 2 is slightly worse than that of Row 3 but the corresponding minDCF is much better than that in Row 3). Also, we observe that combining DSVAE and MoCo can further improve the performance when CN2-dev was used for training (Row 4). The above analysis indicates that the proposed contrastive learning can alleviate language mismatch.

To further demonstrate the robustness of the proposed method to language variation, we investigate the SV performance on the VoxSRC 2021 validation (VoxSRC21-val) set and the customized dataset (Vox1-cus). As observed from

TABLE V
STATISTICS OF WITHIN-LANGUAGE EVALUATION TRIALS IN VOXSRC21-VAL AND VOX1-CUS.

Dataset	No. of trials						
	Eng.	Wel.	Hin.	Fre.	Ger.	Ita.	Spa.
VoxSRC21-val	27,520	1,009	1,286	1,369	689	307	215
Vox1-cus	48,215	7,816	1,808	1,833	1,209	1,052	1,222

TABLE VI
STATISTICS OF TARGET AND NON-TARGET EVALUATION TRIALS IN VOXSRC21-VAL AND VOX1-CUS THAT WERE CREATED FROM MULTILINGUAL SPEAKERS. THE TARGET TRIALS INCLUDES SAME-LANGUAGE AND CROSS-LANGUAGE TRIALS.

Dataset	#Same-tar	#Cross-tar	#Non-tar
VoxSRC21-val	4,540	18,922	14,318
Vox1-cus	9,000	32,000	29,303

Table IV, the overall performance of our contrastive method outperforms the conventional MoCo on both VoxSRC21-val and Vox1-cus.

To better show the language robustness of the proposed method, we also evaluated the performance of within-language trials on some common languages, where both the enrollment and test utterances are from the same language. The statistics of the within-language subsets of VoxSRC21-val and Vox1-cus are summarized in Table V and the corresponding performance is shown in Fig. 2. We observe that the proposed method outperforms MoCo across the listed languages except for French on VoxSRC21-val. On the customized set, the proposed contrastive learning consistently achieves better performance than MoCo. This observation is consistent with that on CN-Celeb, suggestting that the proposed method is less sensitive to language variation than the conventional MoCo.

Next, we investigate the effectiveness of the proposed method on cross-lingual enrollment-test pairs. In VoxCeleb1, there are 695 (out of 1,251) celebrities speaking more than two languages. We regarded the target (positive) trials from these multilingual speakers as hard positives. The rationale is that the speaker embeddings should contain speaker-discriminative information while be robust to language variation. Therefore, a better performance on cross-language trials indicates a more language-invariant SV system. The statistics of these trials are shown in Table VI. Note that for both the same-language target trials and the cross-language target trials, the same non-target trials were used. We used DET curves to compare the performance of MoCo and MoCo+DSVAE (proposed) based on these hard positive and non-target trials. The closer the DET curve to the origin, the better the performance.

As shown in Fig. 3, the DET curves (in blue and red) on the same-language trials are closer to the origin than those on the cross-language trials (in green and black). This indicates that cross-language target trials are more difficult than the same-language target trials. Also, we can see that the gap between the DET curves on the same-language trials and the cross-language trials becomes narrower ($g_{\text{M+D}} < g_{\text{M}}$) when DSVAE was applied. This means that the performance drop due to changing from easy target trials to hard target trials is

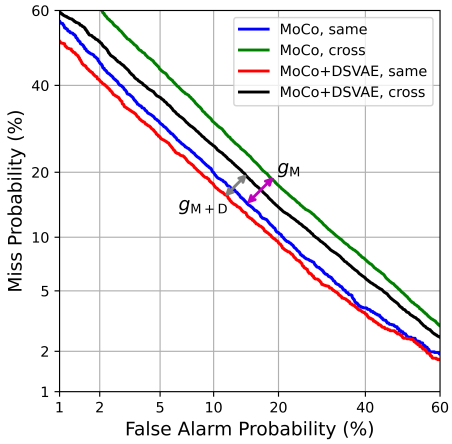


Fig. 3. DET curves of MoCo and MoCo+DSVAE (proposed) on VoxSRC21-val. "same" and "cross" mean that the evaluation is performed on the same-language target trials and cross-language target trials, respectively. The same non-target trials were used for both target trials. g_{M} and $g_{\text{M+D}}$ denote the performance gaps due to changing from same-language target trials to cross-language target trials in MoCo and MoCo+DSVAE, respectively.

smaller in the proposed method, suggesting that the proposed contrastive learning is more robust to language variation than MoCo.

To further verify that our method is more language-invariant than MoCo, we compared the t-SNE plots of the embeddings corresponding to utterances of randomly selected 11 speakers. As shown in Fig. 4, each speaker uses at least 2 languages. Fig. 4(a) shows that some utterances, highlighted by three red circles, are assigned to the incorrect clusters when MoCo was used. However, the proposed method does not have this problem, as shown in Fig. 4(b). According to Fig. 4(c), the incorrect assignments within the three red circles in Fig. 4(a) come from the cross-language target trials. This observation, again, shows that using MoCo alone cannot well verify the cross-language (hard) positives and the proposed method is more robust to such language variation. We have the same conclusion on the customized set.

From the above analysis on CN-Celeb, VoxSRC21-val, and the customized set, we conclude that the proposed contrastive learning can alleviate language mismatch.

D. Impact of Batch Size

The number of negative samples has a significant impact on the performance of contrastive learning. In Section III-A1, we show that the batch size determines the number of negative samples of SimCLR and thus substantially affects its performance. MoCo, on the other hand, decouples the batch size from the negative-sample size by maintaining a queue of dynamically encoded key embeddings. However, the performance of MoCo can still be affected by the batch size if it is too small. We thus investigated the effect of batch size on SimCLR and MoCo.

The results of contrastive speaker embeddings by varying batch size are shown in Fig. 5. From Fig. 5, we see that the SimCLR-based contrastive learning follows a similar trend

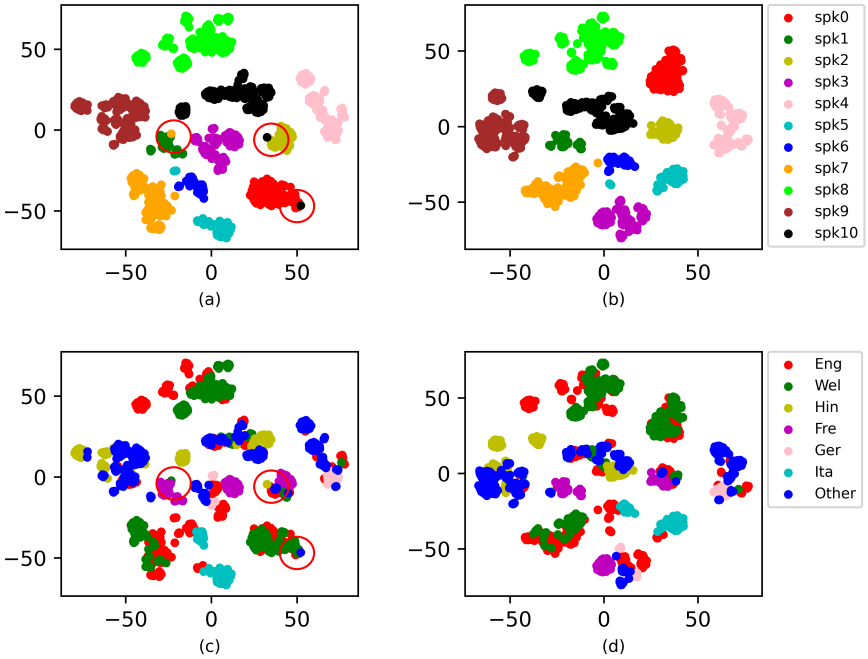


Fig. 4. t-SNE plots of embeddings extracted from (a)(c) MoCo and (b)(d) MoCo+DSVAE (proposed). Each color in (a) and (b) denotes a speaker and each dot represents an utterance. In (c) and (d), each color represents a language. The three red circles in (a) and (c) indicate the regions where some speakers are assigned to the wrong clusters.

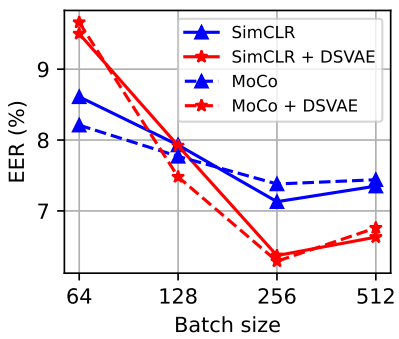


Fig. 5. The effect of varying batch size on the EER of SimCLR- and MoCo-based contrastive speaker embeddings.

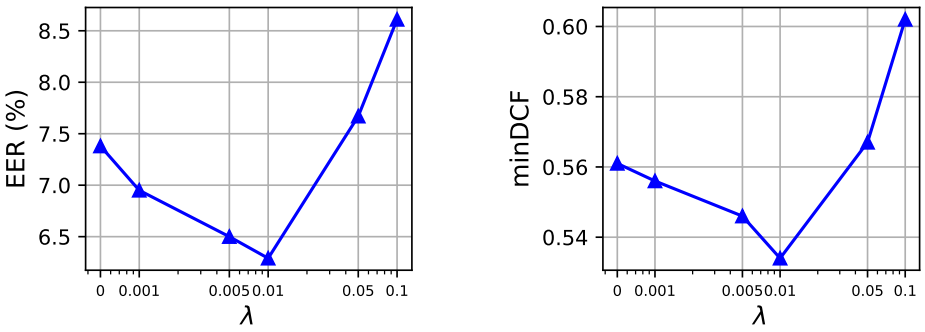
as that based on MoCo and their performance gap is not significant. When the batch size was set to a small value, e.g., 64 and 128, MoCo is slightly better than SimCLR and the variation in EER of MoCo is smaller. This observation suggests that MoCo is less sensitive to small batch size than SimCLR. Also, we observe that for a small batch size of 64, contrastive learning with sequential disentanglement cannot even compete with vanilla contrastive learning. This indicates that the DSVAE requires a large batch size to achieve reasonable disentanglement when incorporated into SimCLR or MoCo. In general, the best performance was obtained at a batch size of 256. This is the reason why we used this configuration in previous sections.

E. Effect of λ

The hyperparameter λ in (11) controls the contribution of DSVAE in the proposed framework. In this subsection, we investigated the effect of λ on MoCo-based contrastive learning. The proposed model was evaluated by choosing λ from $\{0, 0.001, 0.005, 0.01, 0.05, 0.1\}$ and the results are shown in Fig. 6. The best result was obtained at $\lambda = 0.01$ for both EER and minDCF. When λ is larger than 0.05, the performance begins to degrade. This result shows that too much emphasis on sequential disentanglement can prevent the model from learning discriminative speaker embeddings, which agrees with the result in Row 2 of Table II. Recall from Section V-B that using a DSVAE alone cannot achieve performance comparable to a contrastive learning baseline. The above analysis suggests that DSVAE relies on discriminative speaker information to assist the disentanglement between the speaker factors and the content factors. Without discriminative speaker information introduced by the speaker contrast, DSVAE cannot effectively separate the speaker embeddings from the content embeddings.

F. Effect of Content Removal by DSVAE

For text-independent SV, text content is nuisance and such information is supposed to be removed during embedding learning. In [28], the authors proposed frame shuffling to undermine the content dependency in the positive samples. In our method, the content information is discarded from contrastive learning to highlight speaker contrast. Both approaches show performance improvement. However, these observations



(a) (b)

Fig. 6. The effect of varying hyperparameter λ (see (11)) on (a) EER and (b) minDCF of the proposed "DSVAE+MoCo" framework.

TABLE VII
PERFORMANCE OF SUPERVISED AND SELF-SUPERVISED SPEAKER EMBEDDING ON VOX1-O. ALL RESULTS WERE BASED ON MOCO.

Row	Speaker Embedding	Training Style	Vox1-O EER (%)	Vox1-O minDCF
1	E-TDNN	Supervised	1.65	0.172
2	E-TDNN + DSVAE	Supervised	**1.53**	**0.155**
3	ECAPA-TDNN	Supervised	**1.01**	**0.116**
4	ECAPA-TDNN + DSVAE	Supervised	**1.01**	0.125
5	ECAPA-TDNN	Self-Supervised	7.38	0.561
6	ECAPA-TDNN + DSVAE	Self-Supervised	**6.29**	**0.534**

are contradictory to existing works which have shown that incorporating phonetic content into speaker embeddings is beneficial to text-independent SV. For instance, the authors of [56] adopted multi-task learning by combining a phonetic classifier with a speaker classifier and obtained superior performance. In [57], the authors investigated the usefulness of phonetic information at the segment (embedding) level and the frame level. They concluded that although phonetic content at the segment level is detrimental to SV performance, using phonetic information at the frame level is beneficial. In this section, we investigate the effect of content removal through DSVAE on both supervised and self-supervised speaker embedding.

For supervised speaker embedding, we used an E-TDNN [58] and an ECAPA-TDNN as the speaker encoders. These models were trained on Vox2-dev through a combination of speaker classification loss (AMSoftmax [7]) and DSVAE loss. Similar to the proposed contrastive method, only the speaker embeddings were fed into the speaker classifier and the content embeddings were discarded. The results are shown in Rows 1–4 of Table VII. When an E-TDNN was used for speaker embedding, we see that discarding content information through sequential disentanglement (Row 2) is advantageous. This observation verifies the conclusion in [57] that the embedding-level phonetic content should be removed. However, such benefit is not observed when an ECAPA-TDNN was adopted (comparing Row 3 with Row 4). One possible reason is that when a powerful speaker encoder is trained in a supervised manner, the content information has already been substantially suppressed in the embeddings and the DSVAE cannot further separate the speaker factors from the content factors. Under the self-supervised setting (Rows 5–6), we obtained the same conclusion as in Rows 1–2: removing content at the embedding level benefits text-independent SV.

VI. CONCLUSIONS

This paper proposed a contrastive learning framework with sequential disentanglement for text-independent SV. The proposed method adopts a DSVAE for disentangling speaker factors from content factors and uses the speaker factors only for contrastive learning. Two contrastive learning frameworks, SimCLR and MoCo, were exploited in the experiments. Evaluation results on the VoxCeleb1 test and CN-Celeb sets show that the proposed method consistently outperforms the conventional contrastive speaker embedding, suggesting that it is beneficial to incorporate sequential disentanglement into contrastive learning for learning speaker-discriminative embeddings. Also, the results on CN-Celeb demonstrate that the proposed method is able to alleviate language mismatch.

REFERENCES

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

[2] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2019, pp. 5791–5795.

[3] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Annual Conference of the International Speech Communication Association*, 2020, pp. 3830–3834.

[4] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification," in *Proc. Annual Conference of the International Speech Communication Association*, 2018, pp. 3573–3577.

[5] Y. Z. Tu and M. W. Mak, "Short-time spectral aggregation for speaker embedding," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2021, pp. 6708–6712.

[6] Y. Z. Tu and M. W. Mak, "Aggregating frame-level information in the spectral domain with self-attention for speaker embedding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 944–957, 2022.

[7] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 235–238, 2018.

[8] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[9] J. Thienpondt, B. Desplanques, and K. Demuynck, "The IDLAB voxceleb speaker recognition challenge 2020 system description," in *arXiv preprint arXiv:2010.12468*, 2020.

[10] J. Huh, H. Heo, J. Kang, S. Watanabe, and J. Chung, "Augmentation adversarial training for self-supervised speaker recognition," in *Proc. Self-Supervised Learning for Speech and Audio Processing at NeurIPS Workshops*, 2020.

[11] D. Cai, W. Wang, and M. Li, "An iterative framework for self-supervised deep speaker representation learning," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2021, pp. 6728–6732.

[12] H. Zhang, Y. Zou, and H. Wang, "Contrastive self-supervised learning for text-independent speaker verification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2021, pp. 6713–6717.

[13] W. Xia, C. Zhang, C. Weng, M. Yu, and D. Yu, "Self-supervised text-independent speaker verification using prototypical momentum contrastive learning," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2021, pp. 6723–6727.

[14] R. Tao, K. Lee, R. Das, V. Hautamäki, and H. Li, "Self-supervised speaker recognition with loss-gated learning," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2022, pp. 6142–6146.

[15] B. Han, Z. Chen, and Y. Qian, "Self-supervised speaker verification using dynamic loss-gate and label correction," in *Proc. Annual Conference of the International Speech Communication Association*, 2022, pp. 4780–4784.

[16] J. Cho, R. Pappagari, P. Żelasko, L. Moro-Velazquez, J. Villalba, and N. Dehak, "Non-contrastive self-supervised learning of utterance-level speech representations," in *Proc. Annual Conference of the International Speech Communication Association*, 2022, pp. 4028–4032.

[17] Y. Chen, S. Zheng, H. Wang, L. Cheng, and Q. Chen, "Pushing the limits of self-supervised speaker verification using regularized distillation framework," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2023, pp. 1–5.

[18] S. Arora, H. Khandeparkar, M. Khodak, O. Plevrakis, and N. Saunshi, "A theoretical analysis of contrastive unsupervised representation learning," in *Proc. International Conference on Machine Learning*, 2019, pp. 5628–5637.

[19] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, 2020.

[20] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[21] I. Higgins, D. Amos, D. Pfau, S. Racanière, L. Matthey, D. Rezende, and A. Lerchner, "Towards a definition of disentangled representations," in *arXiv preprint arXiv:1812.02230*, 2020.

[22] D. Raj, D. Snyder, and S. Khudanpur, "Probing the information encoded in x-vectors," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 726–733.

[23] R. Peri, H. Li, K. Somandepalli, A. Jati, and S. Narayanan, "An empirical analysis of information encoded in disentangled neural speaker representations," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2020, pp. 194–201.

[24] L. Yi and M. W. Mak, "Disentangled speaker embedding for robust speaker verification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2022, pp. 7662–7666.

[25] F. Locatello, S. Bauer, M. Lucic, G. Rätsch, B. Schölkopf, and O. Bachem, "Challenging common assumptions in the unsupervised learning of disentangled representations," in *Proc. International Conference on Machine Learning*, 2018, pp. 4114–4124.

[26] Y. Li and S. Mandt, "Disentangled sequential autoencoder," in *Proc. International Conference on Machine Learning*, 2018, pp. 5670–5679.

[27] J. Bai, W. Wang, and C. Gomes, "Contrastively disentangled sequential variational autoencoder," in *Advances in Neural Information Processing Systems*, 2021, pp. 10 105–10 118.

[28] W. Lin, L. Li, and D. Wang, "Shuffle is what you need," in *Proc. International Symposium on Chinese Spoken Language Processing*, 2022, pp. 245–249.

[29] H. Li, B. Ma, and K. Lee, "Spoken language recognition: From fundamentals to practice," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1136–1159, 2013.

[30] Y. Z. Tu, M. W. Mak, and J. T. Chien, "Contrastive speaker embedding with sequential disentanglement," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2024, pp. 10 891–10 895.

[31] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. International Conference on Machine Learning*, 2020, pp. 1597–1607.

[32] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738.

[33] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proc. International Conference on Computer Vision*, 2021, pp. 9650–9660.

[34] Q. Garrido, Y. Chen, A. Bardes, L. Najman, and Y. Lecun, "On the duality between contrastive and non-contrastive self-supervised learning," in *Proc. International Conference on Learning Representations*, 2023.

[35] R. Peri, M. Pal, A. Jati, K. Somandepalli, and S. Narayanan, "Robust speaker recognition using unsupervised adversarial invariance," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2020, pp. 6609–6613.

[36] A. Jaiswal, R. Wu, W. Abd-Almageed, and P. Natarajan, "Unsupervised adversarial invariance," in *Advances in Neural Information Processing Systems*, 2018, pp. 5092–5102.

[37] M. K. Nandwana, J. Van Hout, M. McLaren, C. Richey, M. Lawson, and A. Barrios, "The VOiCES from a distance challenge 2019 evaluation plan," in *arXiv preprint arXiv:1902.10828*, 2019.

[38] C. Luu, S. Renals, and P. Bell, "Investigating the contribution of speaker attributes to speaker separability using disentangled speaker representations," in *Proc. Annual Conference of the International Speech Communication Association*, 2022, pp. 610–614.

[39] W. W. Lin, C. H. He, M. W. Mak, and Y. Z. Tu, "Self-supervised neural factor analysis for disentangling utterance-level speech representations," in *Proc. International Conference on Machine Learning*, 2023, pp. 21 065–21 077.

[40] W. Hsu, B. Bolte, Y. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[41] K. Qian, Y. Zhang, H. Gao, J. Ni, C. Lai, D. Cox, M. Hasegawa-Johnson, and S. Chang, "ContentVec: An improved self-supervised speech representation by disentangling speakers," in *Proc. International Conference on Machine Learning*, 2022, pp. 18 003–18 017.

[42] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "Siamese" time delay neural network," in *Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[43] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 750–15 758.

[44] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," in *arXiv preprint arXiv:1807.03748*, 2018.

[45] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. International Conference on Learning Representations*, 2014.

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

[47] M. Tschannen, J. Djolonga, P. Rubenstein, S. Gelly, and M. Lucic, "On mutual information maximization for representation learning," in *Proc. International Conference on Learning Representations*, 2020.

[48] J. Duchi, "Derivations for linear algebra and optimization." [Online]. Available: https://stanford.edu/~jduchi/projects/general_notes.pdf

[49] L. Li, R. Liu, J. Kang, Y. Fan, H. Cui, Y. Cai, R. Vipperla, F. Zheng, and D. Wang, "CN-Celeb: multi-genre speaker recognition," *Speech Communication*, vol. 137, pp. 77–91, 2022.

[50] A. Brown, J. Huh, J. Chung, A. Nagrani, D. Garcia-Romero, and A. Zisserman, "VoxSRC 2021: The third voxceleb speaker recognition challenge," in *arXiv preprint arXiv:2201.04583*, 2022.

[51] J. Valk and T. Alumäe, "VoxLingua107: A dataset for spoken language recognition," in *Proc. IEEE Spoken Language Technology Workshop*, 2021, pp. 652–658.

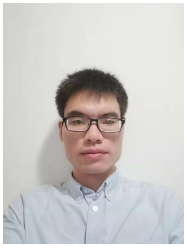
[52] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," in *arXiv preprint arXiv:1510.08484*, 2015.

[53] Y. Z. Tu and M. W. Mak, "Mutual information enhanced training for speaker embedding," in *Proc. Annual Conference of the International Speech Communication Association*, 2021, pp. 91–95.

[54] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conference on Learning Representations*, 2015.

[55] J. Li, X. Fang, F. Chu, T. Gao, Y. Song, and L. Dai, "Acoustic feature shuffling network for text-independent speaker verification," in *Proc. Annual Conference of the International Speech Communication Association*, 2022, pp. 4790–4794.

[56] Y. Liu, L. He, J. Liu, and M. Johnson, "Speaker embedding extraction with phonetic information," in *Proc. Annual Conference of the International Speech Communication Association*, 2018, pp. 2247–2251.
[57] S. Wang, J. Rohdin, L. Burget, O. Plchot, Y. Qian, K. Yu, and J. Černocký, "On the usage of phonetic information for text-independent speaker embedding extraction," in *Proc. Annual Conference of the International Speech Communication Association*, 2019, pp. 1148–1152.
[58] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *Proc. International Conference on Acoustics, Speech, and Signal Processing*, 2019, pp. 5796–5800.



Youzhi TU received a B.Eng. degree and an M.Sc. degree from Harbin Engineering University in 2012 and 2015, respectively. He received a Ph.D. degree in electronic and information engineering at The Hong Kong Polytechnic University in 2022. He is now a postdoctoral fellow at The Hong Kong Polytechnic University. His research interests include speaker recognition and machine learning.

Man-Wai MAK (M'93–SM'15) received a Ph.D. in electronic engineering from the University of Northumbria in 1993. He joined the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University in 1993 and is currently a professor in the same department. He has authored more than 200 technical articles in speaker recognition, machine learning, and bioinformatics. Dr. Mak also coauthored postgraduate textbooks *Biometric Authentication: A Machine Learning Approach*, Prentice-Hall, 2005 and *Machine Learning for Speaker Recognition*, Cambridge University Press, 2020. He served as a member of the IEEE Machine Learning for Signal Processing Technical Committee in 2005–2007. He has served as an associate editor of IEEE/ACM Transactions on Audio, Speech and Language Processing. He is currently an associate editor of *Journal of Signal Processing Systems* and *IEEE Biometrics Compendium*. He also served as Technical Committee Members of a number of international conferences, including ICASSP and Interspeech, and gave a tutorial on machine learning for speaker recognition in Interspeech'2016. Dr. Mak's research interests include speaker recognition, machine learning, and bioinformatics.

Jen-Tzung Chien (M'97–SM'04) is currently the Lifetime Chair Professor in National Yang Ming Chiao Tung University. He has published extensively, including three books and more than 250 peer-reviewed articles, many on machine learning, deep learning, and Bayesian learning with applications on natural language processing and computer vision. He received the Best Paper Award in IEEE Automatic Speech Recognition and Understanding Workshop in 2011 and IEEE Machine Learning and Signal Processing Workshop in 2023.