IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 September 2022, accepted 11 September 2022, date of publication 14 September 2022,
date of current version 23 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3206541

SURVEY

A Survey on Text-Dependent and Text-Independent Speaker Verification

YOUZHI TU, WEIWEI LIN, AND MAN-WAI MAK, (Senior Member, IEEE)
Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong, SAR, China

Corresponding author: Man-Wai Mak (enmwmak@polyu.edu.hk)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61971371; and in part by Huawei Technologies Company Ltd., under Project TC20210903021.

ABSTRACT Speaker verification (SV) aims to detect an individual's identity from his/her voice. SV has been successfully applied in various areas such as access control, remote service customization, financial transactions, etc. Depending on whether the text content is pre-defined or not, SV can be text-dependent or text-independent. This paper reviews recent research on text-dependent SV (TD-SV) and text-independent SV (TI-SV). Because most modern SV systems apply deep learning methods to boost performance, we focus on the studies that use deep speaker embedding, a technique representing a person's identity via a fixed-dimensional vector encoded from a variable-length utterance. Rather than detailing every existing SV system, we make an overview of the representative SV systems that have attracted wide attention. Furthermore, an increasing number of SV systems have been devoted to addressing real-world challenges such as reverberation and noise, and this has driven a large number of studies on practical SV. Therefore, the survey compares the existing SV systems in the Far-Field Speaker Verification Challenge 2020 (FFSVC 2020) to illustrate the most effective techniques for both TD-SV and TI-SV.

INDEX TERMS Text-dependent speaker verification, text-independent speaker verification, deep speaker embedding, far-field speaker verification.

I. INTRODUCTION

Speaker verification (SV) aims to determine whether the identity of a claimed utterance matches a target identity. This technology has been applied in many practical scenarios such as access control, service customization, national security, etc. Modern SV systems generally have two types of structures: (1) a cascaded structure comprising a front-end and a backend [see Fig. 1(a)] and (2) an end-to-end structure where the system directly outputs the verification scores or decisions [see Fig. 1(b)]. The difference between these two types of structures lies in how the decision scores are computed. Specifically, the former requires an embedding model—such as the traditional i-vector extractor [1] or a deep embedding network [2]—to produce speaker embeddings and a backend classifier to compute verification scores. In contrast, the latter directly computes the scores of verification trials.

SV can be text-independent or text-dependent. For text-independent SV (TI-SV), because there is no constraint on the lexical content, the speaker embedding extractor is trained on long utterances to suppress the adverse effect of phonetic variability [3], [4]. TI-SV has been studied extensively due to the ease of collecting large-scale text-independent data. In contrast, the lexicon in text-dependent SV (TD-SV) is constrained to a small set of words or phrases. Because of the low degree of phonetic variability, TD-SV usually outperforms TI-SV under short-duration scenarios. This property makes TD-SV more advantageous when the utterance duration is short and the response should be quick [5]. However, to build a well-performed TD-SV system, we need to collect a large amount of in-domain data, which is very expensive in practice.

The recent advances in deep learning and deep neural networks have changed the landscape of speaker verification. Various backbones, such as ResNets and DenseNets, have been integrated into the speaker embedding networks. Compared with i-vector, deep speaker embedding has achieved

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

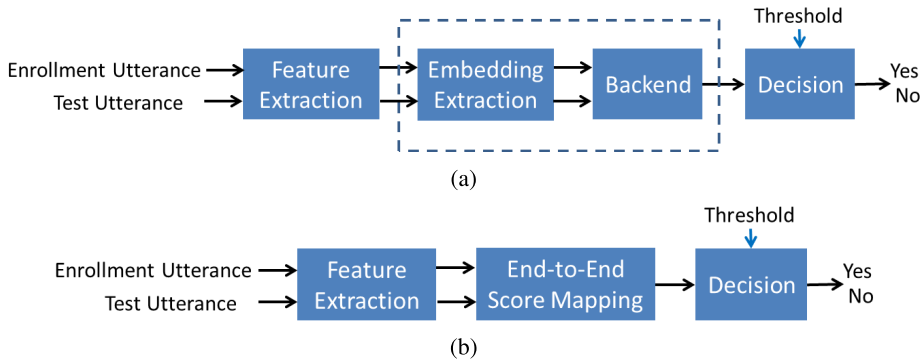


FIGURE 1. (a) A "Front-end + Backend" cascaded SV system and (b) an end-to-end SV system.

state-of-the-art performance and dominated the SV area. However, SV still faces several challenges in practice (see Section II-C for details). Currently, the focus has shifted to using better speaker embedding networks to suppress non-speaker variabilities under various adverse situations. How to develop SV systems that are robust to noise and reverberation, domain mismatch, and short duration remains a difficult problem. Therefore, a comprehensive overview of the current methods and systems is necessary for practitioners and researchers to understand this field better. To this end, this paper surveys the most representative text-dependent and text-independent systems and compares their performance under the same dataset and evaluation protocol. This paper fills the gap in a recent review of speaker recognition [6] by providing comprehensive coverage of text-dependent systems and explaining how they leverage deep speaker embedding to reduce domain mismatch and increase the robustness in the wild.

The paper is organized as follows. Section II briefly introduces the recent development in both TI-SV and TD-SV and the challenges that SV systems face. Then, Section III reviews current representative TI-SV and TD-SV systems. The performance of these systems on a far-field speaker recognition challenge is compared in Section IV. We will give concluding remarks and future trends in Section V.

II. RECENT ADVANCES IN SPEAKER VERIFICATION

In this section, we introduce the recent development of TI-SV and TD-SV and highlight their challenges.

A. TEXT-INDEPENDENT SPEAKER VERIFICATION

TI-SV has long been popular in the speaker recognition community and has received pervasive investigations. Currently, most TI-SV systems use a cascaded structure comprising a front-end and a backend, as shown in Fig. 1(a). The front-end aims to extract speaker characteristics and the backend is responsible for scoring. Typical front-ends are i-vector embedding, x-vector embedding, and the more general deep speaker embedding. Typical backends include cosine similarity measure, probabilistic linear discriminant analysis (PLDA) [7], and heavy-tailed PLDA [8]. Since the era of i-vector [1], TI-SV has been dominated by the i-vector/PLDA framework. With the advancement of deep learning, deep speaker embedding has led to significant performance improvement in TI-SV systems [2], [9], [10], [11].

Classical deep speaker embedding uses a speaker identification network to create a speaker-embedding space. Typically, the embedding network comprises a frame-level subnetwork, a pooling layer, and an utterance-level subnetwork. After frame-level feature extraction and utterance-level aggregation, the speaker embeddings are extracted from the affine output of a fully-connected (FC) layer of the utterance-level subnetwork. Under this framework, various architectures based on convolutional neural networks (CNNs) have been used for frame-level processing. A classic example is the x-vector which uses time delay neural networks (TDNNs) to extract the frame-level features [2]. Later, more advanced networks, such as ResNets [9], DenseNets [10], [12], and Res2Nets [11], [13], were introduced to better model the spectral-temporal relationship across the acoustic frames. Simultaneously, diverse aggregation methods have been proposed to aggregate the frame-level information into utterance-level embeddings, e.g., statistics pooling [2], multi-head attentive pooling [14], NetVLAD-based pooling [9], short-time spectral pooling [15], [16], etc. Also, different training losses besides the softmax loss have been used in deep speaker embedding to achieve better discriminative power. For example, additive margin softmax (AM-Softmax) [17] loss and additive angular margin softmax (AAM-Softmax) loss [18] have been widely used to replace the vanilla softmax counterpart.

Another category of deep speaker embedding uses metric learning for TI-SV [19], [20], [21]. In this category, losses based on some distance measures are used to guide the embedding network so that the speaker embeddings have both large inter-class distance and small intra-class distance. For example, triplet loss [21], prototypical network loss [20], angular prototypical loss [19], and masked proxy loss [22] have been applied to TI-SV and have achieved competitive performance.

Besides the cascade of a front-end and a backend, there are end-to-end architectures with an SV-loss objective (see Fig. 1(b)) for TI-SV [23], [24]. These systems strictly conform to the SV objective in that they directly map an

enrollment-test pair to a score or a decision probability, as shown in Fig. 1(b). In fact, the SV-loss is closely related to deep metric learning in that they both involve mapping a training sample pair/triplet to a similarity score. Their main difference is that the SV-loss uses an additional logistic regression layer to map the similarity score to a decision probability [24].

B. TEXT-DEPENDENT SPEAKER VERIFICATION

Although TI-SV has achieved outstanding performance in various evaluations [2], [11], [25], [26], it suffers severe performance degradation under short-utterance scenarios due to the abundant phonetic mismatch between the enrollment utterances and the test utterances [9], [27]. TD-SV, on the other hand, is more advantageous in short-duration applications due to the limited phonetic variability in the utterances. With the increasing demand for voice-based access control applications, TD-SV has revived recently.

Unlike TI-SV, TD-SV requires the matching of both speakers and text contents. Because TD-SV needs to take text information into account, directly applying a TI-SV model to TD-SV will cause problems. In fact, the authors of [28] have shown that exploiting the text information in utterances can remarkably improve the performance of TD-SV. Different from TI-SV, as shown in Table 1, there are three types of impostor trials in TD-SV: 1) the speaker in the enrollment and verification session matches but the texts do not match, 2) the speakers in the enrollment and verification phases do not match but the text content matches, and 3) neither the speakers nor the contents match [5], [29]. This difference requires TD-SV to adopt additional strategies to deal with the content information, which is ignored in TI-SV. For example, a phrase recognizer may be used to assist the verification of Target-wrong and Impostor-wrong trials in Table 1 [30].

According to whether the text content is fixed or not, TD-SV can be phrase-dependent (phrases are pre-defined by the system) or phrase-independent (the users may customize their own phrases) [29]. For phrase-dependent TD-SV, it is preferable to train the embedding extractor on the matched phrases to make the speaker embeddings reflect the phonetic variability in the pre-defined phrases. Also, the parameters in the conventional channel compensation methods—such as within-class covariance normalization (WCCN) [31], LDA, and PLDA—should be trained on the data with the pre-defined phrases. This strategy helps reject impostors speaking the wrong phrases [32]. For phrase-independent TD-SV, however, because there may be text mismatch between the training utterances and the evaluation utterances, and even mismatch between the enrollment utterances and the verification utterances, specific techniques such as content normalization [28] and text adaptation [33] are required to alleviate the mismatch problem.

Early work on TD-SV was mainly based on the i-vector framework and its variants. In [32], the authors took phonetic variability into consideration when modeling the uncertainty in the i-vectors. To this end, they propagated the i-vectors' uncertainty to a phrase-dependent PLDA model for TD-SV. To incorporate phonetic information into speaker modeling, the authors of [29] proposed the hidden Markov model (HMM) based i-vectors, where mono-phone HMMs were used for frame alignment. The HMM i-vectors are in contrast to the conventional i-vectors where a Gaussian mixture model (GMM) is used for aligning the frames. Another variant of speaker modeling is the bottleneck (BN) feature based i-vector [30], [34]. Rather than extracting i-vectors from the conventional Mel-frequency cepstral coefficients (MFCCs), we extract the i-vectors from the BN features (may be concatenated with MFCCs). Because BN features are obtained from the bottleneck layer of a phonetically-aware DNN trained to classify the phone states, the phonetic information can be incorporated into the sufficient statistics for i-vector extraction. This makes the BN i-vectors both speaker- and phrase-dependent. It was shown that when modeling speakers using BN features, HMM state alignments are not necessary [34]. However, one major drawback of BN i-vectors is that an additional speech recognizer is required to obtain the phone states.

Recently, deep learning has been widely used in TD-SV [33], [35], [36], [37], [38], [39], [40], [41], [42]. One deep learning framework is the end-to-end TD-SV [23], [24]. Another straightforward way is to use the modified architecture of a TI-SV model for TD-SV [37], [40]. For example, under the x-vector framework, the standard deviation vectors rather than the affine output from the utterance-level subnetwork are used as the speaker representations for TD-SV [40]. In [37], a bidirectional attentive pooling layer is incorporated into a DenseNet to better establish the contextual information across the frames. When these models are trained on sufficient in-domain data, good performance can be achieved even though they are expected to perform speaker classification only, i.e., the single-task style.

Although TI-SV models can be well adapted to TD-SV, the text information is actually exploited in an implicit way. A more intuitive strategy is to explicitly explore the phrase information through multi-task learning. In [35], the authors proposed a j-vector system to deal with the contextual information in utterances through multi-task learning. Besides using a speaker classifier as in single-task learning, the j-vector network applies a phrase classifier to explicitly propagate the phrase information to the speaker-embedding layer. To address the text mismatch between the training data and the evaluation data, and also the text mismatch between the enrollment phrases and the test phrases, a speaker-text factorization network was proposed in [33].

TABLE 1. Types of trials in text-dependent speaker verification.

	Target Speaker	Imposter Speaker
Correct Phrase	Target-correct	Imposter-correct
Wrong Phrase	Target-wrong	Imposter-wrong

The network aims to disentangle the speaker information from the text information through two separate subtasks: speaker classification and phoneme classification. Due to the factorization of the speaker and phoneme representations, text-independent speaker embeddings can be adapted to text-dependent ones based on the given phrases. Recently, the authors of [38] proposed a multi-task learning network with phoneme-aware attentive pooling for TD-SV. To exploit the phonetic information in utterances, the posteriors obtained from the frame-level phoneme classifier are used in attentive pooling. Furthermore, attributed to the adversarial training through a segment-level phoneme classification loss, the learned speaker embeddings will be invariant to the phrase variations.

C. CHALLENGES IN SPEAKER VERIFICATION

In practice, we are faced with various challenges. One challenge is that, under noise and reverberation conditions, the performance of both TI-SV and TD-SV will degrade severely [25], [26], [39], [43]. To deal with this problem, researchers applied a speech enhancement module to restore the undistorted speech. For example, in [44], weighted prediction error (WPE) was applied to suppress the late reverberation in multi-channel speech. Later, a neural WPE was proposed in [45] to better estimate the power spectral density for both single- and multi-channel dereverberation. For single-channel speech, the authors of [46] proposed a joint training scheme to optimize a UNet-based speech enhancement front-end and a DenseNet-based speaker-embedding extractor simultaneously. Also, beamforming-based techniques [47], [48] have been used for denoising and dereverberation.

Besides using a speech enhancement front-end, transfer learning is another effective method to address noise and reverberation. In [49] and [50], adversarial learning was exploited to create a noise-invariant embedding space so that the embeddings can generalize to a variety of noise. Also, the authors of [51] implemented a teacher-student learning framework to transfer the knowledge learned from the near-field data to the far-field situation dominated by noise and reverberation.

On the other hand, when the duration of the evaluation utterances becomes short, TI-SV systems will witness a substantial performance drop [5], [9], [27]. As a result, short duration poses another challenge to TI-SV. Although TD-SV is more robust to short utterances, collecting text-dependent training data is expensive.

What is worse, domain mismatch exacerbates the above challenges. Due to the discrepancy between different recording conditions, e.g., differences in channels, languages, noise and so on, the distribution of the training data usually differs from that of the test data. Under this situation, it is necessary to adapt the trained models based on some target-domain data, a strategy known as domain adaptation (DA). On the other hand, due to the high cost of data labeling, only a small amount of labeled data or even no labeled data from the target domain are available. This difficulty necessitates advanced methods to alleviate the domain mismatch challenge.

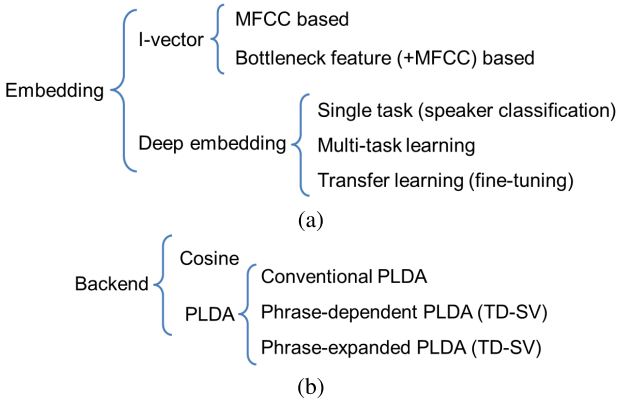


FIGURE 2. Structure of (a) an embedding model and (b) a backend in the cascaded SV systems.

Depending on whether the target-domain data have speaker labels or not, DA can be divided into supervised DA and unsupervised DA. Unsupervised DA has been popular since NIST 2016 SRE, where there is a severe language mismatch between the training and evaluation data [52]. To address this problem, the authors of [53] and [54] minimized the maximum mean discrepancy (MMD) [55] across different languages to create a language-invariant speaker embedding space. Besides, domain adversarial training [56] has been successfully applied [57], [58], [59], [60], [61] to produce language-invariant speaker embeddings. There are also DA methods that directly adapt the PLDA covariance matrices to match the target distribution, e.g., CORAL+ [62] and Kaldi's PLDA adaptation [63].[1]

Recently, studies on DA have focused on the case where a small amount of labeled in-domain data are available. In this situation, many SV systems use transfer learning to fine-tune a source-domain speaker model to the target-domain distribution [39], [41], [42], [43]. In [39], the authors used text-dependent data to fine-tune a text-independent ResNet for the speech in AISHELL-2019B-eval. The same strategy was used to improve the text-dependent evaluation performance in the INTERSPEECH 2020 Far-Field Speaker Verification Challenge (FFSVC 2020). In the latest Short-duration Verification Challenge (SdSVC) 2021, the authors of [41] found that fine-tuning the whole model is more effective than fine-tuning the upper layers only. To explicitly exploit the text information in the fine-tuning operation, a multi-task fine-tuning strategy was introduced for TD-SV in [42], where both a speaker classification head and a phrase classification head were used. Another fine-tuning example for TD-SV is illustrated in [23], where a GE2E contrastive loss was used to fine-tune a text-independent speaker embedding network.

[1] https://github.com/kaldi-asr/kaldi/tree/master/egs/sre16/v2

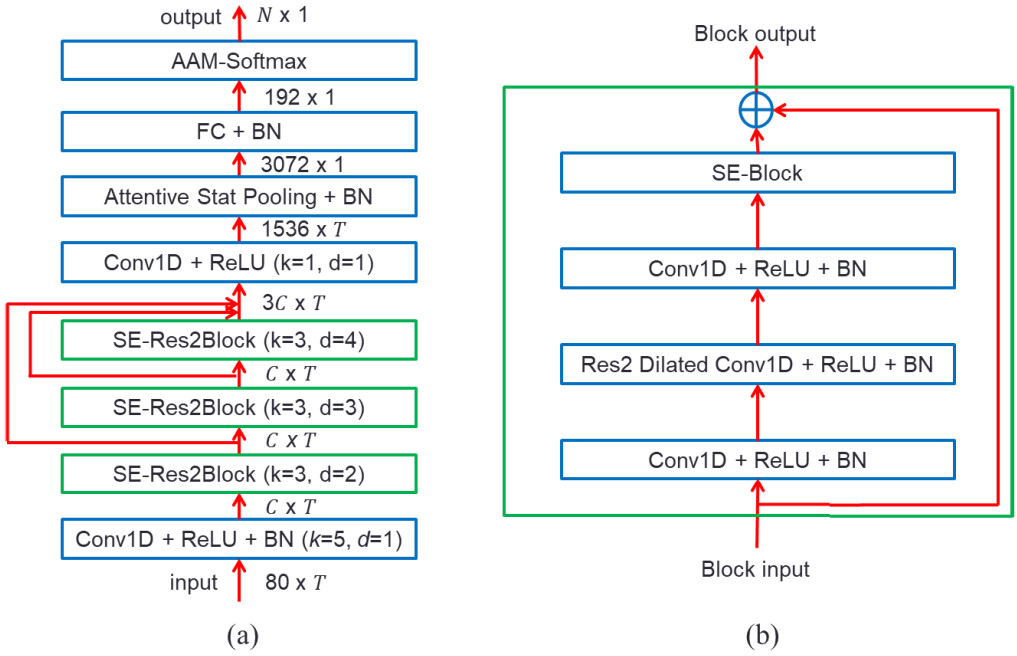


FIGURE 3. (a) Architecture of the ECAPA-TDNN based SV system and (b) detailed structure of the SE-Res2Block used in ECAPA-TDNN. *k* and *d* are the kernel size and dilation rate of the convolutional filters, respectively. *C* and *T* denote the number of channels and the number of frames in the feature map, respectively. *N* is the number of speakers in the training data. The "Res2" in the second layer of (b) means residual-like connections of a standard Res2Net module. (Adapted from [11]).

To make the organization of this paper clear, we illustrate the detailed structure of the "Front-end + Backend" SV system in Fig. 2. Note that the system structure is applicable to both TI-SV and TD-SV except that the phrase-dependent PLDA [30] and phrase-expanded PLDA [40] are designed for TD-SV. Although the TI-SV methods mentioned in Section II-A are mostly investigated on English corpora, they can well be generalized to other languages.

III. REPRESENTATIVE SPEAKER VERIFICATION SYSTEMS

As shown in Fig. 2, there are various categories of SV systems for both TI-SV and TD-SV. In this section, rather than describing every details of these systems, we focus on representative SV systems reported recently.

A. TEXT-INDEPENDENT SV SYSTEMS

State-of-the-art TI-SV systems are mostly based on the cascade of a deep speaker embedding network and a backend. Moreover, we mostly use a single-task network architecture (speaker classifier) to extract speaker embeddings, with possible fine-tuning processes to transfer the source-domain knowledge to the target domain.

1) ECAPA-TDNN

One of the recent advances in deep speaker embedding for TI-SV is the Emphasized Channel Attention, Propagation and Aggregation in TDNN (ECAPA-TDNN) [11], [64]. The architecture of ECAPA-TDNN is shown in Fig. 3, which follows the framework of the x-vector extractor. There are four differences between the ECAPA-TDNN and the x-vector extractor: 1) the former uses Res2Net blocks [65] with multi-layer feature aggregation to replace the conventional TDNN structure for better frame-level information propagation; 2) a channel- and content-dependent statistics pooling layer is used in the ECAPA-TDNN to better emphasize the contribution of the discriminative channels; 3) channel calibration is achieved through the squeeze-and-excitation (SE) blocks [66] to better model the interdependencies across the channels; and 4) additive angular margin softmax (AAM-Softmax) loss [18] rather than the vanilla softmax loss is used in ECAPA-TDNN to better enforce intra-speaker compactness. Attributed to these improvements, ECAPA-TDNN has achieved state-of-the-art TI-SV performance on VoxCeleb1 [11] and on SdSVC 2020 [64].

2) DOMAIN-BALANCED HARD PROTOTYPE MINING

When there is domain mismatch between the training data and the evaluation data, fine-tuning is an effective way to boost SV performance. In [64], a domain-balanced hard prototype mining (HPM) technique was proposed to exploit the "harder" speakers who confuse the system during the fine-tuning process. In contrast to metric learning in which considerable effort is made to mine hard negative samples [19], HPM is easier to implement and nicely adoptable to the AAM-Softmax loss.

Because it is impossible to compute a similarity matrix across all utterances in a mini-batch to deduce speaker confusion, the weights of the AAM-Softmax layer are used as the proxies of the class centers of the training speakers.

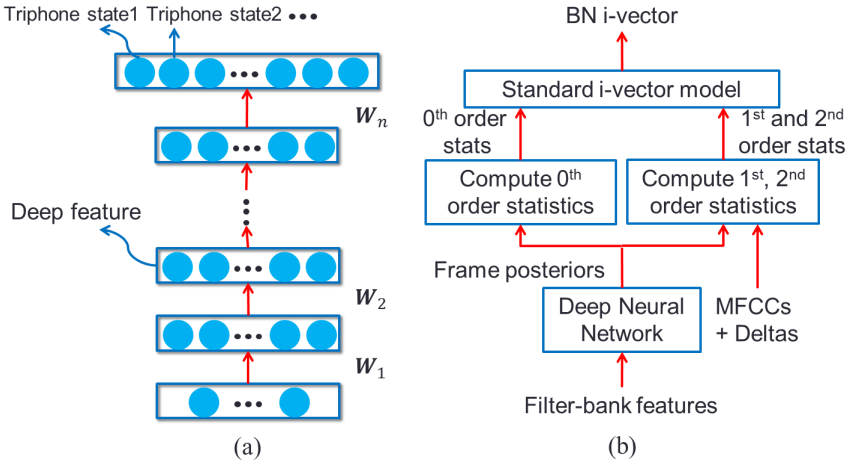


FIGURE 4. (a) Extraction of the bottleneck features from a triphone-state classifier and (b) the process of the BN i-vector extraction. ((a) and (b) are adapted from [34] and [67], respectively).

These weights are referred to as speaker prototypes in HPM. Given N training speakers and a mini-batch of size B, the AAM-Softmax loss is expressed as

$$\mathcal{L} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))}+\sum_{j=1,j\neq y_i}^{N}e^{s(\cos(\theta_j))}}, \quad (1)$$

where m and s denote the angular margin and the scaling factor, respectively. θ_{y_i} represents the angle between the speaker embedding $\mathbf{z}_i$ and the speaker prototype $\mathbf{w}_{y_i}$, where y_i is the corresponding speaker label; whereas θ_j is the angle between $\mathbf{z}_i$ and the speaker prototype $\mathbf{w}_j$. Note that $\mathbf{w}_j$ is the j-th column of the weight matrix $\mathbf{W}$ in the AAM-Softmax layer, i.e., $\mathbf{W} = \{\mathbf{w}_j\}_{j=1}^{N}$, and $\mathbf{w}_j$ is normalized to unit length. Based on the weight matrix $\mathbf{W}$, the speaker similarity matrix $\mathbf{S} \in \mathbb{R}^{N\times N}$ can be computed as $\mathbf{S} = \mathbf{W}^{\top}\mathbf{W}$, whose elements represent the cosine distances between all pairs of speaker prototypes.

Once we obtain the prototype similarity matrix $\mathbf{S}$, we may create mini-batches by sampling the most difficult speakers. This, however, would lead to a problem in which only a small group of speakers will be frequently sampled in the fine-tuning process, reducing the diversity of the training samples. To avoid poor generalization, for each iteration, S speakers are randomly selected from the N training speakers. For each selected speaker, U utterances are sampled from his/her most similar I speakers, including the selected speaker, i.e., $B = S \times U \times I$. The similarity matrix $\mathbf{S}$ will be updated when all the speakers have been iterated over in the mini-batch generation. This process is called HPM.

In reality, the scale of the out-of-domain data is usually much larger than that of the in-domain data, and using a small amount of in-domain data to fine-tune a pre-trained model can easily lead to overfitting. To alleviate this problem, a better idea is to fine-tune a pre-trained model using both out-of-domain data and in-domain data, so that the resulting embeddings are more robust to domain mismatch. Suppose we have N_{in} in-domain speakers ($N_{in} < N$), we follow the HPM strategy for each mini-batch generation. For each mini-batch, we iteratively sample S speakers from both N_{in} in-domain speakers and N_{in} out-of-domain speakers. Once these $2N_{in}$ speakers have been used up, we update the similarity matrix $\mathbf{S}$ and randomly selected N_{in} new speakers from the N out-of-domain speakers again (together with N_{in} in-domain speakers). This process is referred to as domain-balanced HPM, which samples hard speakers not only from the target-domain but also from the source domain. This fine-tuning strategy has shown much better performance than that of fine-tuning on the in-domain data only [64].

B. TEXT-DEPENDENT SV SYSTEMS

TD-SV not only deals with the speaker information but also the text information in the utterances. Therefore, TI-SV methods cannot be directly used in the TD scenarios. In this section, we will introduce several typical TD-SV systems according to the organization in Fig. 2.

1) BOTTLENECK FEATURE BASED I-VECTORS

In [34], the authors investigated BN-feature-based i-vectors that use the activations at the bottleneck layers of a triphone-state classifier as acoustic features for i-vector extraction. The process of extracting BN i-vectors is shown in Fig. 4. Unlike the conventional i-vector extractor, a BN i-vector extractor computes the Baum-Welch statistics (see Fig. 4(b)) from the BN features or their concatenation with MFCCs [34], [67]. Because BN features can capture abundant phonetic information from the triphone-state classifier, the resulting BN i-vectors can better represent the text information in the utterances. Although using BN features for i-vector extraction is not new, it is still competitive in TD-SV. For example, in [30], the authors showed that the BN i-vectors remarkably outperform the x-vectors that were trained on sufficient in-domain data. However, a major disadvantage of BN i-vectors is that to prepare the BN features for i-vector extraction, an additional phonetic-aware DNN needs to be

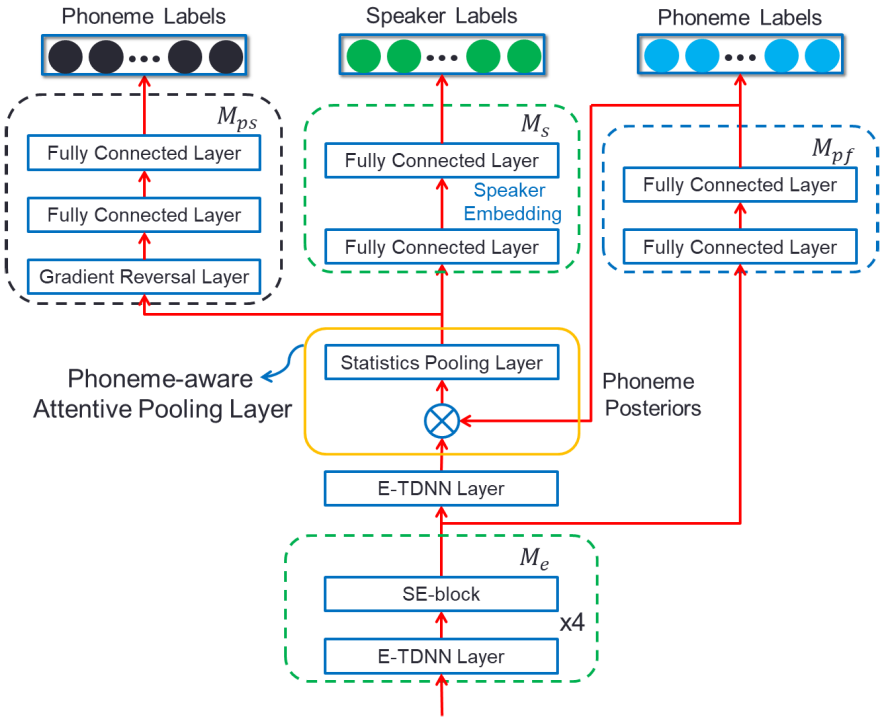


FIGURE 5. Architecture of speaker-phoneme multi-task learning. The pipeline of phoneme-aware attentive pooling is shown in the orange block. (Adapted from [37]).

trained on a large amount of speech data. This will inevitably increase the cost of deploying the SV system.

2) SPEAKER-PHONEME MULTI-TASK LEARNING

Many TD speaker embedding networks are based on single-task learning to classify speakers. In [37], a bidirectional gated recurrent unit (BGRU) layer and an attentive pooling layer are combined to better capture the long-range context information and simultaneously highlight the discriminative frames during aggregation. In this architecture, the phonetic information is implicitly exploited for TD-SV.

To explicitly incorporate text information into the speaker embeddings, we may apply multi-task learning through both speaker classification and phoneme classification. In [38], speaker-phoneme multi-task learning was proposed to produce phoneme-aware speaker embeddings. As shown in Fig. 5, the network is comprised of a shared frame-level encoder M_e, a frame-level phoneme classifier M_{pf}, a speaker classifier M_s, and a segment-level phoneme classifier M_{ps}. To incorporate phonetic information into the segment-level subnetworks shown in Fig. 5(b), the phoneme posteriors produced by M_{pf} are used to weight the convolutional feature maps before statistics pooling:

$$\text{PhoneAttPool} = \text{StatsPool}\left(scale \cdot \text{Softmax}\left(\mathbf{p} \cdot \mathbf{out}^{l_5}\right)\right), \tag{2}$$

where *scale* is a constant, $\mathbf{p}$ is the frame-based phoneme posterior vector produced by M_{pf}, $\mathbf{out}^{l_5}$ is the frame-based output vector at the 5-th TDNN layer, and $(\cdot)$ is the dot product. As a result, phoneme-discriminative frames can be emphasized when producing segment-level embeddings. On the other hand, by adding a phoneme classifier M_{ps}, segment-level adversarial learning is introduced to make the speaker embeddings invariant to the phoneme variations in the utterances. Adversarial learning is accomplished by implementing a gradient reversal layer at the bottom of M_{ps} so that the gradients with respective to the segment-level phoneme classification loss are reversed in backpropagation.

Denote $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ as a sequence of acoustic vectors, $\mathbf{y}^s$ as the speaker label of $\mathbf{X}$, and $\mathbf{Y} = \{\mathbf{y}_1^{pf}, \ldots, \mathbf{y}_N^{pf}\}$ as the phoneme labels of $\mathbf{X}$. The corresponding segment-level phoneme label $\mathbf{y}^{ps}$ is defined as the normalized categorical occurrences of phonemes, i.e.,

$$\mathbf{y}^{ps} = \{y_c\}_{c=1}^{C}, \quad y_c = N_c/N, \tag{3}$$

where N_c is the number of occurrences of the c-th phoneme, N is the number of frames in $\mathbf{X}$, and C is the number of phonemes in the selected phoneme set. To optimize the network, we define the total loss as a combination of the speaker classification loss $\mathcal{L}_s$, the frame-level phoneme classification loss $\mathcal{L}_{pf}$, and the segment-level phoneme classification loss $\mathcal{L}_{ps}$,

$$\mathcal{L} = \mathcal{L}_s + \alpha\mathcal{L}_{pf} + \beta\mathcal{L}_{ps}, \tag{4}$$

where α and β are hyperparameters controlling the contribution of $\mathcal{L}_{pf}$ and $\mathcal{L}_{ps}$, respectively. $\mathcal{L}_s$, $\mathcal{L}_{pf}$, and $\mathcal{L}_{pf}$ are expressed as

$$\mathcal{L}_s = \text{CE}\left(M_s\left(M_e(\mathbf{X})\right), \mathbf{y}^s\right), \tag{5}$$

TABLE 2. TI-SV performance of existing best systems (without fusion) on the development set of FFSVC 2020.

System	Training Data	Front-end	Backend	EER (%)	minDCF	Advantages	Disadvantages
[43]	• SLR33, SLR38, SLR47, SLR49, SLR62, and SLR68 for pre-training • FFSVC20 TI-SV training data for fine-tuning • Pyroomacoustics Aug [69]	• 64-D Mel-frequency filterbank features • Gradient boosting based VAD [70] • ResNet-34 • Softmax loss	• Test embedding averaging • Cosine	5.83	0.580	Advanced VAD	Vanilla softmax loss instead of AAM-Softmax or AM-Softmax loss
[71]	• Concatenated Voxceleb1–2 for pre-training • SLR33, SLR62, SLR82, SLR85, and FFSVC20 TI-SV training data for fine-tuning • Kaldi Aug	• 80-D filterbank features • U-net based VAD • ResNet34 • AM-Softmax loss	• Mean adaptation • Cosine	4.46	0.484	• Advanced VAD • Domain adaptation in the backend	Large number of training speakers (larger training model)
[72]	• SLR18, SLR33, SLR47, SLR49, SLR62, SLR68, SLR85, and Voxceleb2 • Pyroomacoustics Aug, SI Aug, and Kaldi Aug	• WPE [73] • Beamforming • 80-D filterbank features and pitch • ResNet34 • AAM-Softmax loss	• Cosine • ASnorm	3.32	0.435	• Used speech enhancement front-ends • Diverse data augmentations	Large number of training speakers (larger training model)
Ours	• CN-Celeb1–2, AISHELL-2019B-eval, and FFSVC20 TI-SV training data • Kaldi Aug	• 40-D filterbank features • Energy-based VAD • Standard x-vector network • AM-Softmax loss	Cosine	6.93	0.710	• Small amount of training data • Light-weight embedding model (efficient training)	Slightly worse but reasonable performance

$$\mathcal{L}_{pf} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{CE}\left(M_{pf}\left(M_e\left(\mathbf{x}_i\right)\right), \mathbf{y}_i^{pf}\right), \tag{6}$$

and

$$\mathcal{L}_{ps} = \mathrm{KL}\left(M_{ps}\left(M_e\left(\mathbf{X}\right)\right), \mathbf{y}^{ps}\right), \tag{7}$$

respectively, where CE and KL stand for cross-entropy loss and Kullback–Leibler (KL) divergence, respectively. Note that because the segment-level phoneme label $\mathbf{y}^{ps}$ is not in one-hot format, KL divergence instead of cross-entropy loss is used in (7).

Speaker-phoneme multi-task learning has achieved substantial improvement in RSR2015 compared with existing TD-SV systems [38]. However, an ASR model is required to generate phoneme labels, which increases the cost of system deployment.

3) MULTI-TASK FINE-TUNING

Similar to the case in Section III-B2 where multi-task learning is used in pre-training, we can also use multi-task learning in the fine-tuning process to improve TD-SV performance. In [42], the authors investigated two different fine-tuning strategies using both speaker labels and phrase labels: "speaker + phrase" and "speaker × phrase". As shown in Fig. 6, "speaker + phrase" follows a multi-task fine-tuning style with two separate classification heads. In the "speaker × phrase" mode, however, only a single head is used in the output layer of the classifier, and utterances in different phrases with the same speaker identity are considered different classes. It was shown in [42] that the "speaker + phrase" mode outperforms the "speaker × phrase" strategy on the TD task in SdSVC 2021, which verifies the effectiveness of multi-task fine-tuning.

IV. PERFORMANCE COMPARISONS

In this section, we compare the performance on the recent Far-Field Speaker Verification Challenge (FFSVC) 2020 data [68]. FFSVC20 focuses on the smart home scenario where far-field distributed microphone arrays are used in noisy environments. The utterances in FFSVC20 are recorded by one close-talking microphone, one iPhone, and six circular microphone arrays. The language is Mandarin. The enrollment utterances and the test utterances in both TI-SV and TD-SV tasks come from different microphones.

TABLE 3. TD-SV performance of existing best systems (without fusion) on the evaluation set of FFSVC 2020.

System	Training Data	Front-end	Backend	EER (%)	minDCF	Advantages	Disadvantages
[43]	• SLR33, SLR38, SLR47, SLR49, SLR62, and SLR68 for pre-training • SLR85 and FFSVC20 TD-SV training data for fine-tuning • Pyroomacoustics Aug	• 64-D Mel-frequency filterbank features • Gradient boosting based VAD • ResNet-34 • Softmax loss	• Test embedding averaging • Cosine	6.37	0.620	Advanced VAD	Vanilla softmax loss instead of AAM-Softmax or AM-Softmax loss
[74]	• SLR33, SLR38, SLR62, SLR68, and FFSVC20 TD-SV training data for pre-training • FFSVC20 training data for fine-tuning • Kaldi Aug with data selection • SpecAug [75]	• 30-D MFCCs • Energy-based VAD • ReseNet-BAM • Softmax loss • Domain adversarial training	Cosine	4.81	0.454	• Advanced frame-level architecture • Advanced domain adaptation (domain adversarial training)	Large number of training speakers (larger training model)
[37]	• SLR85 and FFSVC20 TD-SV training data • Online data augmentation	• 30-D filterbank features • DenseNet • Bidirectional attentive pooling • AM-Softmax loss	PLDA	5.78	0.570	• Advanced frame-level architecture • Advanced pooling strategy	Used in-domain training data only
Ours	• CN-Celeb1–2, AISHELL-2019B-eval, and FFSVC20 TD-SV training data • Kaldi Aug	• 40-D filterbank features • Energy-based VAD • Standard x-vector network • AM-Softmax loss	Cosine	7.02	0.740	• Small amount of training data • Light-weight embedding model (efficient training)	Slightly worse but reasonable performance

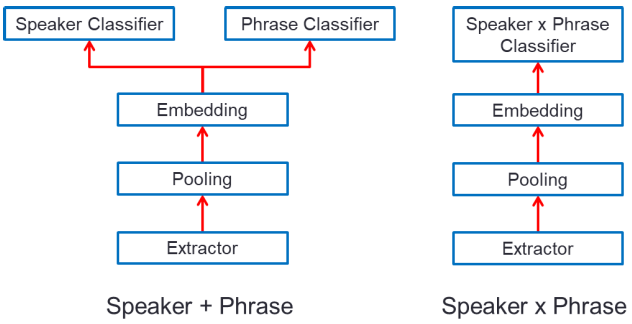


FIGURE 6. Illustration of two fine-tuning strategies in TD-SV. The left subfigure shows the "speaker + phrase" method with a speaker classification head and a phrase classification head; whereas the right "speaker × phrase" method uses a single classification head but with more output nodes. (Adapted from [42]).

A. TEXT-INDEPENDENT EVALUATION

Task 2 of FFSVC 2020 falls into the text-independent category [68]. The training set contains 120 speakers speaking Mandarin. This dataset, together with the SLR-85 HI-MIA data[2] can be used as in-domain data for domain knowledge transfer. Besides, any publicly accessible data shared on openslr.org before 1st February 2020 can be used to develop the TI-SV systems. Because most participants in this challenge only reported the performance of fused systems on the evaluation set, we present the results of single systems on the development set only for fair comparisons. The performance of some top performing systems (without fusion) on the development set is shown in Table 2.

The official baseline system [43] (the first row of Table 2) used public data from openslr.org for pre-training the embedding model and adopted fine-tuning to transfer the knowledge learned from the pre-training data to the FFSVC20 TI-SV task. The pre-training set comprises 10,544 speakers. The system in [71] used a similar number of speakers for pre-training and fine-tuning. Because the system uses more advanced VAD (with a U-net structure) in the front-end and mean adaptation in the backend, its performance is better than

[2]http://openslr.org/85/.

that of the system in [43]. The embedding network in [72] was trained on a 11,120-speaker dataset. Before acoustic feature extraction, WPE and beamforming were applied to alleviate reverberation and to take into account the array information, respectively. Also, diverse augmentation methods were used in data preparation. These improvements contribute to better performance than the system in [71]. For our system (the last row of Table 2), we only used 3,118 speakers to prepare the training data. Besides, we used the standard x-vector network as the embedding model, which is not as capable as the advanced ResNet34. Therefore, we obtained slightly worse but reasonable performance as compared with the system in [43].

B. TEXT-DEPENDENT EVALUATION

Task 1 of FFSVC 2020 is text-dependent. The text content is "ni hao mi ya" in Mandarin. There are 120 speakers in the text-dependent training data. Similar to the TI-SV task, public data from openslr.org can be used in the system development. Table 3 shows the performance of several top performing systems (without fusion) on the evaluation set.

The official TD-SV system [43] is similar to the official TI-SV system in Table 2, except that FFSVC20 TD-SV data were used for fine-tuning. The system in [74] was pre-trained on Mandarin utterances from 3,211 speakers and used ResNet-BAM as the embedding extractor. Also, the authors applied domain adversarial training to further reduce the mismatch between the TI data and the TD data. These implementations contribute to better performance than the system in [43]. Interestingly, although the system in [37] was trained on a smaller number of speakers than the system in [43], it still achieved better performance. This can be due to that the system in [37] uses more advanced embedding network (DenseNet), more effective aggregation strategy (bidirectional attentive pooling), and a more complex backend (PLDA model). Our TD-SV system was based on the same framework as that in the TI-SV task except that we used FFSVC20 TD-SV data for fine-tuning. We obtained reasonable performance as compared with the system in [43] because we neither trained our system on a large number of speakers nor did we use an advanced embedding network and powerful backends.

V. CONCLUDING REMARKS AND FUTURE TRENDS

In this paper, we briefly review the recent studies on TI-SV and TD-SV. Compared with TI-SV, where the context information is considered nuisance variability, TD-SV takes both speaker and phonetic information into account during speaker modeling. With the advances in deep learning, SV has achieved remarkable progress in many aspects such as domain-invariant learning, robust SV in the wild, and short-duration SV. These improvement has been reflected in the recent SV challenges. Specifically, in this paper, we compare the performance of several best performing systems on FFSVC 2020.

A. CONCLUDING REMARKS

The concluding remarks are summarized as follows:

1) Advanced convolutional layers/blocks such as DenseNet and ResNet are prevalent in SV.
2) Most existing SV systems are implemented in a "Front-end + Backend" structure.
3) Fine-tuning is an effective tool to improve the performance of TI-SV and TD-SV.
4) Multi-task learning seems to be unattractive for TD-SV.

B. FUTURE TRENDS

As mentioned in Section II-C, SV faces many challenges in real-world applications. Background noise, reverberation effect, short utterances, microphone mismatches, and language mismatches have always been and will continue to be the critical issues in robust speaker verification. Although the current SV systems can partially address these problems, the solutions are scenario-specific, e.g., an SV system that can address noise could fail miserably when the utterances are very short. Therefore, seeking principled solutions that can generalize across different tasks is essential in the future.

On the other hand, to facilitate system deployment, model compression techniques such as knowledge distillation [76] and network pruning [77] have received increasing attention. However, due to the trade-off between the system performance and the runtime efficiency, developing lightweight and effective SV systems is challenging and worths further research.

Recently, the research on security in SV has also attracted great attention and many studies have been focusing on defending SV systems against malicious spoofing attacks through replay, speech synthesis, voice conversion, and adversarial samples [78], [79], [80]. Unlike previous ASVspoof tasks [78], [79], which aim to develop countermeasures (CMs) for a fixed SV system, the spoofing-aware speaker verification (SASV) challenge [80] focuses on the optimization of both CMs and SV subsystems to improve the SV reliability. In this regard, SASV will attract extensive attention in the future.

REFERENCES

[1] N. Dehak, P. J. Kenny, R. Dehak, D. Pierre, and O. Pierre, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 4, pp. 788–798, May 2011.
[2] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN embeddings for speaker recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5329–5333.
[3] T. Hasan, R. Saeidi, J. Hansen, and D. van Leeuwen, "Duration mismatch compensation for I-vector based speaker recognition systems," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7663–7667.
[4] W. Chen, J. Huang, and T. Bocklet, "Length- and noise-aware training techniques for short-utterance speaker recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3835–3839.
[5] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Short-duration speaker verification (SdSV) challenge 2021: The challenge evaluation plan," 2019, *arXiv:1912.06311*.
[6] Z. Bai and X. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Netw.*, vol. 140, pp. 65–99, Aug. 2021.
[7] S. Ioffe, "Probabilistic linear discriminant analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 531–542.
[8] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. Odyssey, Speaker Lang. Recognit. Workshop*, 2010.

[9] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 5791–5795.
[10] W. W. Lin, M. W. Mak, and L. Yi, "Learning mixture representation for deep speaker embedding using attention," in *Proc. Odyssey, Speaker Lang. Recognit. Workshop*, 2020, pp. 210–214.
[11] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3830–3834.
[12] W. Lin and M.-W. Mak, "Mixture representation learning for deep speaker embedding," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 30, pp. 968–978, 2022.
[13] Y. Z. Tu and M. W. Mak, "Mutual information enhanced training for speaker embedding," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 91–95.
[14] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2018, pp. 3573–3577.
[15] Y. Z. Tu and M. W. Mak, "Short-time spectral aggregation for speaker embedding," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 6708–6712.
[16] Y. Tu and M.-W. Mak, "Aggregating frame-level information in the spectral domain with self-attention for speaker embedding," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 30, pp. 944–957, 2022.
[17] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Process. Lett.*, vol. 25, no. 7, pp. 235–238, Jul. 2018.
[18] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4690–4699.
[19] J. Chung, J. Huh, S. Mun, M. Lee, H. Heo, S. Choe, C. Ham, S. Jung, B. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 2977–2981.
[20] J. Wang, K. Wang, M. Law, F. Rudzicz, and M. Brudno, "Centroid-based deep metric learning for speaker recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 3652–3656.
[21] C. Zhang, K. Koishida, and J. H. L. Hansen, "Text-independent speaker verification based on triplet convolutional neural network embeddings," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 9, pp. 1633–1644, Sep. 2018.
[22] J. Lian, A. Kumar, H. Dhamyal, B. Raj, and R. Singh, "Masked proxy loss for text-independent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 4638–4642.
[23] L. Wan, Q. Wang, A. Papir, and I. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 4879–4883.
[24] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 5115–5119.
[25] D. Snyder, J. Villalba, N. Chen, D. Povey, G. Sell, N. Dehak, and S. Khudanpur, "The JHU speaker recognition system for the VOiCES 2019 challenge," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 2468–2472.
[26] P. Matějka, O. Plchot, H. Zeinali, L. Mošner, A. Silnova, L. Burget, O. Novotný, and O. Glembek, "Analysis of BUT submission in far-field scenarios of VOiCES 2019 challenge," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 2448–2452.
[27] A. Gusev, V. Volokhov, T. Andzhukaev, S. Novoselov, G. Lavrentyeva, M. Volkova, A. Gazizullina, A. Shulipa, A. Avdeeva, A. Ivanov, A. Kozlov, T. Pekhovsky, and Y. Matveev, "Deep speaker embeddings for far-field speaker recognition on short utterances," in *Proc. Odyssey, Speaker Lang. Recognit. Workshop*, 2020, pp. 179–186.
[28] S. Dey, S. Madikeri, P. Motlicek, and M. Ferras, "Content normalization for text-dependent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 1482–1486.
[29] H. Zeinali, H. Sameti, and L. Burget, "HMM-based phrase-independent I-vector extractor for text-dependent speaker verification," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 7, pp. 1421–1435, Jul. 2017.
[30] A. Lozano-Diez, A. Silnova, B. Pulugundla, J. Rohdin, K. Veselỳ, L. Burget, O. Plchot, O. Glembek, O. Novotnỳ, and P. Matejka, "BUT text-dependent speaker verification system for SdSV challenge 2020," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 761–765.
[31] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2006, pp. 1471–1474.
[32] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, and P. Dumouchel, "Text-dependent speaker recognition using PLDA with uncertainty propagation," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2013, pp. 3684–3688.
[33] Y. Yang, S. Wang, X. Gong, Y. Qian, and K. Yu, "Text adaptation for speaker verification with speaker-text factorized embeddings," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 6454–6458.
[34] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Commun.*, vol. 73, pp. 1–13, Oct. 2015.
[35] N. Chen, Y. Qian, and K. Yu, "Multi-task learning for text-dependent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2015.
[36] T. Liu, M. Madhavi, R. Das, and H. Li, "A unified framework for speaker and utterance verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 4320–4324.
[37] P. Zhang, P. Hu, and X. Zhang, "Deep embedding learning for text-dependent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3461–3465.
[38] Y. Liu, Z. Li, and Q. Hong, "Phoneme-aware and channel-wise attentive learning for text dependent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 101–105.
[39] X. Qin, D. Cai, and M. Li, "Far-field end-to-end text-dependent speaker verification based on mixed training data with transfer learning and enrollment data augmentation," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 4045–4049.
[40] Z. Chen and Y. Lin, "Improving X-vector and PLDA for text-dependent speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 726–730.
[41] P. Zhang, P. Hu, and X. Zhang, "Investigation of IMU&Elevoc submission for the short-duration speaker verification challenge 2021," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 2322–2326.
[42] B. Han, Z. Chen, Z. Zhou, and Y. Qian, "The SJTU system for short-duration speaker verification challenge 2021," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 2332–2336.
[43] X. Qin, M. Li, H. Bu, W. Rao, R. Das, S. Narayanan, and H. Li, "The INTERSPEECH 2020 far-field speaker verification challenge," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3456–3460.
[44] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 7, pp. 1717–1731, Sep. 2010.
[45] K. Kinoshita, M. Delcroix, H. Kwon, T. Mori, and T. Nakatani, "Neural network-based spectrum estimation for online WPE dereverberation," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 384–388.
[46] Z. Gao, M. W. Mak, and W. W. Lin, "UNet-DenseNet for robust far-field speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2022, pp. 1–5.
[47] H. Taherian, Z.-Q. Wang, and D. Wang, "Deep learning based multi-channel speaker recognition in noisy and reverberant environments," in *Proc. Interspeech*, Sep. 2019, pp. 4070–4074.
[48] J. Yang and J. Chang, "Joint optimization of neural acoustic beamforming and dereverberation with X-vectors for robust speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 4075–4079.
[49] J. Zhou, T. Jiang, L. Li, Q. Hong, Z. Wang, and B. Xia, "Training multi-task adversarial network for extracting noise-robust speaker embedding," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 6196–6200.
[50] Z. Meng, Y. Zhao, J. Li, and Y. Gong, "Adversarial speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 6216–6220.
[51] L. Zhang, Q. Wang, K. Lee, L. Xie, and H. Li, "Multi-level transfer learning from near-field to far-field speaker verification," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 1094–1098.
[52] NIST. (2016). *NIST 2016 Speaker Recognition Evaluation Plan*. [Online]. Available: https://www.nist.gov/itl/iad/mig/speaker-recognition-evaluation-2016
[53] W. W. Lin, M. W. Mak, and J. T. Chien, "Multi-source I-vectors domain adaptation using maximum mean discrepancy based autoencoders," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 16, no. 12, pp. 2412–2422, Dec. 2018.

[54] W. W. Lin, M. W. Mak, N. Li, D. Su, and D. Yu, "Multi-level deep neural network adaptation for speaker verification using MMD and consistency regularization," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 6839–6843.

[55] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel method for the two-sample-problem," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 513–520.

[56] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2096–2130, 2016.

[57] Q. Wang, W. Rao, S. Sun, L. Xie, E. S. Chng, and H. Li, "Unsupervised domain adaptation via domain adversarial training for speaker recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 4889–4893.

[58] Y. Tu, M.-W. Mak, and J.-T. Chien, "Variational domain adversarial learning for speaker verification," in *Proc. Interspeech*, Sep. 2019, pp. 4315–4319.

[59] J. Rohdin, T. Stafylakis, A. Silnova, H. Zeinali, L. Burget, and O. Plchot, "Speaker verification using end-to-end adversarial language adaptation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6006–6010.

[60] Y. Tu, M.-W. Mak, and J.-T. Chien, "Variational domain adversarial learning with mutual information maximization for speaker verification," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 2013–2024, 2020.

[61] M. Sang, W. Xia, and J. Hansen, "DEAAN: Disentangled embedding and adversarial adaptation network for robust speaker representation learning," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 6154–6158.

[62] K. A. Lee, Q. Wang, and T. Koshinaka, "The CORAL+ algorithm for unsupervised domain adaptation of PLDA," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 5821–5825.

[63] P. Bousquet and M. Rouvier, "On robustness of unsupervised domain adaptation for speaker recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 2958–2962.

[64] J. Thienpondt, B. Desplanques, and K. Demuynck, "Cross-lingual speaker verification with domain-balanced hard prototype mining and language-dependent score normalization," in *Proc. Interspeech*, Oct. 2020, pp. 756–760.

[65] S. H. Gao, M. M. Cheng, and K. Zhao, "Res2Net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

[66] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[67] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 1695–1699.

[68] X. Qin, M. Li, H. Bu, R. Das, W. Rao, S. Narayanan, and H. Li. (2020). *The FFSVC 2020 Evaluation Plan*. [Online]. Available: http://2020.ffsvc.org/The_FFSVC2020_Evaluation_Plan.pdf

[69] R. Scheibler, E. Bezzam, I. Dokmanic, and M. McLaren, "Pyroomacoustics: A Python package for audio room simulation and array processing algorithms," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 351–355.

[70] L. Prokhorenkova, G. Gusev, A. Vorobev, A. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 6638–6648.

[71] A. Gusev, V. Volokhov, A. Vinogradova, T. Andzhukaev, A. Shulipa, S. Novoselov, T. Pekhovsky, and A. Kozlov, "STC-innovation speaker recognition systems for far-field speaker verification challenge 2020," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3466–3470.

[72] Y. Tong, W. Xue, S. Huang, L. Fan, C. Zhang, G. Ding, and X. He, "The JD AI speaker verification system for the FFSVC 2020 challenge," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3476–3480.

[73] L. Drude, J. Heymann, C. Boeddeker, and R. Haeb-Umbach, "NARA-WPE: A Python package for weighted prediction error dereverberation in numpy and tensorflow for online and offline processing," in *Proc. Speech Commun., 13th ITG-Symp.*, 2018, pp. 1–5.

[74] L. Zhang, J. Wu, and L. Xie, "NPU speaker verification system for INTERSPEECH 2020 far-field speaker verification challenge," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3471–3475.

[75] D. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. Cubuk, and Q. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 2613–2617.

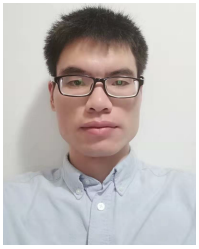
[76] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. NIPS*, 2014.

[77] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. Graf, "Pruning filters for efficient convnets," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–13.

[78] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. Kinnunen, and K. Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2019, pp. 1008–1012.

[79] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. Lee, T. Kinnunen, N. Evans, and H. Delgado, "ASVspoof2021: Accelerating progress in spoofed and deep fake speech detection," in *Proc. ASVspoof Workshop*, 2021, pp. 47–54.

[80] J.-W. Jung, H. Tak, H.-J. Shim, H.-S. Heo, B.-J. Lee, S.-W. Chung, H.-G. Kang, H.-J. Yu, N. Evans, and T. Kinnunen, "SASV challenge 2022: A spoofing aware speaker verification challenge evaluation plan," 2022, *arXiv:2201.10283*.



YOUZHI TU received the B.Eng. and M.Sc. degrees from Harbin Engineering University, in 2012 and 2015, respectively, and the Ph.D. degree in electronic and information engineering from The Hong Kong Polytechnic University, in 2022. His research interests include speaker recognition and machine learning.

WEIWEI LIN received the B.Eng. degree from the Guangdong University of Technology, Guangzhou, China, in 2013, and the M.Sc. and Ph.D. degrees in electronic and information engineering from The Hong Kong Polytechnic University, Hong Kong, in 2016 and 2020, respectively. He is currently a Postdoctoral Researcher with The Hong Kong Polytechnic University. His research interests include speaker recognition, transfer learning, and deep learning.

MAN-WAI MAK (Senior Member, IEEE) received the Ph.D. degree 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, where he is currently a Professor. He has authored more than 200 technical articles in speaker recognition, machine learning, and bioinformatics. He also coauthored postgraduate textbooks titled *Biometric Authentication: A Machine Learning Approach* (Prentice-Hall, 2005) and *Machine Learning for Speaker Recognition* (Cambridge University Press, 2020). His research interests include speaker recognition, machine learning, and bioinformatics. He was a member of the IEEE Machine Learning for Signal Processing Technical Committee from 2005 to 2007. He also served as a Technical Committee Member for a number of international conferences, including ICASSP and Interspeech, and gave a tutorial on machine learning for speaker recognition in Interspeech'2016. He was 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.