Adaptation Algorithms for Neural Network-Based Speech Recognition: An Overview

Peter Bell, *Member, IEEE*, Joachim Fainberg, *Member, IEEE*, Ondrej Klejch, *Member, IEEE*, Jinyu Li, *Member, IEEE*, Steve Renals, *Fellow, IEEE*, and Pawel Swietojanski, *Member, IEEE*

***Abstract*—We present a structured overview of adaptation algorithms for neural network-based speech recognition, considering both hybrid hidden Markov model / neural network systems and end-to-end neural network systems, with a focus on speaker adaptation, domain adaptation, and accent adaptation. The overview characterizes adaptation algorithms as based on embeddings, model parameter adaptation, or data augmentation. We present a meta-analysis of the performance of speech recognition adaptation algorithms, based on relative error rate reductions as reported in the literature.**

***Index Terms*—Speech recognition, speaker adaptation, speaker embeddings, structured linear transforms, regularization, data augmentation, domain adaptation, accent adaptation, semi-supervised learning**

I. Introduction

THE performance of automatic speech recognition (ASR) systems has improved dramatically in recent years thanks to the availability of larger training datasets, the development of neural network based models, and the computational power to train such models on these datasets [1]–[4]. However, the performance of ASR systems can still degrade rapidly when their conditions of use (test conditions) differ from the training data. There are several causes for this, including speaker differences, variability in the acoustic environment, and the domain of use.

Adaptation algorithms attempt to alleviate the mismatch between the test data and an ASR system's training data. Adapting an ASR system is a challenging problem since it requires the modification of large and complex models, typically using only a small amount of target data and without explicit supervision. Speaker adaptation – adapting the system to a target speaker – is the most common form of adaptation, but there are other important adaptation targets such as the domain of use, and the spoken accent. Much of the work in the area has focused on speaker adaptation: it is the case that many approaches developed for speaker adaptation do not explicitly model speaker characteristics, and can be applied to other adaptation targets. Thus our core treatment of adaptation algorithms is in the context of speaker adaptation, with a later discussion of particular approaches for domain adaptation and accent adaptation. Specifically, domain adaptation in this paper refers to the task of adapting the models to the target domain that has either acoustic or content mismatch from the source domain in which the models were trained.

This overview focuses on the adaptation of neural network (NN) based speech recognition systems, although we briefly discuss earlier approaches to speaker adaptation of hidden Markov model (HMM) based systems. NN-based systems [1], [5], [6] have revolutionized the field of speech recognition, and there has been intense activity in the development of adaptation algorithms for such systems. Adaptation of NN-based speech recognition is an exciting research area for at least two reasons: from a practical point of view, it is important to be able to adapt state-of-the-art systems; and from a theoretical point of view the fact that NNs require fewer constraints on the input than a Gaussian-based system, along with the gradient-based discriminative training which is at the heart of most NN-based speech recognition systems, opens a range of possible adaptation algorithms.

A. NN/HMM Hybrid systems

Neural networks were first applied to speech recognition as so-called NN/HMM *hybrid systems*, in which the neural network is used to estimate (scaled) likelihoods that act as the HMM state observation probabilities [5] (Fig. 1a). During the 1990s both feed-forward networks [5] and recurrent neural networks (RNNs) [7] were used in such hybrid systems and close to state-of-the-art results were obtained [8]. These systems were largely context-independent, although context-dependent NN-based acoustic models were also explored [9].

The modeling power of neural network systems at that time was computationally limited, and they were not able to achieve the precise levels of modeling obtained using

Authors have made equal contributions and are listed alphabetically.

Peter Bell, Ondrej Klejch, and Steve Renals are with the Centre for Speech Technology Research, University of Edinburgh, Edinburgh EH8 9AB, UK (email: peter.bell@ed.ac.uk, o.klejch@ed.ac.uk, s.renals@ed.ac.uk).

Joachim Fainberg did this work at the Centre for Speech Technology Research, University of Edinburgh, Edinburgh EH8 9AB, UK. He is now with JPMorgan Chase, New York, NY 10017, USA (email: j.fainberg@ed.ac.uk).

Jinyu Li is with the Microsoft Corporation, Redmond, WA 98052, USA (e-mail: jinyli@microsoft.com)

Pawel Swietojanski did this work at the School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia. He is now with Apple, Cambridge, UK (email: p.swietojanski@unsw.edu.au).

This work was partially supported by the EPSRC Project EP/R012180/1 (SpeechWave), a PhD studentship funded by Bloomberg, and the EU H2020 project ELG (grant agreement 825627).

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Air Force Research Laboratory (AFRL) contract #FA8650-17-C-9117. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, AFRL or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

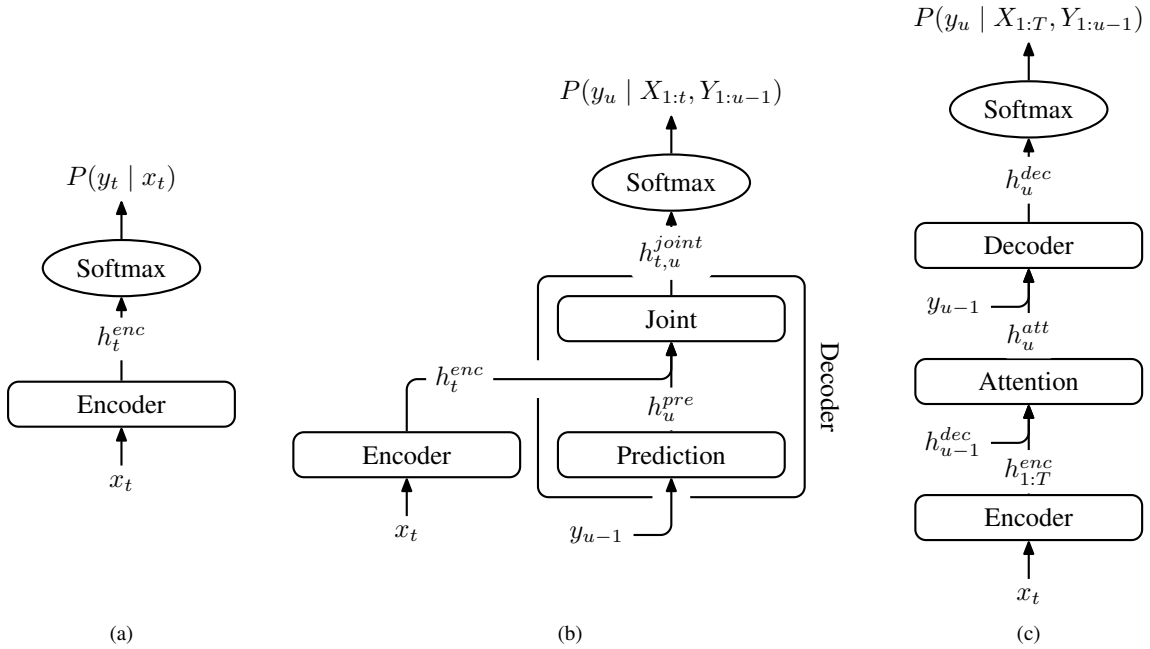


Fig. 1: NN architectures used for hybrid NN/HMM and end-to-end (CTC, RNN-T, AED) speech recognition systems: (a) Scheme of NN architecture used for NN/HMM hybrid systems and for connectionist temporal classification (CTC); (b) architecture for the RNN Transducer (RNN-T); (c) architecture for attention based encoder-decoder (AED) end-to-end systems. Input acoustic feature vectors are denoted by x_t; hidden layers are denoted by h_t, h_u and output labels by y_t, y_u depending on whether they are indexed by time t (in hybrid and CTC systems) or only by output label u (in parts of RNN-T and AED systems). In practice, the encoders use a wide temporal context as input, even the whole acoustic sequence in the case of most CTC and AED models.

context-dependent GMM-based HMM systems which became the dominant approach. However, increases in computational power enabled deeper neural network models to be learned along with context-dependent modeling using the same number of context-dependent HMM tied states (senones) as GMM-based systems [1], [2]. This led to the development of systems surpassing the accuracy of GMM-based systems. This increase in computational power also enabled more powerful neural network models to be employed, in particular time-delay neural networks (TDNNs) [10], [11], convolutional neural networks (CNNs) [12], [13], long short-term memory (LSTM) RNNs [14], [15], and bidirectional LSTMs [16], [17].

B. End-to-end systems

Since 2015, there has been a significant trend in the field moving from hybrid HMM/NN systems to end-to-end (E2E) NN modeling [4], [6], [18]–[24] for ASR. E2E systems are characterized by the use of a single model transforming the input acoustic feature stream to a target stream of output tokens, which might be constructed of characters, subwords, or even words. E2E models are optimized using a single objective function, rather than comprising multiple components (acoustic model, language model, lexicon) that are optimized individually. Currently, the most widely used E2E models are connectionist temporal classification (CTC) [25], [26], the RNN Transducer (RNN-T) model [21], [27], and the attention-based encoder-decoder (AED) model [6], [18].

CTC and the RNN-T both map an input speech feature sequence to an output label sequence, where the label sequence (typically characters) is considerably shorter than the input sequence. Both of these architectures use an additional blank output token to deal with the sequence length differences, with an objective function which sums over all possible alignments using the forward backward algorithm [28]. CTC is an earlier, and simpler, method which assumes frame independence and functions similarly to the acoustic model in hybrid systems without modeling the linguistic dependency across words; its architecture is similar to that of the neural network in the hybrid system (Fig. 1a).

An RNN-T (Fig. 1b) combines an additional prediction network with the acoustic encoder. The prediction network is an RNN modeling linguistic dependencies whose input is the previously output symbol. It is possible to initialize some of its layers from an external language model trained on additional text data. The acoustic encoder and the prediction network are combined using a feed-forward joint network followed by a softmax to predict the next output token given the speech input and the linguistic context.

Together, the RNN-T's prediction and joint networks may be regarded as a decoder, and we can view the RNN-T as a form of encoder-decoder system. The AED architecture (Fig. 1c) enriches the encoder-decoder model with an additional attention network which interfaces the acoustic encoder with the decoder. The attention network operates on the

entire sequence of encoder representations for an utterance, offering the decoder considerably more flexibility. A detailed comparison of popular E2E models in both streaming and non-streaming modes with large scale training data was conducted by Li et al. [29]. It is worth noting that with the recent success in machine translation, there is a trend of using the transformer model [30] to replace LSTM for both the AED [31]–[33] and RNN-T models [34]–[36].

C. Adaptation and transfer learning in related fields

Adaptation and transfer learning have become important and intensively researched topics in other areas related to machine learning, most notably computer vision and natural language processing (NLP). In both these cases the motivation is to train powerful base models using large amounts of training data, then to adapt these to specific tasks or domains, for which considerably less training data is available.

In computer vision, the base model is typically a large convolutional network trained to perform image classification or object recognition using the ImageNet database [37], [38]. The ImageNet model is then adapted to a lower resource task, such as computer-aided detection in medical imaging [39]. Kornblith et al. [40] have investigated empirically how well ImageNet models transfer to different tasks and datasets.

Transfer learning in NLP differs from computer vision, and from the speech recognition approaches discussed in this paper, in that the base model is trained in an unsupervised fashion to perform language modeling or a related task, typically using web-crawled text data. Base models used for NLP include the bidirectional LSTM [41] and Transformers which make use of self-attention [42], [43]. These models are then trained on specific NLP tasks, with supervised training data, which is specified in a common format (*e.g.* text-to-text transfer [43]), often trained in a multi-task setting. Earlier adaptation approaches in NLP focused on feature adaptation (e.g. [44]), but more recently better results have been obtained using model-based adaptation, for instance "adapter layers" [43], [45], in which trainable transform layers are inserted into the pretrained base model.

More broadly there has been extensive work on domain adaptation and transfer learning in machine learning, reviewed by Kouw and Loog [46]. This includes work on few-shot learning [47]–[49] and normalizing flows [50], [51]. Normalizing flows which provide a probabilistic framework for feature transformations, were first developed for speech recognition as Gaussianization [52], and more recently have been applied to speech synthesis [53] and voice transformation [54].

D. Structure of this review

We begin by considering the issues of identifying suitable data and target labels to adapt to in Sec. II. After discussing speaker adaptation of non NN-based HMM systems in Section III, we present a general framework for adaptation of NN-based speech recognition systems (both hybrid and E2E) in Sec. IV, where we organize adaptation algorithms into three general categories: embedding-based approaches (discussed in Sec. V), model-based approaches (discussed in Secs. VI–VIII), and data augmentation approaches (discussed in Sec. IX).

As mentioned above, most of our treatment of adaptation algorithms is in the context of speaker adaptation. In Secs. X and XI we discuss specific approaches to accent adaptation and domain adaptation respectively.

Our primary focus is on the adaptation of acoustic models and end-to-end models. In Sec. XII we provide a summary of work in language model (LM) adaptation, mentioning both n-gram and neural network language models, and the use of LM adaptation in E2E systems.

Finally we provide a meta analysis of experimental studies using the main adaptation algorithms that we have discussed (Sec. XIII). The meta-analysis is based on experiments reported in 47 papers, carried out using 38 datasets, and is primarily based on the relative error rate reduction arising from adaptation approaches. In this section we analyze the performance of the main adaptation algorithms across a variety of adaptation target types (for instance speaker, domain, and accent), in supervised and unsupervised settings, in six different languages, and using six different NN model types in both hybrid and E2E settings. Raw data, aggregated results and the corresponding scripts are available at https://github.com/pswietojanski/ojsp_adaptation_review_2020.

II. Identifying adaptation targets

Adaptation aims to reduce the mismatch between training and test conditions. For an adaptation algorithm to be effective, the distribution of the adaptation data should be close to that encountered in test conditions. For this reason it is important to ensure that the target labels adapted to form coherent classes. For the task of acoustic adaptation this requirement is typically satisfied by forming the adaptation data from one or more speech segments from known testing conditions (*i.e.* the same speaker, accent, domain, or acoustic environment). While for some tasks labels ascribed to speech segments may exist, allowing segments to be grouped into larger adaptation clusters, it is unrealistic to assume the availability of such metadata in general. However, depending on the application and the operating regime of the ASR system, it may be possible to derive reasonable proxies.

Utterance-level adaptation derives adaptation statistics using a single speech segment [55]. This waives the requirement to carry information about speaker identity between utterances, which may simplify deployment of recognition systems – in terms of both engineering and privacy – as one does not need to estimate and store offline speaker-specific information. On the other hand, owing to the small amounts of data available for adaptation the gains are usually lower than one could obtain with speaker-level clusters. While many approaches use utterances to directly extract corresponding embeddings to use as an auxiliary input for the acoustic model [56]–[59], one can also build a fixed inventory of speakers, domains, or topic codes [60] or embeddings [61], [62] when learning the acoustic model or acoustic encoder, and then use the test utterance to select a combination of these at test stage. The latter approach alleviates the necessity of estimating an

accurate representation from small amounts of data. It may be possible to relax the utterance-level constraint by iteratively re-estimating adaptation statistics using a number of preceding segment(s) [57]. Extra care usually needs to be taken to handle silence and speech uttered by different speakers, as failing to do so may deteriorate the overall ASR performance [62]–[64].

Speaker-level adaptation aggregates statistics across two or more segments uttered by the same talker, requiring a way to group adaptation utterances produced by different talkers. In some cases – for example lecture recordings and telephony – speaker information may be available. In other cases potentially inaccurate metadata is available, for instance in the transcription of television or online broadcasts. In many cases (for instance, anonymous voice search) speaker metadata is not available. The generic approach to this problem relies on a speaker diarization system [65], that can identify speakers and accordingly assign their identities to the corresponding segments in the recordings. This is often used in the offline transcription of meetings or broadcast media. Alternative clustering approaches can be used to define the adaptation classes [66], [67].

Domain-level adaptation broadens the speaker-level cluster by including speech produced by multiple talkers characterized by some common characteristic such as accent, age, medical condition, topic, *etc.* . This typically results in more adaptation material and an easier annotation process (cluster labels need to be assigned at batch rather than segment level). As such, domain adaptation can usually leverage adaptation transforms with greater capacity, and thus offer better adaptation gains.

Depending on whether adaptation transforms are estimated on held out data, or adaptation is iteratively derived from test segments, we will refer to these as *enrolment* or *online* modes, respectively. In enrolment mode, the adaptation data would be ideally labeled with a gold-standard transcription, to enable supervised learning algorithms to be used for adaptation. However, supervised data is rarely available: small amounts may be available for some domain adaptation tasks (for example, adapting a system trained on typical speech to disordered speech [68]). In the usual case, where supervised adaptation data is not available, supervised training algorithms can still be used with "pseudo-labels" that are automatically obtained from a seed model, a process which is a type of *semi-supervised training* [69]. Alternatively, unsupervised training can be applied to learn embeddings for the different adaptation classes, such as i-vectors [56] or bottleneck features extracted from an auto-encoder neural network [70]. A *two-pass* system is a special case for which the statistics are estimated from test data using the first pass decoding with a speaker-independent model in order to obtain adaptation labels, followed by a second pass with the speaker-adapted model.

For semi-supervised approaches, it is possible to further filter out regions with low-confidence to avoid the reinforcement of potential errors [71]–[73]. There is some evidence in the literature that, for some limited-in-capacity transforms estimated in a semi-supervised manner, the first pass transcript quality has a small impact on the adapted accuracy as long as these are obtained with the corresponding speaker-independent model [74], [75]. In *lattice supervision* multiple possible transcriptions are used in a semi-supervised setting by generating a lattice, rather than the one-best transcription [76]–[79].

III. Adaptation algorithms for HMM-based ASR

Speaker adaptation of speech recognition systems has been investigated since the 1960s [80], [81]. In the mid-1990s, the influential maximum likelihood linear regression (MLLR) [82] and maximum a posteriori (MAP) [83] approaches to speaker adaptation for HMM/GMM systems were introduced. These methods, described below, stimulated the field leading an intense activity in algorithms for the adaptation of HMM/GMM systems, reviewed by Woodland [84] and Shinoda [85], as well as in section 5 of Gales and Young's broader review of HMM-based speech recognition [86]. As we later discuss, some of the algorithms developed for HMM-based systems, in particular feature transformation approaches have been successfully applied to NN-based systems. In this section we review MAP, MLLR, and related approaches to the adaptation of HMM/GMM systems, along with earlier approaches to speaker adaptation.

A. Speaker normalisation

Many of these early approaches were designed to normalize speaker-specific characteristics, such as vocal tract length, building on linguistic findings relating to speaker normalization in speech perception [87], often casting the problem as one of spectral normalization. This work included formant-based frequency warping approaches [80], [81], [88] and the estimation of linear projections to normalize the spectral representation to a speaker-independent form [89], [90].

Vocal tract length normalization (VTLN) was introduced by Wakita [91] (and again by Andreou [92]) as a form of frequency warping with the aim to compensate for vocal tract length differences across speakers. VTLN was extensively investigated for speech recognition in the 1990s and 2000s [93]–[96], and is discussed further in Sec. V.

B. Model based approaches

In model based adaptation, the speech recognition model is used to drive the adaptation. In work prefiguring subspace models, Furui [97] showed how speaker specific models could be estimated from small amounts of target data in a dynamic time warping setting, learning linear transforms between pre-existing speaker-dependent phonetic templates, and templates for a target speaker. Similar techniques were developed in the 1980s by adapting the vector quantization (VQ) used in discrete HMM systems. Shikano, Nakamura, and Abe [98] showed that mappings between speaker dependent codebooks could be learned to model a target speaker (a technique widely used for voice conversion [99]); Feng et al. [100] developed a VQ-based approach in which speaker-specific mappings were learned between codewords in a speaker-independent codebook, in order to maximize the likelihood of the discrete HMM system. Rigoll [101] introduced a related approach in which the speaker-specific transform took the form of a Markov model. A continuous version of this approach, referred

to as probabilistic spectrum fitting, which aimed to adjust the parameters of a Gaussian phonetic model was introduced by Hunt [102] and further developed by Cox and Bridle [103].

These probabilistic spectral modeling approaches can be viewed as precursors to maximum likelihood linear regression (MLLR) introduced by Leggetter and Woodland [82] and generalized by Gales [104]. MLLR applies to continuous probability density HMM systems, composed of Gaussian probability density functions. In MLLR, linear transforms are estimated to adapt the mean vectors and – in [104] – covariance matrices of the Gaussian components. If μ and Σ are the mean vector and covariance matrix of a particular Gaussian, then MLLR adapts the parameters as follows:

$$\hat{\mu}_s = A_s\,\mu - b_s \tag{1}$$

$$\hat{\Sigma}_s = H_s\,\Sigma\,H_s^\intercal\,. \tag{2}$$

The speaker-specific parameters b_s, A_s and H_s are estimated using maximum likelihood. MLLR is a compact adaptation technique since the transforms are shared across Gaussians: for instance all Gaussians corresponding to the same monophone might share mean and covariance transforms. Very often, especially when target data is sparse, a greater degree of sharing is employed – for instance two shared adaptation transforms, one for Gaussians in speech models and one for Gaussians in non-speech models.

Constrained MLLR [104], [105], is an important variant of MLLR, in which the same transform is used for both the mean and covariance:

$$\hat{\mu}_s = A_s\mu - b_s \tag{3}$$

$$\hat{\Sigma}_s = A_s\,\Sigma\,A_s^\intercal \tag{4}$$

In this case, the log likelihood for a single Gaussian is given by

$$L^{\text{cMLLR}}(x;\hat{\mu}_s,\hat{\Sigma}_s) = \log\mathcal{N}(x;A_s\mu - b_s, A_s\,\Sigma\,A_s^\intercal) \tag{5}$$

$$= \log\mathcal{N}(A_s^{-1}x + A_s^{-1}b_s;\mu,\Sigma) - \log|A_s| \tag{6}$$

It can be seen that this transform of the model parameters is equivalent to applying an affine transform to the data – hence constrained MLLR is often referred to as feature-space MLLR (fMLLR), although it is not strictly feature-space adaptation unless a single transform is shared across all Gaussians in the system, in which case the Jacobian term $-\log|A_s|$ can be ignored. MLLR and its variants have been used extensively in the adaptation of Gaussian mixture model (GMM)-based HMM speech recognition systems [84], [86].

C. Bayesian methods

The above model-based adaptation approaches have aimed to estimate transforms between a speaker independent model and a model adapted to a target speaker. An alternative Bayesian approach attempts to perform the adaptation by using the speaker independent model to inform the prior of a speaker-adapted model. If the set of parameters of a speech recognition model are denoted by θ, then maximum likelihood estimation sets θ to maximize the likelihood $p(X\mid\theta)$. In MAP training, the estimation procedure maximizes the posterior of the parameters given the data $X = \{x_1 \ldots x_T\}$:

$$P(\theta\mid X) \propto p(X\mid\theta)\,p(\theta)^r\,, \tag{7}$$

where $p(\theta)$ is the prior distribution of the parameters, which can be based on speaker independent models, and r is an empirically determined weighting factor. Gauvain and Lee [83] presented an approach using MAP estimation as an adaptation approach for HMM/GMM systems. A convenient choice of function for $p(\theta)$ is the conjugate to the likelihood – the function which ensures the posterior has the same form as the prior. For a GMM, if it is assumed that the mixture weights c_i and the Gaussian parameters (μ_i, Σ_i) are independent, then the conjugate prior may take the form of a mixture model $p_D(c_i)\prod_i p_W(\mu_i,\Sigma_i)$, where $p_D()$ is a Dirichlet distribution (conjugate to the multinomial) and $p_W()$ is the normal-Wishart density (conjugate to the Gaussian). This results in the following intuitively understandable parameter estimate for the adapted mean of a Gaussian $\hat{\mu}\in\mathbb{R}^d$:

$$\hat{\mu} = \frac{\tau\mu_0 + \sum_t \gamma(t)x_t}{\tau + \sum_t \gamma(t)}\,, \tag{8}$$

where $\mu_0\in\mathbb{R}^d$ is the unadapted (speaker-independent) mean, $x_t\in\mathbb{R}^d$ is the adaptation vector at time t, $\gamma(t)\in\mathbb{R}$ is the component occupation probability (responsibility) for the Gaussian component at time t (estimated by the forward-backward algorithm), and τ is a positive scalar-valued parameter of the normal-Wishart density, which is typically set to a constant empirically (although Gauvain and Lee [83] also discuss an empirical Bayes estimation approach for this parameter). The re-estimated means of the Gaussian components take the form of a weighted interpolation between the speaker independent mean and data from the target speaker. When there is no target speaker data for a Gaussian component, the parameters remain speaker-independent; as the amount of target speaker data increases, so the Gaussian parameters approach the target speaker maximum likelihood estimate.

D. Speaker adaptive training

In the model-based approaches discussed above (MLLR and MAP), we have implicitly assumed that adaptation takes place at test time: speaker independent models are trained using recordings of multiple speakers in the usual way, with only the test speakers used for adaptation. In contrast to this, it is possible to employ a model-based adaptive training approach. In speaker adaptive training [106], a transform is estimated for each speaker in the training set, as well as for each speaker in the test set. During training, speaker-specific transforms and a speaker-independent canonical model are updated in an iterative fashion.

Speaker space approaches represent a speaker-adapted model as a weighted sum of a set of individual models which may represent individual speakers or, more commonly, speaker clusters. In cluster-adaptive training (CAT) [66], the mean for a Gaussian component for a specific speaker s is given by:

$$\hat{\mu}_s = \sum_{c=1}^{C} w_c\mu_c \tag{9}$$

where $\mu_c \in \mathbb{R}^d$ is the mean of the particular Gaussian component for speaker cluster c, and $w_c \in \mathbb{R}$ is the cluster weight. This expresses the speaker-adapted mean vector as a point in a speaker space. Given a set of canonical speaker cluster models, CAT is efficient in terms of parameters, since only the set of cluster weights need to be estimated for a new speaker. Eigenvoices [107] are alternative way of constructing speaker spaces, with a speaker model again represented as a weighted sum of canonical models. In the Eigenvoices technique, principal component analysis of "supervectors" (concatenated mean vectors from the set of speaker-specific models) is used to create a basis of the speaker space.

A number of variants of cluster-adaptive training have been presented, including representing a speaker by combining MLLR transforms from the canonical models [66], and using sequence discriminative objective functions such as minimum phone error (MPE) [108]. Techniques closely related to CAT have been used for the adaptation of neural network based systems (Sec. VI).

In contrast to model-based methods, in feature-based adaptation it is usual to adapt or normalize the acoustic features for each speaker in both the training and test sets– this may be viewed as a form of speaker adaptive training. For example, in the case of cepstral mean and variance normalization (CMVN), statistics are computed for each speaker and the features normalized accordingly, during both training and test. Likewise, VTLN is also carried out for all speakers, transforming the acoustic features to a canonical form, with the variation from changes in vocal tract length being normalized away.

IV. Adaptation algorithms for NN-based ASR

The literature describing methods for adaptation of NNs has tended to inherit terminology from the algorithms used to adapt HMM-GMM systems, for which there is an important distinction between *feature space* and *model space* formulations of MLLR-type approaches [104], as discussed in the previous section. In a 2017 review of NN adaptation, Sim et al. [109] divide adaptation algorithms into *feature normalisation*, *feature augmentation* and *structured parameterization*. (They also use a further category termed *constrained adaptation*, discussed further below.)

The task of an ASR model is to map a sequence of acoustic feature vectors, $X = (x_1, \dots x_t, \dots, x_T)$, $x_t \in \mathbb{R}^d$ to a sequence of words W. Although – as we discuss below – most techniques described in this paper apply equally to end-to-end models and hybrid HMM-NN models, we generally treat the model to be adapted as an acoustic model. That is, we ignore aspects of adaptation that affect only $P(W)$, independently of the acoustics X (LM adaptation is discussed in Sec. XII). Further, with only a small loss of generality, in what follows we will assume that the model operates in a framewise manner, thus we can define the model as:

$$y_t = f(x_t; \theta) \tag{10}$$

where $f(x; \theta)$ is the NN model with parameters θ and y_t is the output label at frame t. In a hybrid HMM-NN system, for example, y_t is taken to be a vector of posterior probabilities over a senone set. In a CTC model, y_t would be a vector of posterior probabilities over the output symbol set, plus blank symbol. Note that NN models often operate on a wider windowed set of input features, $x_t(w) = [x_{t-c}, x_{t-c+1}, \dots, x_{t+c-1}, x_{t+c}]$ with the total window size $w = 2c + 1$. For reasons of notational clarity, we generally ignore the distinction between x_t and $x_t(w)$, unless it is specifically relevant to a particular topic.

In this framework, we can define *feature normalisation* approaches as acting to transform the features in a speaker-dependent manner, on which the speaker-independent model operates. For each speaker s, a transformation function $g : \mathbb{R}^d \to \mathbb{R}^{d'}$ computes:

$$x'_t = g(x_t; \phi_s) \tag{11}$$

where ϕ_s is a set of speaker-dependent parameters. Commonly the dimension of the normalised features is the identical to the original (*i.e.* $d = d'$) but this is not required. This family is closely related to feature space methods used in GMM systems described above in Sec. III, including fMLLR (when only a single affine transform is used), VTLN, and CMVN.

Structured parameterization approaches, in contrast, introduce a speaker-dependent transformation of the acoustic model parameters:

$$\theta'_s = h(\theta; \varphi_s) \tag{12}$$

In this case, the function h would typically be structured so as to ensure that the number of speaker-dependent parameters φ_s is sufficiently smaller than the number of parameters of the original model. Such methods are closely related to model-based adaptation of GMMs such as MLLR.

Finally, *feature-augmentation* approaches extend the feature vector x_t with a speaker-dependent embedding λ_s, which we can write as

$$x'_t = \begin{pmatrix} x_t \\ \lambda_s \end{pmatrix} \tag{13}$$

Close variants of this approach use the embedding to augment the input to higher layers of the network. Note that the incorporation of an embedding requires the addition of further parameters to the acoustic model controlling the manner in which the embedding acts to adapt the model, which can be written $f(x_t; \theta, \theta^E)$. The embedding parameters θ^E are themselves speaker-independent.

We suggest that the distinctions described above may not always be helpful when considering NN adaptation specifically, because all three approaches can be seen to be closely related or even special cases of each other. As we saw in Section III this is not the case in HMM-GMM systems, where the distinction between feature-space and model adaptation is important (as noted by Gales [104]) because in the former case, different feature space transformations can be carried out per senone class if the appropriate scaling by a Jacobian is performed; whilst in the latter case, it is necessary for the adapted probability density functions to be re-normalized.

As an example of the equivalence of the close relationship between the three approaches to NN adaptation, the normalisation function g can generally be formulated as shallow NN,

possibly without a non-linearity. If there is a set of "identity transform" parameters ϕ^I such that

$$g(x_t;\phi^I) = x_t,\ \forall x_t \tag{14}$$

then we have

$$y_t = f(x_t;\theta) = f(g(x_t;\phi^I);\theta) = f'(x_t;\theta,\phi^I) \tag{15}$$

where f' is a new network comprising of a copy of the original network f with the layers of g prepended. Applying feature normalization (11) leads to:

$$y_t = f(x'_t;\theta) = f(g(x_t;\phi_s);\theta) = f'(x_t;\theta,\phi_s) \tag{16}$$

which we can write as a structured parameter transformation of f', as defined in (12):

$$\theta'_s = \{\theta,\phi_s\} = h(\{\theta,\phi^I\};\varphi_s) \tag{17}$$

where the transformation $h(\cdot\,;\varphi_s)$ is simply set to replace the parameters pertaining to g with the original normalisation parameters, $\phi_s = \varphi_s$, leaving the other parameters unchanged.

Similarly, feature augmentation approaches may be readily seen to be a further special case of structured adaptation. In the simple case of input feature augmentation (13), we see that the output of the first layer, prior to the non-linearity, can be written as

$$z = Wx' + b = W\begin{pmatrix} x \\ \lambda_s \end{pmatrix} + b \tag{18}$$

where W and b are the weight and bias of the first layer respectively. By introducing a decomposition of W, $W = \begin{pmatrix} U & V \end{pmatrix}$ we write this as

$$z = \begin{pmatrix} U & V \end{pmatrix}\begin{pmatrix} x \\ \lambda_s \end{pmatrix} + b = Ux + b + V\lambda_s \tag{19}$$

with $U \in \theta$ and $V \in \theta^E$ being weight matrices pertaining to the input features and speaker embedding, respectively.

This can be expressed as a structured transformation of the bias:

$$\theta'_s = \{U',b'\} = h(\{U,b\};\varphi_s) = \{U, b + V\lambda_s\} \tag{20}$$

with $\varphi_s = V\lambda_s$. Similar arguments apply to embeddings used in other network layers.

Certain types of feature normalisation approaches can be expressed as feature augmentation. For example, cepstral mean normalisation given by

$$x'_t = g(x_t;\phi_s) = x_t - \mu_s \tag{21}$$

can be expressed as

$$z = W(x-\mu_s) + b = \begin{pmatrix} W & W \end{pmatrix}\begin{pmatrix} x \\ -\mu_s \end{pmatrix} + b \tag{22}$$

with augmented features $\lambda_s = -\mu_s$.

As we have seen, approaches to NN adaptation under the traditional categorization of feature augmentation, structured parameterization and feature normalization can usually be seen as special cases of one another. Therefore, in the remainder of this paper, we adopt an alternative categorization:

- **Embedding-based** approaches in which any speaker-dependent parameters are estimated independently of the model, with the model $f(x_t;\theta)$ itself being unchanged between speakers, other than the possible need for additional embedding parameters θ^E;
- **Model-based** approaches in which the model parameters θ are directly adapted to data from the target speaker according to the primary objective function;
- **Data augmentation** approaches which attempt to synthetically generate additional training data with a close match to the target speaker, by transforming the existing training data.

This distinction is, we believe, particularly important in *speaker* adaptation of NNs because in ASR it has become standard to perform adaptation in a semi-supervised manner, with no transcribed adaptation data for the target speaker. In this setting, as we will discuss, standard objective functions such as cross-entropy, which may be very effective in supervised training or adaptation, are particularly susceptible to transcription errors in semi-supervised settings.

We describe the model-independent approaches as embedding-based because any set of speaker-dependent parameters can be viewed as an embedding. Embedding-based approaches are discussed in Sec. V. Well-known examples of speaker embeddings include i-vectors [56], [110], and x-vectors [111], but can also include parameter sets more classically viewed as normalizing transforms such as CMVN statistics and global fMLLR transforms (see Sec. III above). However, for the reasons mentioned above, we exclude from this category methods where the embedding is simply a subset of the primary model parameters and estimated according to the model's objective function. Note that methods using a one-hot encoding for each speaker are also excluded, since it would be impossible to use these with a speaker-independent model, without each test speaker having been present in training data; such methods might however be useful for closely related tasks such as domain adaptation, discussed in Sec. XI.

The primary benefit of speaker adaptive approaches over simply using speaker-dependent models is the prevention of over-fitting to the adaptation data (and its possibly errorful transcript). A large number of model-based adaptation techniques have been proposed to achieve this; in this paper, we sub-divide them into:

- **Structured transforms**: Methods in which a subset of the parameters are adapted, with many instances structuring the model so as to permit a reduced number of speaker-dependent parameters, as in the Learning Hidden Unit Contributions (LHUC) scheme [75], [112]. The can be viewed as an analogy to MLLR transforms for GMMs. They are discussed in Sec. VI.
- **Regularization**: Methods with explicit regularization of the objective function to prevent over-fitting to the adaptation data, examples including the use of L2 loss or KL divergence terms to penalize the divergence from the speaker-independent parameters [113], [114]. Such methods can be viewed as related to the MAP approach for GMM adaptation. They are discussed in Sec. VII.
- **Variant objective functions**: Methods which adopt vari-

ants of the primary objective function to overcome the problems of noise in the target labels, with examples including the use of lattice supervision [79] or multi-task learning [115]. They are discussed in Sec. VIII.

The second two categories above are collectively termed *constrained adaptation* in the review by Sim et al. [109]. Within this, multi-task learning is labeled by Sim et al. as attribute aware training; however, we do not believe that all multi-task learning approaches to adaptation can be labeled in this way.

Data augmentation methods have proved very successful in adaptation to other sources of variability, particularly those – such as background noise conditions – where the required model transformations are hard to explicitly estimate, but where it is easy to generate realistic data. In the case of speaker adaptation, it is significantly harder to generate sufficiently good-quality synthetic data for a target speaker, given only limited data from the speaker in question. However, there is a growing body of work in this area using, for example, techniques from the field of speech synthesis [116]. Approaches in this area are discussed in Sec. IX.

Most works suitable for adapting hybrid acoustic models can be leveraged to adapt acoustic encoders in E2E models. Both Kullback-Leibler divergence (KLD) regularization (Sec. VII) and multi-task learning (MTL) methods (Sec. VIII) have been used for speaker adaptation for CTC and AED models [117], [118].

Sim et al. [119] updated the acoustic encoder of RNN-T models using speaker-specific adaptation data. Furthermore, by generating text-to-speech (TTS) audio from the target speaker, more data can be used to adapt the acoustic encoder. Such data augmentation adaptation (discussed in Sec. IX) was shown to be an effective way for the speaker adaptation of E2E models [120] even with very limited raw data from the target speaker. Embeddings have also been used to train a speaker-aware AED model [62], [121], [122].

Because AED and RNN-T also have components corresponding to the language model, there are also techniques specific to adapting the language modeling aspect of E2E models, for instance using a text embedding instead of an acoustic embedding to bias an E2E model in order to produce outputs relevant to the particular recognition context [123]–[125]. If the new domain differs from the source domain mainly in content instead of acoustics, domain adaptation on E2E models can be performed by either interpolating the E2E model with an external language model or updating language model related components inside the E2E model with the text-to-speech audio generated from the text in the new domain [126], [127], discussed in Sec. XII.

V. Speaker embeddings

Speaker embeddings map speakers to a continuous space. In this section we consider embeddings that may be extracted in a manner independent of the model, and which are also typically unsupervised with respect to the transcript. They can therefore also be useful in a standalone manner for other tasks such as speaker recognition. When used with an acoustic model, the model learns how to incorporate the embedding information by, in effect, speaker-aware training. Speaker embeddings may encode speaker-level variations that are otherwise difficult for the AM to learn from short-term features [64], and may be included as auxiliary features to the network. Specifically, let $x \in \mathbb{R}^d$ denote the acoustic features, and $\lambda_s \in \mathbb{R}^k$ a k-dimensional speaker embedding. The speaker embeddings may be concatenated with the acoustic input features, as previously seen in (13):

$$x'_t = \begin{pmatrix} x_t \\ \lambda_s \end{pmatrix} \tag{23}$$

Alternatively they may be concatenated with the activations of a hidden layer. In either case the result is bias adaptation of the next hidden layer as discussed in Sec. VI. As noted by Delcroix et al. [128] the auxiliary features may equivalently be added directly to the features using a learned projection matrix P, with the benefit that the downstream architecture can remain unchanged:

$$x'_t = x_t + P\lambda_s \tag{24}$$

There are many other ways to incorporate embeddings into the AM: for example, they may be used to scale neuron activations as in LHUC [75]. More generally we may consider embeddings applied to either biases or activations through context-adaptive [129] or control networks [130]. It is possible to limit connectivity from the auxiliary features to the rest of the network in order to improve robustness at test time or to better incorporate static features [131]–[133]. We will further consider transformations of the features as speaker embeddings, such as with fMLLR [104], [105], and they may also be used as label targets [134].

A. Feature transformations

We may consider speaker-level transformations of the acoustic features as speaker embeddings. These include methods traditionally viewed as normalisation, such as CMVN and fMLLR, which produce affine transformations of the features:

$$x'_s = A_s x + b_s \tag{25}$$

CMVN derives its name from the application to cepstral features, but corresponds to the standardization of the features to zero mean and unit variance (z-score):

$$x'_s = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{26}$$

where $\mu \in \mathbb{R}^d$ is the cepstral mean, $\sigma^2 \in \mathbb{R}^d$ is the cepstral variance, and ϵ is a small constant for numerical stability.

fMLLR [104] belongs to a family of speaker adaptation methods originally developed for HMM-GMM models, as discussed in Section III. The technique has, however, later been used with success to transform features for hybrid models as well [135], [136]. While the fMLLR transforms were traditionally estimated using maximum likelihood and HMM-GMM models, the transforms may also be estimated using a neural network trained to estimate fMLLR features [137]

(in Sec. VI we will further discuss structurally similar transforms estimated using the main objective function). Instead of transforming the input features, some work has explored fMLLR features as an additional, auxiliary, feature stream to the standard features in order to improve robustness to mismatched transforms [133], or to obtain speaker-adapted features derived from GMM log-likelihoods [138], otherwise known as GMM-derived features.

Another technique with a long history is VTLN [91], [92], [94], [139], which was briefly introduced in Section III. To control for varying vocal tract lengths between speakers, VTLN typically uses a piecewise linear warping function to adjust the filterbank in feature extraction. This requires only a single warping factor parameter that can be estimated using any AM with a line search. Alternatively, linear-VTLN (*e.g.* [95]) obtains a corresponding affine transform similar to fMLLR, but chooses from a fixed set of transforms at test time . A related idea is that of the exponential transform [140], which forgoes any notion of vocal tract length, but akin to VTLN is controlled by a single parameter. More recently, adaptation of learnable filterbanks, operating as the first layer in a deep network, has resulted in updates which compensate for vocal tract length differences between speakers [141].

B. i-vectors

Many types of embeddings stem from research in speaker verification and speaker recognition. One such approach is identity vectors, or i-vectors [56], [110], [142], which are estimated using means from GMMs trained on the acoustic features. Specifically, the extraction of a speaker i-vector, $\lambda_s \in \mathbb{R}^k$, assumes a linear relationship between the global means from a background GMM (or universal background model, UBM), $m_g \in \mathbb{R}^m$, and the speaker-specific means, $m_s \in \mathbb{R}^m$

$$m_s = m_g + T\lambda_s \tag{27}$$

where $T \in \mathbb{R}^{m \times k}$ is a matrix that is shared across all speakers which is sometimes called the total variability matrix from its relation to joint factor analysis [143]. An i-vector thus corresponds to coordinates in the column space of T. T is estimated iteratively using the EM algorithm. It is possible to replace the GMM means with posteriors or alignments from the AM [131], [144], [145] although this is no longer independent of the AM and requires transcriptions. The i-vectors are usually concatenated with the acoustic features as discussed above, but have also been used in more elaborate architectures to produce a feature mapping of the input features themselves [146], [147].

C. Neural network embeddings

A number of works proposed to extract low-dimensional embeddings from bottleneck layers in neural network models trained to distinguish between speakers [64], [132] or across multiple layers followed by dimensionality reduction in a separate AM (*e.g.* CNN embeddings [148]). One such approach, using Bottleneck Speaker Vector (BSV) embeddings [64], trains a feed-forward network to predict speaker labels (and silence) from spliced MFCCs (Fig. 2a). Tan et al. [132] proposed to add a second objective to predict monophones in a multi-task setup. The bottleneck layer dimension is typically set to values commonly used for i-vectors. In fact, Huang and Sim [64] note that if the speaker label targets are replaced with speaker deviations from a UBM, then the bottleneck-features may be considered frame-level i-vectors. The extracted features are averaged across all speech frames of a given speaker to produce speaker-level i-vectors.

There are several more recent approaches that we may collectively refer to as $\star$-vectors. Like bottleneck features, these approaches typically extract embeddings from neural networks trained to discriminate between speakers, but not necessarily using a low-dimensional layer. For instance, deep vectors, or d-vectors [149], [150], extract embeddings from feed-forward or LSTM networks trained on filterbank features to predict speaker labels. The activations from the last hidden layer are averaged over time. X-vectors [111], [130] use TDNNs with a pooling layer that collects statistics over time and the embeddings are extracted following a subsequent affine layer. A related approach called r-vectors [151] uses the architecture of x-vectors, but predicts room impulse response (RIR) labels rather than speaker labels. In contrast to the above approaches, label embeddings, or l-vectors [134], are designed to be used as soft output targets for the training of an AM. Each label embedding represents the output distribution for a particular senone target. In this way they are, in effect, uncoupled from the individual data points and can be used for domain adaptation without a requirement of parallel data. We will discuss this idea further in Sec. XI. For completeness we also mention h-vectors [152] which use a hierarchical attention mechanism to produce utterance-level embeddings, but has only been applied to speaker recognition tasks.

X-vector embeddings are not widely used for adaptating ASR algorithms in practice – especially in comparison to commonly used i-vectors – as experiments have not shown consistent improvements in recognition accuracy. One reason for this is related to the speaker identification training objective for the x-vector network which implicitly factors out channel information, which might be beneficial for adaptation. The optimal objective for speaker embeddings used in ASR differs from the objective used in speaker verification.

Summary networks [59], [128] produce sequence level summaries of the input features and are closely related to $\star$-vectors (*cf.* Fig. 2b). Auxiliary features are produced by a neural network that takes as input the same features as the AM, and produces embeddings by taking the time-average of the output. By incorporating the averaging into the graph, the network can be trained jointly with the AM in an end-to-end fashion [128]. A related approach is to produce LHUC feature vectors (Sec. VI) from an independent network with embedded averaging [153].

D. Embeddings for E2E systems

The embedding method is also helpful to the adaptation of E2E systems. Fan et al. [121] and Sari et al. [62] generated a soft embedding vector by combining a set of i-vectors

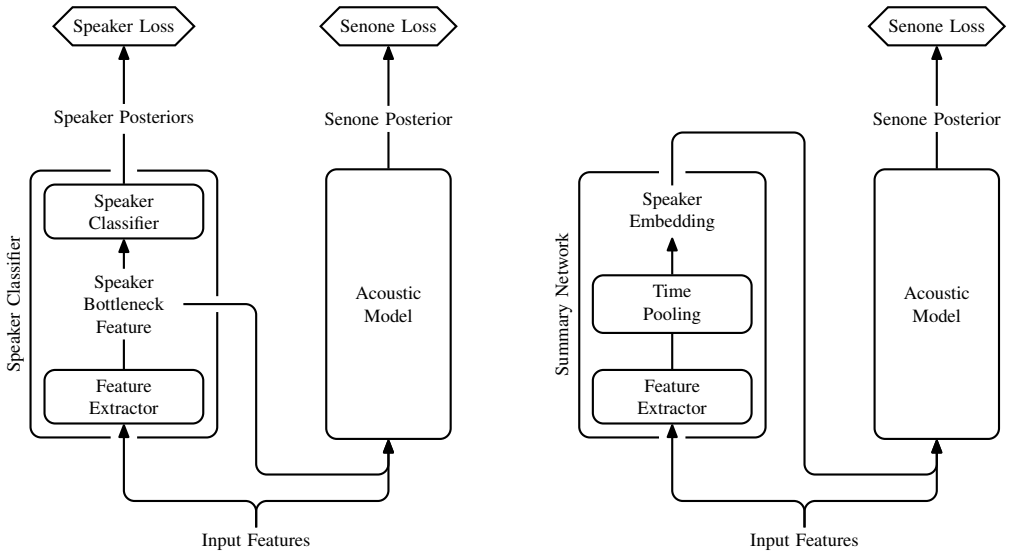


Fig. 2: (a) Bottleneck feature extraction that uses a pretrained speaker classifier. (b) Summary network extracting speaker embeddings which is trained jointly with the acoustic model.

from multiple speakers with the combination weight calculated from the attention mechanism. The soft embedding vector is appended to the acoustic encoder output of the E2E model, helping the model to normalize speaker variations. While the soft embedding vectors in [62], [121] are different at each frame, the speaker i-vectors are concatenated with the speech utterance as the input of every encoder layer in [122] to form a persistent memory through the depth of encoder, hence learning utterance-level speaker knowledge.

In addition to acoustic embedding, E2E models can also leverage text embedding to improve their modeling accuracy. For example, E2E models can be optimized to produce outputs relevant to the particular recognition context, for instance user contacts or device location. One solution is to add a context bias encoder in addition to the original audio encoder into E2E models [123]–[125]. This bias encoder takes a list of biasing phrases as the input. The context vector of the biasing list is generated by using the attention mechanism, and is then concatenated with the context vector of acoustic encoder and is fed into the decoder.

VI. Structured transforms

Methods to adapt the parameters θ of a neural network-based acoustic model $f(x;\theta)$ can be split into two groups. The first group adapts the whole acoustic model or some of its layers [113], [114], [154]. The second group employs structured transformations [109] to transform input features x, hidden activations h or outputs y of the acoustic model. Such transformations include the linear input network (LIN) [155], linear hidden network (LHN) [156] and the linear output network (LON) [157]. These transforms are parameterized with a transformation matrix $A_s \in \mathbb{R}^{n\times n}$ and a bias $b_s \in \mathbb{R}^n$. The transformation matrix A_s is initialized as an identity matrix and the bias b_s is initialized as a zero vector prior to speaker adaptation. The adapted hidden activations then become

$$h' = A_s h + b_s. \tag{28}$$

However, even a single transformation matrix A_s can contain many speaker dependent parameters, making adaptation susceptible to overfitting to the adaptation data. It also limits its practical usage in real world deployment because of memory requirements related to storing speaker dependent parameters for each speaker. Therefore there has been considerable research into how to structure the matrix A_s and the bias b_s to reduce the number of speaker dependent parameters.

The first set of approaches restricts the adaptation matrix A_s to be diagonal. If we denote the diagonal elements as $r_s = diag(A_s)$, then the adapted hidden activations become

$$h' = r_s \odot h + b_s. \tag{29}$$

There are several methods that belong to this set of adaptation methods. LHUC [75], [112] adapts only the parameters r_s:

$$h' = r_s \odot h. \tag{30}$$

Speaker Codes [158], [159] prepend an adaptation neural network to an existing SI model in place of the input features. The adaptation network – which operates somewhat similarly to control networks, described below – uses the acoustic features as inputs, as well as an auxiliary low-dimensional speaker code which essentially adapts speaker dependent biases within the adaptation network:

$$h' = h + b_s. \tag{31}$$

The network and speaker codes are learned by back-propagating through the frozen SI network with transcribed training data. At test time the speaker codes are derived by freezing all but the speaker code parameters and back-propagating on a small amount of adaptation data.

Similarly, Wang and Wang [160] proposed a method that adapts both r_s and b_s as parameters $\beta_s \in \mathbb{R}^n$ and $\gamma_s \in \mathbb{R}^n$ of a batch normalization layer, adapting both the scale and the offset of the hidden layer activations with mean $\mu \in \mathbb{R}^n$ and variance $\sigma^2 \in \mathbb{R}^n$:

$$h' = \gamma_s \frac{h - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta_s. \tag{32}$$

Mana et al. [161] showed that batch normalization layers can be also updated by recomputing the statistics μ and σ^2 in online fashion.

A similar approach with a low-memory footprint adapts the activation functions instead of the scale r_s and offset b_s. Zhang and Woodland [162] proposed the use of parameterised sigmoid and ReLU activation functions. With the parameterised sigmoid function, hidden activations h are computed from hidden pre-activations z as

$$h = \eta_s \frac{1}{1 + e^{-\gamma_s z + \zeta_s}}, \tag{33}$$

where $\eta_s \in \mathbb{R}^n$, $\gamma_s \in \mathbb{R}^n$ and $\zeta_s \in \mathbb{R}^n$ are speaker dependent parameters. $|\eta_s|$ controls the scale of the hidden activations, γ_s controls the slope of the sigmoid function and ζ_s controls the midpoint of the sigmoid function. Similarly, parameterised ReLU activations were defined as

$$h = \begin{cases} \alpha_s z & \text{if } z > 0 \\ \beta_s z & \text{if } z \leq 0 \end{cases}, \tag{34}$$

where $\alpha_s \in \mathbb{R}^n$ and $\beta_s \in \mathbb{R}^n$ are speaker dependent parameters that correspond to slopes for positive and negative pre-activations, respectively.

Other approaches factorize the transformation matrix A_s into a product of low-rank matrices to obtain a compact set of speaker dependent parameters. Zhao et al. [163] proposed the *Low-Rank Plus Diagonal* (LRPD) method, which reduces the number of speaker dependent parameters by approximating the linear transformation matrix $A_s \in \mathbb{R}^{n \times n}$ as

$$A_s \approx D_s + P_s Q_s, \tag{35}$$

where the $D_s \in \mathbb{R}^{n \times n}$, $P_s \in \mathbb{R}^{n \times k}$ and $Q_s \in \mathbb{R}^{k \times n}$ are treated as speaker dependent matrices ($k < n$) and D_s is a diagonal matrix. This approximation was motivated by the assumption that the adapted hidden activations should not be very different from the unadapted hidden activations when only a limited amount of adaptation data is available; hence the adaptation linear transformation should be close to a diagonal matrix. In fact, for $k = 0$ LRPD reduces to LHUC adaptation. LRPD adaptation can be implemented by inserting two hidden linear layers and a skip connection as illustrated in Fig. 3b.

Zhao et al. [164] later presented an extension to LRPD called *Extended LRPD* (eLRPD), which removed the dependency of the number of speaker dependent parameters on the hidden layer size by performing a different approximation of the linear transformation matrix A_s,

$$A_s \approx D_s + P T_s Q, \tag{36}$$

where matrices $D_s \in \mathbb{R}^{n \times n}$ and $T_s \in \mathbb{R}^{k \times k}$ are treated as speaker dependent, and matrices $P \in \mathbb{R}^{n \times k}$ and $Q \in \mathbb{R}^{k \times n}$ are treated as speaker independent. Thus the number of speaker dependent parameters is mostly dependent on k, which can be chosen arbitrarily.

Instead of factorizing the transformation matrix, a technique typically known as feature-space discriminative linear regression (fDLR) [135], [165], [166] imposes a block-diagonal structure such that each input frame shares the same linear transform. This is, in effect, a tied variation of LIN with a reduction in the number of speaker dependent parameters.

Another set of approaches uses the speaker dependent parameters as mixing coefficients $\theta_s = \{\alpha_0 \dots \alpha_k\}$ for a set of k speaker independent bases $\{B_0 \dots B_k\}$ which factorize the transformation matrix A_s. Samarakoon and Sim [167], [168] proposed to use factorized hidden layers (FHL) that allow both speaker-independent and speaker dependent modelling. With this approach, activations of a hidden layer h with an activation function σ are computed as

$$h = \sigma\left((W + \sum_{i=0}^{k} \alpha_i B_i)x + b_s + b\right). \tag{37}$$

Note, that when $\alpha_s = 0$ and $b_s = 0$, the activations correspond to a standard speaker independent model. If the bases B_i are rank-1 matrices, $B_i = \gamma_i \psi_i^T$, then this allows the reparameterization of (37) as [168]:

$$h = \sigma\left((W + \Gamma D \Psi^T)x + b_s + b\right), \tag{38}$$

where vectors γ_i and ψ_i are i-th columns of matrices Γ and Ψ, respectively, and the mixing coefficients α_s correspond to the diagonal of matrix D. This approach is very similar to the factorization of hidden layers used for Cluster Adaptive Training of DNN networks (CAT-DNN) [67] that uses full rank bases instead of rank-1 bases.

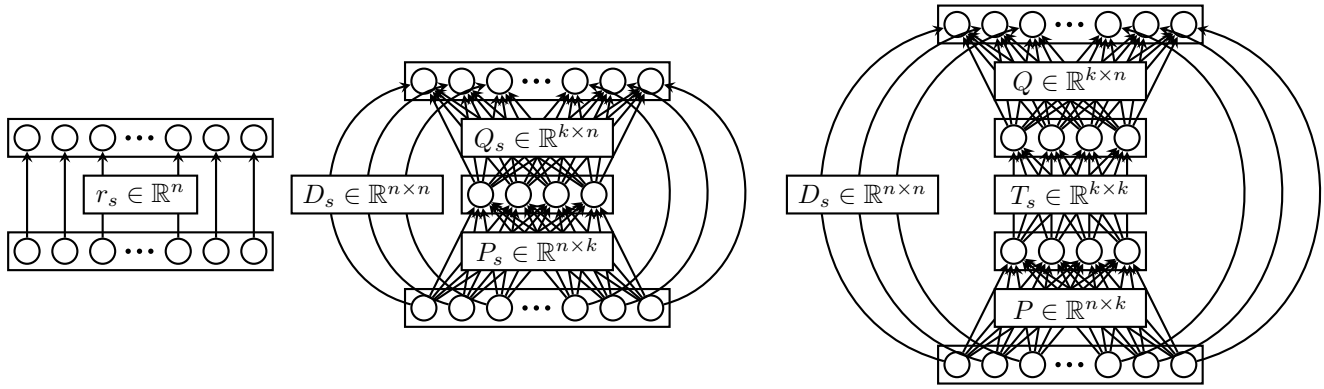
Similarly, Delcroix et al. [129] proposed to adapt the activations of a hidden layer using a mixture of experts [169]. The adapted hidden unit activations are then

$$h' = \sum_{i=0}^{k} \alpha_i B_i h. \tag{39}$$

There have also been approaches, that further reduce the number of speaker dependent parameters by removing the dependency on the hidden layer width by using control networks that predict the speaker-dependent parameters

$$\theta_s = c(\lambda_s; \phi), \tag{40}$$

In contrast to the adaptation network used in the Speaker Codes scheme, the control networks themselves are speaker-independent, taking as input some lower dimensional speaker embedding $\lambda_s \in \mathbb{R}^k$. As such, they form a link between structured transforms and the embedding-based approaches of Sec. V. The control networks $c(\lambda_s, \phi)$ can be implemented as a single linear transformation or as a multi-layer neural network. These control networks are similar to the conditional affine transformations referred to as Feature-wise Linear Modulation (FiLM) [170]. For example, Subspace LHUC [171] uses a control network to predict LHUC parameters r_s from i-vectors λ_s, resulting in a 94% memory footprint reduction compared to standard LHUC adaptation. Cui et al. [172] used auxiliary



(a) (b) (c)

Fig. 3: Structured transforms of an adaptation matrix A_s: (a) Learning Hidden Unit Contributions (LHUC) adapts only diagonal elements of the transformation matrix $r_s = diag(A_s)$; (b) Low-Rank Plus Diagonal factorizes the adaptation matrix as $A_s \approx D_s + P_s Q_s$; (c) Extended LRPD factorizes the adaptation matrix as $A_s \approx D_s + P T_s Q$.

features to adapt both the scale r_s and offset b_s. Other approaches adapted the scale r_s or the offset b_s by leveraging the information extracted with summary networks instead of auxiliary features [173]–[175].

Finally, the number of speaker dependent parameters in all the aforementioned linear transformations can be reduced by applying them to bottleneck layers that have much lower dimensionality than the standard hidden layers. These bottleneck layers can be obtained directly by training a neural network with bottleneck-layers or by applying Singular Value Decomposition (SVD) to the hidden layers [176], [177].

VII. Regularization methods

Even with the small number of speaker dependent parameters required by structured transformations, speaker adaptation can still overfit to the adaptation data. One way to prevent this overfitting is through the use of regularization methods that prevent the adapted model from diverging too far from the original model. This can be achieved by using early stopping and appropriate learning rates, which can be obtained with a hyper-parameter grid-search or by meta-learning [178], [179]. Another way to prevent the adapted model from diverging too far from the original can be achieved by limiting the distance between the original and the adapted model. Liao [113] proposed to use the L2 regularization loss of the distance between the original speaker dependent parameters θ_s and the adapted speaker dependent parameters θ'_s

$$\mathcal{L}_{L2} = |\theta_s - \theta'_s|_2^2. \tag{41}$$

Yu et al. [114] proposed to use Kullback-Leibler (KL) divergence to measure the distance between the senone distributions of the adapted model and the original model

$$\mathcal{L}_{KL} = D_{KL}(f(x;\theta) \,||\, f(x;\theta'_s)). \tag{42}$$

If we consider the overall adaptation loss using cross-entropy:

$$\mathcal{L} = (1-\lambda)\mathcal{L}_{xent} + \lambda \mathcal{L}_{KL}, \tag{43}$$

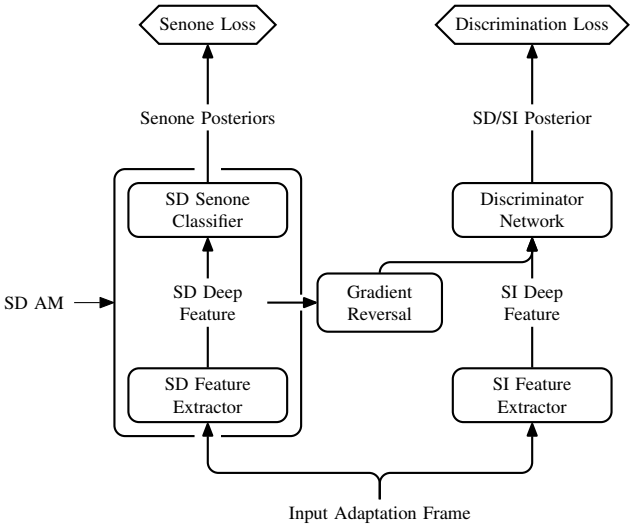


Fig. 4: Adversarial speaker adaptation.

we can show that this loss equals to cross-entropy with the target distribution for a label y given the input frame x_t

$$(1-\lambda)\hat{P}(y \mid x_t) + \lambda f(x_t;\theta), \tag{44}$$

where $\hat{P}(y \mid x_t)$ is a distribution corresponding to the provided labels y^{adapt}. Although initially proposed for adapting hybrid models, the KLD regularization method may also be used for speaker adaptation of E2E models [117], [118], [180].

Meng et al. [181] noted that KL divergence is not a distance metric between distributions because it is asymmetric, and therefore proposed to use adversarial learning which guarantees that the local minimum of the regularization term is reached only if the senone distributions of the speaker independent and the speaker dependent models are identical. They achieve this by adversarially training a discriminator $d(x;\phi)$ whose task is to discriminate between the speaker dependent deep features h' and speaker independent deep features h that are obtained by passing the input adaptation frames through speaker dependent and speaker independent

feature extractor respectively. This process is illustrated in Fig. 4. The regularization loss of the discriminator is

$$\mathcal{L}_{disc} = -\log d(h;\phi) - \log\left[1 - d(h';\phi)\right], \tag{45}$$

where h are hidden layer activations of the speaker independent model and h' are hidden layer activations of the adapted model. The discriminator is trained in a minimax fashion during adaptation by minimizing $\mathcal{L}_{disc}$ with respect to ϕ and maximizing $\mathcal{L}_{disc}$ with respect to θ_s. Consequently, the distribution of activations of the i-th hidden layer of the speaker dependent model will be indistinguishable from the distribution of activations of the i-th hidden layer of the speaker independent model, which ought to result in more robust performance of speaker adaptation.

Other approaches aim to prevent overfitting by leveraging the uncertainty of the speaker-dependent parameter space. Huang et al. [182] proposed Maximum A Posteriori (MAP) adaptation of neural networks, inspired by MAP adaptation of GMM-HMM models [83] (Sec. III). MAP adaptation estimates speaker dependent parameters as a mode of the distribution

$$\hat{\theta}_s = \arg\max_{\theta_s} P(Y \mid X, \theta_s) p(\theta_s), \tag{46}$$

where $p(\theta_s)$ is a prior density of the speaker dependent parameters. In order to obtain this prior density, Huang et al. [182] employed an empirical Bayes approach (following Gauvain and Lee [83]) and treated each speaker in the training data as a data point. They performed speaker adaptation for each speaker and observed that the speaker parameters across speakers resemble Gaussians. Therefore they decided to parameterise the prior density $p(\theta_s)$ as

$$p(\theta_s) = \mathcal{N}(\theta_s; \mu, \Sigma), \tag{47}$$

where μ is the mean of adapted speaker dependent parameters across different speakers, and Σ is the corresponding diagonal covariance matrix. With this parameterisation the regularization term of the prior density $p(\theta_s)$ is

$$\mathcal{L}_{MAP} = \frac{1}{2}(\theta_s - \mu)^T \Sigma^{-1} (\theta_s - \mu), \tag{48}$$

which for the prior density $p(\theta_s) = \mathcal{N}(\theta_s; 0, I)$ degenerates to the L2 regularization loss. Huang et al. investigated their proposed MAP approach with LHN structured transforms, but noted that it may be used in combination with other schemes.

Xie at al [183] proposed a fully Bayesian way of dealing with uncertainty inherent in speaker dependent parameters θ_s, in the context of estimating the LHUC parameters r_s (see Sec. VI). In this method, known as BLHUC, the posterior distribution of the adapted model is approximated as:

$$P(Y \mid X, \mathcal{D}^{\text{adapt}}) \approx P(Y \mid X, \mathbb{E}[r_s \mid \mathcal{D}^{\text{adapt}}]), \tag{49}$$

Xie at al propose to use a distribution $q(r_s)$ as a variational approximation of the posterior distribution of the LHUC parameters, $p(r_s|\mathcal{D}^{\text{adapt}})$. For simplicity, they assume that both $q(r_s)$ and $p(r_s)$ are normal, such that $q(r_s) = \mathcal{N}(r_s; \mu_s, \gamma_s)$ and $p(r_s) = \mathcal{N}(r_s; \mu_0, \gamma_0)$, which results in the expectation for the speaker dependent parameters in (49) being given by :

$$\mathbb{E}[r_s \mid \mathcal{D}^{\text{adapt}}] = \mu_s. \tag{50}$$

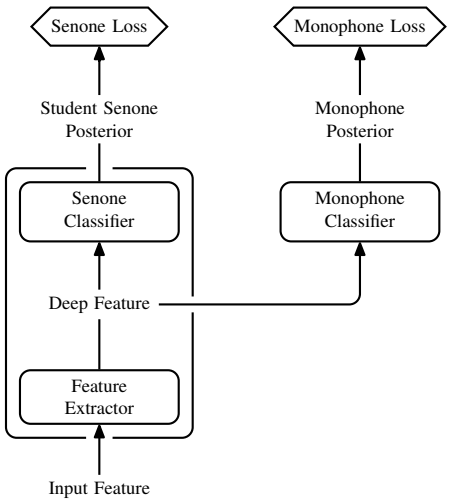


Fig. 5: Multi-task learning speaker adaptation.

The parameters are computed using gradient descent with a Monte Carlo approximation. Similarly to MAP adaptation, the effect is to force the adaptation to stay close to the speaker independent model when we perform adaptation with a small amount of adaptation data.

VIII. Variant objective functions

Another challenge in speaker adaptation is overfitting to targets seen in the adaptation data and to errors in semi-supervised transcriptions. This issue can be mitigated by an appropriate choice of objective function.

Gemello et al. [156] proposed *Conservative Training*, which modifies the target distribution to ensure that labels not seen in the adaptation data will not be catastrophically forgotten. Given a set of labels not seen in the adaptation data U and the reference label $\hat{y}_t$ at a time-step t the adjusted target distribution $\hat{P}$ is defined as

$$\hat{P}(y \mid x_t) = \begin{cases} P(y \mid x_t) & \text{if } y \in U \\ 1 - \sum\limits_{y' \in U} P(y'|x_t) & \text{if } y = \hat{y}_t \\ 0 & \text{otherwise.} \end{cases} \tag{51}$$

To mitigate errors in semi-supervised transcriptions we can replace the transcriptions with a lattice of supervision, which encodes the uncertainty arising from the first pass decoding. Lattice supervision has previously been used in work on unsupervised adaptation [76] and training [77] of GMMs, as well as discriminative [184] and semi-supervised training [78], and adaptation [79], of neural network models. For instance, lattice supervision can be used with the MMI criterion where for a single utterance we have:

$$\mathcal{F}_{MMI}(\theta) = \log \frac{p(X \mid \mathbb{M}^{num}; \theta)}{p(X \mid \mathbb{M}^{den}; \theta)}, \tag{52}$$

where the $\mathbb{M}_r^{num}$ is a numerator lattice containing multiple hypotheses from a first pass decoding and $\mathbb{M}_r^{den}$ is a denominator lattice containing all possible sequences of words.

Another family of methods prevents overfitting to adaptation targets by performing adaptation through the use of a lower

entropy task such as monophone or senone cluster targets. This has the advantage that the unsupervised targets might be less noisy and also that the targets have higher coverage even with small amounts of adaptation data. Price et al. [185] proposed to append a new output layer predicting monophone targets on top of the original output layer predicting senones. The layer can be either full rank or sparse – leveraging knowledge of relationships between monophones and senones. Its parameters are trained on the training data with a fixed speaker independent model. Only the monophone targets are used for the adaptation of the speaker dependent parameters.

Huang et al. [115] presented an approach that used multi-task learning [186] to leverage both senone and monophone/senone clusters targets. It worked by having multiple output layers, each on top of the last hidden layer, that predicted the corresponding targets. These additional output layers were also trained after a complete training pass of the speaker independent model with its parameters fixed. Thus, the adaptation loss was a weighted sum of individual losses, for example monophone and senone losses (Fig. 5). Swietojanski et al. [187] combined these two approaches and used multi-task learning for speaker adaptation through a structured output layer, which predicts both monophone targets and senone targets. Unlike the approach by Price et al. [185], the monophone predictions are used for the prediction of senones.

Li et al. [117] and Meng et al. [118] applied multi-task learning to speaker adaptation of CTC and AED models. These E2E models typically use subword units, such as word piece units, as the output target in order to achieve high recognition accuracy. The number of subword units is usually at the scale of thousands or even more. Given very limited speaker-specific adaptation data, these units may not be fully covered. Multi-task learning using both character and subword units can significantly alleviate such sparseness issues.

IX. Data augmentation

Data augmentation has been proven to be an effective way to decrease the acoustic mismatch between training and testing conditions. Data augmentation approaches supplement the training data with distorted or synthetic variants of speech with characteristics resembling the target acoustic environment, for instance with reverberation or interfering sound sources. Thanks to realistic room acoustic simulators [188] one can generate large numbers of room impulse responses and reuse clean corpora to create multiple copies of the same sentence under different acoustic conditions [189]–[191].

Similar approaches have been proposed for increasing robustness in speaker space by augmenting training data with, typically label-preserving, speaker-related distortions or transforms. Examples include creating multiple copies of clean utterances with perturbed VTL warp factors [192], [193], augmenting related properties such as volume or speaking rate [11], [194], [195], or voice-conversion [196] inspired transformations of speech uttered by one speaker into another speaker using stochastic feature mapping [193], [197], [198].

While voice conversion does not create any new data with respect to unseen acoustic / linguistic complexity (just replicas of the utterances with different voices, often from the same dataset), recent advances in text-to-speech (TTS) allows the rapid building of new multi-speaker TTS voices [199] from small amounts of data. TTS may then be used to arbitrarily expand the adaptation set for a given speaker, possibly to cover unseen acoustic domains [116], [120]. If TTS is coupled with a related natural language generation module, it is possible to generate speech for domain-related texts. In this way, the speaker adaptation uses more data, not only from the speaker's original speech but also from the TTS speech. Because the transcription used for TTS generation is also used for model adaptation, this approach also circumvents the obstacle of the hypothesis error in unsupervised adaptation. Moreover, TTS generated data can also help to adapt E2E models to a new domain which has more discrepancy in contents from the source domain, which will be discussed in Sec. XII.

Finally, for unbalanced data sets the acoustic models may under-perform for certain demographics that are not sufficiently represented in training data. There is an ongoing effort to address this using generative adversarial networks (GANs). For example, Hosseini-Asl et al. [200] used GANs with a cycle-consistency constraint [201] to balance the speaker ratios with respect to gender representation in training set.

X. Accent adaptation

Although there is significant literature on automatic dialect identification from speech (*e.g.* [202]), there has been less work on accent and dialect adaptive speech recognition systems. The MGB–3 [203] and MGB–5 [204] evaluation challenges have used dialectal Arabic test sets, with a modern standard Arabic (MSA) training set, using broadcast and internet video data. The best results reported on these challenges have used a straightforward model-based transfer learning approach in an lattice-free maximum mutual information (LF-MMI) framework [205], adapting MSA trained baseline systems to specific Arabic dialects [206], [207].

Much of the reported work on accent adaptation has taken approaches for speaker adaptation, and applied them using an adaptation set of utterances from the target accent. For instance, Vergyri et al. [208] used MAP adaptation of a GMM/HMM system. Zheng et al. [209] used both MAP and MLLR adaptation, together with features selected to be discriminative towards accent, with the accent adaptation controlled using hard decisions made by an accent classifier.

Earlier work on accent adaptation focused on automatic adaptation of the pronunciation dictionary [210], [211]. These approaches resemble approaches for acoustic adaptation of VQ codebooks (discussed in section III), in that they learn an accent-specific transition matrix between the phonemic symbols in the dictionary. Selection of utterances for accent adaptation has been explored, with Nallasamy et al. [212] proposing an active learning approach.

Approaches to accent adaptation of neural network-based systems have typically employed accent-dependent output layers and shared hidden layers [213], [214], based on a similar approach to the multilingual training of deep neural networks [215]–[217]. Huang et al. [213] combined this with

KL regularization (Sec. VII), and Chen et al. [214] used accent-dependent i-vectors (Sec. V); Yi et al. [218] used accent-dependent bottleneck features in place of i-vectors; and Turan et al. [219] used x-vector accent embeddings in a semi-supervised setting.

Multi-task learning approaches, where the secondary task is accent/dialect identification has been explored by a number of researchers [220]–[224] in the context of both hybrid and end-to-end models. Improvements with multi-task training were observed in some instances, but the evidence indicates that it gives a small adaptation gain. Sun et al. [225] replaced multi-task learning with domain adversarial learning (Sec. VIII), in which the objective function treated accent identification as an adversarial task, finding that this improved accented speech recognition over multi-task learning.

More successfully, Li et al. [226] explored learning multi-dialect sequence-to-sequence models using one-hot dialect information as input. Grace et al. [227] also used one-hot dialect codes and also explored a family of cluster adaptive training and hidden layer factorization approaches. In both cases using one-hot dialect codes as an input augmentation (corresponding to bias adaptation) proved to be the best approach, and cluster-adaptive approaches did not result in a consistent gain. These approaches were extended by Yoo et al. [228] and Viglino et al. [224] who both explored the use of dialect embeddings for multi-accent end-to-end speech recognition. Ghorbani et al. [229] used accent-specific teacher-student learning, and Jain et al. [230] explored a mixture of experts (MoE) approach, using mixtures of experts both at the phonetic and accent levels.

Yoo et al. [228] also applied a method of feature-wise affine transformations on the hidden layers (FiLM), that are dependent both on the network's internal state and the dialect/accent code (discussed in Sec. VI). This approach, which can be viewed as a conditioned normalization, differs from the previous use of one-hot dialect codes and multi-task learning in that it has the goal of learning a single normalized model rather than an implicit combination of specialist models. A related approach is gated accent adaptation [231], although this focused on a single transformation conditioned on accent.

Winata et al. [232] experimented with a meta-learning approach for few-shot adaptation to accented speech, where the meta-learning algorithm learns a good initialization and hyperparameters for the adaptation.

XI. Domain adaptation

The performance of automatic speech recognition (ASR) always drops significantly when the recognition model is evaluated in a mismatched new domain. Domain adaptation is the technology used to adapt the well-trained source domain model to the new domain. The most straightforward way is to collect and label data in the new domain to fine-tune the model. Most adaptation technologies discussed in this paper can also be applied to domain adaptation [154], [233]–[236]. When the amount of adaptation data is limited, a common practice is adapting only partial layers of the network [237]. To let the adapted model still perform well on the source domain, Moriya et al. [238] proposed progressive neural networks by adding an additional model column to the original model for each new domain and only update the new model column with the new domain data. In the following, we focus on technologies more specific to domain adaptation.

A. Teacher-student learning

While conventional adaptation techniques require large amounts of labeled data in the target domain, the teacher-student (T/S) paradigm [239], [240] can better take advantage of large amounts of unlabeled data and has been widely used for industrial scale tasks [241], [242].

The most popular T/S learning strategy was proposed in 2014 by Li et al. [239] to minimize the KL divergence between the output posterior distributions of the teacher network and the student network. This can also be considered as learning soft targets generated by a teacher model instead of 1-hot hard targets

$$-\sum_{t=1}^{T}\sum_{y=1}^{N} P_T(y \mid x_t) \log P_S(y \mid x_t), \tag{53}$$

where P_T and P_S are posteriors of teacher and student networks, x_t and y_t are the input speech and output senone at time t, respectively. T is the number of speech frames in an utterance, and N is the number of senones in the network output layer.

Later, Hinton et al. [240] proposed knowledge distillation by introducing a temperature parameter (like chemical distillation) to scale the posteriors. This has been applied to speech by *e.g.* Asami et al. [243]. There are also variations such as learning the interpolation of soft and hard targets [240] and conditional T/S learning [244]. Although initially proposed for model compression, T/S learning is also widely used for model adaptation if source and target signals are frame-synchronized, which can be realized by simulation. The loss function is [245], [246]

$$-\sum_{t=1}^{T}\sum_{y=1}^{N} P_T(y \mid x_t) \log P_S(y \mid \hat{x}_t), \tag{54}$$

where x_t is the source speech signal while $\hat{x}_t$ is the frame-synchronized target signal. It can be further improved with sequence-level loss function as the speech signal is a sequence signal [247], [248].

The biggest advantage of T/S learning is that it can leverage large amounts of unlabeled data by using soft labels $P_T(y_t = y|x_t)$. This is particularly useful in industrial setups where effectively unlimited unlabeled data is available [241], [242]. Furthermore, soft labels produced by the teacher network carry knowledge learned by the teacher on the difficulty of classifying each sample, while the hard labels do not contain such information. Such knowledge helps the student to generalize better, especially when adaptation data size is small.

E2E models tend to memorize the training data well, and therefore may not generalize well to a new domain. Meng et

al. [249] proposed T/S learning for the domain adaptation of E2E models. The loss function is

$$-\sum_{l=1}^{L}\sum_{y=1}^{N} P_T(y \mid Y_{1:u-1}, X) \log P_S(y \mid Y_{1:u-1}, \hat{X}), \quad (55)$$

where X and $\hat{X}$ are the source and target domain speech sequence, Y is the label sequence of length L which is either the ground truth in the supervised adaptation setup or the hypothesis generated by the decoding of the teacher model with X in the unsupervised adaptation setup. Note that in the unsupervised case, there are two levels of knowledge transfer: the teacher's token posteriors (used as soft labels) and one-best predictions as decoder guidance.

One constraint to T/S adaptation is that it requires paired source and target domain data. While the paired data can be obtained with simulation in most cases, there are scenarios in which it is hard to simulate the target domain data from the source domain data. For example, simulation of children's speech or accented speech remains challenging. In [134], a neural label embedding scheme was proposed for domain adaptation with unpaired data. A label embedding, l-vector, represents the output distribution of the deep network trained in the source domain for each output token, *e.g.* , senone. To adapt the deep network model to the target domain, the l-vectors learned from the source domain are used as the soft targets in the cross entropy criterion.

B. Adversarial learning

It is usually hard to obtain the transcription in the target domain, therefore unsupervised adaptation is critical. Although the transcription can be generated by decoding the target domain data using the source domain model, the generated hypothesis quality is often poor given the domain mismatch. Recently, adversarial training was applied to the area of unsupervised domain adaptation in a form of multi-task learning [250] without the need for transcription in the target domain. Unsupervised adaptation is achieved by learning deep intermediate representations that are both discriminative for the main task on the source domain and invariant with respect to mismatch between source and target domains. Domain invariance is achieved by adversarial training of the domain classification objective functions using a *gradient reversal layer (GRL)* [250]. This GRL approach has been applied to acoustic models for unsupervised adaptation in [251]–[253]. Meng et al. [254] further combine adversarial learning and T/S learning as adversarial T/S learning to improve the robustness against condition variability during adaptation.

There is also increasing interest in the use of GANs with cycle consistency constraints for domain adaptation [255]–[257]. This enables the use of non-parallel data without labels in the target domain by learning to map the acoustic features into the style of the target domain for training. The cycle-consistency constraint also provides the possibility of mapping features from the target to the source style for, in effect, test-time adaptation or speech enhancement.

Unsupervised domain adaptation is more attractive than the supervised one because there is usually large amount of unlabeled data in the new domain while transcribing the new domain data usually is time consuming with large cost. T/S learning and adversarial learning both can utilize unlabeled data well. Specifically, T/S learning has been very successful in industry-scale tasks. In contrast, adversarial learning was reported successful in relatively smaller tasks. Therefore, T/S learning is more promising if the parallel data is available. However, if there is no prior knowledge about the new domain, adversarial learning can be a good choice. There are also other works on unsupervised domain adaptation. For example, Hsu et al. [70] use a variational autoencoder instead of adversarial learning to obtain a latent representation robust to domains. However, similar to adversarial learning, such method is pending examination when large amount of unlabeled training data is available.

XII. Language model adaptation

LM adaptation typically involves updating an LM estimated from a large general corpus, with data from a target domain. Many approaches to LM adaptation were developed in the context of n-gram models, and are reviewed by Bellegarda [258]. Hybrid NN/HMM speech recognition systems still make use of n-gram language models and a finite state structure, at least in the first pass; it is difficult to use neural network LMs (with infinite context) directly in first pass decoding in such systems. Neural network LMs are typically used to rescore lattices in hybrid systems, or may be combined (in a variety of ways) in end-to-end systems.

The main techniques for n-gram language model adaptation include interpolation of multiple language models [259]–[261], updating the model using a cache of recently observed (decoded) text [259], [262]–[264], or merging or interpolating n-gram counts from decoded transcripts [265]. There is also a large body of work incorporating longer scale context, for instance modelling the topic and style of the recorded speech [266]–[269]. LM adaptation approaches making use of wider context have often built on approaches using unigram statistics or bag-of-words models, and a number of approaches for combination with n-gram models have been proposed, for example dynamic marginals [270].

Neural network language modelling [271] has become state-of-the-art, in particular recurrent neural network language models (RNNLMs) [272]. There has been a range of work on adaptation of RNNLMs, including the use of topic or genre information as auxiliary features [273], [274] or combined as marginal distributions [275], domain specific embeddings [276], and the use of curriculum learning and fine-tuning to take account of shifting contexts [277], [278]. Approaches based on acoustic model adaptation, such as LHUC [278] and LHN [274], have also been explored.

There have a been a number of approaches to apply the ideas of cache language model adaptation to neural network language models [275], [279], [280], along with so-called dynamic evaluation approaches in which the recent context is used for fine tuning [275], [281].

E2E models are trained with paired speech and text data. The amount of text data in such a paired setup is much smaller

than the amount of text data used in training a separate external LM. Therefore, it is popular to adjust E2E models by fusing the external LM trained with a large amount of text data. The simplest and most popular approach is shallow fusion [282]–[285], in which the external LM is interpolated log-linearly with the E2E model at inference time only.

However, shallow fusion does not have a clear probabilistic interpretation. McDermott et al. [286] proposed a density ratio approach based on Bayes' rule. An LM is built on text transcripts from the training set which has paired speech and text data, and a second LM is built on the target domain. When decoding on the target domain, the output of the E2E model is modified by the ratio of target/training LMs. While it is well grounded with Bayes' rule, the density ratio method requires the training of two separate LMs, from the training and target data respectively. Variani et al. [287] proposed a hybrid autoregressive transducer (HAT) model to improve the RNN-T model. The HAT model builds a training set LM internally and the label distribution is derived by normalizing the score functions across all labels excluding blank. Therefore, it is mathematically justified to integrate the HAT model with an external or target LM using the density ratio formulation.

In [126], [127], RNN-T models were adapted to a new domain with the TTS data generated from the domain-specific text. Because the prediction network in RNN-T works similarly to a LM, adapting it without updating the acoustic encoder is shown to be more effective than interpolating the RNN-T model with an external LM trained from the domain-specific text [127].

XIII. Meta Analysis

In this section we present an aggregated review of published results in experiments applying adaptation algorithms to speech recognition. This differs from typical experimental reporting that focuses on one-to-one system comparisons typically using a small fixed set of systems and benchmark tasks and data. The proposed meta-analysis approach offers insights into the performance of adaptation algorithms that are difficult to capture from individual experiments.

We divide this section into four main parts. The first, Sec. XIII-A, explains the protocol and overall assumptions of the meta-analysis, followed by a top-level summary of findings in Sec. XIII-B, with a more detailed analysis in Sec. XIII-C. The final part, Sec. XIII-D, aims to quantify the adaptation performance across languages, speaking styles and datasets.

A. Protocol and Literature

The meta-analysis is based on 47 peer-reviewed studies selected such that they cover a wide range of systems, architectures, and adaptation tasks. Each study was required to compare adaptation results versus a baseline, enabling the configurations of interest to be compared quantitatively. There was no fixed target for the total number of papers included, due to our aim to cover as many different methods as possible. Note that the meta-analysis spans several model architectures, languages, and domains; although most studies use word error rate (WER) as the evaluation metric, some studies used character error rate (CER) or phone error rate (PER). Since we are interested in the relative improvement brought by adaptation, we report Relative Error Rate Reductions (RERR).

The meta-analysis is based on the studies shown in Table I, with additional splits into level of operation and top-level system architecture. The positions were selected such that they cover most of the topics mentioned in the review. For an adaptation of end-to-end systems we included all peer-reviewed works we could find (their number is relatively limited). For the hybrid approach, the studies are shortlisted such that they enable the quantification of the gains for the categories outlined in the preceding theoretical sections. As a generic rule, when choosing papers for the analysis we first included works that introduced a specific adaptation method in the context of neural models, or that offered some additional experiments allowing the comparison of different areas of interest - such as the impact of objective functions, the complementarity of adaptation transforms or that show behavior under different operating regimes. In the case of certain more commonly-used techniques, due to the laborious nature of the analysis, it was not always possible to include an exhaustive set of somewhat similar papers. In this situation, the papers selected were those with higher citation counts.

The analysis spans 38 datasets (more than 50 unique $\{$train, test$\}$ pairings), 28 of which are public and 10 are proprietary. These cover different speaking styles, domains, acoustic conditions, applications and languages (though the study is strongly biased towards English resources). The public corpora used include the following: AISHELL2 [298], AMI [299], APASCI [300], Aurora4 [301], CASIA [302], ChildIt [303], Chime4 [304], CSJ [305], ETAPE [306], HKUST [307], MGB [308], RASC863 [309], SWBD [310], TED [311], TED-LIUM [312], TED-LIUM2 [313], TIMIT [314], WSJ [315], PF-STAR [316], Librispeech [317], Intel Accented Mandarin Speech Recognition Corpus [214], UTCRSS-4EnglishAccent [295]. To save space we do not provide detailed corpora statistics in this paper, but make them available via a corresponding repository[1] alongside raw data and scripts used to perform the analysis.

Overall, the meta-analysis is based on ASR systems trained on datasets with a combined duration of over 30,000 hours, while the baseline acoustic models were estimated from as little as 5 hours to around 10,000 hours of speech. Adaptation data varies from a few seconds per speaker to over 25,000 hours of acoustic material used for domain adaptation.

B. Overall findings

Fig. 6 (Top) presents the average adaptation gains for all considered systems, adaptation methods, and adaptation classes. The overall RERR is 9.72%[2]. Since grouping data across attributes of interest may result in unbalanced (or very sparse) sample sizes, we also report additional statistics such as the number of samples, datasets and studies the given statistic is based on. As can be seen in the right part of Fig. 6

[1] https://github.com/pswietojanski/ojsp_adaptation_review_2020.

[2] We do not report exact numbers in tabular form due to space limitations, but they are available in the GitHub repository

TABLE I: Adaptation studies used in the meta-analysis categorized on the level they operate at, and system architecture.

Level	System	References
Model	Hybrid	[74], [75], [113], [141], [153], [154], [168], [178]–[180], [195], [213], [214], [231], [241], [288]–[294]
	E2E	[117]–[119], [180], [229], [232], [249], [295]
Embedding	Hybrid	[56], [57], [61], [74], [130], [132], [138], [148], [150], [153], [159], [161], [168], [214], [296]
	E2E	[62], [128], [218]
Feature	Hybrid	[56], [74], [75], [135], [138], [168], [231], [289], [293], [294], [297]
Data	Hybrid	[116], [193]

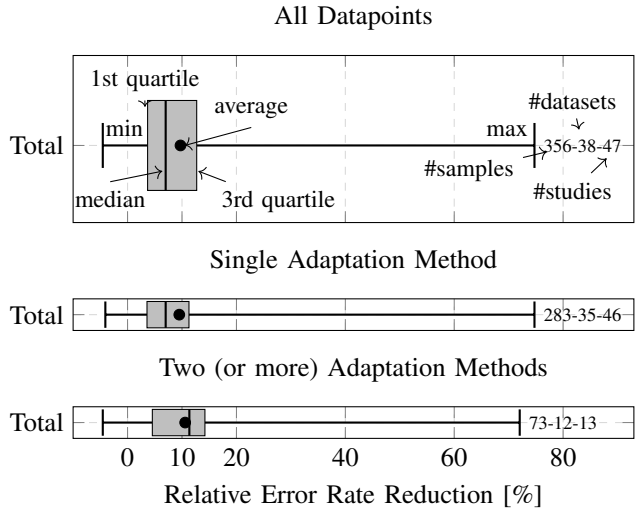


Fig. 6: Aggregated summary of adaptation RERR from all studies (top), considering single method only (middle) and two or more methods stacked (bottom). The top graph is annotated to explain the information presented in each of the boxplot graphs in this section.

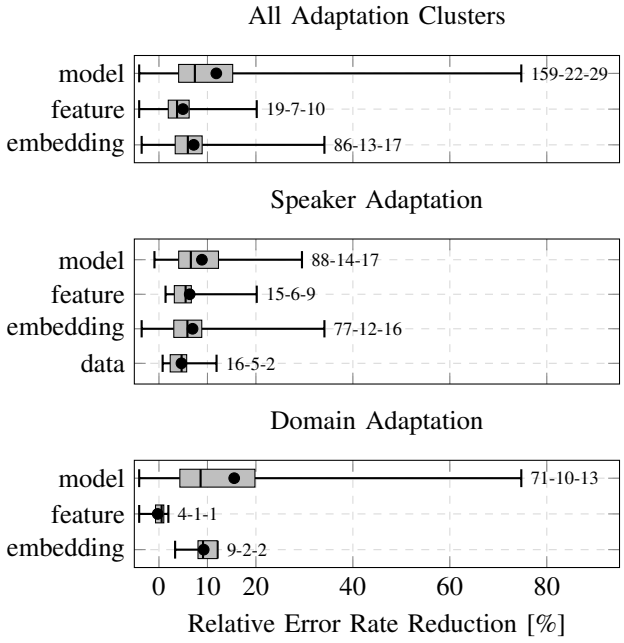


Fig. 7: Comparison of feature, embedding, and model-level adaptation approaches. Speaker (middle) and domain (bottom) adaptations are based on {utterance, speaker} and {accent, child, domain, disordered} clusters, respectively.

(Top), the results in this review were derived from 356 samples produced using 38 datasets reported in 47 studies. A single sample is defined as a 1:1 system comparison for which one can unambiguously state the RERR. Likewise, a dataset refers to a particular training corpus configuration. Note that there may be some data-level overlap between different corpora originating from the same source (*e.g.* TED talks) and we make a distinction for the acoustic condition (*e.g.* AMI close-talking and distant channels are counted as two different datasets when they are used to estimate separate acoustic models). A study refers to a single peer-reviewed publication.

Depending on which property we want to measure the analysis set can be split into smaller subsets, as the ones shown in the lower part of Fig. 6. The majority of analyses in this review are reported for models adapted using a single method with some additional groupings used to better capture further details such as complementarity of adaptation methods or their performance in different operating regimes.

As mentioned in Sec. IV, adaptation methods were historically categorized based on the level they operated at in the speech processing pipeline. Fig. 7 (top) quantifies the ASR performance along this attribute, showing that model-based adaptation obtains the best average improvements of 11.8%, followed by embedding and feature levels at 7.2% and 5.0% RERR, respectively. This is not surprising, as model level adaptation allows large amounts of adaptation data to be leveraged by allowing the update of large portions of the model (including re-training the whole model). In more data-constrained regimes, such as utterance or speaker-level adaptation, where only a limited amount of adaptation data is typically available, differences are less pronounced and model-based speaker adaptation obtains 8.9% RERR while adapting to domains gives 15.5% RERR (*cf.* middle and bottom plots in Fig 7). Embedding approaches stay at a similar level for speaker adaptation, improving to 9.2% RERR for domain adaptation (although based on only two studies). Feature-space domain adaptation was used in only one study, which reported a small deterioration of -0.3% RERR.

Fig. 7 (middle) additionally shows results for speaker-oriented data augmentation as described in Sec. IX. These were found to increase accuracy by 4.6% RERR on average, or by 3.3%, 3.6% and 8.2% RERR for VTL perturbations (VTLP) [192], [193], stochastic feature mapping (SFM) [193],

TABLE II: Amounts of data used to estimate Hybrid and E2E models for `Speaker` and `Domain` adaptation clusters.

Cluster	Avg. Training Data [hours]	
	Hybrid	E2E
All	862	1747
Speaker	874	2640
Domain	824	1033

[197] and when using synthetically generated TTS utterances [116], respectively. Note that the TTS method was used to augment the adaptation set to better estimate additional adaptation transforms while VTLP and SFM were used to directly expand the training data, and were found particularly effective for low-resource training conditions. Data augmentations are beyond the scope of this meta-analysis and will not be further investigated in this review.

The results for different adaptation clusters, introduced in Sec. II, are shown in Fig. 8. Models benefit more when adapting to accent, from adult to child speech, to the domain, and to disordered speech conditions (such as arising from speech motor disorders), as opposed to speaker or utterance adaptation. This is expected, since domain adaptation usually has more adaptation data, and the acoustic mismatch introduced by unseen domains is greater than the mismatch caused by unseen speakers – unless these are substantially mismatched to the training data as is often the case for child or disordered speech recognition. But in the latter case the adaptation is typically not carried out at the speaker level, but at the domain level (*i.e.* tailoring the acoustic model to better handle dysarthric speech, not a single dysarthric speaker).

Fig. 9 aggregates the adaptation along the two main neural network-based ASR approaches - hybrid and E2E. It is interesting to observe that E2E systems gain more from adaptation (12.8% RERR) than hybrid systems (9.2% RERR) in both the overall and speaker-based regimes. This is somewhat expected, as hybrid systems benefit from strong inductive biases – such as access to pronunciation dictionaries and hand engineered modeling constraints – whereas E2E models must learn these from data. Given limited amounts of training data one may expect that E2E may struggle to learn these as well as hybrid models, as such adaptation may bring greater gains. This reverses for domain adaptation, with E2E and hybrid improving by 12.2 and 14.9% RERR, respectively. Note that for domain adaptation, the hybrid approach was studied more often for child and disordered speech applications, which makes adaptation gains bigger (see also Fig. 8). Table II further reports average amounts of training data used to estimate hybrid and E2E models. It is interesting to notice that E2E systems on average leverage twice as much acoustic material when compared to hybrid setups but still seem to substantially benefit from adaptation. These results suggest that adaptation for E2E is a promising direction for future investigations, that remains under-investigated as of now based on the relatively few works published to date.

Next we compare feed-forward (FF) and recurrent neural network (RNN) architectures in both hybrid and E2E models.

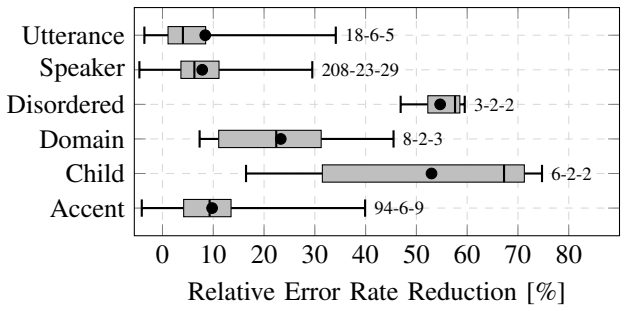


Fig. 8: Adaptation results for different adaptation clusters.

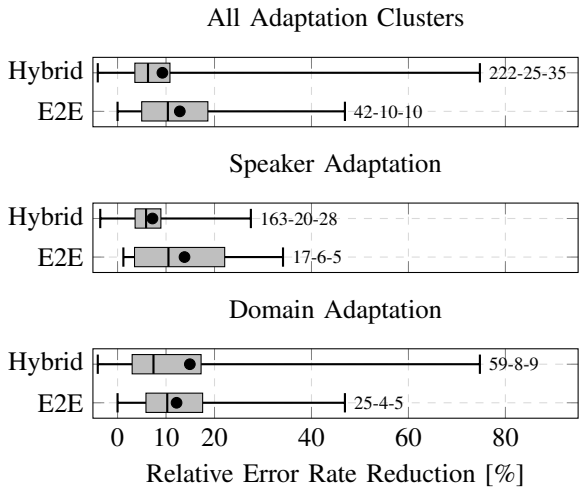


Fig. 9: Comparison of adaptation results for hybrid and E2E systems.

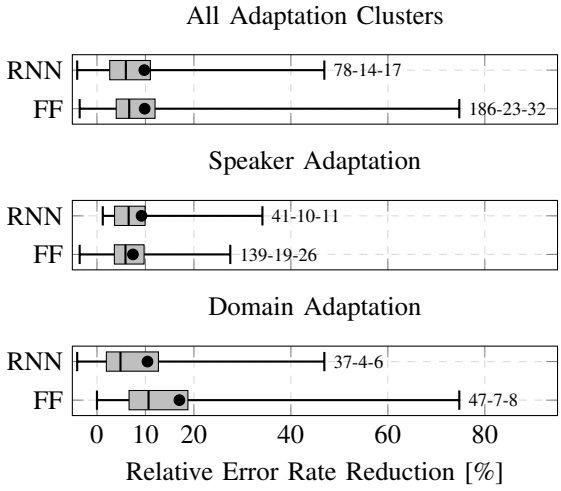


Fig. 10: Comparison of adaptation results for FF and RNN architectures.

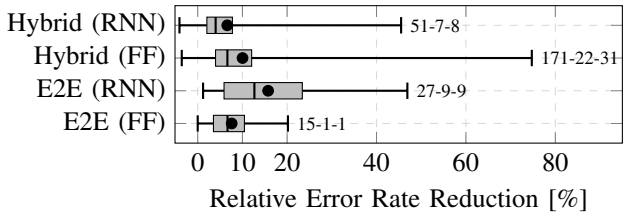


Fig. 11: Comparison of adaptation results for FF and RNN architectures split by hybrid and E2E systems.

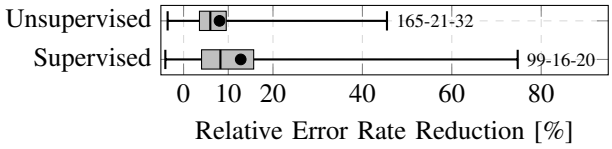


Fig. 12: Comparison of adaptation results for supervision modes.

Hybrid models can leverage either FF or RNN architectures while most E2E systems use some form of RNN. (Note, transformer-based E2E models [30] are built from FF (CNN) modules, however, due to their relative novelty in ASR there is only one accent adaptation study included in our meta-analysis [232]). Fig. 10 reports similar adaptation gains of 9.8% RERR for both FF and RNN architectures. RNNs seem to benefit more when adapting to speakers (9.2% vs 7.4% RERR for RNN and FF, respectively), and less when adapting to domain (10.4% vs 17.0% RERR for RNN and FF, respectively). When controlling for the system paradigm (E2E vs. Hybrid), RNNs mostly benefit through adapting E2E models (*cf.* Fig 11 6.6% vs 15.7% RERR for Hybrid (RNN) and E2E (RNN), respectively). We observed a similar trend for speaker and domain clusters separately (figure not shown).

Fig. 12 compares the RERR for unsupervised and supervised modes of adaptation. Overall, deriving the adaptation transform with manually annotated targets results in an average 12.8% RERR, whereas unsupervised methods result in 8% RERR. Fig. 12 shows results specifically for semi-supervised adaptation, which are captured by the 2pass and enrol (Unsup.) conditions. Fig. 13 also shows further analysis on the modes of deriving adaptation statistics (Sec. II). Both online and two-pass adaptation are unsupervised, while the enrollment mode may be either supervised or unsupervised. The supervised approach offers most accurate adaptation, as expected. Unsupervised enrollment outperforms the other two unsupervised methods mainly due to the T/S domain adaptation study [249] (Sec. XI) that leverages large amounts of data. When considering speaker adaptation only, the two-pass approach obtains 8.2% RERR and is more effective than enrol (Unsup.) (7.3% RERR) and online adaptation (6.5% RERR).

Finally, we consider the overall trends for the considered systems and their operating regions. Fig. 14 reports results obtained with different amounts of adaptation data. Fig. 16 further shows regression trends when splitting by adaptation type, hybrid or E2E, and adaptation clusters. These are in

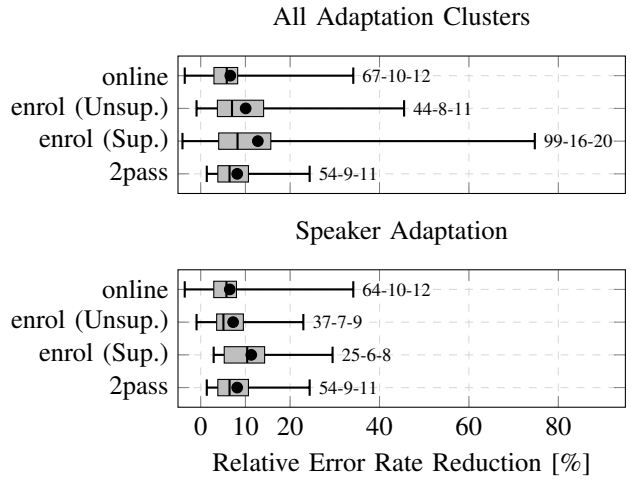


Fig. 13: Comparison of adaptation results for different adaptation targets: online adaptation, supervised and unsupervised enrollment, and two-pass decoding.

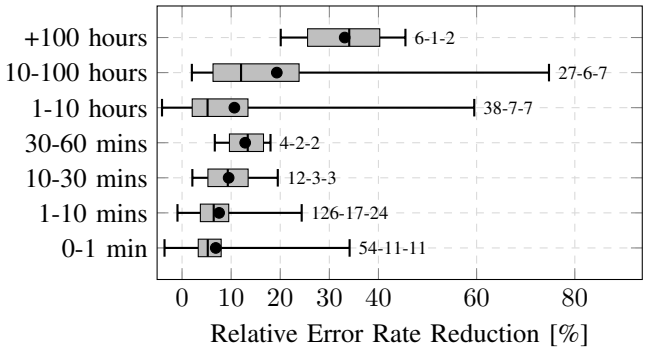


Fig. 14: Comparison of adaptation results for different amount of adaptation data.

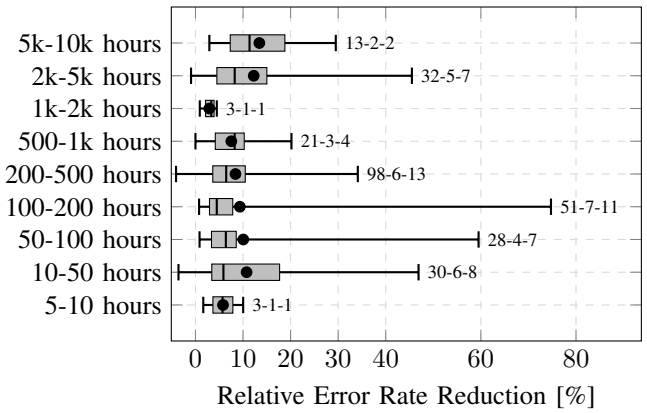
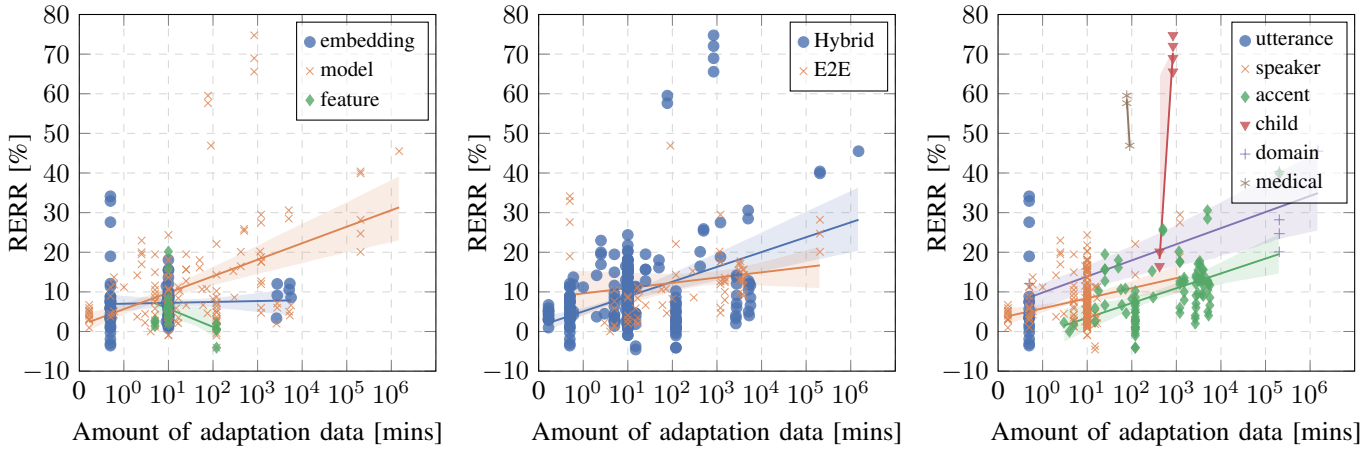


Fig. 15: Comparison of adaptation results for acoustic models estimated from different amounts of training data.



(a) Adaptation Type (b) Hybrid vs. E2E (c) Adaptation Clusters

Fig. 16: Regression analysis for the three major control variables.

line with the observations so far: i) more adaptation data brings (on average) larger improvements; ii) model-based adaptation is more powerful and gives better results than embedding or feature-based approaches; and iii) adaptation is particularly effective in scenarios with a large mismatch and where obtaining matched training data is difficult.

In Fig. 15 we further report adaptability of acoustic models estimated from different amounts of training material. Interestingly, models trained on small amounts of data (up to 50 hours) benefit from adaptation to a similar degree as models estimated from several thousands of hours. This is somewhat an unexpected result - if test sets are kept fixed, increasing the training material typically results in a less mismatched model, thus lowering gains from adaptation (and most experiments evaluating adaptation performance as a function of data are carried out in this way). However, when training from more data one should proportionally increase the complexity of the testing conditions. We hypothesize that this is what implicitly occurs across different datasets in the meta-analysis - someone who has access to a large training set may also sample a more diverse testing set. Note that the acoustic models in this work were trained from relatively limited amounts of data (up to 10k hours), and adaptation protocols between studies may not be exactly comparable. However, this does not change the conclusion that some form of adaptation is beneficial for most considered systems, regardless of how many hours of acoustic data was used to train it.

Since this meta-analysis combines results across many different studies with many reference systems, the results should not necessarily be compared at the sample level, but rather in an aggregated form to outline dominant trends and typical data regimes that each category was tried in. Data amounts for some systems for the purpose of plotting were assumed approximately to be at a given level: *e.g.* two-pass systems unless shown otherwise assumed 10 minutes per speaker, while embedding approaches 30 seconds.

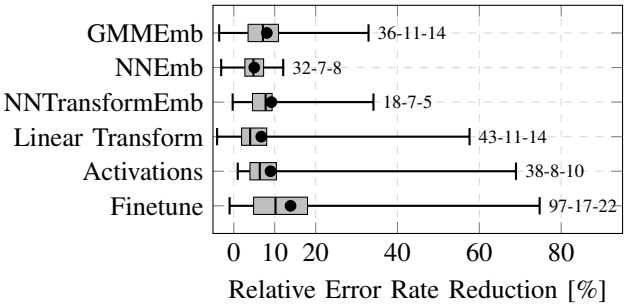
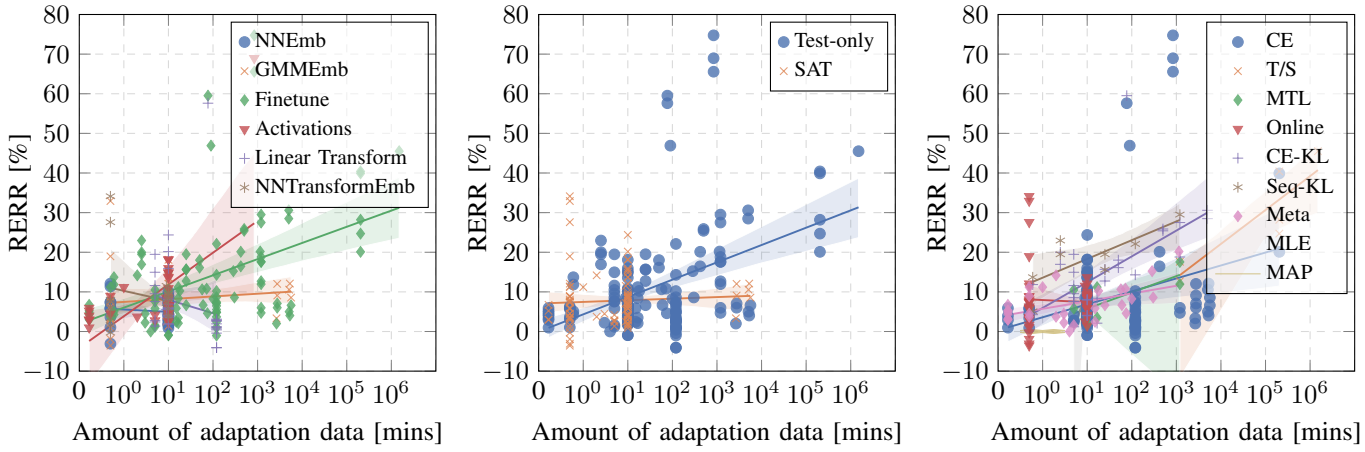


Fig. 17: Comparison of results for different adaptation approaches.

C. Detailed findings

In this subsection we investigate the effect of the specific approach to adaptation, beyond the broad categories discussed above. Fig. 17 reorganizes the earlier split into feature, embedding, and model-level adaptation (Fig. 7) into embedding (*cf.* Sec. V) and model-based transformations (*cf.* Sec. VI).

For the embeddings, we introduce three sub-categories referred to as GMMEmb, NNEmb and NNTransformEmb. GMMEmb comprises GMM-related embedding extractors primarily based on i-vectors [56], [57], [110] but also include adaptation results for other GMM-derived (GMMD) features [138]. NNEmb are neural network-based embedding extractors that estimate speaker/utterance statistics from speaker-independent acoustic features. Examples of NNEmb approaches include $\star$-vector techniques, such as d-vectors [149] and x-vectors [111], discussed in Sec. V, sentence-level embeddings [59], [128], and other bottleneck approaches [130], [132]. NNTransformEmb are transformed embeddings which typically rely on i-vectors as input instead of acoustic features. These have been proposed to help alleviate issues related to inconsistent DNN adaptation performance when using raw i-vectors [57], [58], [318]. The NNTransformEmb group includes studies doing standard i-vector transformations with NNEmb [74], [147], [218] but also more recent memory-based approaches



(a) Adaptation Family (b) SAT vs. Test-only (c) Adaptation Losses

Fig. 18: Regression analysis for adaptation families, speaker-adaptive training and adaptation losses.

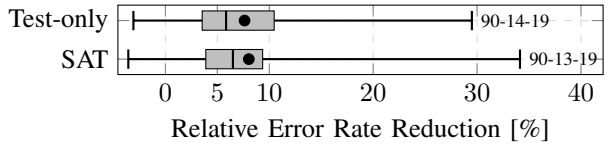


Fig. 19: Comparison of adaptation results for SAT vs Test-only modes.

in which an embedding is selected via attention from a fixed training stage embedding inventory [61], [62]. As shown in Fig. 17 GMMEmb, NNEmb and NNTransformEmb obtain 8.1%, 5.2% and 9.2% average RERR, respectively.

The second group in Fig. 17 comprises model-based approaches split into Linear Transform (LT), Activation, and Finetuning–based methods. LT methods introduce new speaker dependent affine transformations in the model, either in the form of new LIN/LHN/LON layers (*i.e.* [135], [155], [157], [289]) or transforms estimated using a GMM system such as fMLLR [56], [112], [135], [319]. Finetune refers to approaches which assume that the adaptation is carried out by altering a subset of the existing model parameters. This is often done in a similar manner to an LT approach by adapting an input, output and/or one or more hidden layers that are already present in the model [113], [141], [213], [214]. Finally, activation methods perform adaptation by introducing speaker-dependent parameters in the activation functions of the neural network [112], [162], [320]–[322]. Note that, as outlined in Sec. VI, some of activation-based methods can be expressed as constrained LT methods. The results obtained by LT, Activation and Finetune–based methods score 6.7%, 9.0% and 13.9% average RERR, respectively. Fig. 18 (a) shows the regression trends for amounts of adaptation data for each of the six considered categories.

The use of embeddings implies that the acoustic model is trained in a speaker adaptive manner, whereas the majority of model-based techniques are carried out in a test-only manner – meaning that speaker-level information is not used during training – though some methods offer SAT variants [167], [323]. Fig. 19 shows that SAT trained systems offer a small advantage (8% vs. 7.6% RERR) when adapted with limited amounts of data (up to around 10 minutes). When looking at the average performance across all data-points, however, test-only approaches obtain 10.8% RERR, primarily because of greater adaptation gains for larger amounts of data. See also Fig. 18 (b) for operating regions of SAT and non-SAT systems.

Fig. 20 quantifies gains for different adaptation objectives and regularization approaches – results for the online condition are given only for reference, as in this case adaptation information is obtained via an embedding extractor (which is usually not updated, although not always [218]). The second group depicts approaches where the adaptation information is derived by adapting a GMM in model-space using an MLE or MAP criterion when extracting speaker-adapted auxiliary features for NN training [138], [324] or by estimating fMLLR transforms with MLE under a GMM to obtain speaker adapted acoustic features [56], [135], [319].

The third group comprises methods which aim to explicitly match the model's output distribution to the one found in the adaptation data. CE is a non-regularized frame-level cross-entropy baseline obtaining 8.7% average RERR. This can be improved to 14.8% average RERR by penalizing the adapted model's predictions such that they do not deviate too much from the speaker independent variant by KL regularization (CE-KL) [114]. KL regularization can be applied to either CE or sequential objective functions [154], although most models estimated in a sequential discriminative manner can successfully be adapted with a CE (or CE-KL) criterion [75], [168], [195], [297] (see also Fig. 21). Teacher-student (T/S) [239] is a special case (see Sec. XI) where the adaptation is carried with the targets directly produced by a teacher model, rather than the ones obtained from first pass decodes (possibly KL-regularized with the SI model). T/S allows the use of large amounts of unsupervised data and in this analysis was found to offer an average 28.2% RERR when adapting to domains [229], [241], [249].

The final group in Fig. 20 includes objectives that try to

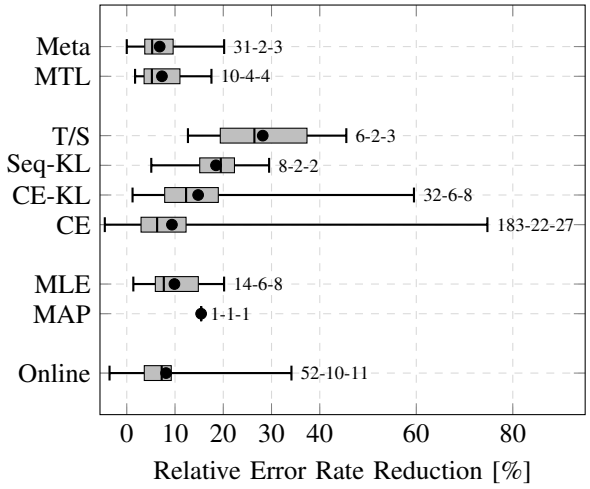


Fig. 20: Comparison of results for different adaptation loss functions.

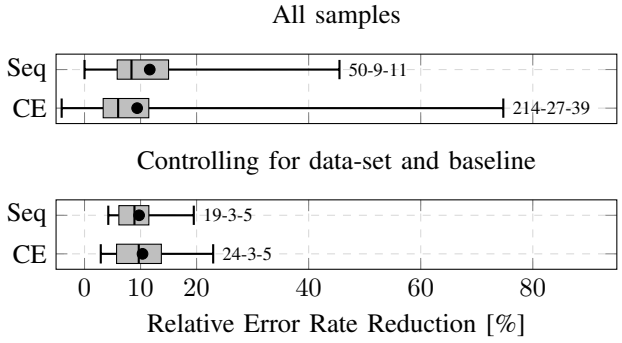


Fig. 21: Comparison of adaptation results for acoustic models trained with CE and Sequence-level objectives.

leverage auxiliary information at the objective function level. Meta-learning [178], [179], [232] estimates the adaptation hyper-parameters jointly with the adaptation transform while multi-task learning [115], [118], [187], [295] leverages additional phonetic priors to circumvent the (potential) sparsity of senones when adapting with small amounts of data. Meta-learning and multi-task adaptation obtain 6.8% and 7.6% average RERR, respectively. See also Fig. 18 (c).

Fig. 21 further summarizes the adaptability of acoustic models trained in a frame-based (CE) or a sequential (Seq) manner. The results indicate that sequential models benefit more from adaptation when compared to frame-based systems (11.6% vs. 9.8% average RERR). However, when controlling for the same dataset and baseline (reference systems were expected to exist for both CE and Seq) the difference decreases to around 0.6% RERR in favor of the frame-based systems.

Fig. 22 compares the adaptation gains obtained using various model architectures. LSTM benefits the most (15.4% average RERR). The feed-forward TDNN, DNN, and ResNet architectures all improve by around 10.5% RERR. Smaller gains were observed for Transformer, CNN and BLSTM, improving by 7.6, 6.5 and 4.9% average RERR, respectively. This result is somewhat expected as the last three architectures

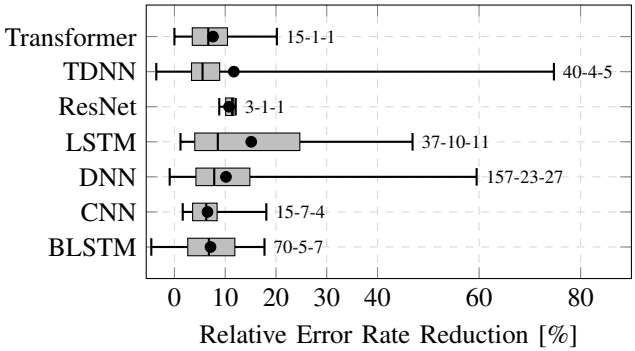


Fig. 22: Comparison of adaptation results for different architectures.

either normalize some of the variability by design, or have access to a larger speech context during recognition.

In Fig. 23 we study the complementarity of the different adaptation techniques. These results are based on 22 samples and 6 studies for which there were a complete set of baseline experiments allowing improvements to be quantified when adapting an SI model with Method1, and then measuring further gains when adding Method2. Fig. 23 shows that, on average, stacking adaptation techniques improved the adaptation performance by an additional 4%, from 8% to 12% RERR.

Finally, in Fig. 24 we report results for all techniques included in the meta-analysis. These are based on samples where only a single method was used to adapt the acoustic model (*cf.* Fig. 6 (middle)), spanning results for all adaptation clusters (*cf.* Fig. 8). These should not be directly compared owing to differences in operating regions, but they offer an indication of the performance of the individual methods.

D. Speech styles, applications, languages

In this subsection, we analyze the efficacy of adaptation methods across acoustic and linguistic dimensions by reporting adaptation gains for different types of speech styles, applications (including ones with a large mismatch to the training conditions), and languages.

Fig. 25 compares gains as obtained for different speech styles. At the top we report three special cases spanning disordered, children's, and accented speech (these are similar to the adaptation clusters from Fig. 8). As expected, acoustic models estimated largely from adult speech of healthy individuals perform poorly in these highly mismatched domains, especially for disordered and children's speech, and domain adaptation improves ASR by over 50% average RERR.

Performance gains from adapting models with accented speech are similar to that obtained on other speech tasks. Note that the presence of non-native speakers in (English) training corpora is fairly common, so the underlying acoustic models may learn to better normalize this variability at the training stage. Interestingly, adaptation brings relatively larger gains in commercial applications such as VoiceSearch and Dictation tasks (14% RERR on average). This is also visible in Fig. 26 comparing performance on public and proprietary data. We hypothesize that commercial data is more likely

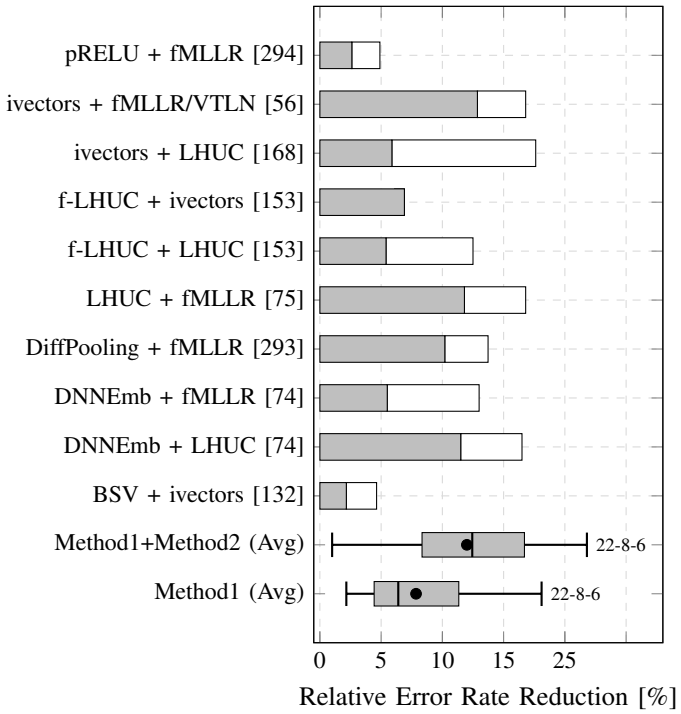


Fig. 23: Complementarity of selected adaptation techniques.

to contain a mix of speech from a diverse set of speakers (including non-native and child speech) and thus benefits more from adaptation. Another explanation could be that the public benchmarks have been around for some time, and systems built on these are likely to be more over-fitted in general.

Finally, Fig. 27 summarizes the adaptation performance for several languages. Note that speaker adaptation was performed on English, French, Japanese, and Mandarin while for Korean and Italian we only report adaptation gains for disordered and children's speech recognition. The overall improvements for non-English languages when adapting to speakers are similar to gains obtained for English when controlling for the adaptation method (*i.e.* improvements are between 6 and 10% average RERR), giving some evidence that adaptation helps to a similar degree for different languages, and that some of these primarily English-based findings generalize across languages.

XIV. Summary and discussion

The rapid developments in speech recognition over the past decade have been driven by deep neural network models of acoustics, deployed in both hybrid and E2E systems. Compared to the previous state-of-the-art approaches based on GMMs, neural network-based systems have less constrained and more flexible models and are open to a richer set of adaptation algorithms, compared to previous approaches based on linear transforms of the model parameters and acoustic features.

In this overview article we have surveyed approaches to the adaptation of neural network-based speech recognition systems. We structured the field into embedding-based, model-based, and data augmentation adaptation approaches, arguing that this organization gives a more coherent understanding of the field compared with the usual split into feature-based and model-based approaches. We presented these adaptation algorithms in the context of speaker adaptation, with a discussion on their application to accent and domain adaptation.

A key aspect of this overview was a meta-analysis of recent published results for the adaptation of speech recognition systems. The meta-analysis indicates that adaptation algorithms apply successfully to both hybrid and E2E systems, across different corpora and adaptation classes.

E2E modeling is less mature than the hybrid approach, and much of the research focus on E2E modeling is to improve the general modeling technology. Therefore, in this overview paper, many more adaptation methods were introduced in the context of hybrid systems. However, most adaptation technologies successfully applied to hybrid models by adapting acoustic model or language model should also work well for E2E models because E2E models usually contain sub-networks corresponding to the acoustic model and language model in hybrid models; this is supported by findings in our meta-analysis.

Different from hybrid models in which components are optimized separately, E2E models are optimized using a single objective function. Therefore, E2E models tend to memorize the training data more and hence the generalization or robustness to unseen data [191] is challenging to E2E models. Consequently, adaptation to new environment or new domain is very important to the large scale application of E2E models. We would expect more research toward this direction as E2E modeling becomes increasingly mainstream in ASR.

Because the size of E2E models is much smaller than that of hybrid models, E2E models have clear advantages when being deployed to device. Therefore, the personalization or adaptation of E2E models [119], [120], [126], [127] is a rapidly growing area. While it is possible to adapt every user's model in the cloud and then push it back to each device, it is more reasonable to adapt the model on device, which requires adjusting the adaptation algorithm to overcome the challenge of limited memory and computation power [119]. Another interesting direction for the adaptation of E2E models is how to leverage unpaired data especially text only data in a new domain. In [127], several methods have been explored in this direction, but we are expecting more innovations there.

Adaptation algorithms are often deployed for conditions in which there is very limited labeled data, or none at all. In this case unsupervised and semi-supervised learning approaches are central, and indeed many current adaptation approaches strongly leverage such algorithms. However there are significant open research challenges in this area, particularly relating to unsupervised and semi-supervised training of E2E systems, using methods which are able to propagate uncertainty. Current approaches often do this indirectly (*e.g.* through T/S training), but more direct modeling of uncertainty would be desirable.

Domain adaptation has become central to work in computer vision and image processing, as discussed in Sec. I, with large scale base models (typically trained on ImageNet) being adapted to specific tasks. The closest analogies to this in speech recognition are some of the domain recognition

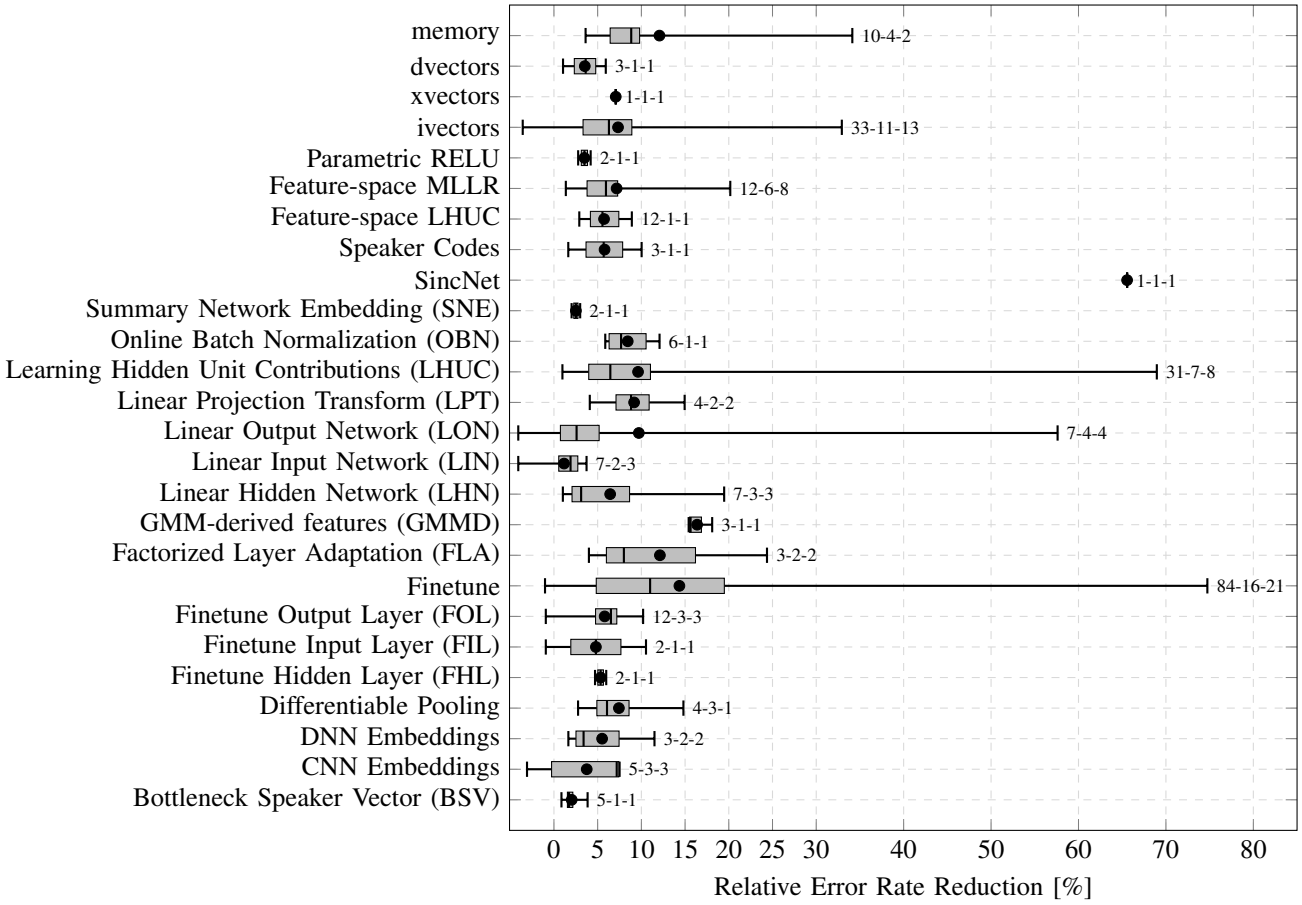


Fig. 24: Comparison of adaptation results for the standalone techniques.

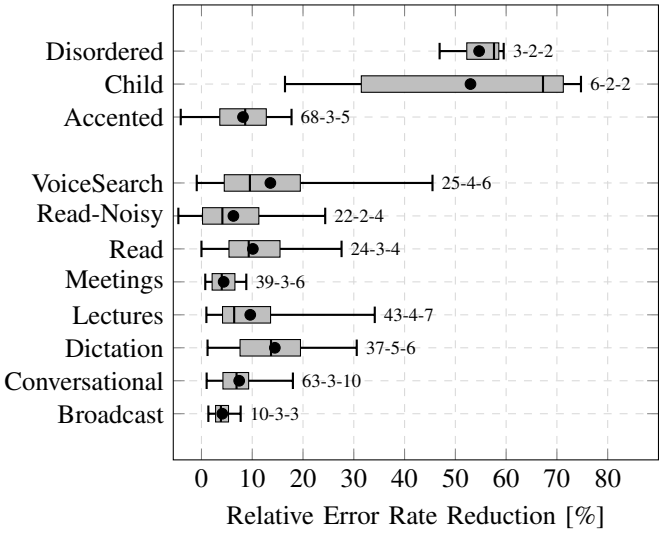


Fig. 25: Comparison of adaptation results for different speech styles

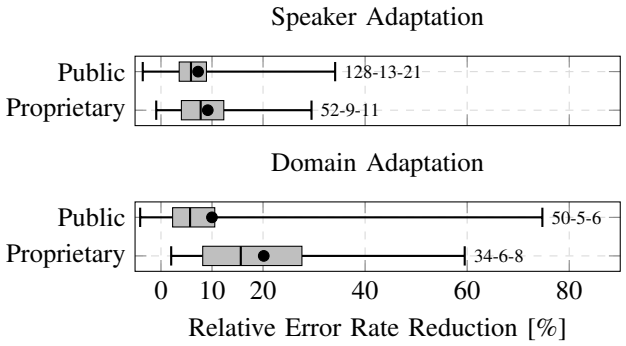


Fig. 26: Performance of adaptation techniques as obtained on public and proprietary datasets.

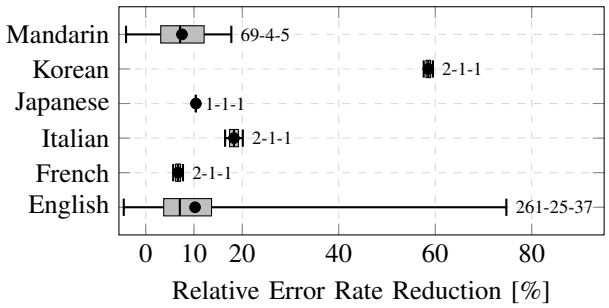


Fig. 27: Adaptation gains for different languages.

approaches discussed in Sec. XI and for multilingual speech recognition. The idea of shared multilingual representations and language-specific or language-adaptive output layers was proposed in 2013 [215]–[217] and has become a standard

architectural pattern. More recently several authors have proposed highly multilingual E2E systems, with a shared multilingual output layer [325]–[328], with the potential to be adapted to new languages.

State-of-the-art NLP systems are characterized by an unsupervised, large-scale base model [30], [42] which may then be adapted to specific domains and tasks [43]. An analogous approach for speech recognition would be based on the unsupervised learning of speech representations, from diverse and potentially multilingual speech recordings. Initial work in this direction includes the unsupervised learning from large-scale multilingual speech data [329], [330]. More generally, deep probabilistic generative modeling has become a highly active research area, in particular through approaches such as normalizing flows [50], [51], [53], [54]. Such deep generative models offer different ways of addressing the problem of adaptation including powerful approaches to data augmentation, and the development of rich adaptation algorithms building on a base model with a joint distribution over acoustics and symbols. This offers the possibility of finetuning general encoders to specific acoustic domains, and adapting the decoder to specific tasks (such as speech recognition, speaker identification, language recognition, or emotion recognition), noting that classic adaptation to speakers can bring further gains [331], [332].

REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012.

[2] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Interspeech*, 2011.

[3] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," in *Interspeech*, 2017.

[4] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani, "State-of-the-art speech recognition with sequence-to-sequence models," in *IEEE ICASSP*, 2018.

[5] N. Morgan and H. A. Bourlard, "Neural networks for statistical recognition of continuous speech," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 742–772, 1995.

[6] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *IEEE ICASSP*, 2016, pp. 4960–4964.

[7] T. Robinson, M. Hochberg, and S. Renals, "The use of recurrent neural networks in continuous speech recognition," in *Automatic Speech and Speaker Recognition*, C.-H. Lee, F. K. Soong, and K. K. Paliwal, Eds. Kluwer, 1996, pp. 233–258.

[8] A. J. Robinson, G. D. Cook, D. P. W. Ellis, E. Fosler-Lussier, S. J. Renals, and D. A. G. Williams, "Connectionist speech recognition of broadcast news," *Speech Communication*, vol. 37, pp. 27–45, 2002.

[9] D. J. Kershaw, A. J. Robinson, and M. Hochberg, "Context-dependent classes in a hybrid recurrent network-HMM speech recognition system," in *Advances in Neural Information Processing Systems*, 1996.

[10] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, 1989.

[11] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts." in *Interspeech*, 2015.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[13] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

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

[15] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *IEEE ICASSP*, 2013.

[16] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[17] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *IEEE ASRU*, 2013.

[18] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015, pp. 577–585.

[19] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *IEEE ASRU*, 2015, pp. 167–174.

[20] L. Lu, X. Zhang, K. Cho, and S. Renals, "A study of the recurrent neural network encoder-decoder for large vocabulary speech recognition," in *Interspeech*, 2015.

[21] K. Rao, H. Sak, and R. Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer," in *IEEE ASRU*, 2017, pp. 193–199.

[22] E. Battenberg, J. Chen, R. Child, A. Coates, Y. G. Y. Li, H. Liu, S. Satheesh, A. Sriram, and Z. Zhu, "Exploring neural transducers for end-to-end speech recognition," in *IEEE ASRU*, 2017, pp. 206–213.

[23] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *IEEE ICASSP*, 2019.

[24] J. Li, R. Zhao, H. Hu, and Y. Gong, "Improving RNN transducer modeling for end-to-end speech recognition," in *IEEE ASRU*, 2019.

[25] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[26] A. Hannun, "Sequence modeling with CTC," *Distill*, 2017, https://distill.pub/2017/ctc.

[27] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv:1211.3711*, 2012.

[28] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[29] J. Li, Y. Wu, Y. Gaur, C. Wang, R. Zhao, and S. Liu, "On the comparison of popular end-to-end models for large scale speech recognition," in *Interspeech*, 2020.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[31] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *IEEE ICASSP*, 2018, pp. 5884–5888.

[32] Y. Zhao, J. Li, X. Wang, and Y. Li, "The speechtransformer for large-scale mandarin chinese speech recognition," in *IEEE ICASSP*, 2019, pp. 7095–7099.

[33] S. Karita, N. E. Y. Soplin, S. Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani, "Improving transformer based end-to-end speech recognition with connectionist temporal classification and language model integration," in *Interspeech*, 2019.

[34] C.-F. Yeh, J. Mahadeokar, K. Kalgaonkar, Y. Wang, D. Le, M. Jain, K. Schubert, C. Fuegen, and M. L. Seltzer, "Transformer-transducer: End-to-end speech recognition with self-attention," *arXiv preprint arXiv:1910.12977*, 2019.

[35] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and RNN-T loss," in *IEEE ICASSP*, 2020, pp. 7829–7833.

[36] X. Chen, Y. Wu, Z. Wang, S. Liu, and J. Li, "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," *arXiv preprint arXiv:2010.11395*, 2020.

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *CVPR*, 2009.

[38] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[39] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[40] S. Kornblith, J. Shlens, and Q. V. Le, "Do better ImageNet models transfer better?" in *CVPR*, 2019.

[41] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *NAACL/HLT*, 2018, pp. 2227–2237.

[42] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL/HLT*, 2019, pp. 4171–4186.

[43] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[44] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil, "Universal sentence encoder for English," in *EMNLP*, 2018, pp. 169–174.

[45] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for NLP," in *ICML*, 2019, pp. 2790–2799.

[46] W. M. Kouw and M. Loog, "A review of domain adaptation without target labels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019, early access.

[47] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *ICLR*, 2016.

[48] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 4077–4087.

[49] X. Li, S. Dalmia, D. R. Mortensen, J. Li, A. W. Black, and F. Metze, "Towards zero-shot learning for automatic phonemic transcription." in *AAAI*, 2020, pp. 8261–8268.

[50] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *ICML*, 2015, pp. 1530–1538.

[51] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *arXiv:1912.02762*, 2019.

[52] S. S. Chen and R. A. Gopinath, "Gaussianization," in *Advances in Neural Information Processing Systems*, 2001, pp. 423–429.

[53] R. Prenger, R. Valle, and B. Catanzaro, "WaveGlow: A flow-based generative network for speech synthesis," in *IEEE ICASSP*, 2019, pp. 3617–3621.

[54] J. Serrà, S. Pascual, and C. S. Perales, "Blow: a single-scale hyperconditioned flow for non-parallel raw-audio voice conversion," in *Advances in Neural Information Processing Systems*, 2019, pp. 6793–6803.

[55] S. Tan and K. C. Sim, "Learning utterance-level normalisation using variational autoencoders for robust automatic speech recognition," in *IEEE SLT*, 2016, pp. 43–49.

[56] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *IEEE ASRU*, 2013.

[57] A. Senior and I. Lopez-Moreno, "Improving DNN speaker independence with i-vector inputs," in *IEEE ICASSP*, 2014, pp. 225–229.

[58] P. Karanasou, Y. Wang, M. J. Gales, and P. C. Woodland, "Adaptation of deep neural network acoustic models using factorised i-vectors," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[59] K. Veselý, S. Watanabe, K. Žmolíková, M. Karafiát, L. Burget, and J. H. Černocký, "Sequence summarizing neural network for speaker adaptation," in *IEEE ICASSP*, 2016.

[60] M. Doulaty, O. Saz, R. W. M. Ng, and T. Hain, "Latent Dirichlet allocation based organisation of broadcast media archives for deep neural network adaptation," in *IEEE ASRU*, 2015, pp. 130–136.

[61] J. Pan, G. Wan, J. Du, and Z. Ye, "Online speaker adaptation using memory-aware networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1025–1037, 2020.

[62] L. Sarı, N. Moritz, T. Hori, and J. Le Roux, "Unsupervised speaker adaptation using attention-based speaker memory for end-to-end ASR," in *IEEE ICASSP*, 2020, pp. 7384–7388.

[63] Z.-P. Zhang, S. Furui, and K. Ohtsuki, "On-line incremental speaker adaptation with automatic speaker change detection," in *IEEE ICASSP*, 2000, pp. II.961–II.964.

[64] H. Huang and K. C. Sim, "An investigation of augmenting speaker representations to improve speaker normalisation for DNN-based speech recognition," in *IEEE ICASSP*, 2015.

[65] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[66] M. J. Gales, "Cluster adaptive training of hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 417–428, 2000.

[67] T. Tan, Y. Qian, and K. Yu, "Cluster adaptive training for deep neural network based acoustic model," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 3, pp. 459–468, 2016.

[68] H. Christensen, S. Cunningham, C. Fox, P. Green, and T. Hain, "A comparative study of adaptive, automatic recognition of disordered speech," in *Interspeech*, 2012.

[69] H. Liao, E. McDermott, and A. Senior, "Large scale deep neural network acoustic modeling with semi-supervised training data for YouTube video transcription," in *IEEE ASRU*, 2013, pp. 368–373.

[70] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation," in *IEEE ASRU*, 2017, pp. 16–23.

[71] L. Mathias, G. Yegnanarayanan, and J. Fritsch, "Discriminative training of acoustic models applied to domains with unreliable transcripts [speech recognition applications]," in *IEEE ICASSP*, 2005.

[72] S.-H. Liu, F.-H. Chu, S.-H. Lin, and B. Chen, "Investigating data selection for minimum phone error training of acoustic models," in *IEEE ICME*, 2007.

[73] S. Walker, M. Pedersen, I. Orife, and J. Flaks, "Semi-supervised model training for unbounded conversational speech recognition," *arXiv:1705.09724*, 2017.

[74] Y. Miao, H. Zhang, and F. Metze, "Speaker adaptive training of deep neural network acoustic models using i-vectors," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 11, pp. 1938–1949, 2015.

[75] P. Swietojanski, J. Li, and S. Renals, "Learning hidden unit contributions for unsupervised acoustic model adaptation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 1450–1463, 2016.

[76] M. Padmanabhan, G. Saon, and G. Zweig, "Lattice-based unsupervised MLLR for speaker adaptation," in *ISCA ASR2000 Workshop*, 2000.

[77] T. Fraga-Silva, J.-L. Gauvain, and L. Lamel, "Lattice-based unsupervised acoustic model training," in *IEEE ICASSP*, 2011.

[78] V. Manohar, H. Hadian, D. Povey, and S. Khudanpur, "Semi-supervised training of acoustic models using lattice-free MMI," in *IEEE ICASSP*, 2018.

[79] O. Klejch, J. Fainberg, P. Bell, and S. Renals, "Lattice-based unsupervised test-time adaptation of neural network acoustic models," *arXiv:1906.11521*, 2019.

[80] H. Suzuki, H. Kasuya, and K. Kido, "The acoustic parameters for vowel recognition without distinction of speakers," in *Proc. 1967 Conf. Speech Comm. and Process*, 1967, pp. 92–96.

[81] L. Gerstman, "Classification of self-normalized vowels," *IEEE Transactions on Audio and Electroacoustics*, vol. 16, no. 1, pp. 78–80, 1968.

[82] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171–185, 1995.

[83] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[84] P. C. Woodland, "Speaker adaptation for continuous density HMMs: A review," in *ISCA Workshop on Adaptation Methods for Speech Recognition*, 2001.

[85] K. Shinoda, "Speaker adaptation techniques for automatic speech recognition," *APSIPA ASC*, 2011.

[86] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," *Foundations and Trends in Signal Processing*, vol. 1, no. 3, pp. 195–304, 2008.

[87] K. Johnson, "Speaker normalization in speech perception," in *The Handbook of Speech Perception*. Wiley Online Library, 2005, pp. 363–389.

[88] K. Paliwal and W. Ainsworth, "Dynamic frequency warping for speaker adaptation in automatic speech recognition," *Journal of Phonetics*, vol. 13, no. 2, pp. 123 – 134, 1985.

[89] Y. Grenier, "Speaker adaptation through canonical correlation analysis," in *IEEE ICASSP*, vol. 5, 1980, pp. 888–891.

[90] K. Choukri and G. Chollet, "Adaptation of automatic speech recognizers to new speakers using canonical correlation analysis techniques," *Computer Speech & Language*, vol. 1, no. 2, pp. 95–107, 1986.

[91] H. Wakita, "Normalization of vowels by vocal-tract length and its application to vowel identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 2, pp. 183–192, 1977.

[92] A. Andreou, "Experiments in vocal tract normalization," in *CAIP Workshop: Frontiers in Speech Recognition II*, 1994.

[93] E. Eide and H. Gish, "A parametric approach to vocal tract length normalization," in *Interspeech*, 1996.

[94] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *IEEE ICASSP*, 1996.

[95] D. Kim, S. Umesh, M. Gales, T. Hain, and P. Woodland, "Using VTLN for broadcast news transcription," in *ICSLP*, 2004.

[96] G. Garau, S. Renals, and T. Hain, "Applying vocal tract length normalization to meeting recordings," in *Interspeech*, 2005.

[97] S. Furui, "A training procedure for isolated word recognition systems," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 129–136, 1980.

[98] K. Shikano, S. Nakamura, and M. Abe, "Speaker adaptation and voice conversion by codebook mapping," in *IEEE International Symposium on Circuits and Systems*, 1991, pp. 594–597.

[99] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *Journal of the Acoustical Society of Japan (E)*, vol. 11, no. 2, pp. 71–76, 1990.

[100] M. Feng, F. Kubala, R. Schwartz, and J. Makhoul, "Improved speaker adaption using text dependent spectral mappings," in *IEEE ICASSP*, vol. 1, 1988, pp. 131–134.

[101] G. Rigoll, "Speaker adaptation for large vocabulary speech recognition systems using speaker markov models," in *IEEE ICASSP*, 1989, pp. 5–8.

[102] M. J. Hunt, "Speaker adaptation for word-based speech recognition systems," *The Journal of the Acoustical Society of America*, vol. 69, no. S1, pp. S41–S42, 1981.

[103] S. J. Cox and J. S. Bridle, "Unsupervised speaker adaptation by probabilistic spectrum fitting," in *IEEE ICASSP*, vol. 1, 1989, pp. 294–297.

[104] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[105] L. Neumeyer, A. Sankar, and V. Digalakis, "A comparative study of speaker adaptation techniques," in *Eurospeech*, 1995.

[106] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *ICSLP*, 1996.

[107] R. Kuhn, J.-C. Junqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, pp. 695–707, 2000.

[108] K. Yu and M. J. Gales, "Discriminative cluster adaptive training," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1694–1703, 2006.

[109] K. C. Sim, Y. Qian, G. Mantena, L. Samarakoon, S. Kundu, and T. Tan, "Adaptation of deep neural network acoustic models for robust automatic speech recognition," in *New Era for Robust Speech Recognition: Exploiting Deep Learning*, S. Watanabe, M. Delcroix, F. Metze, and J. R. Hershey, Eds. Springer, 2017, pp. 219–243.

[110] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[111] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *IEEE ICASSP*, 2018.

[112] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *IEEE SLT*, 2014.

[113] H. Liao, "Speaker adaptation of context dependent deep neural networks," in *IEEE ICASSP*, 2013.

[114] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *IEEE ICASSP*, 2013.

[115] Z. Huang, J. Li, S. M. Siniscalchi, I.-F. Chen, J. Wu, and C.-H. Lee, "Rapid adaptation for deep neural networks through multi-task learning," in *Interspeech*, 2015.

[116] Y. Huang, L. He, W. Wei, W. Gale, J. Li, and Y. Gong, "Using personalized speech synthesis and neural language generator for rapid speaker adaptation," in *IEEE ICASSP*, 2020.

[117] K. Li, J. Li, Y. Zhao, K. Kumar, and Y. Gong, "Speaker adaptation for end-to-end CTC models," in *IEEE SLT*, 2018, pp. 542–549.

[118] Z. Meng, Y. Gaur, J. Li, and Y. Gong, "Speaker adaptation for attention-based end-to-end speech recognition," in *Interspeech*, 2019.

[119] K. C. Sim, P. Zadrazil, and F. Beaufays, "An investigation into on-device personalization of end-to-end automatic speech recognition models," in *Interspeech*, 2019.

[120] Y. Huang, J. Li, L. He, W. Wei, W. Gale, and Y. Gong, "Rapid RNN-T adaptation using personalised speech synthesis and neural language generator," in *Interspeech*, 2020.

[121] Z. Fan, J. Li, S. Zhou, and B. Xu, "Speaker-aware speech-transformer," in *IEEE ASRU*, 2019, pp. 222–229.

[122] Y. Zhao, C. Ni, C.-C. Leung, S. Joty, E. S. Chng, and B. Ma, "Speech transformer with speaker aware persistent memory," *Interspeech*, pp. 1261–1265, 2020.

[123] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao, "Deep context: end-to-end contextual speech recognition," in *IEEE SLT*, 2018, pp. 418–425.

[124] Z. Chen, M. Jain, Y. Wang, M. L. Seltzer, and C. Fuegen, "End-to-end contextual speech recognition using class language models and a token passing decoder," in *IEEE ICASSP*, 2019, pp. 6186–6190.

[125] M. Jain, G. Keren, J. Mahadeokar, and Y. Saraf, "Contextual RNN-T for open domain ASR," *arXiv:2006.03411*, 2020.

[126] K. C. Sim, F. Beaufays, A. Benard *et al.*, "Personalization of end-to-end speech recognition on mobile devices for named entities," *arXiv:1912.09251*, 2019.

[127] J. Li, R. Zhao, Z. Meng, Y. Liu, W. Wei, P. Parthasarathy, V. Mazalov, Z. Wang, L. He, S. Zhao, and Y. Gong, "Developing RNN-T models surpassing high-performance hybrid models with customization capability," in *Interspeech*, 2020.

[128] M. Delcroix, S. Watanabe, A. Ogawa, S. Karita, and T. Nakatani, "Auxiliary feature based adaptation of end-to-end ASR systems," in *Interspeech*, 2018.

[129] M. Delcroix, K. Kinoshita, A. Ogawa, C. Huemmer, and T. Nakatani, "Context adaptive neural network based acoustic models for rapid adaptation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 5, pp. 895–908, 2018.

[130] J. Rownicka, P. Bell, and S. Renals, "Embeddings for DNN speaker adaptive training," *IEEE ASRU*, 2019.

[131] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, "Robust i-vector based adaptation of DNN acoustic model for speech recognition," in *Interspeech*, 2015.

[132] T. Tan, Y. Qian, D. Yu, S. Kundu, L. Lu, K. C. Sim, X. Xiao, and Y. Zhang, "Speaker-aware training of LSTM-RNNs for acoustic modelling," in *IEEE ICASSP*, 2016, pp. 5280–5284.

[133] S. H. K. Parthasarathi, B. Hoffmeister, S. Matsoukas, A. Mandal, N. Strom, and S. Garimella, "fMLLR based feature-space speaker adaptation of DNN acoustic models," in *Interspeech*, 2015.

[134] Z. Meng, H. Hu, J. Li, C. Liu, Y. Huang, Y. Gong, and C.-H. Lee, "L-vector: Neural label embedding for domain adaptation," in *IEEE ICASSP*, 2020, pp. 7389–7393.

[135] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *IEEE ASRU*, 2011, pp. 24–29.

[136] S. P. Rath, D. Povey, K. Veselỳ, and J. Cernockỳ, "Improved feature processing for deep neural networks," in *Interspeech*, 2013, pp. 109–113.

[137] N. M. Joy, M. K. Baskar, S. Umesh, and B. Abraham, "DNNs for unsupervised extraction of pseudo FMLLR features without explicit adaptation data," in *Interspeech*, 2016, pp. 3479–3483.

[138] N. Tomashenko and Y. Khokhlov, "GMM-derived features for effective unsupervised adaptation of deep neural network acoustic models," in *Interspeech*, 2015.

[139] L. F. Uebel and P. C. Woodland, "An investigation into vocal tract length normalisation," in *Eurospeech*, 1999.

[140] D. Povey, G. Zweig, and A. Acero, "Speaker adaptation with an exponential transform," in *IEEE ASRU*, 2011, pp. 158–163.

[141] J. Fainberg, O. Klejch, E. Loweimi, P. Bell, and S. Renals, "Acoustic model adaptation from raw waveforms with SincNet," in *IEEE ASRU*, 2019.

[142] M. Karafiát, L. Burget, P. Matějka, O. Glembek, and J. Černockỳ, "iVector-based discriminative adaptation for automatic speech recognition," in *IEEE ASRU*, 2011, pp. 152–157.

[143] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Speaker and session variability in GMM-based speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1448–1460, 2007.

[144] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *IEEE ICASSP*, 2014, pp. 1695–1699.

[145] P. Kenny, T. Stafylakis, P. Ouellet, V. Gupta, and M. J. Alam, "Deep neural networks for extracting Baum-Welch statistics for speaker recognition." in *Speaker Odyssey*, vol. 2014, 2014, pp. 293–298.

[146] Y. Miao, H. Zhang, and F. Metze, "Towards speaker adaptive training of deep neural network acoustic models," in *Interspeech*, 2014.

[147] Y. Miao, L. Jiang, H. Zhang, and F. Metze, "Improvements to speaker adaptive training of deep neural networks," in *IEEE SLT*, 2014, pp. 165–170.

[148] J. Rownicka, P. Bell, and S. Renals, "Analyzing deep CNN-based utterance embeddings for acoustic model adaptation," in *IEEE SLT*, 2018, pp. 235–241.

[149] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *IEEE ICASSP*, 2014.

[150] X. Li and X. Wu, "Modeling speaker variability using long short-term memory networks for speech recognition," in *Interspeech*, 2015.

[151] Y. Khokhlov, A. Zatvornitskiy, I. Medennikov, I. Sorokin, T. Prisyach, A. Romanenko, A. Mitrofanov, V. Bataev, A. Andrusenko, M. Korenevskaya *et al.*, "R-vectors: New technique for adaptation to room acoustics," *Interspeech*, 2019.

[152] Y. Shi, Q. Huang, and T. Hain, "H-vectors: Utterance-level speaker embedding using a hierarchical attention model," in *IEEE ICASSP*, 2020, pp. 7579–7583.

[153] X. Xie, X. Liu, T. Lee, and L. Wang, "Fast DNN acoustic model speaker adaptation by learning hidden unit contribution features," in *Interspeech*, 2019, pp. 759–763.

[154] Y. Huang and Y. Gong, "Regularized sequence-level deep neural network model adaptation," in *Interspeech*, 2015.

[155] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, S. Renals, and T. Robinson, "Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system," *Eurospeech*, 1995.

[156] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. De Mori, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," *Speech Communication*, vol. 49, no. 10-11, pp. 827–835, 2007.

[157] B. Li and K. C. Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid nn/hmm systems," in *Interspeech*, 2010.

[158] J. S. Bridle and S. J. Cox, "RecNorm: Simultaneous normalisation and classification applied to speech recognition," in *Advances in Neural Information Processing Systems*, 1991, pp. 234–240.

[159] O. Abdel-Hamid and H. Jiang, "Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code," in *IEEE ICASSP*, 2013.

[160] Z. Q. Wang and D. Wang, "Unsupervised speaker adaptation of batch normalized acoustic models for robust ASR," in *IEEE ICASSP*, 2017.

[161] F. Mana, F. Weninger, R. Gemello, and P. Zhan, "Online batch normalization adaptation for automatic speech recognition," in *ASRU*, 2019.

[162] C. Zhang and P. C. Woodland, "Parameterised sigmoid and ReLU hidden activation functions for DNN acoustic modelling," in *Interspeech*, 2015.

[163] Y. Zhao, J. Li, and Y. Gong, "Low-rank plus diagonal adaptation for deep neural networks," in *IEEE ICASSP*, 2016.

[164] Y. Zhao, J. Li, K. Kumar, and Y. Gong, "Extended low-rank plus diagonal adaptation for deep and recurrent neural networks," in *IEEE ICASSP*, 2017.

[165] V. Abrash, H. Franco, A. Sankar, and M. Cohen, "Connectionist speaker normalization and adaptation," in *Eurospeech*, 1995.

[166] K. Yao, D. Yu, F. Seide, H. Su, L. Deng, and Y. Gong, "Adaptation of context-dependent deep neural networks for automatic speech recognition," in *IEEE SLT*, 2012.

[167] L. Samarakoon and K. C. Sim, "Learning factorized feature transforms for speaker normalization," in *IEEE ASRU*, 2015.

[168] ——, "Factorized hidden layer adaptation for deep neural network based acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2241–2250, 2016.

[169] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[170] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "FiLM: Visual reasoning with a general conditioning layer," in *AAAI*, 2018.

[171] L. Samarakoon and K. C. Sim, "Subspace LHUC for fast adaptation of deep neural network acoustic models." in *Interspeech*, 2016.

[172] X. Cui, V. Goel, and G. Saon, "Embedding-based speaker adaptive training of deep neural networks," in *Interspeech*, 2017.

[173] T. Kim, I. Song, and Y. Bengio, "Dynamic layer normalization for adaptive neural acoustic modeling in speech recognition," in *Interspeech*, 2017.

[174] L. Sarı, S. Thomas, M. Hasegawa-Johnson, and M. Picheny, "Speaker adaptation of neural networks with learning speaker aware offsets," in *Interspeech*, 2019.

[175] X. Xie, X. Liu, T. Lee, and L. Wang, "Fast DNN acoustic model speaker adaptation by learning hidden unit contribution features," *Interspeech*, 2019.

[176] J. Xue, J. Li, and Y. Gong, "Restructuring of deep neural network acoustic models with singular value decomposition." in *Interspeech*, 2013, pp. 2365–2369.

[177] J. Xue, J. Li, D. Yu, M. Seltzer, and Y. Gong, "Singular value decomposition based low-footprint speaker adaptation and personalization for deep neural network," in *IEEE ICASSP*, 2014.

[178] O. Klejch, J. Fainberg, and P. Bell, "Learning to adapt: a meta-learning approach for speaker adaptation," in *Interspeech*, 2018.

[179] O. Klejch, J. Fainberg, P. Bell, and S. Renals, "Speaker adaptive training using model agnostic meta-learning," in *IEEE ASRU*, 2019.

[180] F. Weninger, J. Andrés-Ferrer, X. Li, and P. Zhan, "Listen, attend, spell and adapt: Speaker adapted sequence-to-sequence ASR," in *Interspeech*, 2019.

[181] Z. Meng, J. Li, and Y. Gong, "Adversarial speaker adaptation," in *IEEE ICASSP*, 2019.

[182] Z. Huang, S. M. Siniscalchi, I.-F. Chen, J. Wu, and C.-H. Lee, "Maximum a posteriori adaptation of network parameters in deep models," *arXiv:1503.02108*, 2015.

[183] X. Xie, X. Liu, T. Lee, S. Hu, and L. Wang, "BLHUC: Bayesian learning of hidden unit contributions for deep neural network speaker adaptation," in *IEEE ICASSP*, 2019.

[184] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, University of Cambridge, 2005.

[185] R. Price, K.-i. Iso, and K. Shinoda, "Speaker adaptation of deep neural networks using a hierarchy of output layers," in *IEEE SLT*, 2014.

[186] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[187] P. Swietojanski, P. Bell, and S. Renals, "Structured output layer with auxiliary targets for context-dependent acoustic modelling," in *Interspeech*, 2015.

[188] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[189] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *IEEE ICASSP*, 2017, pp. 5220–5224.

[190] C. Kim, A. Misra, K. Chin, T. Hughes, A. Narayanan, T. Sainath, and M. Bacchiani, "Generation of large-scale simulated utterances in virtual rooms to train deep-neural networks for far-field speech recognition in Google Home," in *Interspeech*, 2017.

[191] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 745–777, 2014.

[192] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (VTLP) improves speech recognition," in *ICML Workshop on Deep Learning for Audio, Speech and Language*, 2013.

[193] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1469–1477, 2015.

[194] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Interspeech*, 2015.

[195] Y. Huang and Y. Gong, "Acoustic model adaptation for presentation transcription and intelligent meeting assistant systems," in *IEEE ICASSP*, 2020.

[196] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.

[197] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep convolutional neural network acoustic modeling," in *IEEE ICASSP*. IEEE, 2015, pp. 4545–4549.

[198] J. Fainberg, P. Bell, M. Lincoln, and S. Renals, "Improving children's speech recognition through out-of-domain data augmentation." in *Interspeech*, 2016, pp. 1598–1602.

[199] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in Neural Information Processing Systems*, 2018, pp. 4480–4490.

[200] E. Hosseini-Asl, Y. Zhou, C. Xiong, and R. Socher, "Augmented cyclic adversarial learning for low resource domain adaptation," in *ICLR*, 2019.

[201] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE ICCV*, 2017.

[202] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. Bell, and S. Renals, "Automatic dialect detection in Arabic broadcast speech," in *Interspeech*, 2016.

[203] A. Ali, S. Vogel, and S. Renals, "Speech recognition challenge in the wild: Arabic MGB-3," in *IEEE ASRU*, 2017.

[204] A. Ali, S. Shon, Y. Samih, H. Mubarak, A. Abdelali, J. Glass, S. Renals, and K. Choukri, "The MGB-5 challenge: Recognition and dialect identification of dialectal Arabic speech," in *IEEE ASRU*, 2019.

[205] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," in *Interspeech*, 2016.

[206] P. Smit, S. R. Gangireddy, S. Enarvi, S. Virpioja, and M. Kurimo, "Aalto system for the 2017 Arabic multi-genre broadcast challenge," in *IEEE ASRU*, 2017.

[207] S. Khurana, A. Ali, and J. Glass, "Darts: Dialectal arabic transcription system," *arXiv:1909.12163*, 2019.

[208] D. Vergyri, L. Lamel, and J.-L. Gauvain, "Automatic speech recognition of multiple accented English data," in *Interspeech*, 2010.

[209] Y. Zheng, R. Sproat, L. Gu, I. Shafran, H. Zhou, Y. Su, D. Jurafsky, R. Starr, and S.-Y. Yoon, "Accent detection and speech recognition for Shanghai-accented Mandarin," in *Interspeech*, 2005.

[210] L. W. Kat and P. Fung, "Fast accent identification and accented speech recognition," in *IEEE ICASSP*, vol. 1, 1999, pp. 221–224.

[211] M. Liu, B. Xu, T. Hunng, Y. Deng, and C. Li, "Mandarin accent adaptation based on context-independent/context-dependent pronunciation modeling," in *IEEE ICASSP*, vol. 2, 2000, pp. II1025–II1028.

[212] U. Nallasamy, F. Metze, and T. Schultz, "Active learning for accent adaptation in automatic speech recognition," in *IEEE SLT*, 2012.

[213] Y. Huang, D. Yu, C. Liu, and Y. Gong, "Multi-accent deep neural network acoustic model with accent-specific top layer using the KLD-regularized model adaptation," in *Interspeech*, 2014.

[214] M. Chen, Z. Yang, J. Liang, Y. Li, and W. Liu, "Improving deep neural networks based multi-accent Mandarin speech recognition using i-vectors and accent-specific top layer," in *Interspeech*, 2015.

[215] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in *IEEE ICASSP*, 2013, pp. 7319–7323.

[216] G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, "Multilingual acoustic models using distributed deep neural networks," in *IEEE ICASSP*, 2013, pp. 8619–8623.

[217] J.-T. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *IEEE ICASSP*, 2013, pp. 7304–7308.

[218] J. Yi, H. Ni, Z. Wen, and J. Tao, "Improving BLSTM RNN based Mandarin speech recognition using accent dependent bottleneck features," in *IEEE APSIPA*, 2016, pp. 1–5.

[219] M. T. Turan, E. Vincent, and D. Jouvet, "Achieving multi-accent ASR via unsupervised acoustic model adaptation," in *Interspeech*, 2020.

[220] M. Elfeky, M. Bastani, X. Velez, P. Moreno, and A. Waters, "Towards acoustic model unification across dialects," in *IEEE SLT*, 2016.

[221] X. Yang, K. Audhkhasi, A. Rosenberg, S. Thomas, B. Ramabhadran, and M. Hasegawa-Johnson, "Joint modeling of accents and acoustics for multi-accent speech recognition," in *IEEE ICASSP*, 2018.

[222] A. Jain, M. Upreti, and P. Jyothi, "Improved accented speech recognition using accent embeddings and multi-task learning," in *Interspeech*, 2018.

[223] K. Li, J. Li, Y. Zhao, K. Kumar, and Y. Gong, "Speaker adaptation for end-to-end CTC models," in *IEEE SLT*, 2018.

[224] T. Viglino, P. Motlicek, and M. Cernak, "End-to-end accented speech recognition," in *Interspeech*, 2019.

[225] S. Sun, C.-F. Yeh, M.-Y. Hwang, M. Ostendorf, and L. Xie, "Domain adversarial training for accented speech recognition," in *IEEE ICASSP*, 2018, pp. 4854–4858.

[226] B. Li, T. N. Sainath, K. C. Sim, M. Bacchiani, E. Weinstein, P. Nguyen, Z. Chen, Y. Wu, and K. Rao, "Multi-dialect speech recognition with a single sequence-to-sequence model," in *IEEE ICASSP*, 2018.

[227] M. Grace, M. Bastani, and E. Weinstein, "Occam's adaptation: A comparison of interpolation of bases adaptation methods for multi-dialect acoustic modeling with LSTMs," in *IEEE SLT*, 2018.

[228] S. Yoo, I. Song, and Y. Bengio, "A highly adaptive acoustic model for accurate multi-dialect speech recognition," in *IEEE ICASSP*, 2019, pp. 5716–5720.

[229] S. Ghorbani, A. E. Bulut, and J. H. Hansen, "Advancing multi-accented LSTM-CTC speech recognition using a domain specific student-teacher learning paradigm," in *IEEE SLT*, 2018, pp. 29–35.

[230] A. Jain, V. P. Singh, and S. P. Rath, "A multi-accent acoustic model using mixture of experts for speech recognition," in *Interspeech*, 2019.

[231] H. Zhu, L. Wang, P. Zhang, and Y. Yan, "Multi-accent adaptation based on gate mechanism," in *Interspeech*, 2019.

[232] G. I. Winata, S. Cahyawijaya, Z. Liu, Z. Lin, A. Madotto, P. Xu, and P. Fung, "Learning fast adaptation on cross-accented speech recognition," *arXiv:2003.01901*, 2020.

[233] Y. Long, Y. Li, H. Ye, and H. Mao, "Domain adaptation of lattice-free MMI based TDNN models for speech recognition," *International Journal of Speech Technology*, 2017.

[234] J. Fainberg, S. Renals, and P. Bell, "Factorised representations for neural network adaptation to diverse acoustic environments." in *Interspeech*, 2017.

[235] K. C. Sim, A. Narayanan, A. Misra, A. Tripathi, G. Pundak, T. N. Sainath, P. Haghani, B. Li, and M. Bacchiani, "Domain adaptation using factorized hidden layer for robust automatic speech recognition." in *Interspeech*, 2018, pp. 892–896.

[236] J. Huang, O. Kuchaiev, P. O'Neill, V. Lavrukhin, J. Li, A. Flores, G. Kucsko, and B. Ginsburg, "Cross-language transfer learning, continuous learning, and domain adaptation for end-to-end automatic speech recognition," *arXiv preprint arXiv:2005.04290*, 2020.

[237] S. Ueno, T. Moriya, M. Mimura, S. Sakai, Y. Shinohara, Y. Yamaguchi, Y. Aono, and T. Kawahara, "Encoder transfer for attention-based acoustic-to-word speech recognition." in *Interspeech*, 2018, pp. 2424–2428.

[238] T. Moriya, R. Masumura, T. Asami, Y. Shinohara, M. Delcroix, Y. Yamaguchi, and Y. Aono, "Progressive neural network-based knowledge transfer in acoustic models," in *IEEE APSIPA ASC*, 2018, pp. 998–1002.

[239] J. Li, R. Zhao, J.-T. Huang, and Y. Gong, "Learning small-size DNN with output-distribution-based criteria." in *Interspeech*, 2014.

[240] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv:1503.02531*, 2015.

[241] J. Li, R. Zhao, Z. Chen *et al.*, "Developing far-field speaker system via teacher-student learning," in *IEEE ICASSP*, 2018.

[242] L. Mošner, M. Wu *et al.*, "Improving noise robustness of automatic speech recognition via parallel data and teacher-student learning," in *IEEE ICASSP*, 2019, pp. 6475–6479.

[243] T. Asami, R. Masumura, Y. Yamaguchi, H. Masataki, and Y. Aono, "Domain adaptation of DNN acoustic models using knowledge distillation," in *IEEE ICASSP*, 2017, pp. 5185–5189.

[244] Z. Meng, J. Li, Y. Zhao, and Y. Gong, "Conditional teacher-student learning," in *IEEE ICASSP*, 2019, pp. 6445–6449.

[245] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," in *Interspeech*, 2017.

[246] S. Watanabe, T. Hori, J. Le Roux, and J. R. Hershey, "Student-teacher network learning with enhanced features," in *IEEE ICASSP*, 2017, pp. 5275–5279.

[247] J. H. Wong and M. J. Gales, "Sequence student-teacher training of deep neural networks," in *Interspeech*, 2016.

[248] V. Manohar, P. Ghahremani, D. Povey, and S. Khudanpur, "A teacher-student learning approach for unsupervised domain adaptation of sequence-trained asr models," in *IEEE SLT*, 2018, pp. 250–257.

[249] Z. Meng, J. Li, Y. Gaur, and Y. Gong, "Domain adaptation via teacher-student learning for end-to-end speech recognition," in *IEEE ASRU*, 2019, pp. 268–275.

[250] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *ICML*, 2015, pp. 1180–1189.

[251] S. Sun, B. Zhang, L. Xie, and Y. Zhang, "An unsupervised deep domain adaptation approach for robust speech recognition," *Neurocomputing*, vol. 257, pp. 79 – 87, 2017.

[252] Z. Meng, Z. Chen, V. Mazalov, J. Li, and Y. Gong, "Unsupervised adaptation with domain separation networks for robust speech recognition," in *IEEE ASRU*, 2017, pp. 214–221.

[253] P. Denisov, N. T. Vu, and M. F. Font, "Unsupervised domain adaptation by adversarial learning for robust speech recognition," in *Speech Communication; 13th ITG-Symposium*, 2018, pp. 1–5.

[254] Z. Meng, J. Li, Y. Gong, and B.-H. Juang, "Adversarial teacher-student learning for unsupervised domain adaptation," in *IEEE ICASSP*, 2018, pp. 5949–5953.

[255] M. Mimura, S. Sakai, and T. Kawahara, "Cross-domain speech recognition using nonparallel corpora with cycle-consistent adversarial networks," in *IEEE ASRU*, 2017, pp. 134–140.

[256] E. Hosseini-Asl, Y. Zhou, C. Xiong, and R. Socher, "A multi-discriminator CycleGAN for unsupervised non-parallel speech domain adaptation," in *Interspeech*, 2018.

[257] Z. Meng, J. Li, Y. Gong, and B.-H. F. Juang, "Cycle-consistent speech enhancement," in *Interspeech*, 2018.

[258] J. R. Bellegarda, "Statistical language model adaptation: Review and perspectives," *Speech Communication*, vol. 42, no. 1, pp. 93–108, 2004.

[259] P. R. Clarkson and A. J. Robinson, "Language model adaptation using mixtures and an exponentially decaying cache," in *IEEE ICASSP*, vol. 2, 1997, pp. 799–802.

[260] G. Tur and A. Stolcke, "Unsupervised language model adaptation for meeting recognition," in *IEEE ICASSP*, 2007.

[261] X. Liu, M. J. Gales, and P. C. Woodland, "Context dependent language model adaptation," in *Interspeech*, 2008.

[262] R. Kuhn and R. De Mori, "A cache-based natural language model for speech recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 570–583, 1990.

[263] ——, "Corrections to "a cache-based language model for speech recognition"," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 6, pp. 691–692, 1992.

[264] M. Federico, "Bayesian estimation methods for n-gram language model adaptation," in *ICSLP*, vol. 1, 1996, pp. 240–243.

[265] M. Bacchiani and B. Roark, "Unsupervised language model adaptation," in *IEEE ICASSP*, 2003.

[266] K. Seymore, S. Chen, and R. Rosenfeld, "Nonlinear interpolation of topic models for language model adaptation," in *ICSLP*, 1998.

[267] L. Chen, J.-L. Gauvain, L. Lamel, G. Adda, and M. Adda, "Using information retrieval methods for language model adaptation," in *Interspeech*, 2001.

[268] B.-J. P. Hsu and J. Glass, "Style & topic language model adaptation using HMM-LDA," in *EMNLP*, 2006, pp. 373–381.

[269] S. Huang and S. Renals, "Unsupervised language model adaptation based on topic and role information in multiparty meetings," in *Interspeech*, 2008.

[270] R. Kneser, J. Peters, and D. Klakow, "Language model adaptation using dynamic marginals," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[271] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[272] T. Mikolov, S. Kombrink, L. Burget, J. Černockỳ, and S. Khudanpur, "Extensions of recurrent neural network language model," in *IEEE ICASSP*, 2011, pp. 5528–5531.

[273] X. Chen, T. Tan, X. Liu, P. Lanchantin, M. Wan, M. J. Gales, and P. C. Woodland, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition," in *Interspeech*, 2015.

[274] S. Deena, M. Hasan, M. Doulaty, O. Saz, and T. Hain, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition and alignment," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 572–582, 2018.

[275] K. Li, H. Xu, Y. Wang, D. Povey, and S. Khudanpur, "Recurrent neural network language model adaptation for conversational speech recognition," in *Interspeech*, 2018, pp. 3373–3377.

[276] T. Moriokal, N. Tawara, T. Ogawa, A. Ogawa, T. Iwata, and T. Kobayashi, "Language model domain adaptation via recurrent neural networks with domain-shared and domain-specific representations," in *IEEE ICASSP*, 2018, pp. 6084–6088.

[277] Y. Shi, M. Larson, and C. M. Jonker, "Recurrent neural network language model adaptation with curriculum learning," *Computer Speech & Language*, vol. 33, no. 1, pp. 136–154, 2015.

[278] S. R. Gangireddy, P. Swietojanski, P. Bell, and S. Renals, "Unsupervised adaptation of recurrent neural network language models." in *Interspeech*, 2016.

[279] E. Grave, A. Joulin, and N. Usunier, "Improving neural language models with a continuous cache," in *ICLR*, 2017.

[280] S. Merity, C. Xiong, J. Bradbury, and R. Socher, "Pointer sentinel mixture models," in *ICLR*, 2016.

[281] B. Krause, E. Kahembwe, I. Murray, and S. Renals, "Dynamic evaluation of neural sequence models," in *ICML*, 2018, pp. 2766–2775.

[282] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.-C. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," *arXiv:1503.03535*, 2015.

[283] A. Hannun, A. Maas, D. Jurafsky, and A. Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs," *arXiv preprint arXiv:1408.2873*, 2014.

[284] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *IEEE ICASSP*, 2018, pp. 1–5828.

[285] A. Zeyer, K. Irie, R. Schlüter, and H. Ney, "Improved training of end-to-end attention models for speech recognition," in *Interspeech*, 2018.

[286] E. McDermott, H. Sak, and E. Variani, "A density ratio approach to language model fusion in end-to-end automatic speech recognition," in *IEEE ASRU*, 2019, pp. 434–441.

[287] E. Variani, D. Rybach, C. Allauzen, and M. Riley, "Hybrid autoregressive transducer (HAT)," in *IEEE ICASSP*, 2020, pp. 6139–6143.

[288] M. Kim, Y. Kim, J. Yoo, J. Wang, and H. Kim, "Regularized speaker adaptation of kl-hmm for dysarthric speech recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 9, pp. 1581–1591, 2017.

[289] M. Kitza, R. Schlüter, and H. Ney, "Comparison of BLSTM-layer-specific affine transformations for speaker adaptation," in *Interspeech*, 2018, pp. 877–881.

[290] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of LSTM RNN models for speech recognition," in *IEEE ICASSP*, 2016, pp. 5020–5024.

[291] H. Seki, K. Yamamoto, T. Akiba, and S. Nakagawa, "Rapid speaker adaptation of neural network based filterbank layer for automatic speech recognition," in *IEEE SLT*, 2018, pp. 574–580.

[292] R. Serizel and D. Giuliani, "Deep neural network adaptation for children's and adults' speech recognition," in *Italian Conference on Computational Linguistics*, 2014.

[293] P. Swietojanski and S. Renals, "Differentiable pooling for unsupervised acoustic model adaptation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 10, pp. 1773–1784, 2016.

[294] C. Zhang and P. C. Woodland, "DNN speaker adaptation using parameterised sigmoid and ReLU hidden activation functions," in *IEEE ICASSP*, 2016, pp. 5300–5304.

[295] S. Ghorbani and J. H. Hansen, "Leveraging native language information for improved accented speech recognition," in *Interspeech*, 2018.

[296] V. Gupta, P. Kenny, P. Ouellet, and T. Stafylakis, "I-vector-based speaker adaptation of deep neural networks for French broadcast audio transcription," in *IEEE ICASSP*, 2014, pp. 6334–6338.

[297] P. C. Woodland, X. Liu, Y. Qian, C. Zhang, M. J. Gales, P. Karanasou, P. Lanchantin, and L. Wang, "Cambridge University transcription systems for the multi-genre broadcast challenge," in *IEEE ASRU*, 2015, pp. 639–646.

[298] J. Du, X. Na, X. Liu, and H. Bu, "AISHELL–2: Transforming Mandarin ASR Research Into Industrial Scale," *arXiv:1808.10583*, 2018.

[299] J. Carletta, "Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus," *Language Resources and Evaluation*, vol. 41, no. 2, pp. 181–190, 2007.

[300] B. Angelini, F. Brugnara, D. Falavigna, D. Giuliani, R. Gretter, and M. Omologo, "Speaker independent continuous speech recognition using an acoustic-phonetic Italian corpus," in *ICSLP*, 1994.

[301] N. Parihar, J. Picone, D. Pearce, and H. G. Hirsch, "Performance analysis of the Aurora large vocabulary baseline system," in *EUSIPCO*, 2004, pp. 553–556.

[302] "CASIA Northern accent speech corpus," 2003. [Online]. Available: http://www.chineseldc.org/doc/CLDC-SPC-2004-015/intro.htm

[303] D. Giuliani and M. Gerosa, "Investigating recognition of children's speech," in *IEEE ICASSP*, 2003.

[304] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech & Language*, vol. 46, pp. 535–557, 2017.

[305] K. Maekawa, "Corpus of Spontaneous Japanese: Its design and evaluation," in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003.

[306] G. Gravier, G. Adda, N. Paulsson, M. Carré, A. Giraudel, and O. Galibert, "The ETAPE corpus for the evaluation of speech-based TV content processing in the French language," in *LREC*, 2012.

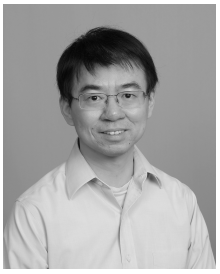
[307] Y. Liu, P. Fung, Y. Yang, C. Cieri, S. Huang, and D. Graff, "HKUST/MTS: A very large scale Mandarin telephone speech corpus,"

in *Chinese Spoken Language Processing*, Q. Huo, B. Ma, E.-S. Chng, and H. Li, Eds., 2006, pp. 724–735.
[308] P. Bell, M. J. F. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, and P. C. Woodland, "The MGB challenge: Evaluating multi-genre broadcast media recognition," in *IEEE ASRU*, 2015, pp. 687–693.
[309] "RASC863: 863 annotated 4 regional accent speech corpus," 2003. [Online]. Available: http://www.chineseldc.org/doc/CLDC-SPC-2004-005/intro.htm
[310] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone speech corpus for research and development," in *IEEE ICASSP*, 1992, p. 517–520.
[311] M. Cettolo, C. Girardi, and M. Federico, "WIT3: Web inventory of transcribed and translated talks," in *EAMT*, 2012, pp. 261–268.
[312] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: an automatic speech recognition dedicated corpus," in *LREC*, 2012.
[313] ——, "Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks," in *LREC*, 2014.
[314] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT acoustic phonetic continuous speech corpus," *Linguistic Data Consortium, LDC93S1*, 1993.
[315] D. B. Paul and J. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Speech and Natural Language Workshop*, 1992.
[316] A. Batliner, M. Blomberg, S. D'Arcy, D. Elenius, D. Giuliani, M. Gerosa, C. Hacker, M. Russell, S. Steidl, and M. Wong, "The PF_STAR children's speech corpus," in *Interspeech*, 2005.
[317] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *IEEE ICASSP*, 2015, pp. 5206–5210.
[318] P. Karanasou, C. Wu, M. Gales, and P. C. Woodland, "I-vectors and structured neural networks for rapid adaptation of acoustic models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 4, pp. 818–828, 2017.
[319] A.-r. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, "Deep belief networks using discriminative features for phone recognition," in *IEEE ICASSP*. IEEE, 2011, pp. 5060–5063.
[320] S. M. Siniscalchi, J. Li, and C.-H. Lee, "Hermitian polynomial for speaker adaptation of connectionist speech recognition systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2152–2161, 2013.
[321] O. Abdel-hamid and H. Jiang, "Rapid and effective speaker adaptation of convolutional neural network based models for speech recognition," in *Interspeech*, 2013.
[322] P. Swietojanski and S. Renals, "Differentiable pooling for unsupervised speaker adaptation," in *IEEE ICASSP*, 2015.
[323] P. Swietojanski and S. Renals, "SAT-LHUC: Speaker adaptive training for learning hidden unit contributions," in *IEEE ICASSP*, 2016.
[324] N. Tomashenko, Y. Khokhlov, A. Larcher, and Y. Estève, "Exploring GMM-derived features for unsupervised adaptation of deep neural network acoustic models," in *International Conference on Speech and Computer*, 2016.
[325] S. Toshniwal, T. N. Sainath, R. J. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, "Multilingual speech recognition with a single end-to-end model," in *IEEE ICASSP*, 2018, pp. 4904–4908.
[326] A. Kannan, A. Datta, T. N. Sainath, E. Weinstein, B. Ramabhadran, Y. Wu, A. Bapna, Z. Chen, and S. Lee, "Large-scale multilingual speech recognition with a streaming end-to-end model," in *Interspeech*, 2019, pp. 2130–2134.
[327] O. Adams, M. Wiesner, S. Watanabe, and D. Yarowsky, "Massively multilingual adversarial speech recognition," in *NAACL/HLT*, 2019.
[328] V. Pratap, A. Sriram, P. Tomasello, A. Hannun, V. Liptchinsky, G. Synnaeve, and R. Collobert, "Massively multilingual ASR: 50 languages, 1 model, 1 billion parameters," *arXiv:2007.03001*, 2020.
[329] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," *arXiv:2006.13979*, 2020.
[330] K. Kawakami, L. Wang, C. Dyer, P. Blunsom, and A. v. d. Oord, "Learning robust and multilingual speech representations," *arXiv:2001.11128*, 2020.
[331] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, "Learning problem-agnostic speech representations from multiple self-supervised tasks," in *Interspeech*, 2019.
[332] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," in *IEEE ICASSP*, 2020, pp. 6989–6993.

Peter Bell (M'09) received a BA in mathematics (2002) and an MPhil in computer speech, text and internet technology (2005) from the University of Cambridge, and a PhD in automatic speech recognition (2010) from the University of Edinburgh. He is a reader in speech technology at the School of Informatics, University of Edinburgh, with research interests that include domain adaptation, regularization, and low-resource methods for acoustic modeling.

Joachim Fainberg (M'20) received a BMus in music and sound recording (Tonmeister) from the University of Surrey (2014), an MSc in artificial intelligence (2015) and a PhD in automatic speech recognition (2020) from the University of Edinburgh. He is currently at the Machine Learning Center of Excellence at JPMorgan Chase. His research interests include domain adaptation, and training methods for acoustic modeling.

Ondrej Klejch (M'17) received a BSc (2013) an MSc (2015) in computer science from the Charles University in Prague, Czech Republic, and a PhD in automatic speech recognition (2020) from the University of Edinburgh. He is a post-doctoral researcher in automatic speech recognition at the University of Edinburgh. His research focuses on acoustic model adaptation using meta-learning approaches.



Jinyu Li (M'08) received the Ph.D. degree from the Georgia Institute of Technology in 2008. From 2000 to 2003, he was a Researcher with the Intel China Research Center and Research Manager in iFlytek, China. Currently, he is a Partner Applied Scientist in Microsoft Corporation, leading a team to design and improve speech modeling algorithms and technologies that ensure industry state-of-the-art speech recognition accuracy for Microsoft products. Dr. Li is a member of the IEEE Speech and Language Processing Technical Committee. He also serves as Associate Editor of the IEEE/ACM Transactions on Audio, Speech, and Language Processing.

Steve Renals (M'91 – SM'11 – F'14) received a BSc in chemistry (1986) from the University of Sheffield, and an MSc in artificial intelligence (1987) and a PhD in neural networks and speech recognition (1991) from the University of Edinburgh. He is professor of speech technology at the School of Informatics, University of Edinburgh, having previously held positions at ICSI Berkeley, the University of Cambridge, and the University of Sheffield. He has research interests in speech recognition, spoken language processing, neural networks, and machine learning. Dr Renals is a fellow of ISCA (2016) and a senior area editor of the IEEE Open Journal of Signal Processing.

Pawel Swietojanski (M'12) received an MSc from AGH-UST in Cracow, Poland and a Ph.D. degree in computer science from the University of Edinburgh, U.K. He is currently a research scientist at Apple, and in the past held academic and research positions at UNSW Sydney and University of Edinburgh. His main research interests include machine learning and its applications in spoken language processing, with a particular focus on learning neural representations for acoustic modeling in speech recognition.