Achieving Multi-Accent ASR via Unsupervised Acoustic Model Adaptation

Mehmet Ali Tuğtekin Turan, Emmanuel Vincent, Denis Jouvet

Achieving Multi-Accent ASR via Unsupervised Acoustic Model Adaptation

M. A. Tuğtekin Turan *Emmanuel Vincent* *Denis Jouvet*

Université de Lorraine, CNRS, Inria, Loria, F-54000 Nancy, France

{tugtekin.turan,emmanuel.vincent,denis.jouvet}@inria.fr

Abstract

Current automatic speech recognition (ASR) systems trained on native speech often perform poorly when applied to non-native or accented speech. In this work, we propose to compute x-vector-like accent embeddings and use them as auxiliary inputs to an acoustic model trained on native data only in order to improve the recognition of multi-accent data comprising native, non-native, and accented speech. In addition, we leverage untranscribed accented training data by means of semi-supervised learning. Our experiments show that acoustic models trained with the proposed accent embeddings outperform those trained with conventional i-vector or x-vector speaker embeddings, and achieve a 15% relative word error rate (WER) reduction on non-native and accented speech w.r.t. acoustic models trained with regular spectral features only. Semi-supervised training using just 1 hour of untranscribed speech per accent yields an additional 15% relative WER reduction w.r.t. models trained on native data only.

Index Terms: ASR, accent adaptation, speaker embedding

1. Introduction

Voice interfaces are widely used for daily tasks such as booking tickets, setting up calendar items, or finding restaurants, and for other applications like education or healthcare. Despite these successes, automatic speech recognition (ASR) systems still perform poorly for speakers whose characteristics do not match those of the training speakers. Accent is considered as one of the most important factors of this mismatch [1]. Most speakers exhibit a very wide variety of accents, usually influenced by their native language or their region. Current ASR models trained on native speech often experience a dramatic loss of accuracy for speakers with strong accents [2]. This affects the overall performance and inclusiveness of voice interfaces. Yet, there is relatively little research on accented speech recognition.

Efforts to address accent and other factors of speaker variability have focused on data transformation and model adaptation methods. For Gaussian mixture model - hidden Markov model (GMM-HMM) based acoustic models, speaker adaptation has proven to be effective for many years. Depiste their superior generalization ability, deep neural network (DNN) based acoustic models also suffer from speaker mismatch [3]. Many adaptation methods have been proposed lately. A simple approach is to retrain the entire DNN using a regularized objective [4]. Another approach is to augment speaker-independent DNNs by speaker-dependent layers [5, 6], activation functions [7] or neuron weights [8] that are trained on adaptation data. These methods are prone to overfitting [9] and induce a large computational overhead at test time. Hence, the most popular approach today is to augment the DNN's input features with i-vectors [10] or other auxiliary features which embed speaker information and can be quickly computed at test time [11–13]. Variants of this approach have been explored, e.g., [14]. In [15], multiple DNNs are trained to form a speaker-independent parametric space. An interpolation vector is estimated for each speaker to combine the DNNs during adaptation. A set of sub-networks is introduced in [16] to capture different acoustic properties where the outputs of those sub-networks are combined by speaker-dependent interpolation weights.

By contrast with the above speaker adaptation methods, research targeted to non-native or accented ASR is much scarcer. Early studies focused on adapting the parameters of GMM-HMM acoustic models trained on native speech by, e.g., maximum likelihood linear regression [17]. Later, methods that train a specific model for each accent have been investigated [18]. Similarly, in [19], a unified model is trained on a limited number of accents, and adapted to any accent using grapheme-based acoustic models. Although these approaches perform well, a separate model per accent induces computational and storage costs [20]. This calls for *multi-accent* methods able to recognize non-native or accented speech via a single model.

Several attempts have recently been made inside the multi-accent setting. One such approach relies on multi-task learning, where the model is trained not only to discriminate phones but also to identify accents [21]. Accent embeddings computed via a time-delay neural network (TDNN) may be used as auxiliary inputs [22]. Another approach is to train with all the available data and fine-tune the last layer on accent-specific data [23, 24]. This is similar to the adaptation of top layers to different languages in DNN-based multilingual ASR [25]. To avoid overfitting when adjusting the network parameters on a small adaptation set, a regularization term may be added [26]. A fundamental limitation of all these approaches is that they assume the availability of transcribed training or adaptation data for the targeted accents. Due to the cost of transcription, this assumption is unrealistic when the number of accents is large.

In this paper, we propose to train an x-vector-like model [27] to compute accent embeddings and use them as auxiliary features for acoustic model adaptation. In addition, we explore the impact of semi-supervised training based on small amounts of untranscribed speech from different accents. Compared to [22], our embeddings are computed using an x-vector like architecture instead of a vanilla TDNN followed by average pooling and, most crucially, our adaptation methodology is fully unsupervised: the accented data used to train the embedding model and the acoustic model is totally untranscribed. For the evaluations, we perform a series of experiments using TDNN acoustic models trained with the lattice-free maximum mutual information (LF-MMI) criterion using Kaldi [28]. For the sake of simplicity, we focus on accented English only. Nevertheless, we believe that our proposed approach will also be helpful in dealing with accented speech from other languages.

The paper is structured as follows. In Section 2, we present our unsupervised model adaptation methodology. Experiments involving four different accents are presented in Section 3. Section 4 provides final remarks and conclusions.

2. Proposed Methodology

Our proposed method consists of training a DNN to compute accent embeddings, and subsequently using them as auxiliary inputs to the acoustic model. This method can adapt on-the-fly to a non-native or accented test utterance. The embedding DNN requires accented data with accent labels for training, but this data doesn't need to be transcribed. Conversely, the training data for the acoustic model must be transcribed, but it doesn't need to be accented. As surprising at it may seem, we will show that accent embeddings do improve ASR performance even when the acoustic model is trained on unaccented data only. In addition, we leverage the untranscribed accented data available at training time by means of semi-supervised learning.

2.1. Accent Embedding Model

In the field of speaker recognition, DNN-based embeddings referred to as x-vectors [27] have replaced i-vectors in many application scenarios. The DNN models the short-term context thanks to its TDNN-based architecture and it is trained to identify individual speakers from variable-length segments.

Following this, we propose to train a DNN with the same architecture as in [27] to classify the accent of input speech segments. The DNN architecture is depicted in Fig. 1. The lower TDNN layers operate at the frame level, their outputs are summarized by a statistical pooling layer, and they are followed by upper layers at the segment level with a softmax output layer. Pooling operation accepts the final frame-level layer as input and calculates the mean as well as the second-order statistics, and the standard deviation. The statistical pooling layer and the upper layers are key in the performance of x-vectors compared to vanilla TDNNs. We train the network to classify accents using a multi-class cross-entropy objective. Denoting as $p(\alpha_{k(n)}|\mathbf{x}_{1:T}^{(n)})$ the estimated probability of accent k given the T input frames $x_1^{(n)}, \ldots, x_T^{(n)}$ in segment n, the training objective is defined as

$$\mathcal{L} = \sum_{n=1}^{N} \log p(\alpha_{k(n)}|\mathbf{x}_{1:T}^{(n)}) \quad (1)$$

where $k(n)$ is the ground truth accent for segment n and N is the number of training segments. Segment-level embeddings are extracted from the last hidden layer.

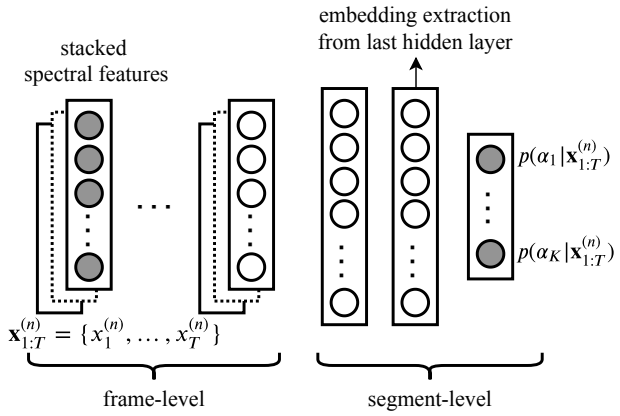

Figure 1: *DNN model for accent embedding. Statistical pooling is performed between frame-level and segment-level layers that aggregates over the frame-level representations.*

2.2. Acoustic Modeling

The proposed accent embeddings can be used as auxiliary inputs for a wide variety of acoustic model topologies to achieve on-the-fly adaptation at test time. In the following, we use a TDNN-based acoustic model [29] trained with the LF-MMI sequence-discriminative criterion [30], which has shown to be effective in many ASR tasks. LF-MMI involves the ratio of two posteriors computed from a numerator graph which represents the phone sequence in the reference transcript and a denominator graph which approximates all possible word sequences by all possible state sequences of a phone-level language model.

2.3. Semi-Supervised Learning

Independently from on-the-fly adaptation, any non-native or accented speech data available at training time (including the data used to train the embeddings) can be leveraged for acoustic model training. Due to the cost of transcribing large amounts of non-native speech, we assume that this data is untranscribed. This situation can be addressed by semi-supervised learning.

We explore this idea using the general semi-supervised LF-MMI training method in [31]. A seed acoustic model is trained on transcribed data and used it to decode the untranscribed training data into lattices that represent the possible transcription hypotheses. A new acoustic model is then trained on both transcribed and untranscribed data by modifying the numerator graph in the LF-MMI criterion for untranscribed utterances so as to span all possible sequences in the corresponding lattices, while the denominator graph remains unchanged. This approach is also referred to as lattice-based semi-supervision.

In the following, the seed model is trained on transcribed unaccented data and the final acoustic model is trained on both transcribed unaccented data and untranscribed accented data covering all accents. We compare it with fully supervised training on transcribed unaccented and accented data, which naturally performs better but incurs a significant transcription cost.

3. Experimental Analysis

3.1. Dataset

We evaluate our method on native, non-native and accented speech. For the native and non-native data, we use the Verbmobil corpus [32] which contains spontaneous speech from human meeting scheduling dialogs. In Verbmobil, we selected American English dialogs as native data and English dialogs from German speakers as non-native data. We also gathered British, Indian, and Australian non-professional accented speech recordings from VoxForge [33]. We created training, test, and adaptation sets with disjoint speakers as shown in Table 1. Non-native and accented datasets are much smaller than native ones, which is a common setting in multi-dialect speech recognition [34].

3.2. Acoustic Model and Lexicon

The TDNN acoustic model takes 40 mel frequency cepstral coefficients (MFCCs) over 25 ms frames with 10 ms stride as inputs. Its architecture is similar to [35], except for the chosen splicing indexes. Denoting as t the current frame index, the input layer splices together frames $\{t-2, t-1, t, t+1, t+2\}$ (or, more compactly, $[-2, 2]$). The i-vector/x-vector speaker embedding or the accent embedding for the considered utterance, if any, is concatenated with the spliced features. The five following hidden layers splice frames at different offsets, namely $\{-1, 1\}$, $\{-1, 1\}$, $\{-3, 3\}$, $\{-3, 3\}$, and $\{-6, 3\}$. Note that

Table 1: *Statistics of the acoustic model training, test, and adaptation sets in terms of the number of speakers. The numbers in parentheses refer to duration in hours.*

Data	Training	Test	Adaptation
Native (US)	235 (25.4)	25 (1.1)	–
British (UK)	–	25 (1.2)	25 (1.0)
Indian (IN)	–	25 (1.4)	25 (1.0)
Australian (AU)	–	25 (1.3)	25 (1.0)
German (DE)	–	25 (1.1)	25 (1.0)

the differences between these offsets were chosen to be multiples of 3 as in [30]. Speed perturbation [36] with speed factors of 0.9, 1.0 and 1.1 is also used for 3-fold augmentation.

Training relies on the supervised or semi-supervised LF-MMI criterion. For decoding, we use a 3-gram language model trained over the native (US) data with a lexicon consisting of 6,945 unique words and a perplexity of 42.7. Decoding parameters are kept fixed for all experiments. In particular, we do not perform lexicon or language model adaptation on top of acoustic model adaptation.

3.3. Computation of Accent Embeddings

The x-vector-like DNN architecture used to extract accent embeddings is outlined in Table 2. The input is a sequence of 30-dimensional MFCCs with 25 ms frame length and 10 ms stride. Cepstral mean normalization is applied over a sliding window of 0.5 s. Speech frames where the first five layers operate at the frame level, with a small temporal context centered at the current frame t. After the four following layers, *frame5* sees a total context of 15 frames. A 512-dimensional accent embedding is extracted from layer *segment7* before the nonlinearity.

In the following, non-overlapping chunks of length 0.5 s are utilized with an online extraction scheme. In other words, we extract the accent embedding for a given 0.5 s chunk by inputting to the network all T frames from the beginning of the utterance up to that point. The embedding network is trained on the utterances of all native (US) and accented (UK, IN, AU, DE) speakers in Verbmobil and VoxForge, excluding those in the test set for the acoustic model. These utterances include the adaptation set for the acoustic model. We achieve 88.5% classification accuracy over the validation data after the training of our embedding network.

Table 2: *DNN architecture for accent embedding.*

Layer	Context	Frames	Input x Output
frame1	$[t-2, t+2]$	5	$5F$ x 512
frame2	$\{t-2, t, t+2\}$	9	1536 x 512
frame3	$\{t-3, t, t+3\}$	15	1536 x 512
frame4	$\{t\}$	15	512 x 512
frame5	$\{t\}$	15	512 x 1500
stat pool	$[0, T)$	T	1500 T x 3000
segment6	$\{0\}$	T	3000 x 512
segment7	$\{0\}$	T	512 x 512
softmax	$\{0\}$	T	512 x K

3.4. Results

We evaluate 9 different acoustic models on native, non-native, and accented data. All of them follow exactly the same TDNN topology defined in Section 3.2. Only the auxiliary features and the training strategy vary. We employed standard Kaldi tools for all the experiments. For further implementation details, see the provided code[1]. The word error rates (WERs) achieved by the 9 systems are reported in Table 3.

Baseline — Model M1 is an accent-unaware model trained on transcribed native English speech only, and can be considered as our baseline. The higher WER observed on accented (VoxForge) data is notably due to the higher language model perplexity. The measured perplexities are 49.2, 94.2, 163.3, 116.5, 66.5 for the English (US), British (UK), Indian (IN), Australian (AU) and German (DE) test sets, respectively.

Impact of Accent Embeddings — Model M4 implements the proposed accent embedding based adaptation method. Compared with M1, we observe relative WER improvements of 7% on non-native data, 11% on native data, and 14 to 20% on accented data. These improvements are remarkable since M4 has been trained on native data only.

Impact of Semi-Supervised Training — Model M5 implements the proposed semi-supervised training method on top of M4. The seed ASR model is trained on the transcribed native training data and used to decode the untranscribed non-native and accented adaptation data. A new model is then trained on both training and adaptation data. Despite the small amount of adaptation data (4 h, i.e., 1 h per accent as shown in Table 1), M5 achieves a relative WER improvement of 15% on non-native speech and 13 to 20% on accented speech compared to M4. The overall WER improvement from M1 to M5 resulting from the combination of our two unsupervised acoustic model adaptation techniques reaches 10% relative for native speech, 21% for non-native speech, and 26 to 36% for accented speech.

Toplines — For comparison, we also evaluate fully supervised toplines. Model M6 is a supervised model trained on both the native training data and the non-native and accented adaptation data, assuming that the latter have been fully transcribed. Model M9 is also a supervised model, which exploits accent embeddings in addition. Unsurprisingly, M6 outperforms M1. More remarkably, it performs comparably and sometimes worse than M5. This means that our two combined unsupervised model adaptation techniques managed to close the gap and achieve the same performance as a standard fully supervised approach. Interestingly also, M9 appears significantly better than M6, indicating that accent embedding based adaptation provides some benefit in the supervised setting too.

3.5. Easy- vs. hard-to-recognize speakers

To further analyze the impact of accent embeddings, we compare the WER reduction achieved for various groups of speakers, from easy- to hard-to-recognize ones. To do so, the test speakers for each accent are divided into 5 groups sorted by increasing WER for M1+M6. The relative WER reduction from M1 to M4 in the unsupervised case and from M6 to M9 in the supervised case is shown in Fig. 2. The WER improvement is consistent across all groups of speakers, and appears to be larger for harder-to-recognize speakers (the largest improvement being observed for group G5).

[1] `https://gitlab.inria.fr/mturan/is-2020`

Table 3: *WERs (%) achieved by different acoustic models on accented British, Indian, Australian, non-native German, and native English speech. M1–M4: Supervised training on native speech only. M5: Semi-supervised training on transcribed native speech and 1 h of untranscribed speech for all accents. M6–M9: Supervised training on native speech and 1 h of transcribed speech for all accents.*

Embedding Model		Adaptation Data	British (UK)	Indian (IN)	Australian (AU)	Non-Native German (DE)	Average Acc./Non-Nat.	Native (US)
[M1]	no embedding	none	38.8	47.9	39.5	33.3	39.4	14.5
[M2]	i-vector embedding [11]		37.7	43.4	36.8	33.1	37.6	13.3
[M3]	x-vector embedding [27]		39.5	33.2	37.4	32.7	34.8	12.9
[M4]	accent embedding		32.4	41.3	31.7	31.1	33.5	12.9
[M5]	accent embedding	1 h per accent (untranscribed)	28.1	35.4	25.4	26.3	28.6	13.1
[M6]	no embedding	1 h per accent (transcribed)	27.9	35.6	28.1	23.8	28.4	14.0
[M7]	i-vector embedding [11]		23.7	31.8	24.3	21.9	24.8	12.8
[M8]	x-vector embedding [27]		24.4	32.1	23.6	20.3	24.5	12.5
[M9]	accent embedding		22.2	30.3	21.2	20.1	23.1	12.4

3.6. Comparison with Other Embeddings

Finally, we compare the proposed accent embeddings with i-vectors [10], which have been successfully applied to speaker and channel adaptation in many ASR tasks [11], and conventional x-vectors [27], which have not been used for acoustic model adaptation to the best of our knowledge but are known to better characterize speaker information. The i-vector and x-vector embeddings have been trained on the same data as the accent embeddings, and they are also computed in an online fashion on 0.5 s chunks. The i-vectors have dimension 100 and are derived from a 512-component GMM taking 40-dimensional MFCCs after linear discriminant analysis as inputs.

In Table 3, we observe that the proposed accent embeddings (models M4 and M9) outperform i-vectors (models M2 and M7) and x-vectors (models M3 and M8). This means that internal representations play an important role in adaptating to multiple accents. This can be understood by visualizing the embeddings using t-SNE [37] in Fig. 3. Our proposed embeddings tend to be better clustered according to the accent which shows that they are indeed able to catch accent-specific aspects.

4. Conclusions

In this paper, we focus on the task of multi-accent speech recognition. We proposed a simple but effective acoustic model adaptation method which combines x-vector-like accent embeddings and semi-supervised LF-MMI training. Unlike previous work, our method is fully unsupervised: the accented data used to train the embeddings and the acoustic model is untranscribed. We show that our proposed accent embeddings outperform classical i-vectors and x-vectors for accented ASR, and that semi-supervised training brings further improvement, closing the gap with a fully supervised approach without auxiliary embeddings. As future work, we plan to combine the proposed acoustic modeling scheme with language model adaptation. We expect this to benefit non-native ASR, since non-native speakers may also use specific language variants.

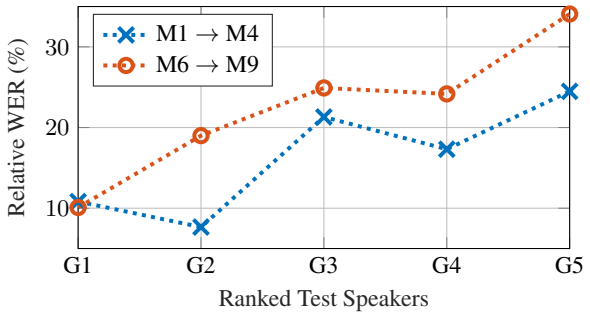

Figure 2: *Relative WER reduction (%) of the accent embedding experiments. G1–G5 denote groups of speakers ranked by increasing WER.*

5. Acknowledgments

This work was supported by the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 825081 COMPRISE (`https://www.compriseh2020.eu/`). Experiments were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see `https://www.grid5000.fr`).

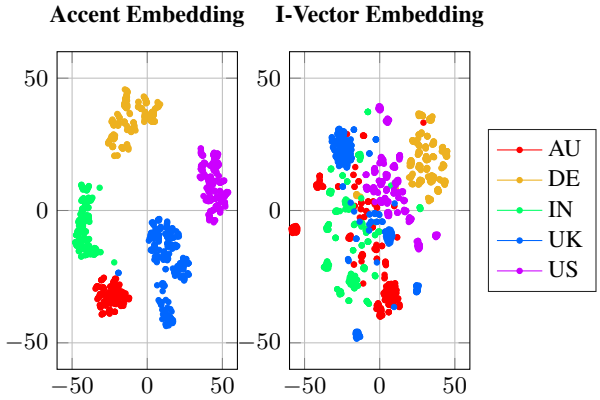

Figure 3: *2D t-SNE projection of the proposed accent embeddings and i-vectors. Each color represents a different accent.*

6. References

[1] K. Yao, D. Yu, L. Deng, and Y. Gong, "A fast maximum likelihood feature transformation method for GMM–HMM speaker adaptation," *Neurocomputing*, vol. 128, pp. 145–152, 2014.

[2] T. Viglino, P. Motlicek, and M. Cernak, "End-to-end accented speech recognition," in *Interspeech*, 2019, pp. 2140–2144.

[3] Y. Zhao, J. Li, S. Zhang, L. Chen, and Y. Gong, "Domain and speaker adaptation for Cortana speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5984–5988.

[4] 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 International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013, pp. 7893–7897.

[5] 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," in *Eurospeech*, 1995, pp. 2171–2174.

[6] K. Kumar, C. Liu, K. Yao, and Y. Gong, "Intermediate-layer DNN adaptation for offline and session-based iterative speaker adaptation," in *Interspeech*, 2015, pp. 1091–1095.

[7] S. M. Siniscalchi, J. Li, and C.-H. Lee, "Hermitian polynomial for speaker adaptation of connectionist speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 2152–2161, 2013.

[8] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 171–176.

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

[10] N. Dehak, P. 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.

[11] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2013, pp. 55–59.

[12] A. Senior and I. Lopez-Moreno, "Improving DNN speaker independence with i-vector inputs," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 225–229.

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

[14] L. Samarakoon and K. C. Sim, "Factorized hidden layer adaptation for deep neural network based acoustic modeling," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2241–2250, 2016.

[15] T. Tan, Y. Qian, M. Yin, Y. Zhuang, and K. Yu, "Cluster adaptive training for deep neural network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4325–4329.

[16] C. Wu and M. Gales, "Multi-basis adaptive neural network for rapid adaptation in speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4315–4319.

[17] L. M. Tomokiyo and A. Waibel, "Adaptation methods for non-native speech," in *Multilinguality in Spoken Language Processing*, 2001.

[18] M. Elfeky, M. Bastani, X. Velez, P. Moreno, and A. Waters, "Towards acoustic model unification across dialects," in *IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 624–628.

[19] K. Rao and H. Sak, "Multi-accent speech recognition with hierarchical grapheme based models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4815–4819.

[20] T. S. Nguyen, K. Kilgour, M. Sperber, and A. Waibel, "Improved speaker adaptation by combining i-vector and fMLLR with deep bottleneck networks," in *International Conference on Speech and Computer (SPECOM)*, 2017, pp. 417–426.

[21] 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 International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1–5.

[22] A. Jain, M. Upreti, and P. Jyothi, "Improved accented speech recognition using accent embeddings and multi-task learning," in *Interspeech*, 2018, pp. 2454–2458.

[23] Y. Huang, D. Yu, C. Liu, and Y. Gong, "Multi-accent deep neural network acoustic model with accent-specific top layer," in *Interspeech*, 2014, pp. 2977–2981.

[24] 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.

[25] 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 International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7304–7308.

[26] J. Yi, Z. Wen, J. Tao, H. Ni, and B. Liu, "Ctc regularized model adaptation for improving lstm rnn based multi-accent mandarin speech recognition," *Journal of Signal Processing Systems*, vol. 90, no. 7, pp. 985–997, 2018.

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

[28] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," Tech. Rep., 2011.

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

[30] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Interspeech*, 2016, pp. 2751–2755.

[31] V. Manohar, H. Hadian, D. Povey, and S. Khudanpur, "Semi-supervised training of acoustic models using lattice-free MMI," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4844–4848.

[32] W. Hess, K. Kohler, and H.-G. Tillmann, "Phondat-Verbmobil speech corpus," in *European Conference on Speech Communication and Technology*, 1995.

[33] "Voxforge: an open and free speech corpus for speaker recognition," http://www.voxforge.org, accessed: 2020-03-12.

[34] S. Yoo, I. Song, and Y. Bengio, "A highly adaptive acoustic model for accurate multi-dialect speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5716–5720.

[35] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Interspeech*, 2015, pp. 3214–3218.

[36] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Interspeech*, 2015, pp. 3586–3589.

[37] L. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.