Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

Robust Unsupervised Adaptation of a Speech Recogniser Using Entropy Minimisation and Speaker Codes

Rogier C. van Dalen, Shucong Zhang*, Titouan Parcollet*, Sourav Bhattacharya*

AI Center Cambridge, Samsung, United Kingdom

{r.vandalen,s1.zhang,t.parcollet,sourav.b1}@samsung.com

Abstract

Speech recognisers usually perform optimally only in a specific environment and need to be adapted to work well in another. For adaptation to a new speaker, there is often too little data for fine-tuning to be robust, and that data is usually unlabelled. This paper proposes a combination of approaches to make adaptation to a single minute of data robust. First, instead of estimating the adaptation parameters with cross-entropy on a single error-prone hypothesis or "pseudo-label", this paper proposes a novel loss function, the conditional entropy over complete hypotheses. Using multiple hypotheses makes adaptation more robust to errors in the initial recognition. Second, a "speaker code" characterises a speaker in a vector short enough that it requires little data to estimate. On a far-field noise-augmented version of Common Voice, the proposed scheme yields a 20 % relative improvement in word error rate on one minute of adaptation data, increasing on 10 minutes to 29 %.

Index Terms: speech recognition, adaptation, minimum entropy, speaker codes

1. Introduction

New environments, whether a new domain or a new speaker, often cause speech recogniser performance to degrade. Adaptation can improve performance, but requires data from the appropriate environment. This paper will focus on unsupervised adaptation to a new speaker, a scenario which brings two problems: first, the adaptation data is usually unlabelled, making adaptation sensitive to errors in the hypotheses; and second, the amount of data per speaker is often limited to a minute or so.

The standard method for dealing with the first problem, unsupervised data, is to run the speech recogniser unadapted to generate a single hypothesis which is used as "pseudolabel". However, adaptation is not robust to errors in the pseudolabel. Section 2 will therefore propose to exploit the fact that a speech recogniser outputs a distribution over word sequences, by minimising the entropy of this distribution [1, 2, 3]. Minimum-entropy adaptation has an effect similar to pseudolabel adaptation in the case where the recogniser is certain, but it will hedge its bets where there are multiple plausible hypotheses. Minimum-entropy adaptation is therefore more robust.

The second problem, the small amount of adaptation data, brings the risk of overfitting the adaptation parameters. A popular adaptation method for neural networks, low-rank adaptation (LoRA) [4], may train only, say, 1 % of model parameters, but they still number in the hundreds of thousands. Section 3 will propose to use speaker codes [5, 6] to make adaptation more robust. Speaker codes are small conditioning vectors, in this work of length 1024, learned with backpropagation. I-vectors and x-vectors [7, 8] are also small conditioning vectors, but computed with a model optimised for e.g. speaker verification. In contrast, speaker codes are learned with a speech recognition objective.

Section 4 will show experimental results on a version of Common Voice, a dataset chosen because it has multiple utterances from each speaker. To make it realistic for far-field recognition, background noise is added with room simulation. In experiments, adapting the low-dimensional speaker codes is more robust than LoRA. The best method of increasing robustness is to minimise the conditional entropy.

1.1. Related work

Outside of the speech literature, [1] proposed minimising the conditional entropy for semi-supervised training, and [2, 3] for adaptation. This is what section 2 will apply to an encoder-decoder speech recogniser, where each output class is a word sequence. For speech, there has been work that minimises the entropy not of the distribution over word sequences, but of intermediate distributions that crop up inside a CTC or Transducer speech recogniser: [9] minimises the entropy of per-frame label distribution of a CTC recogniser; [10] minimises the entropy of distributions over the next token conditioned on partial hypotheses for CTC and Transducer recognisers. There also exists work that finds multiple hypotheses for adaptation by perturbing the input [11] or the model [12]. Finally, there is work that is only superficially related and adds in training (not adaptation) a loss term involving the entropy over CTC frame alignments for a single hypothesis (not over hypotheses): [13] improves performance by minimising and [14] by maximising this entropy.

Given the limited amount of data available for speaker adaptation, a long-standing question is how to select a small set of parameters to estimate robustly (see [15] for an overview). In the neural-network era, fine-tuning has been proposed for subsets of model parameters [16] or with linear transformations of inputs or of outputs of intermediate layers, with easily millions of parameters [17, 18] or with limited expressivity [19]. A recent proposal from general machine learning, low-rank adaptation (LoRA) [4], gives flexibility yet a smaller number of parameters, though still often numbering in the hundreds of thousands. In contrast, speaker embeddings, such i-vectors, x-vectors [7, 8] or bottleneck features [20, 21], have a smaller dimensionality, but are optimised e.g. for speaker verification instead of speech recognition, the task at hand. Speaker codes [5, 6] are computed with backpropagation to improve speech recognition, but are small like embeddings. They were originally proposed to adapt the input for an otherwise frozen DNN-HMM system. Section 3 retains only the core idea, which is to perform fine-tuning on a conditioning vector.

*These authors contributed equally.

10.21437/Interspeech.2025-903

2. Minimum-entropy adaptation

One problem that makes adaptation of a speech recogniser less robust is the reliance of a single hypothesis per utterance. Standard adaptation runs a speech recogniser on an utterance, treats the resulting hypothesis as a "pseudolabel", and then adapts the parameters to make the pseudolabel more likely. The procedure is like adaptation of a HMM model, where this can be seen as a variational approximation. However, current-day speech recognisers are discriminative models $q(\mathbf{w}|\mathbf{X})$ that classify utterance $\mathbf{X}$ from adaptation data $\mathcal{A}$ into word sequence $\mathbf{w}$. The effect of the same procedure is different: pseudolabel adaptation finds the model q^* that makes pseudolabel $\mathbf{w}^*$ more likely:

$$q^* := \arg\max_q \sum_{\mathbf{X}\in\mathcal{A}} q(\mathbf{w}^*(\mathbf{X})|\mathbf{X}), \quad \text{(adaptation)}; \tag{1a}$$

$$\text{where } \mathbf{w}^*(\mathbf{X}) \triangleq \arg\max_{\mathbf{w}} q(\mathbf{w}|\mathbf{X}) \quad \text{(recognition)}. \tag{1b}$$

This procedure is sensitive to errors in the initial hypothesis, since these errors are likely to be reinforced. This paper proposes to increase robustness by using multiple hypotheses, with as a loss function the conditional entropy [1, 2, 3].

To express the conditional entropy mathematically, first produce an auxiliary joint distribution q_A over audio and words by multiplying the distribution p_A of audio from the target domain and the speech recogniser:

$$q_A(\mathbf{X}, \mathbf{w}) \triangleq p_A(\mathbf{X})q(\mathbf{w}|\mathbf{X}). \tag{2}$$

The entropy of this distribution can be written as a sum of conditional entropy $\mathcal{H}_A(q)$, to be minimised, and the entropy over the audio $\mathcal{H}(p_A)$, which is constant w.r.t. the parameters of q:

$$\begin{aligned}\mathcal{H}(q_A) &= -\int \sum_{\mathbf{w}} q_A(\mathbf{X}, \mathbf{w}) \log q_A(\mathbf{X}, \mathbf{w}) \, \mathrm{d}\mathbf{X} \\ &= -\mathbb{E}_{\mathbf{X}\sim p_A}\left[\sum_{\mathbf{w}} q(\mathbf{w}|\mathbf{X}) \log(p_A(\mathbf{X})q(\mathbf{w}|\mathbf{X}))\right] \\ &= \underbrace{-\mathbb{E}_{\mathbf{X}\sim p_A}\left[\sum_{\mathbf{w}} q(\mathbf{w}|\mathbf{X}) \log q(\mathbf{w}|\mathbf{X})\right]}_{\triangleq \mathcal{H}_A(q), \text{ to be minimised}} + \mathcal{H}(p_A). \end{aligned} \tag{3}$$

The first term in (3), the negated expected value, needs to be approximated in two ways. First, the expectation over the distribution p_A over audio from the target domain is approximated using the "empirical distribution", i.e. as a sum over the observed adaptation data $\mathcal{A}$. Second, the unfeasible sum over all possible word sequences $\mathbf{w}$ will be restricted to run over just an N-best list $\mathcal{W}_N$. The N-best list can be found through beam search with the unadapted system.

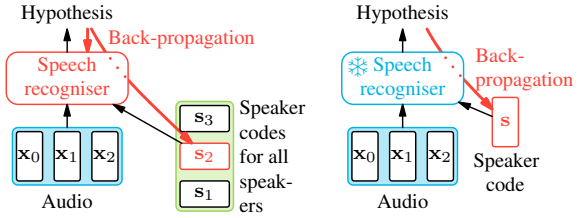
Straightforwardly restricting the sum to an N-best list would open a loophole: the loss could be minimised by shifting all probability mass away from the N-best list. To close this loophole, this paper proposes to re-normalise the sum. Define the normalisation as a function of q:

$$Z(\mathbf{X}) \triangleq \sum_{\mathbf{w}\in\mathcal{W}_N(\mathbf{X})} q(\mathbf{w}|\mathbf{X}). \tag{4}$$

The final approximate loss function with the loophole closed is

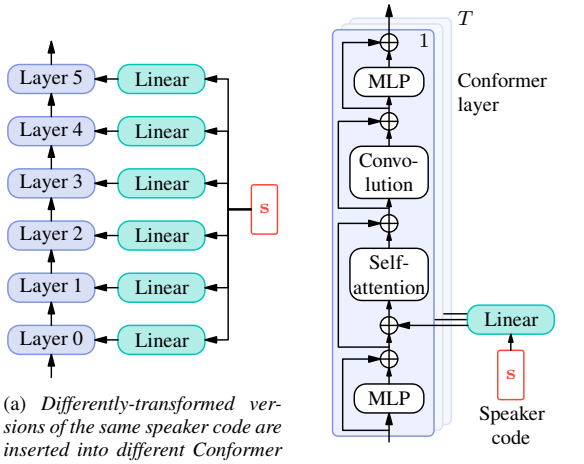
$$\mathcal{H}_A(q) \simeq -\frac{1}{|\mathcal{A}|}\sum_{\mathbf{X}\in\mathcal{A}} \frac{1}{Z(\mathbf{X})} \sum_{\mathbf{w}\in\mathcal{W}_N(\mathbf{X})} q(\mathbf{w}|\mathbf{X}) \log q(\mathbf{w}|\mathbf{X}). \tag{5}$$

Now, the overall weight is always 1, and the $\log q(\cdot)$ term penalises any reduction of mass on the N-best list.



(a) *Jointly training the speech recogniser and all speaker codes.* (b) *Adaptation to a speaker: fine-tuning the speaker code.*

Figure 1: *Using speaker codes, at a high level.*



(a) *Differently-transformed versions of the same speaker code are inserted into different Conformer layers, here layers 0 to 5.* (b) *Conformer layer, zoomed in.*

Figure 2: *Injecting a speaker code into a Conformer speech recogniser.*

3. Adaptation using speaker codes

The more parameters are estimated, the greater the risk of overfitting. Therefore, this paper adapts only 1024 parameters, contained in a vector called a "speaker code" [5, 6]. To make this handful of parameters meaningful, their space is optimised during training. This has been called "adaptive training" [22].

Figure 1a shows how training with speaker codes works. For each speaker in the training set, a separate speaker code vector is added to the trainable parameters. The appropriate speaker code (for each speaker in the batch) is trained with backpropagation, jointly with the speech recogniser. Thus, the speaker code space learns to capture inter-speaker variability, at the same time as the speech recogniser learns to exploit the speaker code to improve recognition.

In adaptation, in Figure 1b, only the speaker code is fine-tuned, using a pseudolabel or the more sophisticated objective from section 2. Before adaptation, the recogniser must come up with a hypothesis in a speaker-independent mode. To support this, this paper uses multi-task learning to train the speech recogniser in speaker-independent and speaker-dependent mode at the same time. For a fraction of utterances ($\frac{1}{2}$ in the experiments), the speaker code is set to $\mathbf{0}$ and not trained.

The next question is how to inject the speaker code into the speech recogniser. The experiments will use an encoder-decoder Conformer model. Since this paper focuses on acoustic personalisation, the speaker code should affect the encoder, specifically the initial layers of the model. From initial experi-

ments with a few points of injection, the architecture illustrated in Figure 2b worked best. The speaker code is transformed with a separate linear projection (without bias) for each layer it is injected into (Figure 2a). The linear layers are part of the model and not adapted. In each of those Conformer layers (Figure 2b), the transformed speaker code is added just before self-attention, after the residual connection branches off, enabling the model to bypass the speaker code if necessary.

4. Experiments

To evaluate speaker adaptation, a large data set that keeps track of speaker identities is necessary. This work uses the Common Voice dataset [23], but augmented with noise and room simulation to make it similar to a realistic farfield scenario. The recognition system is based on a SpeechBrain recipe [24].

The Common Voice dataset, version 18, is used. Speakers with less than 10 minutes of data are removed. The number of samples per speaker is limited to 5,000 as a few contributors recorded hundreds of hours of speech for Common Voice and we wish to limit biases towards them during ASR training.

The "train" set has 1470 speakers, for 855 hours of data. The "train-dev" and "train-test" sets, each 27 hours, are used to evaluate while training. They contain the same speakers as the "train" set, so that not only the canonical model parameters but also the speaker codes can be evaluated during training. The remaining 100 speakers are used for testing. The "adapt", "adapt-dev" and "test" sets each contain 10 minutes of data from those speakers. To confirm that there is no overlap, a SpeechBrain speaker recognition system[1] is used to flag similar speakers, and it is manually confirmed that there is no overlap. The "adapt" set will be used to perform adaptation, the "adapt-dev" set to determine the best epoch to use, and the "test" set for the word error rates. Since the sets for adaptation and testing are separate, the amount of adaptation data from "adapt" can be varied while reporting word error rates on a consistent set: "test".

Background noises for augmentation are taken from the Musan dataset,[2] "music" (with weight $\frac{1}{3}$) and "noise" (with weight $\frac{2}{3}$) subsets. All music files are included, but noise files are selected based on length (at least 20 seconds), the variance in loudness across the waveform ($< 20\,\text{dB}$), and the statistical skew of the loudness (< -1). This leaves 88 files from the "noise" subset. For each speaker, one type of background music/noise is randomly chosen, and added to all their data, repeating the noise where necessary, modelling a situation where any one speaker tends to use speech recognition in a particular environment, be it their car, an office, or on the street. For one scenario, "CommonVoice-Augmented" the noise is added without reverberation. For the main scenario, "CommonVoice-Farfield", the speech and background noise are rendered in a virtual room, roughly following [25]. Rooms are sampled uniformly with horizontal sizes 1.5 to 20 metres, and height 1.5 to 5 metres. The speech, the noise, and the virtual microphone are placed uniformly but at least 0.2 metre from any wall or floor. The resulting distribution of "RT60", the time for the noise to drop by 60 dB, is similar to the one in [25]. After adding room reverb, levels are set to make the signal-to-noise ratio distributed according to $\text{Unif}[\infty, 20, 10, 0]$.

The model is a 100M-parameter Conformer-based encoder-decoder system with joint CTC-Attention training [26], implemented in SpeechBrain [24]. The LoRA adaptation has with rank 16 into layers 1–5, for 1.3M parameters in total. The speaker codes are injected, after initial experiments, as in Figure 2b. The models are trained for 175,000 steps with each batch containing 1,200 seconds based on dynamic duration batching. For systems with speaker codes, a warm-up is performed for the first 5 epochs where speaker codes are kept at $\mathbf{0}$. After that, all parameters, including speaker codes and linear layers, are trained, using the Adam optimiser. In each batch only speaker codes for the speakers in that batch are selected and trained. The model thus jointly learns a meaningful space for speaker codes at the same time as the model that uses the speaker code.

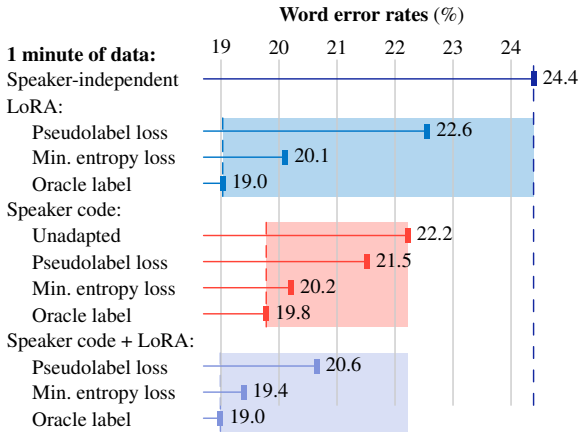


Figure 3: *Average word error rates on CommonVoice-Farfield; adaptation on only one minute of data.*

However, one concern is the performance of the model before adaptation, i.e. with the speaker code set to $\mathbf{0}$. Therefore, multi-task training is used: for some of the utterances ($\frac{1}{2}$ in this work), the speaker code is clamped to $\mathbf{0}$, and not trained.

Adaptation and decoding is run independently per speaker. For experiments with pseudolabels, the adaptation data "adapt" is decoded with the unadapted model. Then, the adaptation parameters are fine-tuned, with all other parameters fixed and the reference transcription set to the pseudolabel. For minimum-entropy training, the loss function is (5), where beam search yields a 5-best list for each utterance. For all adaptation methods, the epoch out of 50 is found that maximises the unsupervised accuracy on the "adapt-dev", averaged over all speakers. This number of epochs is then used to adapt the recogniser. The word error rate on the "test" subset for each of the 100 speakers is then averaged.

4.1. Results

The headline results for adaptation on only one minute of data are shown in Figure 3. There are two aspects to the results: the adaptation parameters (LoRA, speaker code, or both) and the loss function (pseudolabel, minimum-entropy, or oracle label). The shaded areas show the windows of opportunity for each choice of adaptation parameters, between unadapted and oracle label, i.e. supervised, adaptation. Without adaptation, on CommonVoice-Farfield, a hard task, the word error rate is 24.4 %. Intriguingly, training with speaker codes helps the system even if the speaker codes are set to zero ("Unadapted"), with performance at 22.2 %. The hypothesis is that training

[1] https://huggingface.co/speechbrain/spkrec-ecapa-voxceleb

[2] David Snyder, Guoguo Chen, Daniel Povey, "MUSAN: A Music, Speech, and Noise Corpus", https://arxiv.org/abs/1510.08484

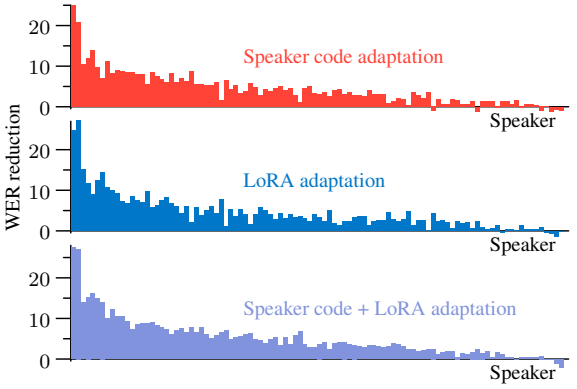


Figure 4: *Per-speaker improvement in word error rate from adapting with the minimum-entropy loss on one minute of data. Speakers are sorted by descending average improvement.*

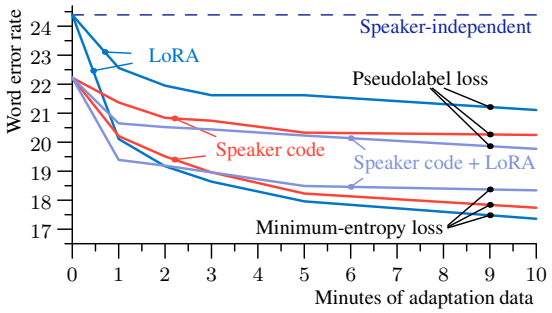


Figure 5: *Average word error rates on CommonVoice-Farfield for adaptation with different amounts of adaptation data.*

with speaker codes has an effect like multi-task learning, but instead of having the speaker identity as an additional output, the speaker identity is given as a conditioning vector.

Two aspects stand out from Figure 3. First, irrespective of which parameters are adapted, the minimum-entropy loss outperforms pseudolabel adaptation. As explained in section 2, the minimum-entropy loss exploits uncertainty in the recogniser output and thus makes adaptation robust to initial recognition errors. When adapting both speaker codes and LoRA, the word error rate drops to 19.4 %, a 20 % relative improvement compared to the baseline. Second, with the pseudolabel loss it is noticeable that adapting the 1024-dimensional speaker code, introduced in section 3, instead of LoRA is more robust. The space of speaker codes, learned while training, is clearly a much terser description of speaker variation than LoRA matrices.

Figure 4 shows the improvement per speaker. For all sets of adaptation parameters, performance improvements are seen for the vast majority of speakers, which underlines the robustness of the minimum-entropy loss.

Figure 5 shows word error rates as a function of the amount of adaptation data. The improvement that the minimum-entropy loss gives is consistent for all amounts of adaptation data. For best performance, if there is a small amount of adaptation data, about one minute, it is useful to adapt both speaker code as well as LoRA. As more adaptation data becomes available, it is worthwhile switching to just LoRA.

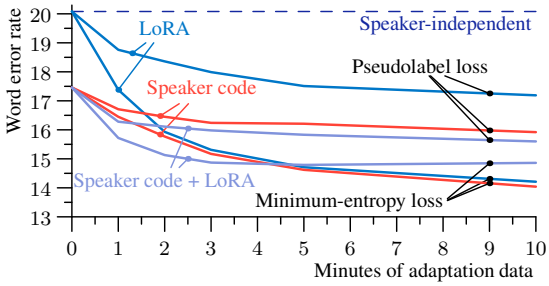


Figure 6: *Ablation: average word error rates on CommonVoice-Augmented for adaptation with different amounts of adaptation data. Unlike Figure 5, the data has no reverberation.*

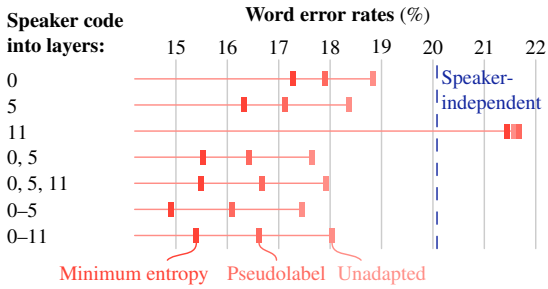


Figure 7: *Ablation: average word error rates on CommonVoice-Augmented for injecting the speaker code into different layers, for 5 minutes of adaptation data.*

4.2. Ablation studies

Figure 6, like Figure 5, shows word error rates from different amounts of adaptation data, but on nearfield data from CommonVoice-Augmented. In this scenario, minimum-entropy adaptation still provides significant improvements, but the harder scenario with reverberation, like in Figure 5, benefits more from the minimum-entropy loss.

Finally, Figure 7 examines recognition performance when the speaker code is injected into different blocks. These results are on CommonVoice-Augmented again, and the procedure for choosing the adaptation epoch is different from the other experiments: here, for each speaker, the epoch is chosen to maximise the accuracy on that speaker's data from adapt-dev. The expectation is that it is more important to inject information about the speaker in the lower layers, whereas in higher layers the differences between speakers are more likely to have been erased. The results match this intuition: from Figure 7, adapting the top layer, layer 11, consistently degrades performance, and in contrast the lower layers 0–5 are important for effective adaptation.

5. Conclusion

This paper has examined how to adapt a speech recogniser to an unseen speaker, using only a small amount of unlabelled adaptation data. It has proposed two methods to improve robustness of adaptation. First, instead of using the standard pseudolabel loss, a new loss, the conditional entropy over the sequence of words that the speech recogniser outputs has been proposed. Second, it has proposed to use speaker codes on an encoder-decoder recogniser. Both methods help improve performance on small amounts of data.

6. References

[1] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Advances in Neural Information Processing Systems*, 2004.

[2] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *Proceedings of Conference on Learning Representations*, 2021.

[3] M. Zhang, S. Levine, and C. Finn, "MEMO: Test time robustness via adaptation and augmentation," in *Advances in Neural Information Processing Systems*, 2022.

[4] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *Proceedings of Conference on Learning Representations*, 2022.

[5] 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 *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2013.

[6] S. Xue, O. Abdel-Hamid, H. Jiang, and L. Dai, "Direct adaptation of hybrid DNN/HMM model for fast speaker adaptation in LVCSR based on speaker code," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[7] S. Tan and K. C. Sim, "Learning utterance-level normalisation using variational autoencoders for robust automatic speech recognition," in *Proceedings of Spoken Language Technology Workshop*, 2016.

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

[9] G.-T. Lin, S.-W. Li, and H. Lee, "Listen, adapt, better WER: Source-free single-utterance test-time adaptation for automatic speech recognition," in *Proceedings of Interspeech*, 2022.

[10] C. Kim, J. Park, H. Shim, and E. Yang, "SGEM: Test-time adaptation for automatic speech recognition via sequential-level generalized entropy minimization," in *Proceedings of Interspeech*, 2023.

[11] P. Teja Sivaprasad and F. Fleuret, "Test time adaptation through perturbation robustness," in *NeurIPS Workshop on Distribution Shifts*, 2021.

[12] S. Khurana, N. Moritz, T. Hori, and J. Le Roux, "Unsupervised domain adaptation for speech recognition via uncertainty driven self-training," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2021.

[13] E. Variani, K. Wu, D. Rybach, C. Allauzen, and M. Riley, "Alignment entropy regularization," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2023.

[14] O. Chang, D. Hwang, and O. Siohan, "Revisiting the entropy semiring for neural speech recognition," in *Proceedings of Conference on Learning Representations*, 2023.

[15] P. Bell, J. Fainberg, O. Klejch, J. Li, S. Renals, and P. Swietojanski, "Adaptation algorithms for neural network-based speech recognition: An overview," *IEEE Open Journal of Signal Processing*, vol. 2, 2021.

[16] Y. Huang, G. Ye, J. Li, and Y. Gong, "Rapid speaker adaptation for Conformer Transducer: Attention and bias are all you need," in *Proceedings of Interspeech*, 2021.

[17] 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 *Proceedings of Eurospeech*, 1995.

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

[19] P. Swietojanski, J. Li, and S. Renals, "Learning hidden unit contributions for unsupervised acoustic model adaptation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 8, 2016.

[20] H. Huang and K. C. Sim, "An investigation of augmenting speaker representations to improve speaker normalisation for dnn-based speech recognition," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, 2015.

[21] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation," in *Proceedings of Automatic Speech Recognition and Understanding Workshop*, 2017.

[22] M. J. F. Gales, "Cluster adaptive training for speech recognition," in *Proceedings of the International Conference on Spoken Language Processing*, 1998.

[23] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common Voice: A massively-multilingual speech corpus," in *Proceedings of the Language Resources and Evaluation Conference*, May 2020. [Online]. Available: https://aclanthology.org/2020.lrec-1.520/

[24] M. Ravanelli, T. Parcollet, A. Moumen, S. de Langen, C. Subakan, P. Plantinga, Y. Wang, P. Mousavi, L. D. Libera, A. Ploujnikov, F. Paissan, D. Borra, S. Zaiem, Z. Zhao, S. Zhang, G. Karakasidis, S.-L. Yeh, P. Champion, A. Rouhe, R. Braun, F. Mai, J. Zuluaga-Gomez, S. M. Mousavi, A. Nautsch, H. Nguyen, X. Liu, S. Sagar, J. Duret, S. Mdhaffar, G. Laperrière, M. Rouvier, R. D. Mori, and Y. Estève, "Open-source conversational AI with SpeechBrain 1.0," *Journal of Machine Learning Research*, vol. 25, no. 333, pp. 1–11, 2024. [Online]. Available: http://jmlr.org/papers/v25/24-0991.html

[25] C. Kim, A. Misra, K. Chin, T. Hughes, A. Narayanan, T. N. 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 *Proceedings of Interspeech*, 2017.

[26] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 4835–4839.