DOMAIN EXPANSION IN DNN-BASED ACOUSTIC MODELS FOR ROBUST SPEECH RECOGNITION

Shahram Ghorbani, Soheil Khorram, John H.L. Hansen

Center for Robust Speech Systems (CRSS), University of Texas at Dallas

ABSTRACT

Training acoustic models with sequentially incoming data – *while both leveraging new data and avoiding the forgetting effect* – is an essential obstacle to achieving human intelligence level in speech recognition. An obvious approach to leverage data from a new domain (e.g., new accented speech) is to first generate a comprehensive dataset of all domains, by combining all available data, and then use this dataset to retrain the acoustic models. However, as the amount of training data grows, storing and retraining on such a large-scale dataset becomes practically impossible. To deal with this problem, in this study, we study several domain expansion techniques which exploit only the data of the new domain to build a stronger model for all domains. These techniques are aimed at learning the new domain with a minimal forgetting effect (i.e., they maintain original model performance). These techniques modify the adaptation procedure by imposing new constraints including (1) *weight constraint adaptation (WCA)*: keeping the model parameters close to the original model parameters; (2) *elastic weight consolidation (EWC)*: slowing down training for parameters that are important for previously established domains; (3) *soft KL-divergence (SKLD)*: restricting the KL-divergence between the original and the adapted model output distributions; and (4) *hybrid SKLD-EWC*: incorporating both SKLD and EWC constraints. We evaluate these techniques in an accent adaptation task in which we adapt a deep neural network (DNN) acoustic model trained with native English to three different English accents: Australian, Hispanic, and Indian. The experimental results show that SKLD significantly outperforms EWC, and EWC works better than WCA. The hybrid SKLD-EWC technique results in the best overall performance.

Index Terms— domain expansion,domain adaptation, DNN-based acoustic models, speech recognition

1. INTRODUCTION

Current state-of-the-art neural network-based ASR systems have advanced to nearly human performance in several evaluation settings [1, 2]; however, these systems perform poorly for domains[1] that are not included in the original training data [3, 4, 5, 6]. For example, if we train an ASR system using a U.S. English dataset, the performance of the system significantly degrades for other English accents (e.g., Australian, Indian, and Hispanic). In order to improve performance of the system for an unseen domain, we can adapt the previously trained model to capture the statistics of the new domain. However, adaptation techniques suffer from the *forgetting effect*: previously learned information will be lost by learning the new information. We need an ASR system that not only performs well for the new domain, but also retains performance for previously seen domains. This is the goal of *domain expansion* methods.

Domain Expansion – In a domain expansion scenario, we are given a model trained on an initial domain and a dataset for an unseen domain, the goal is to modify the model such that it performs well for both domains. The main difficulty of domain expansion is to preserve the functionality (input-output mapping) of the original model (mitigating the forgetting problem). Many approaches have been proposed to deal with the forgetting problem in neural networks. These approaches can be divided into three categories: architectural, rehearsal, and regularization strategies.

1.1. Architectural strategies

In this class of methods, architectures of neural networks are modified to mitigate the forgetting problem. Progressive neural network (PNN) [7] is a popular architectural strategy; it freezes the previously trained network and uses its intermediate representations as inputs into a new smaller network. PNN has been applied in many different applications including speech synthesis [8], speaker identification [9] and speech emotion recognition [9]. However, it has been shown that PNN is not efficient for long sequences of domains, since the number of weights in PNN increases linearly with the number of domains [7].

---

[1]In this paper, we use the term "domain" to refer to a group of utterances that share some common characteristics.

1.2. Rehearsal strategies

These approaches store part of the previous training data and periodically replay them for future training. A full rehearsal strategy can alleviate the forgetting effect, but it is very slow and memory intensive. Tylor et al. proposed EXSTREAM, a new partitioning-based approach, to address the memory problem of the full rehearsal strategy [10]. In another approach, [11] proposed to train an encoder-decoder model that distills information which exists in the previous domains. Their method uses the trained encoder-decoder to simulate pseudo patterns of the previous domains and exploits these pseudo patterns during the training of the new domain.

1.3. Regularization strategies

Regularization refers to a set of techniques that alleviate the forgetting effect by imposing additional constraints on updating parameters. A straightforward constraint is *weight constraint adaptation (WCA)* which penalizes the deviation of the model parameters from the original model parameters; it adds an l_2 distance between the original and adapted weights [12]. Another popular regularization approach is *learning without forgetting (LWF)* [13] that tries to learn a sequence of relevant tasks without losing performance for the older ones by imposing output stability. Jung et al. [14] explored the domain expansion problem for image classification tasks. They used an l_2 distance between the final hidden representations of the original network and the adapted network. Kirkpatrick et al. [15] introduced *elastic weight consolidation (EWC)* which selectively slows down the training for weights that are important for older domains.

In this study, we explore approaches to address the domain expansion problem for the deep neural network (DNN)-based acoustic models. To the best of our knowledge, this is the first study that explores domain expansion for speech recognition. We investigate several existing and proposed regularization strategies to alleviate the forgetting effect in domain expansion. We employ WCA and EWC as the baseline techniques for the domain expansion problem; we also propose two new domain expansion techniques: *soft KL-divergence (SKLD)* and hybrid SKLD-EWC. SKLD penalizes the KL-divergence (KLD) between the original model's output and the adapted model's output as a measure of the deviation of the model. We will demonstrate that the proposed SKLD and EWC are complementary to each other, and combining them can lead to a better domain expansion technique which we refer to as SKLD-EWC. We will compare the efficacy of these methods in an accent adaptation task in which we adapt a DNN acoustic model trained with native English to three different English accents: Australian, Hispanic, and Indian. Our results will show that the proposed hybrid technique, SKLD-EWC, results in the best overall performance and SKLD performs significantly better than EWC and WCA.

2. DOMAIN EXPANSION APPROACHES

In this section, we explain details of four domain expansion techniques (i.e., weight constraint adaptation (WCA), elastic weight consolidation (EWC), soft KL-Divergence (SKLD), and hybrid SKLD-EWC) that we investigate in this study.

Problem Setup – In the domain expansion task, we are given an original model $\mathcal{M}^o$, trained on an original domain $\mathcal{D}^o$, and a dataset for an unseen domain $\mathcal{D}^n$, where the goal is to find a new model $\mathcal{M}^n$ that performs well for both $\mathcal{D}^o$ and $\mathcal{D}^n$.

2.1. Weight Constraint Adaptation (WCA)

WCA was first proposed in [12] to regularize the adaptation process for discriminative classifiers. In another study [13], WCA was employed for continual learning in a sequence of disjoint tasks. This technique tries to find a solution that performs well for the new domain, $\mathcal{D}^n$, which is also close to the original model, $\mathcal{M}^o$.

According to [15], for a given neural network architecture, there are many configurations of model parameters that lead to comparable performance. Therefore, there are many configurations that can efficiently represent our new domain $\mathcal{D}^n$. Among such configurations, an effective solution for domain expansion is the one that stands closer to the original model $\mathcal{M}^o$. Different distance metrics can be used to measure the similarity between models. WCA benefits from the Euclidean distance between the learnable parameters of $\mathcal{D}^o$ and $\mathcal{D}^n$. This idea can be implemented by imposing an additional L_2 constraint on the optimization loss function of the neural network:

$$J_{WCA}(\theta^n) = J_{cross}(\theta^n) + \frac{\lambda_w}{2}||\theta^n - \theta^o||_2, \qquad (1)$$

where θ^o and θ^n are the learnable parameters of $\mathcal{M}^o$ and $\mathcal{M}^n$, respectively; $J_{cross}(\theta^n)$ is the main optimization loss (cross-entropy loss function); $J_{WCA}(\theta^n)$ is the regularized loss with the WCA technique; $||.||_2$ is the L_2 norm; and λ_w is a regularization parameter that determines how far the parameters could diverge from their initial values to learn the new domain.

2.2. Elastic Weight Consolidation (EWC)

The WCA technique considers all weights equally. Therefore, it is unable to find an efficient compromise to maintain the model performance for the original domain $\mathcal{D}^o$ and learning the new domain $\mathcal{D}^n$. However, all weights are not equally important, and using an approach that takes weight importance into account would perform better than a naive WCA.

Intuitively, after training a DNN with sufficient iterations, the model converges to a local minimum point of the optimization landscape. At such a point, the sensitivity of the loss function w.r.t. the i-th learnable weight, θ_i^n, can be calculated

108

by the curvature of the loss function along the direction specified by θ_i^n changes. High curvature for a weight means that the loss function is sensitive to small changes to that weight. Therefore, to preserve the performance of the network for the previous domain, we must prevent modifying the parameters with high curvature. On the other hand, parameters with low curvature values are proper choices to be tuned with new data without losing the model performance for the original data.

The curvature of the loss function is equivalent to the diagonal of the Fisher information matrix F [16]. EWC offers a straightforward method to incorporate the importance of the learnable weights (curvature of the loss function w.r.t. the weights) in the adaptation process. The method is similar to WCA; the only difference is that EWC employs a weighted L_2 norm instead of the regular L_2 norm in WCA:

$$J_{EWC}(\theta^n) = J_{cross}(\theta^n) + \frac{\lambda_e}{2} \sum_i diag\{F\}_i (\theta_i^n - \theta_i^o)^2, \quad (2)$$

where $diag\{F\}_i$ is the i-th element of the diagonal of the Fisher information matrix F (representing the importance of the i-th learnable weight); θ_i^n and θ_i^o are the i-th weight of the new and original models, respectively; and the summation is taken over all learnable weights of the network. $diag\{F\}$ can be easily calculated by the variance of the first order derivatives of the loss function w.r.t. the learnable weights (i.e., $Var\{\partial J(\theta)/\partial \theta_i\}$) [16].

2.3. Soft KL-Divergence (SKLD)

A major difficulty of the domain expansion task is to preserve the functionality (input-output mapping) of the original model $\mathcal{M}^o$. WCA and EWC achieve this by providing a link between the learnable weights of the new model $\mathcal{M}^n$ and the original model $\mathcal{M}^o$. According to the experiments performed in [14], linking the learnable parameters is not an efficient way of preserving the functionality of the parameters, since applying slight changes to some of the parameters may significantly modify the input-output mapping of the network. Another method for preserving the functionality of $\mathcal{M}^o$ is to impose new constraints on the outputs of the model [17]. By constraining the outputs of $\mathcal{M}^n$ to be consistent with the outputs of $\mathcal{M}^o$, we can assure that these two models are similar to each other. SKLD leverages this idea through two steps: (1) it takes the original model $\mathcal{M}^o$ and the data of the new domain $\mathcal{D}^n$; it then generates the output of $\mathcal{M}^o$ for all samples of the dataset. (2) next, SKLD trains the new model $\mathcal{M}^n$ by initializing it from $\mathcal{M}^o$ and using a regularized loss function that can be expressed by:

$$J_{SKLD}(\theta^n) = (1 - \lambda_s)J_{cross}(\theta^n) +$$
$$\lambda_s \sum_{i \in I} D_{KL}(\mathcal{M}^n(x_i), \mathcal{M}^o(x_i)), \quad (3)$$

where I is the total number of samples; D_{KL} is the KL distance; x_i is the i-th input feature vector; $\mathcal{M}^o(x_i)$ and $\mathcal{M}^n(x_i)$

are the outputs of the original and the new models obtained for the i-th sample x_i; and $0 \leq \lambda_s \leq 1$ is a regularization hyperparameter that provides a compromise between learning the new domain (by optimizing J_{cross}) and preserving the input-output mapping of the original model (by optimizing D_{KL}). $\lambda_s = 0$ results in the conventional pre-training/fine-tuning adaptation. By increasing the value of λ_s, we can ensure a balanced trade-off between learning the new domain and mitigating the forgetting effect problem. For this study, we tune λ_s to achieve the best performance for both domains.

Equation (3) uses the KL divergence between $\mathcal{M}^o(x_i)$ and $\mathcal{M}^n(x_i)$ to deal with the forgetting problem. However, some parts of the KL divergence are not related to the learnable parameters of $\mathcal{M}^n$. In [18], it is demonstrated that by removing these parts, the KL divergence will be simplified to the cross-entropy:

$$J_{SKLD}(\theta^n) = (1 - \lambda_s)J_{cross}(\theta^n) +$$
$$\lambda_s \sum_{i \in I} J_{cross}(\mathcal{M}^o(x_i), \mathcal{M}^n(x_i)),$$
$$J_{cross}(\mathcal{M}^o(x_i), \mathcal{M}^n(x_i)) =$$
$$\sum_{c \in C} \mathcal{M}_c^o(x_i) \log(\mathcal{M}_c^n(x_i)), \quad (4)$$

where C is the total number of classes; $\mathcal{M}_c^o(x_i)$ and $\mathcal{M}_c^n(x_i)$ are the probability of the c-th class generated by $\mathcal{M}^o$ and $\mathcal{M}^n$ for an input vector x_i.

In neural networks, we typically use a softmax with temperature $T = 1$ to produce the probability for each class. However, Hinton et al. [19] suggested that using $T > 1$ that increases the probability of small logits, performs better in transferring the functionality of a large network to a smaller one. Therefore, we also consider tuning the temperature to examine its effects in preserving the functionality of the model for the original data. We consider using a softmax with adjustable temperature to produce the output distribution for both $\mathcal{M}^o$ and $\mathcal{M}^n$ in the constraint term of equations (4). Note that we use $T = 1$ for the tuning loss as well as in the evaluation phase.

2.4. Hybrid SKLD-EWC

In the previous sections, we explained both SKLD and EWC approaches. Each one has its advantages and disadvantages. The advantage of EWC is that it computes the Fisher information matrix (that quantifies the importance of the weights) based on the original data during the initial training. However, SKLD does not exploit such information about the importance of the weights. On the other hand, EWC uses a fixed fisher matrix that is estimated for the initial model. However, fisher matrix changes during the adaptation procedure and therefore the fixed assumption of the EWC method is not reliable. The advantage of SKLD is that it is more efficient in preserving the functionality of the original model as the

109

efficacy of SKLD does not change during the adaptation procedure. We propose to combine these two techniques into a new hybrid approach SKLD-EWC. Our proposed technique can be implemented by imposing both SKLD and EWC constraints on the tuning loss:

$$J_{SKLD-EWC}(\theta^n) = (1 - \lambda_s)J_{cross}(\theta^n) +$$
$$\lambda_s \sum_{i \in I} J_{cross}(\mathcal{M}^o(x_i), \mathcal{M}^n(x_i)) +$$
$$\lambda_e \sum_i diag\{F\}_i(\theta_i^n - \theta_i^o)^2. \quad (5)$$

This hybrid method requires two regularization parameters: λ_s and λ_e defined for regularizing the outputs and the weights, respectively. These two parameters provide a more flexible data expansion technique, but at the expense of more difficult hyper-parameter tuning.

3. EXPERIMENTS AND RESULTS

3.1. Experimental Setup

Dataset – We evaluate the efficacy of the domain expansion techniques for a DNN-HMM-based ASR system. To train the original model for native English, we use the 100h part of the LIBRISPEECH corpus (Libri) [20]. This part has higher recording quality and the speakers' accents are closer to the native English compared to the rest of the corpus. For the domain expansion experiments, we use UT-CRSS-4EnglishAccent corpus [21] that contains speech data from 3 non-US English accents, namely Hispanic (HIS), Indian (IND) and Australian (AUS). The data for each accent consists of 100 speakers, with session content that consists of read and spontaneous speech. In this corpus, for each accent, there is more than 28h of training data, 5h of development and 5h of evaluation data. We use the standard language model (LM) provided for Libri to decode the original data (i.e., Libri) [20]. However, for the other accents, since we have spontaneous utterances too, we train a 3-gram and a 4-gram LM by pooling transcriptions of Fisher, Switchboard, and UT-CRSS-4EnglishAccent. The decoding procedure used for Libri and other accents is the same.

Model structure – We implement the domain expansion techniques for a DNN-HMM based ASR system using Kaldi [22] and Tensorflow [23]. In all experiments, we extract 40-dimensional Mel-filterbank coefficients [22] for each 25ms frame with a skip rate of 10ms. Each frame is expanded by stacking 5 frames from each side; therefore, the input to the network is the Mel-filterbank coefficients of 11 successive frames. The acoustic model is a 5-layer fully connected network with 1024 neurons at each hidden layer and 3440 output units that produce a distribution over senones. We use "ReLU" activation function in intermediate layers and "softmax" in the output layer that generates the probabilities

Table 1: WERs of different domain expansion methods on the original (Org) and the new domains (New). For each approach, among the different settings of regularization parameters, the one that results in the best overall performance for both original and new domains (i.e., the lowest average of WERs) is reported. The relative WER increase of the methods compared to multi-condition (MC) training is also reported (Rel-MC)

	Method	Org	New	Avg	Rel-MC
AUS	MC	8.35	10.7	9.52	—
	Original	8.10	23.04	15.57	63.5
	Fine-Tuned	20.3	9.64	14.97	57.2
	WCA	10.23	12.2	11.22	17.74
	EWC	9.27	12.71	11	15.38
	SKLD	8.84	11.23	**10.03**	**5.35**
	SKLD-EWC	8.49	11.48	**9.98**	**4.8**
IND	MC	8.29	15.96	12.12	—
	Original	8.10	28.68	18.39	51.6
	Fine-Tuned	22.54	15.56	19.05	57.1
	WCA	10.66	17.61	14.13	16.5
	EWC	10.26	17.6	13.93	14.8
	SKLD	9.29	16.45	**12.87**	**6.14**
	SKLD-EWC	8.92	16.61	**12.76**	**5.2**
HIS	MC	8.21	11.65	9.93	—
	Original	8.10	20.09	14.1	41.9
	Fine-Tuned	16.14	11.52	13.83	39
	WCA	8.65	12.84	10.74	8.2
	EWC	8.72	12.61	10.66	7.4
	SKLD	9.26	12.0	10.63	7.0
	SKLD-EWC	8.29	12.5	**10.39**	**4.7**

of senones. We initialize all weights using the "he-normal" initialization technique [24]. The loss function for training the baseline model and also adaptation (i.e., $J_{cross}(\theta)$ in our equations) is the cross-entropy between the forced aligned senone labels and the model outputs.

Model training – In all experiments, we use Adam optimizer to train or adapt the DNN models [25]. For training the original model with Libri, we use a learning rate (LR) of 0.001; however, we found that smaller learning rates perform better for adapting the original model. Our initial experiments showed that the learning rate of 0.0001 is an effective choice for the model expansion experiments. To train the original model, we employ the early-stopping technique to deal with the over-training problem. Early-stopping is performed by monitoring the performance of the model on a held-out validation set [26, 27]. However, applying early-stopping is not efficient for the domain expansion task [13]; it is because the data of the original domain is not available and performing the early-stopping only on the data of the new domain is just beneficial for the new domain, and it may significantly reduce the model performance on the original domain [13]. In

110
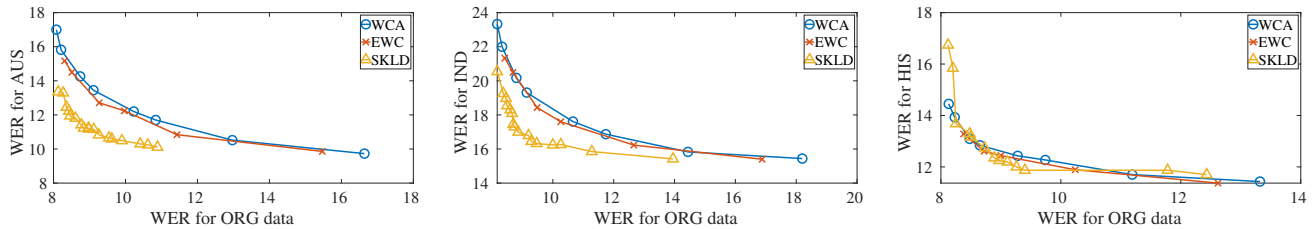
Fig. 1: Visualizing WER of different techniques on original and new datasets. These curves are generated by changing the hyper-parameters that control the trade-off between the performance of the original and the new domains.
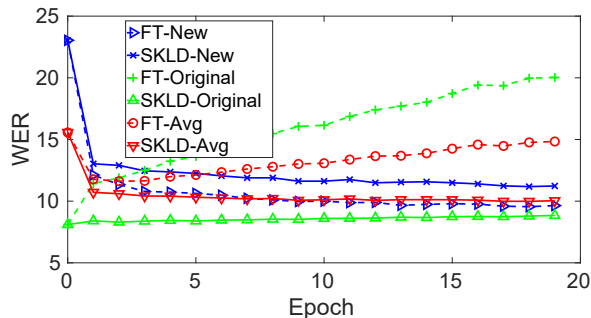
Fig. 2: Visualizing the forgetting effect of fine-tuning the original model to the AUS accent. The performance of SKLD in the same setting is also reported, which demonstrates the efficacy of SKLD to preserve the learned knowledge while adapting to the new domain. FT-Original: performance of the fine-tuned model on Libri; FT-New: performance of the fine-tuned model on AUS; FT-Avg: average of FT-Original and FT-new; SKLD-Original: performance of SKLD on Libri; SKLD-New: performance of SKLD on AUS; SKLD-Avg: average of SKLD-Original and SKLD-New

continual learning, a common approach is to perform a fixed number of iterations to train the new model [28]. In this study, we found that fine-tuning the original model converges to an optimum solution in 20 epochs. To investigate the efficacy of each approach, we evaluate their performance in three independent scenarios that consider IND, AUS, and HIS accents as the new domains.

3.2. Results

We conduct several experiments to evaluate the performance of the domain expansion methods explored in this study. As mention in previous sections, each domain expansion technique has a controlling hyper-parameter that provides a compromise between keeping the model performance for the original data and learning the new domains. In the first set of experiments, we tune these hyper-parameters to achieve the best overall performance for both the original and the new domains. The results for all approaches are summarized in Table 1. We also report the results of multi-condition training [29] in which the model is trained with both original and new

domains by pooling their data. The performance of the multi-condition system can be considered as an upper bound for the performance of domain-expansion methods.

In figure 1, we study the effect of changing the above-mentioned hyper-parameters for three model expansion techniques: WCA, EWC and SKLD. This figure shows which method is better in proving a trade-off between retaining the original model and learning the new domain.

Forgetting effect. The original model performs well for the Libri clean test set that matches the training set conditions; however, for the unseen domains, the performance of the model degrades significantly. Fine-tuning (FT) this model to the unseen domains results in a significant improvement in WER for the new domains, but the model performance for the original domain drops dramatically (Table 1). Figure 2 shows the rate of forgetting the information of the original data as we fine-tune the model to AUS accent (FT-Original). We also show how the SKLD approach performs in the same setting. SKLD can successfully preserve the model performance for the original data while learning the new data. The overall performance of the model on both old and new domains demonstrates that SKLD performs significantly better than naive fine-tuning for domain expansion.

The performance of WCA and EWC. WCA as a naive domain expansion method performs significantly better than fine-tuning in finding a compromise between the performance of the original and new domains. For EWC experiments, since the diagonal of the Fisher information matrix F is zero for many of the original model's weights, simply applying EWC does not preserve the model performance. We found that adding an empirically determined value of 1 to the elements of the matrix addresses the problem. EWC outperforms WCA in all the experiments (Table 1 and Figure 1), which demonstrates the efficacy of the Fisher information matrix in preserving the learned information of the original data. For example, for IND accent in Table 1, both approaches achieve 17.6 WER for the new data, while EWC achieves a relative WER improvement of +3.8% vs. WCA.

The performance of SKLD and Hybrid SKLD-EWC. SKLD significantly outperforms all other single domain-expansion approaches yielding a relative WER improvement of +8.8% and +7.6% vs. EWC for AUS and IND accent,

111

respectively (Table 1). For HIS accent, SKLD is still slightly better than EWC. The hybrid SKLD-EWC that benefits from both SKLD and EWC results in the best overall performance. Comparing the performance of domain expansion techniques with multi-condition training in Table 1 indicates that we have achieved comparable performance with multi-condition training which uses the original training data we consider unavailable for the domain expansion approaches.

4. CONCLUSIONS

In this paper, we explore several continual learning-based domain expansion techniques as an effective solution for domain mismatch problem in ASR systems. We examine the efficacy of the approaches through experiments on adapting a model trained with native English to three different English accents: Australian, Hispanic and Indian. We demonstrate that simply adapting the original model to the target domains results in a significant performance degradation of the adapted model for the original data. However, we demonstrate that SKLD and hybrid SKLD-EWC are effective in adapting the native English model to the new accents while retaining the performance of the adapted model for native English. The proposed SKLD-EWC outperformed other existing approaches such as fine-tuning, WCA, and EWC.

5. REFERENCES

[1] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, et al., "English conversational telephone speech recognition by humans and machines," *arXiv preprint arXiv:1703.02136*, 2017.

[2] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig, "Achieving human parity in conversational speech recognition," *arXiv preprint arXiv:1610.05256*, 2016.

[3] Shahram Ghorbani, Ahmet E Bulut, and John HL Hansen, "Advancing multi-accented lstm-ctc speech recognition using a domain specific student-teacher learning paradigm," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 29–35.

[4] Sining Sun, Binbin Zhang, Lei Xie, and Yanning Zhang, "An unsupervised deep domain adaptation approach for robust speech recognition," *Neurocomputing*, vol. 257, pp. 79–87, 2017.

[5] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation,"

in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 16–23.

[6] Salar Jafarlou, Soheil Khorram, Vinay Kothapally, and John H.L. Hansen, "Analyzing large receptive field convolutional networks for distant speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019.

[7] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[8] Zack Hodari, Oliver Watts, Srikanth Ronanki, and Simon King, "Learning interpretable control dimensions for speech synthesis by using external data," *Proc. Interspeech 2018*, pp. 32–36, 2018.

[9] John Gideon, Soheil Khorram, Zakaria Aldeneh, Dimitrios Dimitriadis, and Emily Mower Provost, "Progressive neural networks for transfer learning in emotion recognition," *arXiv preprint arXiv:1706.03256*, 2017.

[10] Tyler L Hayes, Nathan D Cahill, and Christopher Kanan, "Memory efficient experience replay for streaming learning," *arXiv preprint arXiv:1809.05922*, 2018.

[11] Timothy J Draelos, Nadine E Miner, Christopher C Lamb, Jonathan A Cox, Craig M Vineyard, Kristofor D Carlson, William M Severa, Conrad D James, and James B Aimone, "Neurogenesis deep learning: Extending deep networks to accommodate new classes," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 526–533.

[12] Xiao Li and Jeff Bilmes, "Regularized adaptation of discriminative classifiers," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. IEEE, 2006, vol. 1, pp. I–I.

[13] Zhizhong Li and Derek Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2018.

[14] Heechul Jung, Jeongwoo Ju, Minju Jung, and Junmo Kim, "Less-forgetful learning for domain expansion in deep neural networks," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[15] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[16] Davide Maltoni and Vincenzo Lomonaco, "Continuous learning in single-incremental-task scenarios," *arXiv preprint arXiv:1806.08568*, 2018.

[17] Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide, "Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7893–7897.

[18] Yevgen Chebotar and Austin Waters, "Distilling knowledge from ensembles of neural networks for speech recognition.," in *Interspeech*, 2016, pp. 3439–3443.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[20] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[21] Shahram Ghorbani and John HL Hansen, "Leveraging native language information for improved accented speech recognition," *Proc. Interspeech*, 2018.

[22] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," Tech. Rep., IEEE Signal Processing Society, 2011.

[23] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[25] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[26] Soheil Khorram, Mimansa Jaiswal, John Gideon, Melvin McInnis, and Emily Mower Provost, "The priori emotion dataset: Linking mood to emotion detected in-the-wild," *arXiv preprint arXiv:1806.10658*, 2018.

[27] Soheil Khorram, Melvin McInnis, and Emily Mower Provost, "Jointly aligning and predicting continuous emotion annotations," *IEEE Transactions on Affective Computing*, 2019.

[28] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler L Hayes, and Christopher Kanan, "Measuring catastrophic forgetting in neural networks," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[29] Michael L Seltzer, Dong Yu, and Yongqiang Wang, "An investigation of deep neural networks for noise robust speech recognition," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 7398–7402.