Towards Noise Robust Speech Emotion Recognition Using Dynamic Layer Customization

Alex Wilf
Computer Science and Engineering
University of Michigan
Ann Arbor, Michigan
abwilf@umich.edu

Emily Mower Provost
Computer Science and Engineering
University of Michigan
Ann Arbor, Michigan
emilykmp@umich.edu

***Abstract*—Robustness to environmental noise is important to creating automatic speech emotion recognition systems that are deployable in the real world. In this work, we experiment with two paradigms, one where we can anticipate noise sources that will be seen at test time and one where we cannot. In our first experiment, we assume that we have advance knowledge of the noise conditions that will be seen at test time. We show that we can use this knowledge to create "expert" feature encoders for each noise condition. If the noise condition is unchanging, data can be routed to a single encoder to improve robustness. However, if the noise source is variant, this paradigm is too restrictive. Instead, we introduce a new approach, dynamic layer customization (DLC), that allows the data to be dynamically routed to noise-matched encoders and then recombined. Critically, this process maintains temporal order, enabling extensions for multimodal models that generally benefit from long-term context. In our second experiment, we investigate whether partial knowledge of noise seen at test time can still be used to train systems that generalize well to unseen noise conditions using state-of-the-art domain adaptation algorithms. We find that DLC enables performance increases in both cases, highlighting the utility of mixture-of-expert approaches, domain adaptation methods and DLC to noise robust automatic speech emotion recognition.**

***Index Terms*—Deep learning, domain adaptation, affective computing, speech emotion recognition**

I. Introduction

Automatic emotion recognition provides an opportunity to understand how emotion patterns in daily life are associated with health, both mental and physical [1], [2]. The inexpensive production of audio recording-capable devices has made speech emotion recognition (SER) an attractive avenue for the deployment of emotion recognition systems. However, while recent advances in machine learning have led to improved accuracy in state-of-the-art SER systems, robustness to environmental noise in SER is still an open problem. In this work, we demonstrate that we can successfully customize feature encoders to noise conditions known at training time; apply domain adaptation methods commonly used to generalize performance across datasets to the task of generalizing across noise conditions; and extend these improvements to the multimodal setting using a process we describe as Dynamic Layer Customization (DLC).

In considering how a SER system bound for real-world deployment would be developed, it is reasonable to assume that the system's designers may have some knowledge of the noise conditions that the system will face at deployment time, either through empirical studies or expert knowledge. If designers do not have this knowledge, we assume that they will also not have access to unlabelled data from the test distribution to fine tune their models. To the best of our knowledge, our work is the first to use knowledge of the type of noise present in each sample to customize feature encoders for each noise condition in an end-to-end trained speech emotion recognition network where all noise conditions are seen during training.

In our first Experiment, we ask the question: *How well could we recognize emotion from speech if we had access to all test noise conditions at train time?* This may seem like a quixotic scenario, but we posit that it is not as far fetched as it may appear. The situation where a system designer has access to all test noise conditions at train time may be justified when a small number of noise conditions constitutes a large majority of samples. This is a common paradigm for technology deployed in a static environment, for example, a single clinic.

In this Experiment, we find that by training a mixture of experts model with individual feature encoders for each noise condition, we can improve performance over a system with a single feature encoder. A simple method for implementing a mixture-of-experts model in this setting would be to group utterances within the dataset by noise condition, and train each "expert" on its associated subdataset. We call this *Static Layer Customization (SLC)*. Consider a potential training instance consisting of an individual recording a video diary entry. First, their dog barks in the background (noise type 1), then a fan turns on (noise type 2), then their family member walks by laughing (noise type 3). This type of noise variation is natural in real life deployments [3]. We could handle this variation by segmenting this instance into three regions of uniform noise, but doing so would potentially disrupt the temporal flow of the diary entry. In fact, it is this very flow of temporal information that allows many end-to-end trained state-of-the-art multimodal (text+acoustic) networks to understand long term context [4], [5]. Therefore, although models performing SLC to route samples to experts may enable noise robustness in individual samples, this approach is likely not optimal in end-to-end trained state-of-the-art multimodal (text+acoustic) networks.

978-1-6654-0019-0/21/$31.00

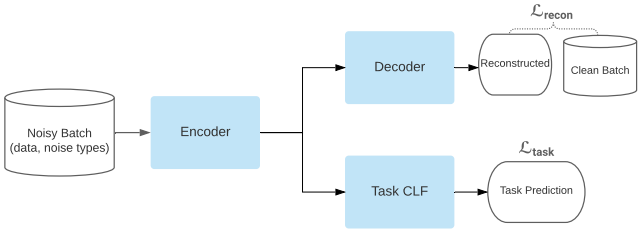


Fig. 1: Our baseline model. Samples are passed through a feature encoder, then through a decoder and an emotion classifier without any generalization on noise condition.

We present an alternative approach, one that dynamically routes samples during training to different feature encoders based on properties of the noise conditions that are present. It then recombines the outputs in the original order before passing the outputs to the remainder of the network. We call this process *Dynamic Layer Customization (DLC)*.

In our second Experiment, we ask: *How can we use knowledge of the types of noise present in each sample at train time to generalize across noise conditions seen at train time to unseen noises at test time?* We support the finding, from related speech tasks, that domain adaptation can be used to generalize across noise conditions [6], [7], and extend these findings to the case where unlabelled target information is not provided to the network during training. We show that domain adaptation methods can lead to significant performance improvements over their ungeneralized counterparts. We run preliminary experiments to identify the most effective domain adaptation method on unimodal data. We then choose the highest performing domain adaptation method over these experiments, Domain Separation Networks (DSN) [8], to extend to the multimodal setting. DSN uses separate feature encoders for each domain. However, as discussed earlier, this strict adherence to uniformity in the noise samples renders it brittle to real world noise-varying conditions. Instead, we augment DSN with DLC (DSN-DLC) to enable it to generalize across noise conditions without disrupting the temporal ordering of samples.

In the remainder of this paper, we detail the related work (Section II), data (Section III), network architectures (Section IV), training procedure (Section V), experiments (Section VI) and results (Section VII). In these sections, we consider whether a mixture-of-experts approach and domain adaptation methods can improve speech emotion recognition networks' performance in the presence of noise, both when that noise is seen beforehand and when it is left unseen. Through DLC, our novel approach which dynamically routes samples through a network without disrupting their temporal ordering, we show that both mixture-of-experts and domain adaptation methods can be extended to the multimodal case, resulting in performance improvements in that setting as well.

II. Related Work

Our work builds on recent advances in domain adaptation methods such as Domain Adversarial Neural Network (DANN) [9], Multiclass Adversarial Discriminative Domain Generalization (MADDoG) [10] and Domain Separation Networks (DSN) [8]. We leverage each of these approaches in our second Experiment and describe them in Section IV. Our work is also inspired by the promising performance of mixture-of-experts models applied to deep learning problems [11], [12]. Finally, as we extend our results to the multimodal setting, there are many related works to the state-of-the-art model we use. Zadeh et al.'s TensorFusion [13] introduced a method of outer product pooling that, when combined with Poria et al.'s approach to cross-utterance modeling [4] yielded the foundation of Hierarchical Feature Fusion (HFFN) [5], the state-of-the-art model we use to extend our findings to the multimodal setting. Our work is also influenced by the literature on speech enhancement. Liao's paper [6] similarly used domain adaptation methods to generalize across noises, but did so by using unlabelled target data - an assumption we do not make in this work.

III. Data

A. Datasets

We consider three datasets in our experiments. **MSP-Improv** is an acted, audiovisual emotional database which aims to approach the naturalness of unsolicited human interactions by asking the actors to embed a "target sentence" into an improvised interaction [14]. MSP-Improv was collected over six sessions with twelve actors and contains 8,438 utterances, each labelled for valence and activation on a scale of 1-5. We convert the n valence ratings into a three bin vector describing the distribution of the sample over "low", "medium", and "high" valences by binning ratings below, equal to, and greater than the midpoint (3) and dividing by n as in [10]. **MOSEI** contains 23,500 utterances extracted from "in the wild" videos on YouTube, labelled for sentiment in the range -3 to 3 [15]. We also consider negative, neutral, and positive bins for MOSEI, this time by partitioning ratings with 0 as the midpoint. **IEMOCAP** was collected over five sessions from ten actors (five male, five female) [16]. Each of the 10,039 utterances is labelled with emotional categories (e.g., anger, happiness, sadness, neutrality) and dimensional labels (i.e., valence, activation, dominance). Though we use dimensional labels to evaluate results on MSP-Improv and MOSEI, we evaluate performance on IEMOCAP using categorical labels to be consistent with prior work [4], [5].

B. Noise Augmentation

We use Librosa, along with the ESC-50 environmental noise dataset [17], to overlay environmental noise with different signal to noise ratios (SNR). In our experiments, we add noise to the instances in each dataset in different profiles, selecting randomly from among three noise conditions comprising both a noise type – either "natural", "human", or "interior" – and a signal to noise ratio (SNR). After selecting a noise condition,

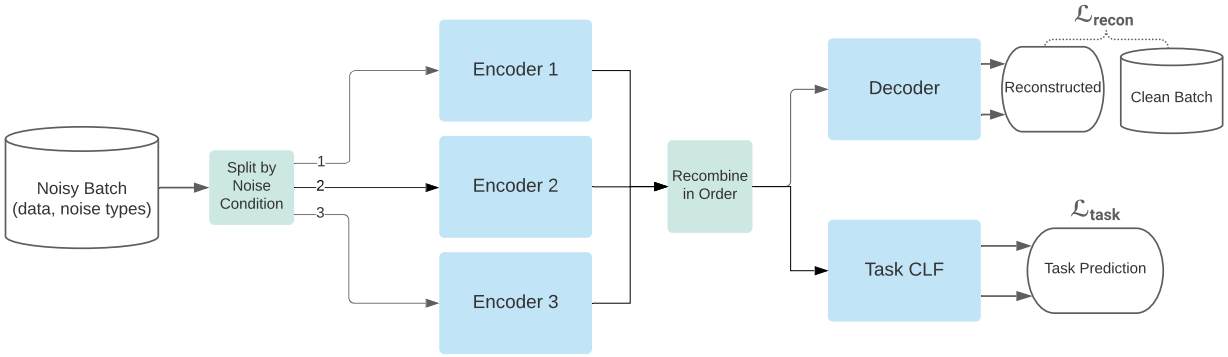


Fig. 2: The Mixture-of-Experts (MoE) DLC model. During training, batches are split by noise condition type, and samples are routed to "expert" encoders for based on the related noise condition dynamically.

we overlay the original sample with a randomly chosen audio file from that category (natural, human, or interior) at the given SNR (lower means noisier). Our heterogeneous noise profiles (h1, h2, and h3) are inspired by real life situations. h1 is inspired by a grocery store environment, with (natural, interior, human) SNR values of (-5, -20, -20). h2 is inspired by a sidewalk environment, with values of (-20, -1, -5). h3 is inspired by an interior environment, with values of (-5, -30, -10). Each instance in each dataset will be noised three times, once for each noise profile.

C. Feature Extraction

We use the Librosa Python library [18] to extract 40 dimensional log Mel Filterbanks (MFB), which have shown effectiveness in SER [10], [19]. We extract transcripts over the noise enhanced audio files using Deepspeech [20] and pre-process the transcripts using a pretrained 512-dimensional BERT network [21].

IV. Network Architectures

In this section, we first provide an overview for the emotion recognition architectures that we use. Next, we describe the dynamic layer customization (DLC) and static layer customization (SLC) approaches. Then, we describe how DLC and SLC can be used to augment existing architectures. Finally, we provide a summary of the models that will be used in Experiments 1 and 2 (Section VI).

A. Emotion Recognition Architectures

1) Baseline: Our baseline network consists of a feature encoder linked to both an emotion classifier and a decoder. Our architectures for each component (feature encoder, classifier, and decoder) are constant across all our networks and are based on Khorram et al.'s approach to unimodal acoustic emotion recognition using dilated convolutions [19]. In other words, each time we refer to a feature encoder and classifier, those architectures are the same as described below.

The **feature encoder** is implemented with three 1-D convolution layers with kernel size 16, 128 feature maps, and dilation rates increasing by powers of two with each successive layer as in [10], [19], followed by a 1-D MaxPool with pool size 4 and 4 strides. The **decoder layer** for feature reconstruction consists of two 1-D convolution layers with 128 and 40 feature maps, kernel size 3, and 2 strides, followed by a single 1-D convolution layer with 40 feature maps, kernel size 3, and 1 stride to mirror the encoder. The **classifier** layer consists of three dense layers with two 128 unit layers followed by a softmax layer where the number of units is the number of emotion bins (classifier).

We test unimodal and multimodal variations of this architecture that leverage customization and domain adaptation for our experiments. In the subsections that follow, we describe the details of the domain adaptation methods we test and the multimodal setting of the task.

2) Mixture of Experts (MoE) Model: The MoE model is an augmentation of the baseline model. In the baseline model there is a single feature encoder that is assumed to function over all noise categories. The MoE model does not make that assumption and introduces separate feature encoders that are specifically trained for each category of noise. We will discuss the gating strategies for noise type later in this section. The network still relies upon a shared classifier and decoder, identical to the one seen in the baseline system.

3) Domain Adversarial Neural Network (DANN): The Domain Adversarial Neural Network [9] is an approach to domain adaptation in which features are passed through a feature encoder, then the encoded features are passed through a task classifier and an adversarial domain classifier (the latter is preceded by a gradient reversal layer). The adversary is structured sinilarly to the classifier layer, except that the number of units in the final layer is the number of noise conditions. In this way, the feature encoder is encouraged to output encodings such that using those encodings, the task classifier is able to predict the task, but the domain classifier is unable to predict

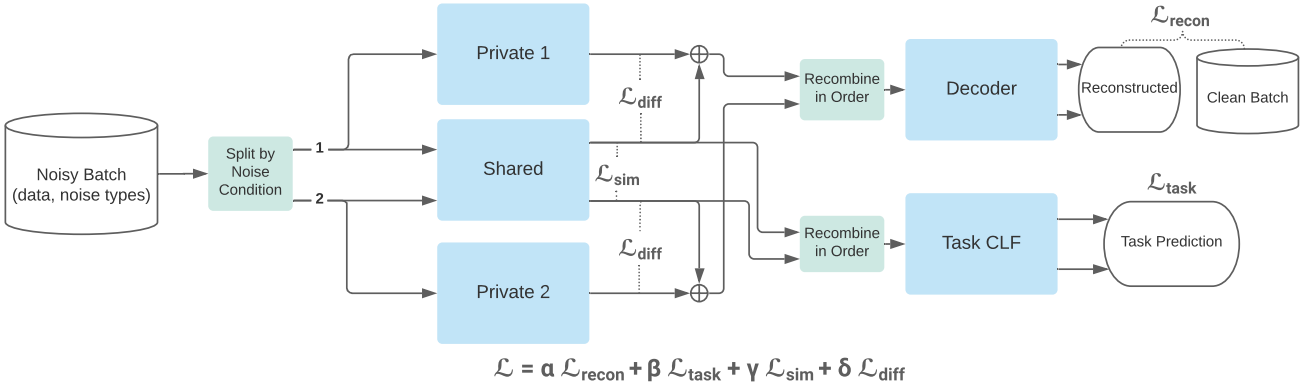


Fig. 3: The Domain Separation Network with Dynamic Layer Customization. Samples overlaid with different environmental noises are first split by noise condition, then passed through separate encoders and a shared emotion classifier (CLF) designed to separate signal from noise for robust emotion classification. $\mathfrak{L}$ represents the total loss, $\mathfrak{L}_{diff}$ is the difference between shared and private encoders, $\mathfrak{L}_{recon}$ is the reconstruction loss, $\mathfrak{L}_{task}$ is the task classification loss, and $\mathfrak{L}_{sim}$ is the loss between samples from different domains. The parameters α, β, γ, and δ are hyperparameters.

the domain. DANN has shown promising results on cross-corpus vision tasks, and has been successfully applied to SER, though the authors noted that it had difficulty converging [22]. We use DANN in Experiment 2 to generalize to our left out noise condition.

4) Multiclass Adversarial Discriminative Domain Generalization (MADDoG): The Multiclass Adversarial Discriminative Domain Generalization network [10] is a variation on DANN where the adversary (called a "critic") uses a linear activation with loss based on WGAN-style "earth mover's distance" [23] instead of cross-entropy loss with a softmax activation. The critic is trained separately at the beginning of each epoch and then is frozen. MADDoG has shown promising results on the task of SER in domain generalization – a domain adaptation variation where some labelled test samples are available at training. We use MADDoG in Experiment 2 as well to generalize to unseen noise conditions.

5) Domain Separation Network (DSN): The purpose of the Domain Separation Network (DSN) [8] is to learn a "shared" encoder that extracts features that are *generalizable* across domains. DSN achieves this by learning "private" feature encoders that encode the parts of a sample *unique* to each domain, trained with losses to encourage that, for each sample: the shared and private encoders yield different representations ($\mathfrak{L}_{diff}$); all information relevant to reconstruction is captured by the normalized sum of the outputs of the shared and private encoders ($\mathfrak{L}_{recon}$); the shared encoding is sufficient to classify the task ($\mathfrak{L}_{task}$); and the outputs of the shared encoder for samples from different domains are so similar as to be indistinguishable by a DANN-style adversary ($\mathfrak{L}_{sim}$). In the original paper, DSN created batches by randomly sampling from each domain.

6) Hierarchical Feature Fusion Network (HFFN): We use a state-of-the-art multimodal fusion network: Hierarchical Feature Fusion Network (HFFN) [5]. HFFN extracts independent features for each modality (i.e., lexical, acoustic) before "fusing" them and learning to recognize emotion using cross-utterance context. HFFN achieves this in three steps: *divide*, *conquer*, and *combine*. HFFN *divides* the intermediate feature space up into local blocks corresponding to neighboring utterances, *conquers* by taking the outer product of the features from neighboring utterances across modalities, and *combines* by using a modified LSTM layer to learn contextual relationships between utterances close to and farther from each other. We refer the reader to Mai et al. [5] for additional details.

B. SLC and DLC Noise Gating Functions

In this section, we describe the different gating functions that are used to augment the existing architectures. In all cases, we use a noise classifier that is trained to recognize noise type. This classifier uses the same architecture as the feature encoder and classifier, described in the baseline model.

We use this classifier for Mixture-of-Experts models at test time in both SLC and DLC. SLC uses a common assumption while training noise robust systems: that the data can be segmented into groups with common noise types. This assumption is valid when there is no value to retaining the temporal ordering of the utterances or when the noise types are consistent over all instances in a section of data. During training, SLC uses oracle noise labels to separate the data into groups of consistent noise categories. During testing, it uses the noise classifier to automatically do this separation.

DLC addresses this challenge associated with segmenting out noise categories by providing dynamic routing to the different feature encoders based on the properties of the noise.

During training, DLC takes in a batch of heterogeneously noised utterances, with the oracle noise labels. It separates the instances by noise category and individually trains the feature encoders. Then, it reorders the utterances, retaining the temporal ordering of the data. During testing, DLC then automatically estimates noise type, routing the instances to the correct encoder, and then recombining in the original temporal order.

The advantage to using DLC, over SLC, comes from the preservation of temporal ordering. Therefore, it is expected that the results will show improvements in using DLC vs. SLC only in the multimodal test cases, where temporal ordering has been shown to substantially improves performance [4]. In the unimodal cases, the models that we use do not consider the temporal ordering of the instances and we therefore do not expect to see improvement gains in this context. We may see subtle differences resulting for variations due to batches being split dynamically and other sources of randomness seen during training. We present results from both unimodal and multimodal tests for completeness.

C. Altering Existing Networks with SLC and DLC

The MoE and DSN unimodal networks already have the notion of routing built in. However, the routing has historically taken the SLC approach during training and testing. We replace the SLC routing with DLC routing during both training and testing. This leads to MoE-SLC/DLC and DSN-SLC/DLC.

The HFFN multimodal network is the network where the maintenance of temporal ordering is assumed. In this case, we replace the acoustic feature encoder, but not the lexical feature encoder, with a domain adaptive/generalized encoder augmented with DLC. Future work will explore how to do an equivalent process for lexical information.

V. Model Training

We split the videos in each dataset into train (70%), validation (10%) and test (20%) sets. The data are split before noise enhancement, thus an original and noise enhanced sample are always in the same set. These splits are used for both the emotion classification and noise classification model training.

For each emotion classification system, we use the Adadelta optimizer with learning rate $1e{-}3$, cross entropy loss for the emotion classifier, and mean squared error loss between the clean input and the output, given noise enhanced input, to encourage denoising. We use early stopping training with a batch size of 32 and a patience of ten epochs (max 1000) and report results from the model with the best validation metric. Here we use Unweighted Average Recall (UAR) as our metric, in line with work in the field [10], [24]. The noise classifier, used as the gating function for MoE models at test time (Section IV-B), is trained with the same optimizer and learning rate over the same splits as the emotion classification models. All experiments were run with Tensorflow 2.0 on a single machine across 3 GPUs: 1x GTX 1080, 2x Titan X. The code to reproduce our results can be found online.[1]

[1]https://github.com/abwilf/mmfusion

VI. Experiments

We present two experiments to understand how we can increase noise robustness in SER. In our first Experiment, we assume that all noise conditions are known in advance. In our second Experiment, we again assume multiple noise conditions, but one of the noise conditions is not known a priori.

A. Experiment 1

We first consider the case where all test noise conditions are seen at train time, and test whether the SLC and DLC routing approaches, permitting multiple feature encoders – one for each noise condition – will outperform the baseline network. The models we test are detailed below.

1) *Single*: Our baseline network consists of an acoustic feature encoder connected to a decoder (to encourage denoising through reconstruction loss with the clean input), and a task classifier (to learn to predict emotion). Each of these components is described in detail in Section IV.
2) *MoE-SLC*: In this network, we examine statically splitting based on noise condition. We train separate feature encoders for each noise condition, with shared classifier and decoder layers, and use the noise predictor described above for gating at test time.
3) *MoE-DLC*: Next, we examine dynamically splitting based on noise condition using DLC. The difference, moving from *MoE-SLC* to *MoE-DLC*, is that *MoE-DLC* splits batches apart by noise condition during training instead of splitting the full dataset into batches by noise condition before training. We would expect that the two models perform similarly in the unimodal condition in which they are tested.
4) *HFFN-Single*: We extend the baseline unimodal model to the multimodal setting using both lexical and acoustic features, using a single acoustic feature encoder and BERT embeddings.
5) *HFFN-MoE-DLC*: We incorporate a mixture-of-experts approach to our acoustic feature encoder, using DLC to route samples based on noise condition at train time so our mixture-of-experts approach can be compatible with HFFN's end-to-end training without disrupting the order of samples. A *HFFN-MoE-SLC* model would not be possible because it would require a disruption to the sample order of the data. We anticipate that this model will outperform *HFFN-Single* due to the strength of its mixture-of-experts approach to noise robustness in its acoustic feature encoders.

B. Experiment 2

Next, we investigate whether performance can be improved in the case where we leave one noise condition out during training (a common testing approach in domain adaptation problems [8], [10]). In this Experiment, we train the network on data from two noise conditions and test on the third and examine whether the domain adaptation methods described in

Section IV can encourage the feature encoders to generalize across noise conditions. We expect that networks trained using domain generalization algorithms will generalize better to the held out noise than our baseline. We identify the method with the highest average UAR across datasets from the unimodal tests, DSN, and use it to extend to the multimodal setting in combination with HFFN. Again, in this Experiment we report the average performance of the model, leaving each noise condition out once as the unseen noise condition.

In this Experiment, we use models 1-4 from Experiment 1. We add to these models the following:

1) *DANN*: We test whether DANN, a known and effective state-of-the-art domain adaptation method [25], can be used effectively in noise adaptation in the unimodal context.
2) *MADDoG*: We test whether MADDoG, a known and effective state-of-the-art domain generalization method [10], can be used effectively in noise adaptation in the unimodal context.
3) *DSN-SLC*: The original implementation of DSN created batches by randomly sampling from each domain. This is the protocol used in the standard SLC routing protocol.
4) *DSN-DLC*: DSN-SLC's random sampling disrupts the temporal ordering of the data. Through DLC, we can both randomly sample and maintain temporal ordering, allowing us to investigate how DSN performs in the multimodal setting with HFFN. This network architecture is described in Figure 3.
5) *HFFN-DSN-DLC*: HFFN requires the maintenance of temporal ordering, which is disrupted by the random sampling inherent in DSN batching. We leverage the advantages of HFFN and DSN by using DLC to dynamically route samples to noise-specific private encoders and back for reconstruction and classification.

VII. Results

We first discuss the accuracy of the noise-source classification, necessary to accurately route samples to different feature encoders. The noise classifier obtained an average over datasets and noise profiles of 87% accuracy at predicting noise condition labels on the test set (across noise profiles, see Section IV-B) over the three datasets (see Section III).

A. Experiment 1

We first analyze the performance of our system on unimodal acoustic data. Our results support our hypothesis that a mixture of experts network with multiple feature encoders specialized for particular noise conditions (MoE-SLC) outperforms our baseline network with a single feature encoder (Single). In the unimodal setting, the mixture of experts network implemented dynamically, MoE-DLC, performs similarly to the same network implemented statically, MoE-SLC, and shows an average improvement of 2.13% UAR across noise profiles and datasets over the baseline network. This is expected as SLC and DLC are functionally equivalent in the unimodal setting.

In the multimodal setting, the average difference between HFFN-Single and HFFN-MoE-DLC is 1.63%. We note that the DLC routing takes into account only the noise-variations in the acoustic content, not the lexical content. Yet, lexical content contributes strongly to SER performance. Therefore, we anticipate that a larger performance gain would be possible if lexical embeddings were also targeted. Future work may find more improvements in the multimodal setting by using denoised audio for transcription.

B. Experiment 2

Again, we first analyze the performance of our system on unimodal acoustic data. In contrast with Experiment 1, we found that the mixture of experts network with multiple feature encoders, MoE-SLC and MoE-DLC, did not outperform the baseline with a single feature encoder. We believe that this is because there is no generalization within the specialized feature encoders, so each encoder is poorly equipped to handle samples that contained novel sources. Put another way, the network is attempting to fit the test noise condition into one of the train noise condition "buckets", where it does not belong.

We found that unimodal domain adaptation methods significantly improved upon the baseline (Single) and mixture of experts (MoE) approaches. Of these methods, DSN performed the best, improving over a single, ungeneralized feature encoder by an average of 2.49%. We implement both DSN-SLC and DSN-DLC, anticipating that their results will be similar, and indeed they are, differing in their $\bar{h}$ values (performance averaged over noise profiles and datasets) by only 0.13%.

In the multimodal setting, we implemented DSN with DLC as part of HFFN (HFFN-DSN-DLC), and found an average improvement of 2.06% over our multimodal network with the single ungeneralized feature encoder (HFFN-Single).

VIII. Discussion

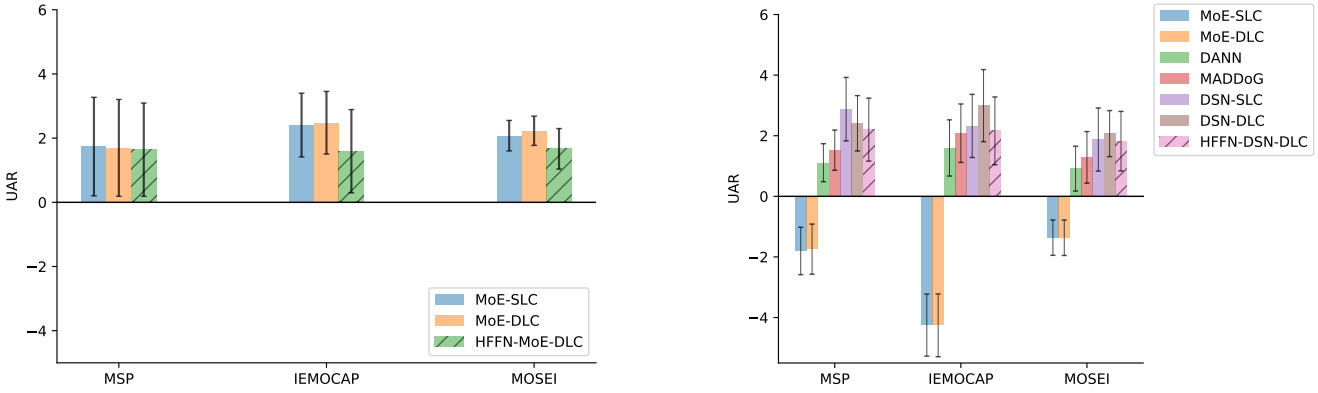
In our experiments, we find that our DLC approach unlocks significant performance increases by extending methods' applicability to the multimodal setting. We further observe that a mixture-of-experts approach can improve noise robustness when deployment noise conditions are seen during training (Figure 4a). However, when noise conditions are novel at test time, approaches that leverage encoders trained for particular noise profiles (e.g., MoE) are not effective. Instead, domain adaptation or generalization methods can be augmented with DLC, to improve over the baseline, even when the baseline includes a multimodal implementation (Figure 4b).

In Experiment 1, our two unimodal Mixture-of-Experts models perform similarly as expected. Yet, it is in the multimodal setting that we both expect and observe performance changes. Through DLC, we can extend performance gains derived from the HFFN cross-utterance multimodal emotion recognition network to a noise-enhanced condition.

In Experiment 2, we observe that when our Mixture-of-Experts models are asked to assign an unseen noise to one of their encoders trained on a different noise type, they perform substantially worse than the baseline system because they

TABLE I: UAR results for all methods on both experiments across datasets. Non-italic text indicates the unimodal setting, italic text indicates a multimodal setting. The final column shows the average performance by noise conditions and then over the three noise conditions, h1, h2, and h3 and averaged over all noise conditions, $\bar{h}$.

	Dataset	**MSP**			**IEMOCAP**			**MOSEI**			**Overall**			
	Noise Profile	h1	h2	h3	h1	h2	h3	h1	h2	h3	h1	h2	h3	$\bar{h}$
Experiment 1	Single	51.39	53.51	54.99	58.29	59.01	60.55	42.25	42.98	43.40	50.64	51.83	52.98	51.82
	MoE-SLC	53.05	55.10	56.95	60.21	62.41	62.45	44.50	44.75	45.61	52.59	54.09	55.00	53.89
	MoE-DLC	53.12	54.97	56.89	60.36	62.23	62.70	44.74	44.86	45.72	52.74	54.02	55.10	53.95
	HFFN-Single	57.47	59.75	60.68	63.98	66.17	66.88	49.41	48.97	50.22	56.95	58.30	59.26	58.17
	HFFN-MoE-DLC	**58.98**	**61.05**	**62.79**	**65.37**	**67.96**	**68.48**	**51.44**	**50.19**	**51.97**	**58.60**	**59.73**	**61.08**	**59.80**
Experiment 2	Single	49.58	51.28	51.70	55.81	56.94	57.92	39.38	40.53	40.91	48.26	49.58	50.18	49.34
	MoE-SLC	48.21	49.78	49.16	52.24	51.43	54.25	38.21	39.08	39.44	46.22	46.76	47.62	46.87
	MoE-DLC	48.16	49.96	49.21	52.18	51.43	54.29	38.19	39.12	39.41	46.18	46.84	47.64	46.89
	DANN	51.50	52.10	52.29	57.40	58.26	59.80	40.15	41.23	42.18	49.68	50.53	51.42	50.54
	MADDoG	51.88	52.88	52.37	58.16	58.28	60.48	40.19	41.75	42.74	50.08	50.97	51.86	50.97
	DSN-SLC	53.43	52.46	55.3	58.59	60.92	58.13	41.74	40.63	44.07	51.25	51.34	52.50	51.70
	DSN-DLC	52.22	53.13	54.44	57.92	60.12	61.61	41.21	42.50	43.32	50.45	51.92	53.12	51.83
	HFFN-Single	55.83	57.29	58.53	62.70	64.83	66.02	45.62	46.66	47.23	54.72	56.26	57.26	56.08
	HFFN-DSN-DLC	**58.28**	**59.31**	**60.66**	**66.08**	**66.05**	**67.90**	**46.66**	**48.48**	**49.83**	**57.01**	**57.95**	**59.46**	**58.14**



(a) Experiment 1.

(b) Experiment 2.

Fig. 4: Results relative to baselines, averaged across noise profiles, visualized with one standard deviation error bands depicted in black. The relevant baseline is "Single" for unimodal networks, and "HFFN-Single" for multimodal networks. Striped bars indicate a multimodal model, meaning that its baseline is "HFFN-Single", not "Single".

lack the capability to generalize. Yet, we find that domain adaptation algorithms can be used to generalize to unseen noise conditions, and that dynamically routing samples to noise-matched feature encoders can be used to extend the performance improvements derived from temporal ordering in the multimodal setting to these noise-enhanced data.

We anticipate that future work may find further improvements by exploring a network that leverages both the mixture-of-experts approach and domain generalization, assigning a sample to a "known" encoder if the noise predictor is confident enough that the type of noise has been seen during training, and assigning the sample to the generalized encoder if not. We are also interested in whether future work may find that customizing to and generalizing across noise conditions in the ways here could inform noise robust speech recognition tasks which, apart from their own importance, could also lead to more noise-robust multimodal emotion recognition systems.

IX. Conclusion

In this paper, we present the findings of our investigation into mixture of experts models and domain adaptation in noise robust speech emotion recognition networks, both unimodal and multimodal. We show that specializing feature encoders to noise conditions by using sample-level noise information during training when all noise conditions are known and by using domain adaptation algorithms (without unlabelled test data) when one noise condition is left out can improve our algorithms' robustness to noise. To extend our findings to the multimodal setting, we present Dynamic Layer Customization (DLC) as a way to route and recombine samples to preserve their temporal ordering. We believe that this work will provide an important baseline for future work in noise robustness to test against, and that our findings will help shape the deployment of SER systems in the real world.

X. Acknowledgments

This work was supported by Cisco Research, the National Science Foundation (NSF CAREER 1651740), and Precision Health at the University of Michigan. Special thanks go to Amir Zadeh, Zakaria Aldeneh, and Katie Matton.

REFERENCES

[1] E. L.-C. Law, S. Soleimani, D. Watkins, and J. Barwick, "Automatic voice emotion recognition of child-parent conversations in natural settings," *Behaviour & Information Technology*, pp. 1–18, 2020.

[2] R. Beale and C. Peter, "The role of affect and emotion in hci," in *Affect and emotion in human-computer interaction*. Springer, 2008, pp. 1–11.

[3] Z. Zhang, F. Ringeval, J. Han, J. Deng, E. Marchi, and B. Schuller, "Facing realism in spontaneous emotion recognition from speech: Feature enhancement by autoencoder with lstm neural networks," in *Interspeech 2016*. International Speech Communication Association (ISCA), 2016.

[4] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, and L.-P. Morency, "Context-dependent sentiment analysis in user-generated videos," in *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*, 2017, pp. 873–883.

[5] S. Mai, H. Hu, and S. Xing, "Divide, conquer and combine: Hierarchical feature fusion network with local and global perspectives for multimodal affective computing," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 481–492.

[6] C.-F. Liao, Y. Tsao, H.-Y. Lee, and H.-M. Wang, "Noise adaptive speech enhancement using domain adversarial training," *arXiv preprint arXiv:1807.07501*, 2018.

[7] Z. Meng, Z. Chen, V. Mazalov, J. Li, and Y. Gong, "Unsupervised adaptation with domain separation networks for robust speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 214–221.

[8] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Advances in neural information processing systems*, 2016, pp. 343–351.

[9] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[10] J. Gideon, M. McInnis, and E. M. Provost, "Improving cross-corpus speech emotion recognition with adversarial discriminative domain generalization (addog)," *IEEE Transactions on Affective Computing*, 2019.

[11] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," *arXiv preprint arXiv:1701.06538*, 2017.

[12] M. Perez, Z. Aldeneh, and E. M. Provost, "Aphasic speech recognition using a mixture of speech intelligibility experts," *arXiv preprint arXiv:2008.10788*, 2020.

[13] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Tensor fusion network for multimodal sentiment analysis," *arXiv preprint arXiv:1707.07250*, 2017.

[14] C. Busso, S. Parthasarathy, A. Burmania, M. AbdelWahab, N. Sadoughi, and E. M. Provost, "Msp-improv: An acted corpus of dyadic interactions to study emotion perception," *IEEE Transactions on Affective Computing*, vol. 8, no. 1, pp. 67–80, 2016.

[15] A. B. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L.-P. Morency, "Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 2236–2246.

[16] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, p. 335, 2008.

[17] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1015–1018.

[18] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, 2015, pp. 18–25.

[19] S. Khorram, Z. Aldeneh, D. Dimitriadis, M. McInnis, and E. M. Provost, "Capturing long-term temporal dependencies with convolutional networks for continuous emotion recognition," *arXiv preprint arXiv:1708.07050*, 2017.

[20] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[22] M. Abdelwahab and C. Busso, "Domain adversarial for acoustic emotion recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 12, pp. 2423–2435, 2018.

[23] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[24] M. Chen, X. He, J. Yang, and H. Zhang, "3-d convolutional recurrent neural networks with attention model for speech emotion recognition," *IEEE Signal Processing Letters*, vol. 25, no. 10, pp. 1440–1444, 2018.

[25] H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, and M. Marchand, "Domain-adversarial neural networks," *arXiv preprint arXiv:1412.4446*, 2014.