SMOOTHING MODEL PREDICTIONS USING ADVERSARIAL TRAINING PROCEDURES FOR SPEECH BASED EMOTION RECOGNITION

Saurabh Sahu[1]*, Rahul Gupta*[2]*, Ganesh Sivaraman*[1]*, Carol Espy-Wilson*[1]

[1]Speech Communication Laboratory, University of Maryland, College Park, MD, USA
[2]Amazon.com, USA

ABSTRACT

Training discriminative classifiers involves learning a conditional distribution $p(y_i|\boldsymbol{x}_i)$, given a set of feature vectors $\boldsymbol{x}_i$ and the corresponding labels $y_i, i = 1..N$. For a classifier to be generalizable and not overfit to training data, the resulting conditional distribution $p(y_i|\boldsymbol{x}_i)$ is desired to be smoothly varying over the inputs $\boldsymbol{x}_i$. Adversarial training procedures enforce this smoothness using manifold regularization techniques. Manifold regularization makes the model's output distribution more robust to local perturbation added to a datapoint $\boldsymbol{x}_i$. In this paper, we experiment with the application of adversarial training procedures to increase the accuracy of a deep neural network based emotion recognition system using speech cues. Specifically, we investigate two training procedures: (i) adversarial training where we determine the adversarial direction based on the given labels for the training data and, (ii) virtual adversarial training where we determine the adversarial direction based only on the output distribution of the training data. We demonstrate the efficacy of adversarial training procedures by performing a k-fold cross validation experiment on the Interactive Emotional Dyadic Motion Capture (IEMOCAP) and a cross-corpus performance analysis on three separate corpora. Results show improvement over a purely supervised approach, as well as better generalization capability to cross-corpus settings.

Index Terms— Adversarial training, Manifold regularization, Speech emotion recognition

1. INTRODUCTION

Design of emotion recognition systems is a classical problem with applications in several fields including medicine, analysis of human interaction and other behavioral studies [1]. An emotion recognition system design involves extracting cues from facial expressions, speech or body language and expressions and depict them as feature representations. This is followed by training a classification algorithm using existing supervised/semi-supervised methods [2, 3]. We consider a setting with N training examples $\{\boldsymbol{x}_i, y_i\}, i = 1, .., N$, where $\boldsymbol{x}_i$ is the obtained feature representation for example i and y_i is the corresponding label. Let $\boldsymbol{x}$ and y denote the random variables of which $\boldsymbol{x}_i$ and y_i are instances. A typical supervised learning approach involves modeling the probability $p(y|\boldsymbol{x})$ using a chosen functional form (e.g. neural networks, support vector machine classifier). For the chosen model (trained on finite training data) to generalize well to unseen data, the probability $p(y|\boldsymbol{x})$ is desired to have certain properties [4]. One such property is the smoothness of the distribution $p(y|\boldsymbol{x})$ which states that if two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are close to each other in feature space (based on some distance metric) then so should be their corresponding model outputs $p(y_i|\boldsymbol{x}_i)$ and $p(y_j|\boldsymbol{x}_j)$. The underlying idea is for the classifier to be generalizable and not overfit to training data. Enforcing this smoothness can be particularly useful for low resource tasks such as emotion recognition, where collecting a large number of labeled data instances may not always be possible. Methods such as manifold regularization impose this smoothness by modifying the optimization objective [5]. Manifold regularization exploits the distribution $p(\boldsymbol{x})$ as available through a set of labeled/unlabeled points to better estimate $p(y|\boldsymbol{x})$ thereby leveraging the concept of manifold learning to enforce model smoothness.

In the past, researchers have investigated manifold learning methods for speech based emotion recognition. Most of these methods attempt to learn the manifold by reducing the dimensionality of the input feature space and subsequently feeding them to a classifier. For example, [6, 7] employed isometric feature embedding for deriving the manifolds and then used Gaussian Mixture Models as classifiers. You et al. [8] employed Lipschitz embedding for non-linear manifold learning in an unsupervised way followed by using support vector machines for classification. Qian et al. [9] applied a supervised manifold learning method by considering the difference between feature subsets of different classes and reported improvement in recognition accuracy. However, none of them have investigated manifold regularization techniques that jointly optimize a manifold regularization loss along with supervised classification loss. In particular, jointly optimizing the two losses has shown promise with deep neural networks (DNNs) for improving ASR [10] and sentiment classification [11]. Researchers have proposed several manifold regularization techniques, starting from Belkin et al. [5] and Geng et al. [12]. These methods make use of available labeled/unlabeled data points for regularization for better performance of classification models. Another way of ensuring smoothness is to train the model to produce similar outputs for a set of inputs obtained by adding a small amount of random perturbation to the training data [13]. This makes the model generalize better and it has been employed for semi-supervised learning [14]. Goodfellow et al. [15] suggested an improved method called Adversarial Training (AT) in which the perturbations are added only along an adversarial direction. Adversarial direction for a certain training data point is the direction along which the label probability of the model for that data point is most sensitive. Miyato et al. [16] proposed an extension of adversarial training wherein determining the adversarial direction didn't depend on the availability of labels, termed as Virtual Adversarial Training (VAT). We refer to the training methods proposed by Goodfellow et al. [15] and Miyato et al. [16] as adversarial training procedures, and investigate their applicability for improving the performance of emotion recognition systems.

In this paper we compare the performance of AT and VAT to that of a baseline DNN model for emotion recognition. After training the model using the aforementioned procedures, we evaluate its perfor-

mance under two settings: (i) Running a cross validation experiment on a single corpora (ii) Doing a cross corpora study. Under the single corpora setting, we aim to understand the impact of adversarial training on system performance under matched conditions. We note that we train a neural network on a few thousand samples, and aim to harness manifold regularization techniques to achieve a lower generalization error. In the cross corpora setting, we train the model on a single dataset and evaluate performance on three separate unseen datasets. We hypothesize that since manifold regularization imposes smoothness constraint on the model's outputs for the data points that are in the neighborhood of each other, it can also make the model more robust to noise arising due to difference in data distributions. In the next section, we provide a background of the adversarial training procedures (AT and VAT) , followed by a detailed explanation of the experiments in Section 3. We finally present our conclusions in Section 4.

2. ADVERSARIAL TRAINING PROCEDURES FOR EMOTION RECOGNITION

Given the set N labeled data points $\{\boldsymbol{x}_i, y_i\}, i = 1, .., N$, we represent the DNN output for the point $\boldsymbol{x}_i$ as $\theta(\boldsymbol{x}_i)$. $\theta(\boldsymbol{x}_i)$ is a vector of probabilities that the DNN assigns to each class in the label space spanned by y. We define a loss function based on the DNN outputs and the one hot-vectors $\boldsymbol{y}_i$ corresponding to labels y_i, as shown below. $V(\theta(\boldsymbol{x}_i), \boldsymbol{y}_i)$ is the loss for the data point $\boldsymbol{x}_i$ and typical choices include cross-entropy, mean squared error or the hinge loss.

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} V(\theta(\boldsymbol{x}_i), \boldsymbol{y}_i) \tag{1}$$

Generalizing the performance of a model trained solely on the loss above is challenging, particularly with a small number of training instances N. Studies have used L1 or L2 regularizers on DNN parameters or dropout to prevent overfitting to the training set [17]. Manifold learning and smoothing is an alternate way to build models that generalize better. These approaches add a regularization term that penalizes large differences in model outputs when a small perturbation is added to a data point. We determine the nature of this perturbation based on two existing methods: (i) adversarial training and, (ii) virtual adversarial training, as discussed next.

2.1. Adversarial training

Adversarial training [15, 16] penalizes large perturbations in model outputs when small perturbations are added to the training data points $\boldsymbol{x}_i$. We determine a perturbation vector $\boldsymbol{r}_i^a$ for every datapoint $\boldsymbol{x}_i$, and optimize the loss $\mathcal{L}_{\text{adv}}$ to train a DNN, as shown in Equation 2. D is a non-negative function that quantifies the distance between the predictions $\theta(\boldsymbol{x}_i + \boldsymbol{r}_i^a)$ and targets $\boldsymbol{y}_i$. α is a tunable hyper-parameter, determining the trade-off between $\mathcal{L}$ and the adversarial loss.

$$\mathcal{L}_{\text{adv}} = \mathcal{L} + \alpha \times \frac{1}{N}\sum_{i=1}^{N} D(\boldsymbol{y}_i, \theta(\boldsymbol{x}_i + \boldsymbol{r}_i^a)) \tag{2}$$

We determine the perturbation $\boldsymbol{r}_i^a$ based on Equation 3. ϵ is a hyper-parameter that determines the search neighborhood for $\boldsymbol{r}_i^a$.

$$\boldsymbol{r}_i^a = \arg\max_{\boldsymbol{r}:\|\boldsymbol{r}\|\leq\epsilon} D(\boldsymbol{y}_i, \theta(\boldsymbol{x}_i + \boldsymbol{r})) \tag{3}$$

Considering $||r||$ to be the Euclidean norm, $\boldsymbol{r}_i^a$ in Equation 3 can be approximated as shown below.

$$\boldsymbol{r}_{adv} \approx \epsilon\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|_2}, where\ \boldsymbol{g} = \nabla_{\boldsymbol{x}_i} D(\boldsymbol{y}_i, \theta(\boldsymbol{x}_i)) \tag{4}$$

The gradient term in Equation 4 can be easily computed during back-propagation. We note that this optimization has two hyper-parameters to tune, α and ϵ. We investigate the impact of these hyper-parameters on the model performance in one of our experiments.

2.2. Virtual adversarial training

Similar to adversarial training, virtual adversarial training [16] penalizes large variations in model outputs given small perturbations in the input. However, in this case the optimization is performed as shown in Equation 5. Instead of computing D using the labels $\boldsymbol{y}_i$, we use the predictions $\theta(\boldsymbol{x}_i)$ on the actual datapoints.

$$\mathcal{L}_{\textbf{vadv}} = \mathcal{L} + \alpha * \frac{1}{N}\sum_{i=1}^{N} D(\theta(\boldsymbol{x}_i), \theta(\boldsymbol{x}_i + \boldsymbol{r}_i^v)) \tag{5}$$

where the adversarial perturbation $\boldsymbol{r}_i^v$ for training example $\boldsymbol{x}_i$ is defined as following

$$\boldsymbol{r}_i^v = \arg\max_{\boldsymbol{r}:\|\boldsymbol{r}\|\leq\epsilon} D(\theta(\boldsymbol{x}_i), \theta(\boldsymbol{x}_i + \boldsymbol{r})) \tag{6}$$

The algorithm to compute $\boldsymbol{r}_i^v$ is described in detail [16]. As can be seen from Equation 5, the regularization loss term doesn't depend on labels. So, it can be used in semi-supervised training scenarios where the first term $\mathcal{L}$ is computed using labeled data and the second term is computed using both labeled and unlabeled data. Since in our experiments we do not have access to huge amounts of unlabeled data, the adversarial loss is calculated only using the available labeled dataset.

3. EXPERIMENTS

We perform experimental investigations under two settings: (i) a single corpora setting using a cross validation setup and, (ii) a cross corpora setting involving training on one corpus and testing on the other. In the single corpora setting, we aim to test improvements in the generalized performance of the model under matched dataset conditions. However, in the case of cross-corpora evaluation, representations for emotional utterances tend to be dissimilar due to factors such as differences in data collection protocol and noise conditions. Through cross-corpora evaluation, we aim to investigate if manifold regularization can yield models robust to the corpus specific variations.

3.1. Single corpora setting

We use the Interactive Emotional Dyadic Motion Capture (IEMOCAP) [18] dataset for our single corpora evaluation. The dataset consists of approximately 12 hours of speech from 10 subjects. For our classification experiments we only focused on a set of 4490 utterances spanning four emotional labels: neutral (1708), angry (1103), sad (1084), and happy (595). These utterances have a majority agreement amongst the annotators (at least two out of three annotators) regarding the emotion label. Please refer to [18] for more information regarding the dataset.

3.2. Cross corpus evaluation

We use a set of three datasets for the cross corpora evaluation. We train a DNN on the IEMOCAP dataset to identify four classes of emotion, followed by predictions on these datasets.

Surrey Audio Visual Expressed Emotion (SAVEE) database: Surrey Audio-Visual Expressed Emotion (SAVEE) database [19] has recordings of four male speakers reciting IEEE sentences in seven different emotions. For the purpose of our evaluation, we only select the subset of utterances belonging to one of the four target emotions, as predicted by the model trained on the IEMOCAP dataset. The dataset consists 60 utterances each belonging to the angry, sad, happy classes and 120 neutral utterances. We acknowledge that transfer of models across corpora spanning different label spaces is a challenge. By selecting a subset of utterances in our experiments, we simulate a study that assumes that the two datasets span the same label space.

Electromagnetic Articulography (EMA) database: Electromagnetic Articulography (EMA) database [20] contains a set of 680 utterances spoken in four different target emotions, such as anger, happiness, sadness and neutrality. Speakers are native speakers of American English: two females and one male. Note that the label space spanned by this dataset is equivalent to the one spanned by utterances in the training set.

Linguistic Data Consortium's (LDC) emotional prosody dataset: This database [21] was developed by LDC and contains the recordings of professional actors reading a series of semantically neutral utterances (dates and numbers) spanning fourteen distinct emotional categories. We select a subset of 714 utterances from the dataset that span the four emotion labels as modeled using training on the IEMOCAP dataset.

We note that there are several dissimilarities between the IEMOCAP dataset and the datasets used in the cross corpora study. Whereas the speakers in EMA and LDC have an American accent, SAVEE has speakers having a British accent. Unlike IEMOCAP, these databases aren't dyadic conversations. While EMA and SAVEE have speakers speaking different sentences emulating different emotions, in the LDC database we have speakers reading out numbers while emulating different emotions. We next discuss the features extracted on these datasets.

3.3. Features

We use the openSMILE toolkit to extract a set of 1582 features [22]. This feature set consists of various functionals computed for spectral, prosody and energy based features. Same feature set has also been used in several previous works including the INTERSPEECH Paralinguistic Challenges [23]. Similar sets of spectral, prosodic and energy based features has shown considerable success in emotion classification and affect tracking [24]. However an increased feature count leads to the "curse of dimensionality", a problem that manifold learning and smoothing can mitigate.

3.4. Experimental setup

We use a DNN as our classification model, such that the output layer consists of four nodes (each corresponding to an emotion), with softmax activation function. The DNN has three hidden layers with the number of neurons in each layer set to 128. The objective function $V(\theta(\boldsymbol{x}_i), \boldsymbol{y}_i)$ is chosen to be the cross entropy loss in our experiments [25]. While performing AT we chose the D function to be cross entropy between $\boldsymbol{y}_i$ and $\theta(\boldsymbol{x}_i + \boldsymbol{r}_i^a)$, while in the case of VAT D is set to be the cross entropy between $\theta(\boldsymbol{x}_i)$ and $\theta(\boldsymbol{x}_i + \boldsymbol{r}_i^v)$. As a baseline, we use a DNN with purely supervised loss (by setting $\alpha = 0$ in Equation 2). In [16], the authors considered two different distance functions D for VAT training: (i) Kullback-Leibler divergence between $\theta(\boldsymbol{x}_i)$ and $\theta(\boldsymbol{x}_i + \boldsymbol{r}_i^v)$, (ii) cross entropy between $\theta(\boldsymbol{x}_i)$ and $\theta(\boldsymbol{x}_i + \boldsymbol{r}_i^v)$. We also experimented with the Kullback-Leibler divergence as the distance function D, without observing significant differences in the model performances.

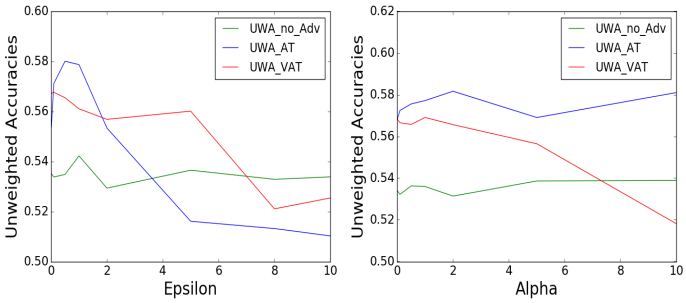

Fig. 1. Unweighted accuracies vs the hyper-parameters ϵ (left) and α (right) for baseline DNN (green), DNN trained with AT (blue) and DNN trained with VAT (red)

We implemented the AT and VAT model training in Keras [26] with a Tensorflow backend and perform optimization using stochastic gradient descent [27]. Our evaluation metric is Unweighted Accuracies (UWA) which has been used previously in emotion classification tasks [28]. Since the distribution of emotion classes are unbalanced in the datasets of interest, the UWA metric assigns equal weight to each emotion class during evaluation. Next, we present further details regarding the single corpus and cross corpus evaluation.

3.4.1. Results: Single corpus setting

We perform a leave one session out cross validation experiment on IEMOCAP. Through this experiment, we aim to understand the impact of the two hyper-parameters ϵ and α on the model performance. In order to study their impact individually, we perform evaluation by perturbing one of the two parameters, while keeping the other constant. By altering ϵ, we aim to understand the impact of smoothing radius around the data-points on the model performance and perturbing α impacts the weight of the adversarial loss on the overall optimization. The plots comparing the UWA of baseline DNN with that of DNN with adversarial training procedures for different values of hyper-parameters is shown in Figure 1.

It is evident that DNN trained with adversarial training procedures perform better than the baseline DNN. First, the value of α was kept fixed at 2 and ϵ was varied. For DNN trained with adversarial procedures, the model shows a higher performance for lower value of ϵ peaking at ϵ = 0.5. As we increase the value of ϵ, the model's performance starts deteriorating. This is expected since ϵ defines the neighborhood around an input feature vector over which the conditional distribution $p(y|\boldsymbol{x})$ is smoothed. Increasing the radius of this neighborhood forces our model to learn smoother functions that cannot capture the complexity of the conditional distribution function $p(y|\boldsymbol{x})$ thereby decreasing its performance on the validation set. For lower values of ϵ, AT outperforms VAT which may be due to the fact that AT is a supervised learning scheme where we use actual labels to find the adversarial direction. However, for higher values of ϵ, the trend reverses which leads us to believe that for larger values of search radius we are better off smoothening the output of the per-

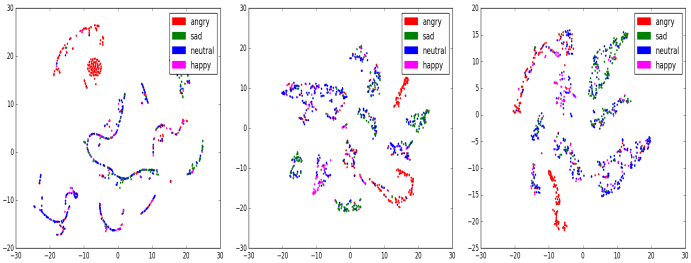

Fig. 2. TSNE plots comparing the output of the baseline DNN model (left), DNN trained with AT (center) and DNN trained with VAT (right)

Table 1. Class-wise accuracies (%) for one of the cross validation iterations showing how the adversarial procedures smoothen the conditional probability distribution $p(y|\boldsymbol{x})$. The "happy" class with least number of training samples has higher UWA for adversarial training procedures

Class	Baseline DNN	DNN with AT	DNN with VAT
Angry	80.78	80.34	79.48
Sad	93.81	92.78	92.27
Neutral	36.19	33.59	39.06
Happy	0.00	33.33	25.92

turbed input with respect to the output of the actual input rather than the label. Changing the weight α while keeping ϵ fixed at 0.5 did not seem to affect the accuracies of AT very much. For VAT however, increasing the weight of the VAT loss parameter in the loss function decreases the performance of the system. It was observed that for α = 2 and ϵ = 0.5 performance of AT was the best giving us UWA of 58.00% as compared to baseline DNN's performance of 53.47%. The VAT model was close to AT model at 56.54%.

As our goal in this work is to smoothen the posterior probability distribution of the labels given the feature vectors (expressed by $p(y|\boldsymbol{x})$), we perform further analysis by projecting and visualizing the four dimensional output vector $\theta(\boldsymbol{x}_i)$ using t-Stochastic Neighbor Embedding (t-SNE) approach [29]. t-SNE is a dimension reduction technique that clusters similar vector values together. Figure 2 shows the results with ϵ and α values fixed at 0.5 and 2, respectively. We observe that while the baseline DNN's output has very sharp boundaries, the output that we get from the model trained with adversarial training procedures has a wider spread. This is indicative of the smoothing effect of the manifold regularization losses. We further investigate the impact of adversarial losses on the class-wise performance, as shown in Table 1. In particular, the adversarial procedures were better at classifying the "happy" samples than the baseline DNN (Table 1). The class "happy" has the least number of samples compared to any other class in the training set. Smoothing the conditional distribution $p(y|\boldsymbol{x})$ yields better generalization and the minority class is not confused with other classes. In the next section, we discuss further results on the cross-corpus settings.

3.4.2. Results: Cross corpus evaluation

Since the adversarial procedures make the model robust to small perturbations to the input training points, we hypothesize that the regularized models are also robust to variation across datasets arising due to dissimilar noise conditions. Hence a model trained on an external corpus can achieve better performance on a dataset of interest.

Table 2. Cross-corpus accuracies (%) obtained using baseline DNN and DNN models trained with adversarial procedures. The training was performed using IEMOCAP in all cases.

Test Dataset	Baseline DNN	DNN with AT	DNN with VAT
SAVEE	42.5	46.25	46.04
EMA	58.91	61.65	60.75
LDC	37.91	43.18	42.29

To verify this, we did a cross corpus analysis where the whole of IEMOCAP dataset was used for training and a different corpus was used for testing. We extract the openSMILE features for the three external corpora, followed by mean-variance normalization using in-corpus statistics. We compare the UWA for three datasets as shown in Table 2 and show the superior performance of models trained with adversarial training procedures than baseline DNN. This indicates that the adversarial procedures increase model robustness to cross-corpus differences. Similar to the the single corpus results, an absolute increase of 15-25% was observed on the classification results of the "happy" class. We also note that the IEMOCAP trained models perform better on EMA compared to the other two datasets. This can be explained by domain variabilities. While SAVEE has British accented speech, in LDC the actors are reading out just numbers instead of actual English sentences. EMA being an American English corpus where participants are reading out sentences, comes closest to IEMOCAP which has actors having conversations in English. This observation suggests that despite better model generalization across datasets, data specific characteristics still play a part in determining the model performance.

4. CONCLUSIONS

This paper shows the effectiveness of adversarial training procedures for emotion classification using a DNN model. Adversarial training enforces the smoothness of the output probabilities $p(y|\boldsymbol{x})$, a case particularly applicable to low resource tasks such as emotion classification. We perform two sets of evaluation, a single corpus evaluation on the IEMOCAP dataset and three evaluations using a cross-corpus setup. In both the cases, we observe an improvement in the classification performance using the adversarial methods. We perform further investigation to understand the impact of the model hyper-parameters on the model performance and analyze the model outputs using t-SNE projections of the model outputs.

In the future, we aim to conduct further investigations using the adversarial loss. In particular, the VAT training can be used for semi-supervised optimization. This can be performed using an in-domain/external source for unlabeled data. We also aim to investigate other distance metrics D and its impact on the performance. Another pertinent problem is making the cross-corpus study compatible to different output label spaces across the datasets. Finally, one can also test the adversarial methods to other low resource problems.

5. REFERENCES

[1] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[2] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, and S. Narayanan, "Analysis of emotion recognition using facial expressions,

speech and multimodal information," in *Proceedings of the 6th international conference on Multimodal interfaces*. ACM, 2004, pp. 205–211.

[3] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras, "Deap: A database for emotion analysis; using physiological signals," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 18–31, 2012.

[4] O. Chapelle, B. Schlkopf, and A. Zien, *Semi-Supervised Learning*, The MIT Press, 1st edition, 2010.

[5] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of machine learning research*, vol. 7, no. Nov, pp. 2399–2434, 2006.

[6] J. Kim, S. Lee, and S. S. Narayanan, "An exploratory study of manifolds of emotional speech," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 5142–5145.

[7] F. Ping, J. Dongmei, W. Fengna, R. Ilse, and S. Hichem, "Manifold analysis for subject independent dynamic emotion recognition in video sequences," in *Image and Graphics, 2009. ICIG'09. Fifth International Conference on*. IEEE, 2009, pp. 896–901.

[8] M. You, C. Chen, J. Bu, J. Liu, and J. Tao, "Manifolds based emotion recognition in speech," *Computational Linguistics and Chinese Language Processing*, vol. 12, no. 1, pp. 49–64, 2007.

[9] Y. Qian, L. Ying, and J. Pingping, "Speech emotion recognition using supervised manifold learning based on all-class and pairwise-class feature extraction," in *Conference Anthology, IEEE*. IEEE, 2013, pp. 1–5.

[10] V. S. Tomar and R. C. Rose, "Graph based manifold regularized deep neural networks for automatic speech recognition," *arXiv preprint arXiv:1606.05925*, 2016.

[11] S. Zhou, Q. Chen, and X. Wang, "Active deep networks for semi-supervised sentiment classification," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, 2010, pp. 1515–1523.

[12] B. Geng, D. Tao, C. Xu, L. Yang, and X.-S. Hua, "Ensemble manifold regularization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1227–1233, 2012.

[13] C. M. Bishop, "Training with noise is equivalent to tikhonov regularization," *Training*, vol. 7, no. 1, 2008.

[14] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with stochastic transformations and perturbations for deep semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2016, pp. 1163–1171.

[15] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[16] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *arXiv preprint arXiv:1704.03976*, 2017.

[17] N. Tripathi and A. Jadeja, "A survey of regularization methods for deep neural network," 2014.

[18] 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, pp. 335, 2008.

[19] S. Haq, P. J. Jackson, and J. Edge, "Speaker-dependent audio-visual emotion recognition.," in *AVSP*, 2009, pp. 53–58.

[20] S. Lee, S. Yildirim, A. Kazemzadeh, and S. Narayanan, "An articulatory study of emotional speech production.," in *Interspeech*, 2005, pp. 497–500.

[21] M. Liberman, "Emotional prosody speech and transcripts," *http://www. ldc. upenn. edu/Catalog/CatalogEntry. jsp? catalogId= LDC2002S28*, 2002.

[22] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in opensmile, the munich open-source multimedia feature extractor," in *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 2013, pp. 835–838.

[23] B. W. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. A. Müller, S. S. Narayanan, et al., "The interspeech 2010 paralinguistic challenge.," in *Interspeech*, 2010, vol. 2010, pp. 2795–2798.

[24] R. Gupta, N. Malandrakis, B. Xiao, T. Guha, M. Van Segbroeck, M. Black, A. Potamianos, and S. Narayanan, "Multimodal prediction of affective dimensions and depression in human-computer interactions," in *Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2014, pp. 33–40.

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, MIT press, 2016.

[26] F. Chollet et al., "Keras," 2015.

[27] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 116.

[28] S. Sahu, R. Gupta, G. Sivaraman, W. AbdAlmageed, and C. Espy-Wilson, "Adversarial auto-encoders for speech based emotion recognition," in *InterSpeech*, 2017, pp. 1243–1247.

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