DATA AUGMENTATION USING GENERATIVE NETWORKS TO IDENTIFY DEMENTIA

Bahman Mirheidari[1] *Yilin Pan*[1] *Daniel Blackburn*[2] *Ronan O'Malley*[2] *Traci Walker*[3]
Annalena Venneri[4] *Markus Reuber*[5] *Heidi Christensen*[1]

[1]Department of Computer Science, [2]Sheffield Institute for Translational Neuroscience (SITraN)
[3]Department of Human Communication Sciences, [4]Department of Neuroscience, Royal Hallamshire Hospital
[5]Academic Neurology Unit, Royal Hallamshire Hospital
{b.mirheidari,yilin.pan, heidi.christensen}@sheffield.ac.uk

ABSTRACT

Data limitation is one of the most common issues in training machine learning classifiers for medical application. Due to ethical concerns and data privacy, the number of people that can be recruited to such experiments is generally smaller than the number of participants contributing to non-healthcare datasets. Recent research showed that generative models can be used as an effective approach for data augmentation, which can ultimately help to train more robust classifiers in sparse data domains. A number of studies proved that this data augmentation technique works for image and audio data sets. In this paper, we investigate the application of a similar approach to different types of speech and audio-based features extracted from interactions recorded with our automatic dementia detection system. Using two generative models we show how the generated synthesized samples can improve the performance of a DNN based classifier. The variational autoencoder increased the F-score of a four-way classifier distinguishing the typical patient groups seen in memory clinics from 58% to around 74%, a 16% improvement.

Index Terms— clinical applications of speech technology, sparse data, automatic speech recognition, data augmentation

1. INTRODUCTION

Dementia is a disorder of cognitive skills affecting memory, everyday functionalities, speech, language and communication abilities. The number of people developing dementia is increasing drastically. It is estimated that there are around 850 thousand people living with dementia in the UK. Dementia is the leading cause of death in the country accounting for over 12 percent of total deaths. The figure has grown by threefold from 2017 to 2005 [1]. The early diagnosis of dementia is of great clinical importance, and there is a need for an automatic, easy-to-use, low-cost and accurate stratification tool.

Recent studies using the qualitative methodology of conversation analysis (CA) demonstrated that communication problems may be picked up during conversations between patients and neurologists and that this can be used to differentiate between patients with neurodegenerative disorder (ND) and functional memory disorder (FMD; exhibiting problems with memory not caused by dementia) [2, 3]. However, conducting manual CA is expensive and difficult to scale up for routine clinical use. We have therefore developed a fully automatic system based on analysing a person's speech and language as they speak to an Intelligent Virtual Agent (IVA). The IVA asks a series of memory-probing questions that have been found to be cognitively demanding to answers. These questions are mimicking the style of questions often using during the *history taking* part of a normal face-to-face consultation. A number of features routed in conversation analysis were extracted and high accuracy levels were achieved when evaluating the system in a real memory clinic on patients with ND and FMD [4, 5, 6, 7]. We have recently expanded our data collection to include two more diagnostic classes: healthy controls (HC), and patients with mild cognitive impairment (MCI; a promodal condition to Alzheimer's disease (AD) indicating cognitive decline worse than normal aging but not consistent with an AD diagnosis.) [6]. This changed the task of binary decision for the classifier to a four-way classification, which naturally increased the difficulty due to the large overlap between symptoms (and extracted features) from the HC, FMD and MCI participants. In addition, the amount of data is still limited (in total 60 samples altogether, around 11 hours speech, 3.5 K utterances), which makes it challenging to train a very robust classifier and to apply state-of-the-art deep learning based machine learning techniques successfully.

It is very well-known that to train robust machine learning models, there should be a large number of samples for each class in the training data set; large enough to generalise the model, i.e. predict the classes of unseen samples correctly. However, in the medical domain, the number of people recruited to studies is often limited and the collected datasets are relatively small. Training classifiers with sparse data is therefore a major issue when applying state-of-the-art machine learning in medical applications. Therefore, most research in this field resort to using conventional classifiers rather than the recently introduced deep neural network (DNN) based models.

One of the common approaches to increase the number of samples is data augmentation. Data augmentation is widely used in image ([8]), speech ([9]), and text ([10]) processing to alleviate problems with limited data. The standard augmentation techniques, for instance in image processing, includes rotation, cropping, scaling and transformation of images [11]. There are increasing number of studies applying generative models such as generative adversarial networks (GANs) for data augmentation. For instance in the speech area, GANs used for different tasks such as speech synthesis [12], speech recognition [13, 14], speech emotion recognition [15], speech enhancement [16], and speaker verification [17].

In this paper we investigate using three recent generative models to produce synthesized samples of the features extracted from the conversation participants Adding the generated features to the original features, we train a new DNN-based classifier to distinguish between the four classes (FMD, ND, MCI and HC). To the best of our knowledge, this is the fist study on direct augmentation of variant statistical features extracted from speech for dementia detection.

The majority of generative models introduced for speech, image

and text applications are based on CNN or long short-term memory (LSTM), where the order or position of features in each sample is important, and the network learns the context. However, in this paper we aim to use the generative models as an augmentation technique to produce more samples of the features. These features are inherently different in nature to image pixels, speech waveforms or word sequences. Therefore, standard dense layers of neural networks will be used instead of CNNs and LSTMs in the generative models.

2. GENERATIVE MODELS FOR DATA AUGMENTATION

Machine learning models generally fall into two major approaches: discriminative verses generative models. For the input features, x and the corresponding labels y, the discriminative models try to directly make decision boundaries from the features to determine the labels (i.e., directly predict the probability of y given x: $P(y|x)$), while the generative models focus on feature distribution and generation of features (probability of x: $P(x)$). In addition to the naive Bayes generative models which have been known for a long time in the machine learning community, there are two recently introduced techniques: GANs and variational autoencoders (VAEs).

The GAN model, originally introduced by ([18]), consists of two main components: the generator and the discriminator. The generator generates samples (e.g., new images in the case of the MNIST data set[1]), while the discriminator authenticates samples, i.e., decides that a sample either comes from the real data set or not. So the task of the discriminator is simply a binary classification. The generator, on the other hand, attempts to deceive the discriminator by creating better samples, as realistic as possible, in order to pass the evaluation of the discriminator (the discriminator gets confused and treats them as the authentic samples). Normally in training GANs, the generator maps randomly made numbers (noise, hidden/latent code) into samples. The synthesised and real samples are both fed to the discriminator, and the discriminator returns a probability indicating authenticity (1:real, 0:fake). The generator model, technically, is built in a reverse network as the discriminator with the opposition loss function. For instance, if the discriminator model is a convolutional network, the generator is the inverse convolutional network. The low resolution of the synthesized sample and the difficulties in stabilising the model are the two main issues of GANs. There have been different improvements to address these issues, including using the Wasserstein distance for the function loss (WGAN [20]), conditional GAN (CGAN [21]) and semi-supervised GANs (SGAN) (forcing the discriminator to produce the labels [22]).

Autoencoder (AE), another generative model, consists of two components: the encoder and the decoder. The encoder encodes the input samples into a compressed representation (latent vectors which are dimensionally reduced version of samples), while the decoder reconstruct the samples from the compressed representations. The aim of AE is to reconstruct the input samples as similar as possible to the real samples. Basically, AE uses an unsupervised training regime to reconstruct the original data. VAEs are the extension to AEs, which normalise the latent vectors. Unlike the conventional AE, VAEs assume Gaussian distribution for the input samples and tries to capture the distribution of the original samples and they are much more similar to GANs than the normal AEs [23].

Generative models for data augmentation have been used for medical applications, where the data limitation issue is of particular concern. Synthesised samples can help in training more robust classifiers improving the generalisation and reducing the overfitting problem. For instance, GANs have been shown to improve performance in image segmentation tasks such as the computed tomography (CT) cerebrospinal fluid (CSF) and the fluid-attenuated inversion recovery (FLAIR) magnetic resonance (MR) [11]. [24] reported significant improvement in classification of CT images of liver lesions, when data augmentation was carried out after applying the standard augmentation techniques on the images. Although, these studies were in the medical domain, the data they worked with was very different to our speech data, such as brain and liver images.

[1]Large dataset of handwritten digits widely used in the machine learning community [19]

3. EXPERIMENTAL SETUP

The data was collected using the IVA during summers of 2016, 2017, 2018 and 2019 at the Department of Neurology, University of Sheffield, UK based at the Royal Hallamshire Hospital. Of the total number of 93 participants, 60 were chosen for the study (the rest were found to not have memory problems, however we made use of that data for training the speaker diarisation and the ASR). Table 1 shows the demographic information of the participants in the study. Comparing to the previous experiment ([6]), in this study we use a balanced number of conversations for each class of our four groups (i.e. 15 FMD, 15 ND, 15 MCI and 15 ND).

Table 1. Demographic information of the participants (15 in each group). FMD: Functional Memory Disorder, ND: Neurodegenerative Disorder, MCI: Mild Cognitive Impairment, HC: Healthy Control.

Class	Age	Education (Years)	Male
FMD	54.9 (+/- 4.1)	16.4 (+/- 0.6)	40.0%
ND	67.8 (+/- 4.2)	18.0 (+/- 1.6)	66.7%
MCI	63.0 (+/- 4.3)	17.3 (+/- 1.1)	66.7%
HC	69.5 (+/- 4.0)	18.1 (+/- 1.0)	40.0%

Table 2 shows the information of the two datasets used: DR INTVWS (295 doctor-patient interviews) and IVA (93 IVA-patient recordings). The DR INTVWS data set was only used for training the i-vector based diarisation module (the CALL_HOME recipe [25]) and the Bidirectional Long Short Term Memory/Time-Delay Neural Network (BLSTM)-TDNN based ASR using the Kaldi toolkit [26]. The 10 fold cross validation approach was used for training the diarisation and the ASRs. The diarisation error rate (**DER**) was **26.2%**, and the word error rate (**WER**) was **38.2%**.

3.1. Extended features

In addition to the initial features (78 including CA-inspired, only-acoustic, only-lexical, word vector and verbal fluency) introduced in [6]), 104 MFCC acoustic features were extracted. Then the min, max, average and standard deviation were applied $(4(\text{func.}) \times 13(\text{MFCC}) \times 2(\text{speakers}) = 104)$. The initial acoustic-only and lexical-only features included the average as the only statistic function, therefore the other remaining three statistic features were also applied on the features, resulting in additional 72 acoustic-only $(3(\text{func.}) \times 24(\text{acoustic-only}) = 72)$ and 72 lexical-only features $(3(\text{func.}) \times 24(\text{lexical-only}) = 72)$. This resulted in a total number of 324 features.

Table 2. Datasets used for training the ASRs, including Len.:the total length in hours/mins, Utts.:number of utterances, Spks.:number of speakers, and Avg. Utts.:Average utterance length in seconds.

Dataset(No)	Len.	Utts.	Spks.	Avg Utts.
Dr intvws (295)	64h 21m	39184	736	5.9s
IVA (93)	17h 18m	5637	103	11.05s

3.2. Details of the generative models

For training the generative models we used the Keras python library ([27]) back-ended by Tensorflow([28]). Three candidate generative models were selected: CGAN, VAE and VAE combined with SGAN. This is similar to the AE-GAN introduced by [29], but the CNN layers were replaced with dense layers, and the AE with VAE; we refer to this model as VAE-SGAN. The encoder and decoder parts of the VAE-SGAN were similar to the encoder and decoder of the VAE. In addition to the normal dense layers, and to reduce overfitting, layers of BatchNormalization, LeakyReLU, and Dropout were used in between the layers. The Adam optimizer was used for training, as well as a two layer standard DNN classifier which is used separately to evaluate the synthesized samples.

Algorithm 1 shows how we use the generative models to make synthesized samples and add them to the training set. We can repeat this N times and keep the results for both the test and evaluation (eval) sets separately. In order to see how well the reconstructed samples do, a DNN based classifier is used and the F-score is calculated based on its performance on both the test and eval sets.

Algorithm 1: Reconstructed samples from a generative model.

Result: Best scores and reconstruction numbers for the eval and test data: $Score_{eval}, Score_{test}$

Input: train, eval and test data: $X_{train}, Y_{train}, X_{eval}, Y_{eval}, X_{test}, Y_{test}$;

$reconX = X_{train}$;$reconY = Y_{train}$;

$Score_{eval} = (0, 0)$; $Score_{test} = (0, 0)$;

for $recon = 1, 2, ..., N$ **do**

  Train a generative model, M(Enc,Dec,Dis) with X_{train}, Y_{train};

  $lat = Enc(X_{train})$;

  $X' = Dec(lat)$;

  $reconX, reconY = reconX + X', reconY + Y_{train}$;

  $lat2 = Enc(X_{eval})$;

  $X2' = Dec(lat2)$;

  $reconX2, reconY2 = reconX + X2', reconY2 + Y_{eval}$;

  Train a DNN-based model, DM with $reconX, reconY$ tuned by $reconX2, reconY2$;

  $S_{eval} = DM.score(X_{eval}, Y_{eval})$;

  $S_{test} = DM.score(X_{test}, Y_{test})$;

  if $S_{eval} \geq Score_{eval}[0]$ **then**

    $Score_{eval} = (S_{eval}, rec)$;

  end

  if $S_{test} \geq Score_{test}[0]$ **then**

    $Score_{test} = (S_{test}, rec)$;

  end

end

return $Score_{test}, Score_{test}$;

4. RESULTS

This section compares the performance on a normal classifier baseline (logistic regression - LR) and a DNN-based trained using the adding the synthesized samples.

4.1. Normal classifier

Using the LR classifier and the five fold cross validation approach the precision, recall and F-score of the classifier were calculated first on the original 78 features and then on the 324 features (original+extended features). The columns with majority of zero values were omitted from the feature sets (we call them non-zeros (NZ)). We observed that using the NZ can result in a better performance for the recursive feature elimination (RFE, a standard approach for feature selection) ([30]). Based on the five fold cross-validation, in each fold out of the total 60 samples, 40 were used in train set, 8 for evaluation and 12 for test. Table 3 shows the details of the performance of the classifier in terms of precision, recall and F-score for the original 78 features, the NZ original features, the top 13 original features selected by RFE, all features (original+extended), the NZ for all features, and the top 68 all features selected by RFE. It can be seen, that the NZ features from the original set can achieve around 40% F-score (3.5% increase), which then can be improved further by RFE up to 59%. However, using all features together resulted in a better performance than the original features (F-score of 45.6% compared to the 36.6% F-score). Applying RFE (68 top features) this was further improved to an F-score of 64% (F-score of fold 5 was 58.3%, the closest F-scores to the average). On the last row of the table, the results for the average fold (number 5) is shown. We will refer to this fold as (HALLAM (F5)). This fold will be used in the following experiments as a fixed train/test partition.

Table 3. Precision (Pr), recall (Rc) and F-scores (Fs) of the Logistic Regression classifier trained using different sets of features extracted from the 60 conversations with 5-fold cross validation. NZ: Non-zero features. F5: Fold 5, the fold close to the average.)

Feature set	Feat. No.	Pr %	Rc %	Fs %
Original	78	36.7	40.0	36.6
Original (NZ)	64	41.5	43.3	40.1
Original (RFE)	13	60.5	60.0	59.1
ALL	324	45.9	46.6	45.6
ALL (NZ)	261	45.1	45.0	44.4
ALL (RFE)	68	64.5	**65.0**	**64.0**
ALL (F5) (RFE)	68	**68.3**	58.3	58.3

4.2. Generative models on MNIST

Before we start using the augmentation techniques on our dataset, we demonstrate the approach on a widely used dataset, MNIST. This experiment will show how the technique generally works on a standard data set. MNIST contains 60000 train and 10000 test hand written digit images (in 10 classes, each 28 by 28 pixels). Generative models have been shown to work well for tasks involving images, speech and text where there are sequences of features in which the neighbouring features may be co-related to each other. As mentioned before, we removed all CNN or LSTM layers (which capture context information very well). So as expected, this reduces the performance of the generative model significantly. From the train set of MNIST, 1500

samples (150 for each digit) were selected. 10 percent was chosen as the eval set (150 samples) and 90 percent (1350 samples) as the train set (we refer to this as MNIST-1500). The standard 10K samples of MNIST was used as the test in our experiments.

The F-score, when training the normal LR classifier (baseline) on the MNIST-1500 subset, was around 90%. The algorithm for augmentation was applied on the dataset using the three generative models. Figure 1 shows the F-scores gained using the algorithm over 20 times reconstructions (up to 27000 additional samples). As can be seen, all three generative models (CGAN, VAE and VAE-SGAN) improved the F-score up to around 93%. The improvement seen is not steady though, and the results fluctuate, however on average, VAE-SGAN performed slightly better than CGAN, while VAE was not as good as the other two and had the highest fluctuation.

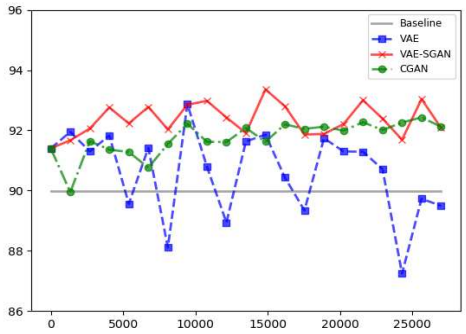

Fig. 1. *F-scores (%) of the DNN classifier for the MNIST-1500 data for different numbers of reconstructed samples tested on the standard 10K samples of the test set (number of reconstructions: 20).*

4.3. Generative models on HALLAM (F5)

The algorithm was repeated for the HALLAM (F5) data set. The baseline classifier F-score was around 58%. Figure 2 shows the F-scores when applying the three generative models. As the number of reconstructed features increased (up to 2000), the F-scores of the VAE-SGAN varies between 60% to around 75%, VAE fluctuated between 40% to 74%, and CGAN between 40% to 70%. Compared to MNIST-1500, these fluctuations were much higher. VAE-SGAN, however, performed better.

4.4. Optimum number of reconstructions

Based on the previous two figures, finding the optimum number of reconstructions is challenging. One approach is to use the eval set to find the best F-score, although naturally, this might not be the best for the test set. The high fluctuation, especially in HALLAM (F5) might indicate that not all of the reconstructed samples are useful for the classification. Therefore, we modified the Algorithm 1, to first check the quality of the reconstructed features, and only if they are good enough, are they then added to the train set. We used a similarity measure (normalised pair-wise distance) between the two individual features in the feature set (similar to the pixel-based similarity in image processing). We observed, that we can get better results if in each iteration of the algorithm, we check whether the similarity improves or not between the reconstructed features and the original features in the train set. Table 4 shows the best F-scores gained

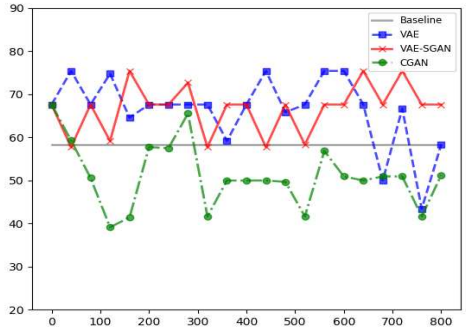

Fig. 2. *F-scores (%) of the DNN classifier for the* HALLAM *(F5) data for different numbers of reconstructed samples tested on 12 samples of the test set (number of reconstructions: 25).*

Table 4. F-scores (Fs) and the number of reconstructed samples after modifying the algorithm (see text for details).

Data set	Model	Train no.	Fs
MNIST-1500	VAE	20250	90.3%
MNIST-1500	CGAN	2700	**92.3%**
MNIST-1500	VAE-SGAN	4050	**92.3%**
HALLAM (F5)	VAE	160	**75.4%**
HALLAM (F5)	CGAN	200	74.8%
HALLAM (F5)	VAE-SGAN	120	67.6%

for MNIST and HALLAM (F5) after modifying the algorithm. Using VAE, MNIST achieved a 90.3% F-score, while the VAE-SGAN and CGAN achieved 92.3%. For HALLAM (F5), VAE performed the best with a 75.4% F-score. CGAN gained 74.8%. Comparing to the baseline of 58.3%, all three generative models saw improvements with VAE performing the best at 17.1%, and CGAN and VAE-SGAN following with 16.5% 9.3%, respectively.

5. CONCLUSIONS

We introduced the concept of using data augmentation on features extracted from a person's speech and language using the recent generative models CGAN, VAE and VAE-SGAN. Finding the optimum number of reconstructed samples is the challenging part of this technique, although the evaluation set can help us to find a local optimum number. For the two tasks using Algorithm 1 and the modification, each generative model performed well, if slightly differently. However more work is needed to investigate the use of generative models. Reported experiments were carried out on a representative fold; future work will expand this to all fold as well as exploring the effect of different generative models with more reconstructions.

6. ACKNOWLEDGEMENTS

This research has been partly supported under the European Union's H2020 Marie Skodowska-Curie programme TAPAS (Training Network for Pathological Speech processing; Grant Agreement No. 766287)

7. REFERENCES

[1] Dementia Statistics, "Deaths due to dementia," 2018, Accessed on October 12, 2019.

[2] C. Elsey, P. Drew, D. Jones, D. Blackburn, S. Wakefield, K. Harkness, A. Venneri, and M. Reuber, "Towards diagnostic conversational profiles of patients presenting with dementia or functional memory disorders to memory clinics," *Patient Education and Counseling*, vol. 98, pp. 1071–1077, 2015.

[3] D. Jones, P. Drew, C. Elsey, D. Blackburn, S. Wakefield, K. Harkness, and M. Reuber, "Conversational assessment in memory clinic encounters: interactional profiling for differentiating dementia from functional memory disorders," *Aging & Mental Health*, vol. 7863, pp. 1–10, 2015.

[4] B. Mirheidari, D. Blackburn, K. Harkness, T. Walker, A. Venneri, M. Reuber, and H. Christensen, "An avatar-based system for identifying individuals likely to develop dementia," *Proc. Interspeech*, pp. 3147–3151, 2017.

[5] B. Mirheidari, D. Blackburn, A. Venneri, M. Reuber, T. Walker, and H. Christensen, "Detecting signs of dementia using word vector representations," in *Proc. Interspeech*. ISCA, 2018.

[6] B. Mirheidari, D. Blackburn, R. OMalley, T. Walker, A. Venneri, M. Reuber, and H. Christensen, "Computational cognitive assessment: Investigating the use of an intelligent virtual agent for the detection of early signs of dementia," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2732–2736.

[7] R OMalley, B. Mirheidari, A. Harkness, K.and Venneri, M. Reuber, T. Walker, H. Christensen, and D. Blackburn, "A fully automated cognitive screening tool based on assessment of speech and language," *Journal of Neurology, Neurosurgery & Psychiatry*, 2019, In preparation.

[8] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[9] T. Ko, V. Peddinti, D. Povey, M. L Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5220–5224.

[10] S. Semeniuta, A. Severyn, and E. Barth, "A hybrid convolutional variational autoencoder for text generation," *arXiv preprint arXiv:1702.02390*, 2017.

[11] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, Maria V. Dickie, D. Alexander H., J. Wardlaw, and D. Rueckert, "Gan augmentation: augmenting training data using generative adversarial networks," *arXiv preprint arXiv:1810.10863*, 2018.

[12] T. Kaneko, H. Kameoka, N. Hojo, Y. Ijima, K. Hiramatsu, and K. Kashino, "Generative adversarial network-based postfilter for statistical parametric speech synthesis," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4910–4914.

[13] Hu Hu, Tian Tan, and Yanmin Qian, "Generative adversarial networks based data augmentation for noise robust speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5044–5048.

[14] P. Sheng, Z. Yang, H. Hu, T. Tan, and Y. Qian, "Data augmentation using conditional generative adversarial networks for robust speech recognition," in *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2018, pp. 121–125.

[15] A. Chatziagapi, G. Paraskevopoulos, D. Sgouropoulos, G. Pantazopoulos, M. Nikandrou, T. Giannakopoulos, A. Katsamanis, A. Potamianos, and S. Narayanan, "Data augmentation using gans for speech emotion recognition," *Proc. Interspeech 2019*, pp. 171–175, 2019.

[16] S. Pascual, A. Bonafonte, and J. Serra, "Segan: Speech enhancement generative adversarial network," *arXiv preprint arXiv:1703.09452*, 2017.

[17] D. Michelsanti and Z. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," *arXiv preprint arXiv:1709.01703*, 2017.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[19] Y. LeCun, "The mnist database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*.

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

[21] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[22] A. Odena, "Semi-supervised learning with generative adversarial networks," *arXiv preprint arXiv:1606.01583*, 2016.

[23] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[24] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Synthetic data augmentation using gan for improved liver lesion classification," in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 2018, pp. 289–293.

[25] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision. IEEE 11th International Conference*, 2007, pp. 1–8.

[26] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.

[27] F. Chollet et al., "Keras: Deep learning library for theano and tensorflow," *URL: https://keras. io/k*, vol. 7, pp. 8, 2015.

[28] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, Software available from tensorflow.org.

[29] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.