Deepfake Audio Detection: A Deep Learning Based Solution for Group Conversations

R.L.M.A.P.C. Wijethunga
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it17136716@my.sliit.lk

D.M.K. Matheesha
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it17123228@my.sliit.lk

Abdullah Al Noman
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it17155908@my.sliit.lk

K.H.V.T.A. De Silva
Department of Information Technology
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
it17144704@my.sliit.lk

Muditha Tissera
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
muditha.t@sliit.lk

Lakmal Rupasinghe
Department of Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
lakmal.r@sliit.lk

2020 2nd International Conference on Advancements in Computing (ICAC) | 978-1-7281-8412-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICAC51239.2020.9357161

***Abstract*—The recent advancements in deep learning and other related technologies have led to improvements in various areas such as computer vision, bio-informatics, and speech recognition etc. This research mainly focuses on a problem with synthetic speech and speaker diarization. The developments in audio have resulted in deep learning models capable of replicating natural-sounding voice also known as text-to-speech (TTS) systems. This technology could be manipulated for malicious purposes such as deepfakes, impersonation, or spoofing attacks. We propose a system that has the capability of distinguishing between real and synthetic speech in group conversations.We built Deep Neural Network models and integrated them into a single solution using different datasets, including but not limited to Urban-Sound8K (5.6GB), Conversational (12.2GB), AMI-Corpus (5GB), and FakeOrReal (4GB). Our proposed approach consists of four main components. The speech-denoising component cleans and preprocesses the audio using Multilayer-Perceptron and Convolutional Neural Network architectures, with 93% and 94% accuracies accordingly. The speaker diarization was implemented using two different approaches, Natural Language Processing for text conversion with 93% accuracy and Recurrent Neural Network model for speaker labeling with 80% accuracy and 0.52 Diarization-Error-Rate. The final component distinguishes between real and fake audio using a CNN architecture with 94% accuracy. With these findings, this research will contribute immensely to the domain of speech analysis.**

***Index Terms*—Deep Neural Networks, Natural Language Processing, Speaker Diarization, Deepfake, Deep learning**

I. Introduction

Machine-generated voices are mostly populated in our day-to-day lives. With the automation of technology, people are more intended to control daily works over voice. Machine-generated voice or synthetic voice is rapidly used in virtual assistants such as Google Assistant, Alexa, Siri, Bixby, and or some other. Despite having advantages of such technologies, people such as celebrities, politicians, or some other famous persons are victimized. The major concern goes to an advanced technology called Deepfake which mostly uses Generative Adversarial Network (GAN) [1] to generate synthetic audio to impersonate real people.

Studies proved the worst case of Deepfake audio used as a shred of crucial evidence to let criminals go free. Raising of Deepfake audio in a group conversation is the next level of crime. In a group conversation, multiple user voices are integrated. Previous research studies clearly showed how a Deepfake mostly Deepfake image or Deepfake video can be identified. Now, the challenge is to detect Deepfake audio in a conversation.

Deepfake audio detection starts with audio signal processing [2]. An audio signal is the representation of sound and using signal processing mechanisms, spectrogram and wavelength are processed. By studying previous research, the audio signal should be processed in such a way that all the speakers must be diarized before the detection of Deepfake. Preprocessing of audio is to reduce noise and other unnecessary factors from the audio signal.

Natural Language Processing deals with the text format retrieved from the audio. Mechanisms of both classification and clustering are to reach an acceptable accuracy of diarization. According to previous studies, the outcome of NLP based diarization can be integrated with Machine Learning [3] to cluster the speakers with improved accuracy levels.

Diarized contents can be used in the detection phase. In terms of synthetic speech detection, the research problem discusses the drawbacks of synthetic speech in more detail. For example, a TTS system can be used to train a targeted individual voice. Once trained, it could be used for impersonation attacks. Therefore, to detect these malicious utterances when authenticity is required, there must be a mechanism to

978-1-7281-8412-8/20/$31.00 ©2020 IEEE

tell the difference.

The aim is to build a model with DNN which could extract the dynamic acoustic features [4] of the voice, which will determine whether given audio is artificially generated or real human speech. CNN and RNN will be used in the development of the model. CNN has proven to be the most efficient technology used in Computer Vision and Image Processing, therefore the Spectrum will be analyzed with CNN. An RNN is a generalization of Feedforward Neural Network with internal memory, in which RNN can use it to process sequences of inputs, that is applicable in determining a new state.

II. Related Work

Most of the researchers were focused on reducing one specific noise reduction in audio in their noise reduction algorithms [5]. They have proposed adaptive filtering for echo cancellation and based on the denoising functions, wavelet transform has been used to remove noises [2].

Spectral Subtraction also plays a major role in noise removal systems. Spectral Subtraction is a method to remove background noise from noisy speech signals in the frequency domain. This approach consists of calculating the noise spectrum using the Fast Fourier Transform (FFT) and subtracting the average noise level from the noisy speech spectrum [6].

Apart from those methods, the researchers have used different filters for the removal of unwanted noises such as the Gaussian filter, Butterworth filter, Comb filter, Chebyshev I and II filters, Elliptic filter, etc [7]. Butterworth filter, Chebyshev I filter, and Elliptic filter is used in one research paper to remove the noises in ECG signals using MATLAB software and suggested that the Butterworth filter is the best filter when comparing the others [8].

Natural Language Processing is one of the most trending technologies which has been applied in most of the areas to revolutionize the modern world. NLP is proposed to use to diarize the different speakers from a group conversation. Analyzing the raw text obtained from audio format based on the important relationships and factors, the speakers will be diarized.

Previous studies were analyzed to understand the flexibility of the different approaches in NLP. According to the acquired knowledge from those, it is understood that to diarize the speakers there are two different approaches in NLP – text classification which requires pre-labeled data that means supervised learning and text clustering which deals with unsupervised data that is known as unsupervised learning.

Since the text from audio format is mostly unsupervised, the clustering approach will get a higher priority. Along with text clustering, the classification can be used for better accuracy. Clustering algorithms analyze the unlabeled data and grouping similar data points together [6].

1) Convolutional Neural Networks (CNN): First published in the '80s, later researchers were able to use the architecture to learn positional relation between pixels, enabling neural networks to identify shapes and patterns. The scheme behind a convolutional layer is to break down an image into miniature squares, and with the use of a convolution operator to differentiate each square to learnable filters. Stacking convolutional layers with fully-connected layers results in forming a powerful architecture that has the ability to identify patterns, from shapes to complex objects; lowest layers to highest layers [9].

2) Recurrent Neural Networks (RNN): The concept of RNNs is that the output of one layer will be the input to the same layer. What happens is that this mechanism provides the architecture a "memory" and allows us to comprehend correlation in sequences. That helps in remembering the past and its present decisions which are influenced by what it learned from past experiences [13], [12]. There is a modified version of RNN, Long Short-Term Memory (LSTM), instead of using classic neurons it uses memory cells, composed by several components and processes to provide long term memory. And these states could be changed accordingly [12].

The idea behind RNNs is the output of one layer will be the input to the same layer. By doing this, the mechanism provides a memory to the architecture and allows comprehending correlation in sequences. This helps the RNN to make decisions based on past decisions it has made in the same state, this induces the memory state. Long Short-Term Memory (LSTM) is the modified version of RNN; the states could be changed in LSTM so that at given times the memory of states would be more or less [9], [10].

I-Vector Framework - The i-vector subspace modeling is a recent state-of-the-art technique. It is proven to be the most effective feature for speaker diarization according to a recent study. Speaker or session variability is the variability exhibited by a given speaker from one recording session to another. This type of variability is usually attributed to channel effects, although this is not strictly accurate since intra-speaker variation and phonetic variation are also involved [11]. In this approach, a speech segment is represented by a low-dimensional "identity vector" (i-vector for short) extracted by Factor Analysis [12].

III. Methodology

A. Speech Denoising using DNN

Speech denoising aims to remove noise from speech signals while enhancing the quality and intelligibility of speech. The system does not know what background noises are in the audio. So, a background noise dataset (UrbanSound8K) was used here to check what noises are there using different deep learning techniques. When an audio sample (.wav format) which contains some background noises was given as an input to the algorithm, it can determine if it contains one of the sounds with a corresponding likelihood score. Conversely, if none of the target sounds were detected, it will be presented with an unknown score. To achieve this, different neural network architectures such as Multi-Layer Perceptron's (MLPs) and Convolutional Neural Networks (CNNs) were used. Adaptive filters were used to remove the predicted noise from the original audio to make clean audio which can be used to diarize the speakers easily.

a) Dataset: UrbanSound8K is a background noise dataset which contains 8732 labeled sound excerpts of urban sounds from 10 classes: air conditioner, car horn, children playing, dog bark, drilling, enginge idling, gun shot, jackhammer, siren, and street music. All excerpts are taken from field recordings uploaded to freesound.org [15]. In addition to the sound excerpts, a CSV file containing metadata about each excerpt is also provided.

b) Preprocessing Stage: The preprocessing of the data set is performed using sample rate conversion and merging audio channels. Mel-Frequency Cepstral Coefficients (MFCC) were extracted on a per-frame basis from the audio samples. To analyze frequency and time characteristics, MFCC was used because it summarizes the frequency distribution, across the window-size. The dataset is split using train_test_split which allocates 20% as the testing set and 80% as the training set [16].

c) Train and Test Model : Train a Deep Neural Network with the dataset to predict the noises was the next step. A simple neural network architecture like MLP was used before experimenting with a more complex neural architecture like CNN [16]. CNNs build upon the architecture of MLPs but with several important changes. Next, they group the layers into three dimensions: width, height, and depth. Second, the nodes in one layer do not necessarily connect to all nodes in the layer that follows, but often only a sub-region of it [16]. This allows CNN to perform two important stages named Feature Extraction and Classification.

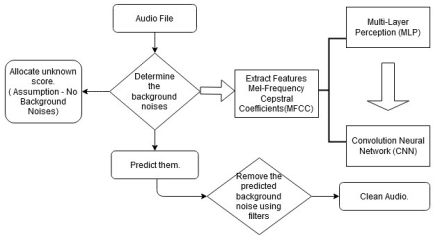

Fig. 1. System Diagram for Audio Classification

Adaptive filters are used to remove the predicted background noises from the source and generate clean audio without damaging the features of the audio. For initial section of adaptive filters require two inputs and create the adaptive filter according to the parameters. After that data filtering process is started to remove the background noises and smooth the audio.

B. Speaker Diarization with NLP

1) Audio Dataset Conversion to Text Dataset Conversion: The process of converting spoken words into written texts is called a speech-to-text (STT) conversion and it is used to get a wider understanding of the speech process [17]. Like speech recognition, STT follows the same principles and steps, with different combinations of various techniques for each step [17].

The ability to identify the words and phrases in spoken language and convert them to human-readable text is called speech recognition and it can be done in Python using the Speech-Recognition library [11].

- Picking a Python Speech Recognition Package – Google Speech Recognition
- Installing SpeechRecognition in Python
- Working with Audio Files – Loading the audio file and converting the speech into text using Google Speech Recognition

Diarization of the speaker in a conversation is more challenging where more speakers are present in the conversation. The random conversation often differs from a formal and sophisticated conversation. A raw transcript of a group conversation contains different noises and if the noises are properly handled, it will make the diarization job one step ahead. Natural Language Processing plays an important role to deal with such a scenario.

a) Dataset: A conversational dataset from Opensubtitles [19] is used in this research. Raw data of this dataset is preprocessed and normalized according to this research criteria. The final dataset is split into 4415 segments containing 100000 lines each and placed in Google Cloud storage.

b) Train and Test Model: The large conversational dataset is split into the train and test model using the Apache Beam pipeline. Considering the dataset size, TensorFlow framework is implemented with Python programming language. The training model includes the features extracted from the dataset and correlation among the speeches. The system can detect the speakers based on the behavioral analysis, common words usages, and the names provided in the transcripts. The testing model is prepared to the system to measure how accurate the system works. To build the models, Google Dataflow Engine is needed and to initiate, a suitable apache pipeline is built.

Dataset is stored and retrieved from Google Cloud Storage and processed in Google Dataflow environment, then models are built and written back to the Cloud Storage. The system is more convenient to use since there is less risk in the cloud.

A transcript is taken as input, and based on the trained model the system finds the relationships and features from the transcript. Once the context is understood by the system, it does the clustering of common speeches. K-Means clustering algorithm is used to cluster the speeches from the transcript. The whole idea about clustering the speeches are based on Natural Language Processing techniques where Machine Learning is significantly used. Machine Learning applies to diarize the speakers based on the factors mentioned above.

C. Speaker Diarization with DNN

Speaker diarization can be defined as the identification of the number of different speakers in a conversation. This task can be achieved using different technologies, but the accuracy levels will not be the same. Here the task is accomplished by training a Deep Neural Network model, which will then be

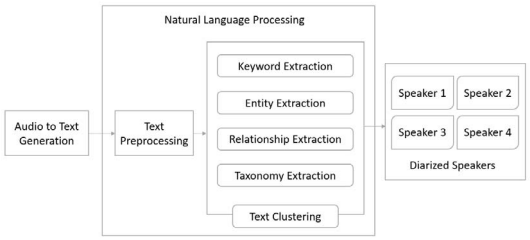

Fig. 2. System Diagram for Speaker Diarization with NLP

used to predict the various number of speakers in each audio sample.

a) Dataset and Model Training: The dataset used to train the model was freely available from International Computer Science Institute (The ICSI Meeting Corpus), the AMI Corpus Dataset. This dataset contained meeting recordings of different speakers with a size of 5GB all files in the .wav format. With the dataset the model was trained, evaluated and tested [20]. After training the model, it can be used for predictions. Generally the speaker diarization task is achieved undergoing several sub processes;

- Speech segmentation – The process in which input audio is segmented to short sections assuming they are from the same speaker
- Audio embedding extraction – i-vector feature used for extraction of audio clips
- Homogeneous segmentation – Identification of segments per speaker and aggregates them
- Clustering – Final number of speakers are determined, and homogeneous segments clustered accordingly [21]-[23]

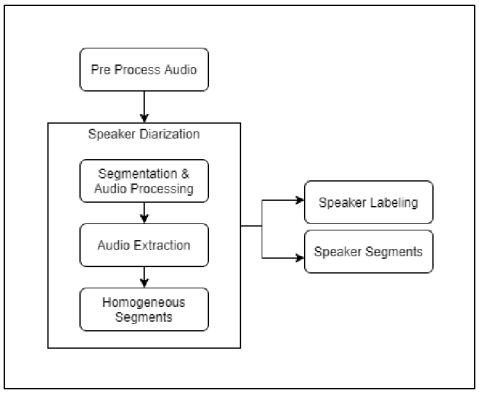

Fig. 3. System Diagram for Speaker Diarization with DNN

The model was trained in Google Colab platform. This provides free Linux environment, supporting all tensorflow and python commands. The platform also gives users access to install other packages and libraries. Google also provides free GPU for deep neural network training. But the lab environment is only saved on session based, so every command needs to be recorded and executed if the session ends suddenly. To overcome this problem, the Google Drive is mounted, and processed data and models are transferred to the Google Drive.

The model which produced the best accuracy results contained the following layers; 4 CNNs for image processing, a max pooling function and dropout. The audio feature extraction for i-vectors was done with a basic flatten layer. Then, 3 dense layers and a final layer would predict the probability of the number of speakers in the given audio. The model was trained with 50 epochs, with 100 steps per epoch. The following structure of the model was the most accurate.

D. Synthetic Speech Detection with DNN

The approach was initiated by analyzing several research papers and publications to understand which architecture is best suited for the synthetic speech detection domain. The subsections describe the relevant methodology.

a) Dataset: The dataset was taken from a group of researchers (APTLY Lab) which was freely available Fake-OrReal Dataset of size 4GB. It was readily labelled and pre-processed for direct use in the training phase. The dataset contained utterances labelled as real and fake (synthetic) in the format of .wav files. The audio samples labelled 'real' were collected from online sources like YouTube and other streaming platforms. Audio samples labelled 'fake' were obtained from the output of some of the latest state-of-the-art text-to-speech systems.

b) Training and Validation: To train the model, the architecture and build of it had to be decided. A thorough study was done before concluding the architecture. The architecture was built upon Convolution Neural Networks (CNN) and Recurrent Neural Networks (RNN). The architecture of CNNs which was first published in the '80s, was later used by researchers to learn positional relation between pixels, enabling neural networks to identify shapes and patterns. The scheme behind a convolutional layer is to break down an image into miniature squares, and with the use of convolution operator to differentiate each square to learnable filters. Stacking convolutional layers with fully connected layers results in forming a powerful architecture that has the ability to identify patterns from shapes to complex objects, lowest layers to highest layers [9]. The concept of RNNs is that the output of one layer will be the input to the same layer. What happens is that this mechanism provides the architecture a "memory" and allows to comprehend correlation in sequences. This helps in remembering past and its present decisions which are influenced by what it learnt from past experiences [13], [12].The idea of using both architectures in a single system is that CNNs are good at feature extraction and RNNs are good at identifying long-term dependencies in a time domain. Therefore, both architectures could produce better accurate results. This model was also trained in the Google Colab platform.

The structure of the model was designed with 4 CNNs, followed by a max pooling and dropout. And then a flatten layer is used to extract the different features of the audio.

Next, 4 variants of dense layers and finally an activation layer with 2 classes for the fake and real predictions. The best model containing the most accurate results had undergone 50 epochs and 100 steps per each epoch.

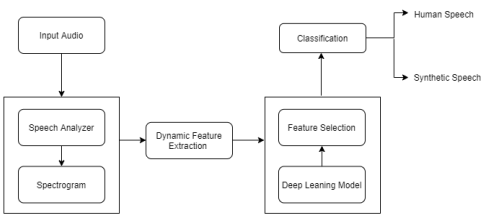

Fig. 4. System Diagram for Synthetic Speech Detection with DNN

IV. Results and Discussion

A. Speech Denoising using DNN

This research is not focused on removing the previously added noises but removing any kind of background noise from the audio file which can be used in day-to-day life scenarios. A different approach is used here to predict background noises before reducing them that has been explained in the methodology section. Adaptive filters have also given good results compared to other filtering methods in echo cancelation as stated in the literature review. Spectral Subtraction has not been used for this research since adaptive filters, Gaussian filter, and Butterworth filter covered the noise reduction perfectly as mentioned in the literature review section.

According to the methodology section, after training the model using two different neural networks such as MLP and CNN, the testing and validation process were carried out separately. Accuracy of the model on both the training and test data sets are as below.

TABLE I
Training and Testing Accuracy

Model	Training Accuracy	Testing Accuracy
MLP Architecture	93%	88%
CNN Architecture	94%	89%

Classification Accuracy(a) can be calculated using correct classifications(c) and number of classifications(n) by following equation. [14]

$$c/n = a \tag{1}$$

The validation and prediction process was done using two separate dataset such as a sample test dataset and various copyright-free audio set from the internet and both validation processes were succeeded.

B. Speaker Diarization with NLP

Speech-to-text conversion achieved 93% accuracy. Preprocessed audio provides more accuracy.

TF-IDF (term frequency-inverse document frequency), is a numerical statistic used to describe documents in the Vector Space Model, especially on IR problems. Term frequency is calculated for a word as the ratio of the number of times the word occurs in the document to the total number of words in a document. The inverse document frequency is calculated by dividing the total number of documents by the number of documents obtaining the term and taking the logarithm of that quotient.

$$tf - idf_{t,d} = tf_{t,d} \times idf_t \tag{2}$$

The complete training of models excludes sections of unidentified speakers. Speaker detection error rates are found higher when identifying the next or previous speakers. The change of speakers is quite challenging. Some of the errors could not be traced due to proper names or missing places. The speaker identification error rate on the 65hours of training transcripts was unexpectedly low. This approach needs to be combined with the automatic partitioning of the procedure of homogeneous speakers. The next step would be an automatic filter for the diarization of the whole process.

C. Speaker Diarization with DNN

Just as the proposed methodology, a model was designed, and the necessary pre-processing was done to the downloaded dataset. After successfully training, a moderate accuracy level was achieved. During the testing phase, some flaws were identified, and few assumptions had to be made. In the audio analysis for speaker separation, there is another problem that was identified after identifying the flaw. The cocktail party problem- the human brain can focus one's auditory attention on a specific stimulus while filtering out a variety of other stimuli. An example would be a person who has entered a noisy party could focus on a single conversation. The denoising process is done so the background noises are filtered before the speaker diarization process starts. The problem faced was if the audio sample had two speakers speaking at the same time. To overcome this problem, the dataset went through some reductions and the model was trained again. Due to this limitation, an assumption was made that the audio input had speakers which uttered one at a time. The proposed model achieved an accuracy of 80% and a loss value of 0.47.

Diarization error rate (DER) is the standard metric for comparing and evaluating speaker diarization systems. Its definition is as below:

$$DER = (falsealarm + misseddetection + confusion)/total \tag{3}$$

DER can be defined as the sum of; false alarm- the time period of incorrectly classified non-speech as speech; missed detection- duration of speech that is incorrectly specified as non-speech; and confusion- the duration of speaker confusion divided by the total duration of speech. The best diarization error rate that I achieved for the speaker diarization system was 0.52. Previous studies related to Speaker Diarization has achieved DER up to 0.56 [23]. Fine-tuning and optimization of the system prove to upgrade the model, but these results are proven to be time-consuming.

D. Synthetic Speech Detection with DNN.

The implementation was done in the Google Colab platform which is a free cloud service, provides a dedicated runtime environment also supports free GPU for machine learning/ deep learning. The results for the proposed model was measured by the loss function and accuracy, during and after training. It is the measure of how accurate your model's prediction is compared to the true data and the loss value implies how poorly/well a model behaves after each iteration of optimization. The result of the model achieved an accuracy of 94% and a loss value of 0.691.

We have used three different approaches on the audio for extracting features: Short-Term-Fourier-Transformation (STFT): type of Fourier transform which calculates the frequency content of local sections of the signal, Mel-Spectrogram: it is similar to STFT but the non-linear mel-scale frequency and Mel-Frequency Cepstral Coefficients (MFCC): derived from the cepstral representation of an audio clip, the coefficients that jointly make the MFC. This was performed for the proposed deep learning architecture. The results achieved for each feature compared to the previous [9] are as follows:

TABLE II
MODEL COMPARISON

Algorithm	SFTF	Mel	MFCC
Proposed Model	47.55%	41.14%	88.80%
Previous Studies [9]	50.10%	41.57%	90.48%

However, the outcome did not meet expectations. To overcome this problem a new model was trained using a pre-trained model, here the last layer is added to a pre-trained model. This approach is known as "Transfer learning", the advantage is the pre-trained models have been trained with thousands of data and have better accuracy results. The pre-trained model chosen was VGG19, the specialty of this model is that it was trained with audio data. That is images of spectrograms of the audio dataset. Out of the two trained models the model trained with Transfer learning had better accuracy results.

V. CONCLUSION

Within this research work, an innovative approach is made to train and validate the models to achieve some of the best accuracy results. To accomplish this, we have used some of the best available datasets prepared by researchers which were freely available for public use. This paper has included audio signal processing techniques, speaker diarization approaches, and synthetic speech detection mechanisms. Though signal denoising tasks were accomplished with a satisfactory result, the methods can be improved by implementing better filtering techniques and improving the datasets as well. Audio to text generation could be more accurate since this is a vital input to deal with Natural Language Processing based algorithms. Audio-based speaker diarization comes to an important play for this research. Reduced false positive can help to achieve a better outcome. The fine-tuned solution will be suitable for different security-critical organizations as well as government organizations to improve security measures. The next steps could be taken to implement fully automated features such as automatic audio processing, filtering, diarization, and detection with improved models that might add great advantage to this solution.

REFERENCES

[1] F. Ahmad, "The Role of Deepfake Audio in the Growing AI Voice Market," Medium, 2019. [Online]. Available: https://medium.com/@mfaizan.ahmad/the-role-of-deepfake-audio-in-the-growing-ai-voice-market-9246365442f5.

[2] P. Rao, "Audio signal processing," Stud. Comput. Intell., vol. 83, pp. 169–189, 2008.

[3] N. Melethadathil, P. Chellaiah, B. Nair, and S. Diwakar, "Classification and clustering for neuroinformatics: Assessing the efficacy on reverse-mapped NeuroNLP data using standard ML techniques," 2015 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2015, pp. 1065–1070, 2015.

[4] H. Yu, Z. H. Tan, Z. Ma, R. Martin, and J. Guo, "Spoofing Detection in Automatic Speaker Verification Systems Using DNN Classifiers and Dynamic Acoustic Features," IEEE Trans. Neural Networks Learn. Syst., vol. 29, no. 10, pp. 4633–4644, 2018.

[5] J. Vijayakumar, "A Systematic Algorithm for Denoising Audio Signal Using Savitzky - Golay Method," no. April, pp. 676–679, 2018.

[6] C. Cole, M. Karam, and H. Aglan, "Increasing additive noise removal in speech processing using spectral subtraction," Proc. - Int. Conf. Inf. Technol. New Gener. ITNG 2008, no. May, pp. 1146–1147, 2008.

[7] H. Magsi, A. H. Sodhro, F. A. Chachar, and S. A. K. Abro, "Analysis of signal noise reduction by using filters," 2018 Int. Conf. Comput. Math. Eng. Technol. Inven. Innov. Integr. Socioecon. Dev. iCoMET 2018 - Proc., vol. 2018-Janua, pp. 1–6, 2018.

[8] P. Podder, M. Mehedi Hasan, M. Rafiqul Islam, and M. Sayeed, "Design and Implementation of Butterworth, Chebyshev-I and Elliptic Filter for Speech Signal Analysis," Int. J. Comput. Appl., vol. 98, no. 7, pp. 12–18, 2014.

[9] R. A. M. Reimao, "Synthetic Speech Detection Using Deep Neural Networks," 2019.

[10] A. Deshpande, "A Beginner's Guide To Understanding Convolutional Neural Networks," 2016. [Online]. Available: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/.

[11] A. Rockikz, "How to Convert Speech to Text in Python," 2020. [Online]. Available: https://www.thepythoncode.com/article/using-speech-recognition-to-convert-speech-to-text-python.

[12] W. Feng, N. Guan, Y. Li, X. Zhang, and Z. Luo, "Audiovisual speech recognition with multimodal recurrent neural networks," Proc. Int. Jt. Conf. Neural Networks, vol. 2017-May, no. March 2018, pp. 681–688, 2017.

[13] M. Venkatachalam, "Recurrent Neural Networks," 2019. [Online]. Available: https://towardsdatascience.com/recurrent-neural-networks-d4642c9bc7ce.

[14] C. J. and J. P. B. J. Salamon, "URBAN-SOUND8K DATASET," 2014. [Online]. Available: https://urbansounddataset.weebly.com/urbansound8k.html.

[15] "Free Sound." [Online]. Available: https://freesound.org/.

[16] M. Smales, "Classifying Urban sounds using Deep Learning," 2018.

[17] A. Trivedi, N. Pant, P. Shah, S. Sonik, and S. Agrawal, "Speech to text and text to speech recognition systems-Areview," IOSR J. Comput. Eng., vol. 20, no. 2, pp. 36–43, 2018.

[18] A. Rockikz, "How to Convert Speech to Text in Python," 2020. [Online]. Available: https://www.thepythoncode.com/article/using-speech-recognition-to-convert-speech-to-text-python.

[19] Jörg Tiedemann, "Conversational Dataset."

[20] "ICSI Corpus Meeting." [Online]. Available: http://groups.inf.ed.ac.uk/ami/icsi/download/.

[21] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., vol. 2018-April, pp. 5239–5243, 2018.

[22] C. Wooters and M. Huijbregts, "The ICSI RT07s speaker diarization system," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 4625 LNCS, pp. 509–519, 2008.

[23] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," IEEE Trans. Audio, Speech Lang. Process., vol. 14, no. 5, pp. 1557–1565, 2006.