SPEECH EMOTION RECOGNITION

PROJECT REPORT

Submitted by

JOSIKA P - 2019202020

submitted to the Faculty of

INFORMATION SCIENCE AND TECHNOLOGY

in partial fulfillment for the award of the degree

of

MASTER OF COMPUTER APPLICATON

in

INFORMATION SCIENCE AND TECHNOLOGY

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY

CHENNAI 600 025

MAY 2022

BONAFIDE CERTIFICATE

Certified that this project report titled SPEECH EMOTION RECOGNITION is the bonafide work of JOSIKA P(2019202020) carried out project work under my supervision. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on this or any other candidate.

PLACE:

DATE:

MS.R.L.JASMINE

TEACHING FELLOW

PROJECT GUIDE

DEPARTMENT OF IST, CEG

ANNA UNIVERSITY

CHENNAI 600025

DR.S.SRIDHAR

HEAD OF THE DEPARTMENT

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY CHENNAI

600025

ABSTRACT

This project aims at building and training speech and emotion recognition system by using machine learning and deep learning algorithm which uses CNN which is a type of artificial neural network which is widely used for image/object recognition. This project is to predict emotion based on the speech used to perform analytical research by applying different machine learning algorithms and neural networks with different architecture and compare their results for insights. This project uses four kinds of data sets RAVDESS,TESS,SAVEE,CREMA-D for classification and uses emotions like neutral,calm,happy,sad,angry,fear,disgust,pleasant surprise and boredom. Feature Extraction is done using librosa library which comes under pre processing data. The next step is to build model using Convolution Neural Network then prediction is done to find accuracy with various algorithms to find the better one. Testing is done with live voices.

ACKNOWLEDGEMENT

First and foremost, we would like to express our deep sense of gratitude to our guide MS.R.L.JASMINE, Teaching Fellow, Department of Information Science and Technology, Anna University, for her excellent guidance, counsel, continuous support and patience. She has helped us to come up with this topic and guided us in the development of this project. She gave us the moral support to finish our mini project in a successful manner.

We express our gratitude to Dr.S.Sridhar,Head of the Department, Department of Information Science and Technology, Anna University, for her kind support and for providing necessary facilities to carry out the work.We are thankful to the project committee members Dr.S.Saswati Mukherjee,Professor,Dr.M.Vijalakshmi, Associate Professor, Dr.E.Uma, Assistant Professor, MS.P.S.Apirajitha,Teaching Fellow, MS.C.M.Sowmiya,Teaching Fellow Department of Information Science and Technology, Anna University, Chennai, for their valuable guidance and technical support.

We also thank all the faculty and non-teaching staff members of theInformation Science and Technology, CEG Campus, Anna University, Chennai for their valuable support throughout the course of our project work.

JOSIKA P

CHAPTER 1

INTRODUCTION

1.1 DOMAIN OF THE PROJECT

Deep Learning

Deep Learning in a single term can understand as human nervous System. Machine vision deep learning sets are made to learn over a collection of audio/image also known as training data, in order to rectify a problem. The various deep learning models trains a computer to visualize like a human.

Deep learning models based on the inputs to the nodes can visualize. Hence network type is like that of a human nervous system, with every node performing under a larger network as a neuron. So, deep learning models are basically a part of artificial neural networks. Algorithms of deep learning learns in depth about the input audio/image as it passes over every neural network Layer. Low-level characteristics like edges are detected by learning given to the initial layers, and successive layers collaborate characteristics from prior layers in a more philosophical representation.

Images, sounds, censor data and other data are those digital forms patterns which deep learning recognizes. For prediction pre-training the data and constructing a training set and testing set . As prediction obtains an optimum node such that the predicted node provides the satisfactory output.

Basis of the neurons are in different levels and created to predict at every level and the most-optimum predictions, and thereafter for the best-fit outcome use the data. It is treated as true machine intelligence.

A Convolutional Neural Network (CNN)[2] is a sort of feed-ahead artificial network in which the joining sequence among its nodes is motivated by presenting an animal visual-cortex.

Single cortical neurons give response to the stimuli at a prohibited area of region known as the receptive areas. The receptive areas of various nodes semi-overlap so that they can match the visual area. The reply of a single node for stimuli among its receptive area could be mathematically through the convolution operations. Convolutional network was motivated by natural procedures and are varieties of multi-layer perceptron formulated to use least quantity of pre-processing. They have broad use in image and video recognition, recommendation systems and NLP.

The dimensions of the Characteristics Map (convolved features) is regulated by following parameters:

- Depth refers to the filter count used in the operation.
- Stride refers to the size of the filter if the size is 5*5 the the stride will be 5.
- Zero-padding is padding the input matrix with often convenient around the border in order to apply filterto 'Input Audio' matrix's bordering elements. Using zero padding size of the characteristics map can be governed.

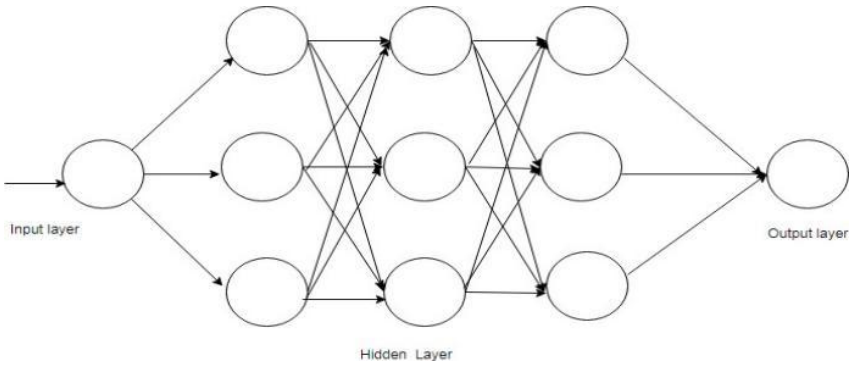


Figure 1.1 CNN overview

1.2 PROBLEM STATEMENT

- Recognition of emotions in audio signals has been a field of study in the past. Previous work in this area included use of various classifiers like SVM, Neural Networks, Bayes Classifier etc.
- The number of emotions classified varied from study to study, they play an important aspect in evaluating the accuracy of the different classifiers.
- Using machine learning models to speech emotion recognition has less accuracy in order to overcome this issue have been using a model from deep learning called convolution neural network(CNN).

1.3 MOTIVATION AND OBJECTIVE

MOTIVATION:

Speech Emotion Recognition (SER) is the task of recognizing the emotion from speech irrespective of the semantic contents. However, emotions are subjective and even for humans it is hard to notate them in natural speech communication regardless of the meaning. The ability to automatically conduct it is a very difficult task and still an ongoing subject of research.This project aims to help in building an emotion recognizer from speech data using a deep neural network.

OBJECTIVE:

The objective is to identify emotions by using various sample audio files available. In order to achieve this, convolution neural network is used which has steps like feature learning and classification. The audio files will be converted into image files and feature extraction is done using mel frequency cepstral coefficients which is a technique to extract features from the audio files then training and testing of data will be done in order to use convolution neural network.

1.4 IMPLEMENTATION PLATFORM

- Development Platform: Jupyter Notebook

- Language: Python 3

1.5 DATASET DETAILS

- Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)

This portion of the RAVDESS contains 1440 files: 60 trials per actor x 24 actors = 1440. The RAVDESS contains 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. Speech emotions includes calm, happy, sad, angry, fearful, surprise, and disgust expressions. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression.

- Surrey Audio-Visual Expressed Emotion (SAVEE)

The SAVEE database was recorded from four native English male speakers (identified as DC, JE, JK, KL), postgraduate students and researchers at the University of Surrey aged from 27 to 31 years. Emotion has been described psychologically in discrete categories: anger, disgust, fear, happiness, sadness and surprise. A neutral category is also added to provide recordings of 7 emotion categories.

- Toronto Emotional Speech Set(TESS)

There are a set of 200 target words were spoken in the carrier phrase "Say the word _' by two actresses (aged 26 and 64 years) and recordings were made of the set portraying each of seven emotions (anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral). There are 2800 data points (audio files) in total. The dataset is organised such that each of the two female actor and their emotions are contain within its own folder. And within that, all 200 target words audio file can be found. The format of the audio file is a WAV format.

- Crowd Sourced Emotional Multimodal Actors Dataset(CREMA-D)

CREMA-D is a data set of 7,442 original clips from 91 actors. These clips were from 48 male and 43 female actors between the ages of 20 and 74 coming from a variety of races and ethnicities (African America, Asian, Caucasian, Hispanic, and Unspecified). Actors spoke from a selection of 12 sentences. The sentences were presented using one of six different emotions (Anger, Disgust, Fear, Happy, Neutral, and Sad) and four different emotion levels (Low, Medium, High, and Unspecified).

CHAPTER 2

LITERATURE SURVEY

Here, Xinzhou Xu [3] et al generalized the spectral regression model exploiting the joins of extreme leaning machines (ELMs) and subspace learning (SL) was expected for overlooking the disadvantages of spectral regression based graph embedding (GE) and ELM. Using the GSR model, in the execution of speech emotion recognition (SER) had to precisely represent theses relations among data. These multiple embedded graphs were constructed to same.Demonstration over speech emotional corpora determined that the impact and feasibility of the techniques compared to prior methods that includes ELM and subspace learning (SL)techniques. The system output can be improved by exploring embedded graphs at more precise levels. Only least-square regression along with l2-norm minimization was considered in the regression stage.

Zhaocheng Huang[4] et al uses a heterogeneous token-used system to detect the speech depression. Abrupt changes and acoustic areas are solely and collectively figured out in joins among different embedding methods. Contributions towards the detection of depression were used and probably various health problems that would affects vocal generation. Landmarks are used to pull out the information particular to individual type of articulation

at a time. This is a hybrid system. LWs and AWs hold various information. AW holds section of acoustic area into single token per frame, and on the contemporary the abrupt changes in speech articulation are shown by LWs. The hybrid join of the LWs and AWs permits exploitation of various details, more specifically, articulatory dysfunction into conventional acoustic characteristics are also incorporated.

Peng Song [5] offers transfer linear subspace learning (TLSL) framework for cross corpus recognition of speech. TLSL approaches, TULSL and TSLSL were taken in count. TLSL aims to extract robust characteristics representations over corpora into the trained estimated subspace. TLSL enhances the currently used transfer learning techniques which only focuses on searching the most portable components of characteristics TLSL can reach even better results compared to the 6 baseline techniques with stats significance, and TSLSL gives better outcomes compared to TULSL, in fact all the transfer learning is more accurate than usual learning techniques. TLSL significantly excels TLDA,TPCA, TNMF and TCA, the excellent transfer learning techniques based on characteristics transformation. A big set back that these early transfer learning methods possess was that they concentrate on searching the portable components of characteristics that tend to ignore less informative section. The less informative parts are also significant when it comes to transfer learning results experimented that TLSL is implemented for cross-corpus recognition of speech emotion.

With this paper Jun Deng [6] et al focused on unsupervised learning with automatic encoders of speech emotion recognition. Significantly work was on joining generative and discriminative training, by partially supervised learning algorithms designed to settings where non-labeled data was available. The process had been sequentially evaluated with 5 databases in different settings. The proposed technique enhances recognition performance by learning the prior knowledge from non-labeled data in conditions with a smaller number of libeled examples. These techniques can solve the problems in mismatched

settings and incorporate the learnings from different domains into the classifiers, eventually resulting in outstanding performance. This shows that the model is having the capacity to make good use of the combination of labeled and non-labeled data for speech emotion recognition. The residual neural network displayed that intense architectures make the classifier beneficial to pull out complicated structure in image processing.

Ying Qin[7] et al presented Cantonese-speaking PWA narrative speech which is a base of completely automated assessment system. Experiments on the text characteristics driven by the proposed data could detect out the impairment of language in the aphasic speech. The AQ scores were significantly correlated with the text characteristics learned by the Siamese network. The improvised representation of ASR output was leveraged as the confusion network and the robustness of text characteristics were felicitated to it. There was an immediate requirement of improving the performance of ASR on aphasic speech for generation speech that has more robust characteristics. It was necessary that the databases of pathological speech and other languages to apply this proposed methodology. As seen clinically the most desirable one is automatic classification of aphasia variant along with this large-scale accumulation of data is needed substantially.

CHAPTER 3

SYSTEM DESIGN AND ARCHITECTURE

3.1 DETAILED ARCHITECTURE DIAGRAM

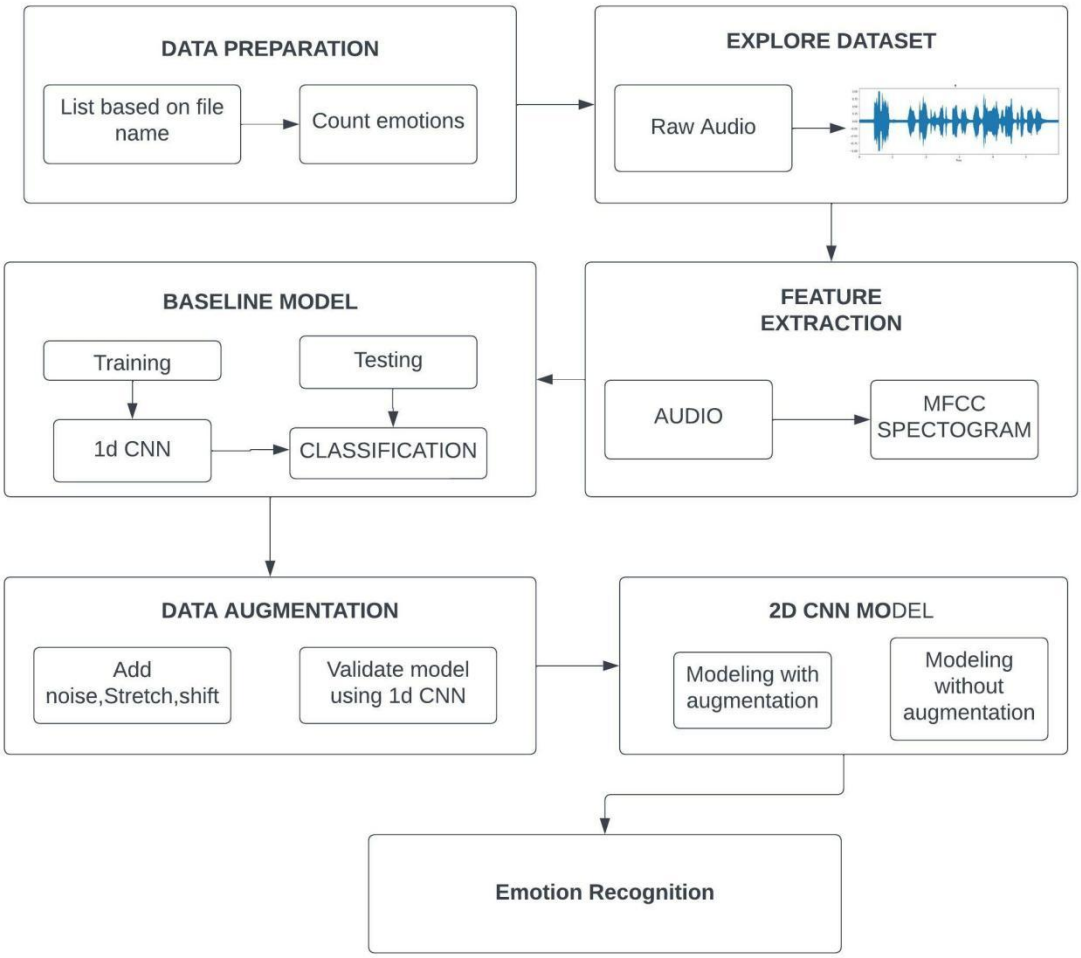


Figure 3.1:Detailed Architecture Diagram

3.2 WORKFLOW OF PROJECT

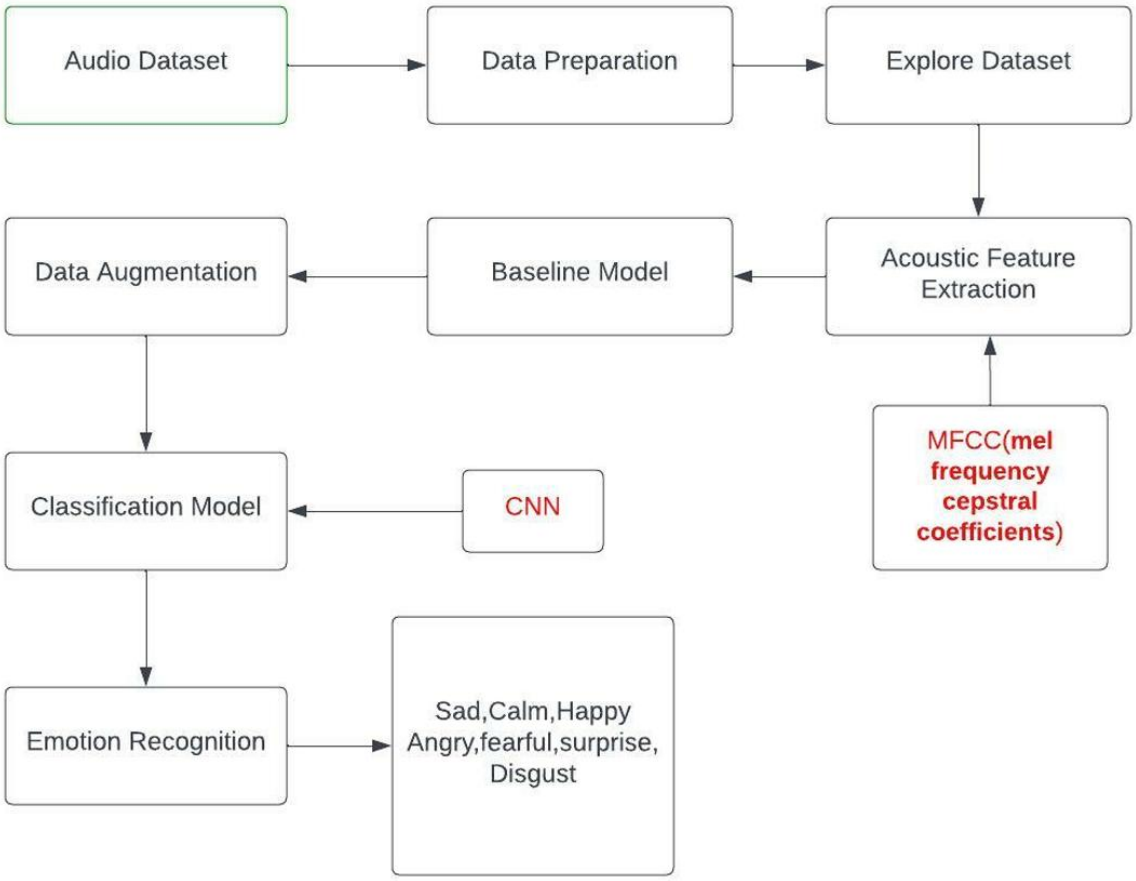


Figure 3.2 Workflow of the project

3.3 MODULES OF THE PROJECT:

1) Data Preparation
2) Explore Data set
3) Feature Extraction
4) Baseline Model
5) Data Augmentation
6) 2D CNN Classification model

Data Preparation:

Here we are about to load the 4 different datasets from kaggle which contain audio files will be saving the datasets with a variable name and will group them into a list based on the file name with explains about the audio file then inorder to proceed count the emotions based on gender.

FLOW DIAGRAM OF DATA PREPARATION:

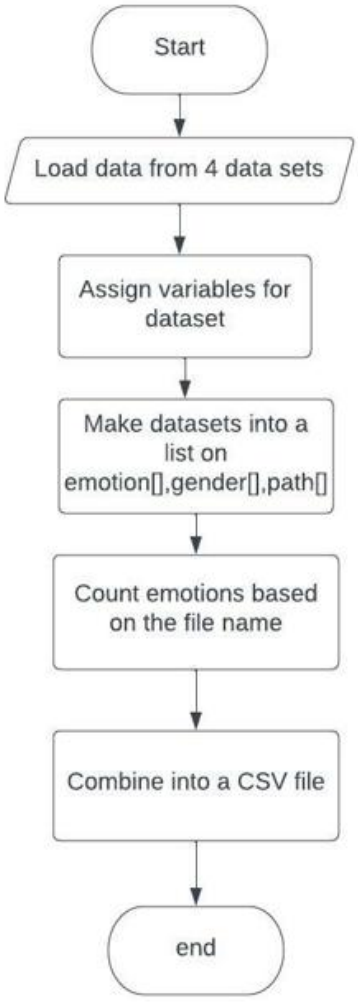


Figure 3.3 Flow diagram of data preparation.

Explore Data set:

Here will explore the audio files and plot them based on time and amplitude to check the fluxations based on different emotions and different genders.

FLOW DIAGRAM OF EXPLORE DATASET:

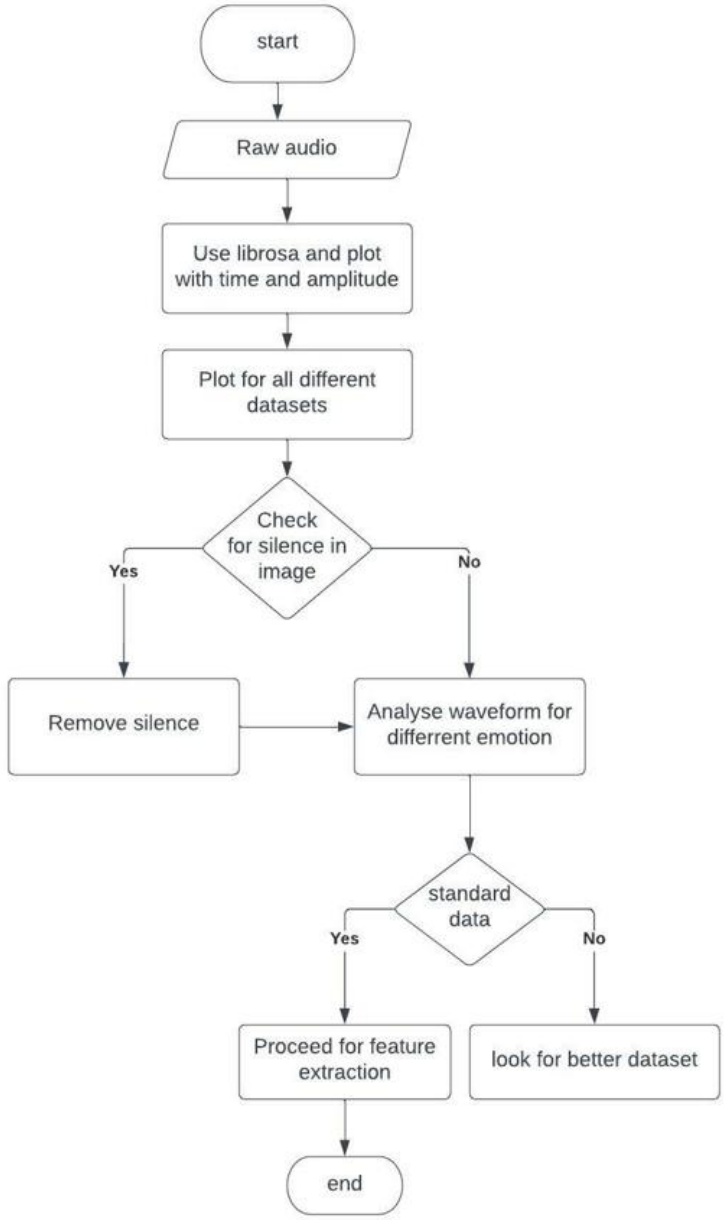


Figure 3.4 Flow diagram of explore dataset.

Feature Extraction

In this feature extraction part going to use the fast fourier transform,melscale filter bank and plot the MFCC bands with respect to time and make it into a spectogram. A spectrogram is an image that displays the variation of energy at different frequencies across time. The vertical axis (ordinate) represents frequency and the horizontal axis (abscissa) represents time. The energy or in- tensity is encoded either by the level of darkness or by the colors. There are two general types of spectrograms: wide-band spectrograms and narrow-band spectro- grams. Wide-band spectrograms has a higher time resolution than narrow-band spectrograms. This property enables the wide-band spectrograms to show individ- ual glottal pulses. In contrast, narrow-band spectrograms have higher frequency resolution than wide-band spectrograms. This feature enables the narrow-band spectrograms to resolve individual harmonics.Considering the importance of vocal fold vibration, along with the fact that glottal pulse is associated with one period of vocal fold vibration, decidedto convert all utterances into wide-band spectrograms. In doing so, the length of hamming windows were set to 5 ms with ms overlap. The number of DFT points was set to 512. Also, discarded the frequency information greater than4 kHz from

spectrograms since frequencies below 4000 Hz are sufficient for speech perception in many situations . In pilot studies, eliminating energy above 4000 Hz improved the performance of the algorithms. This gave 129 frequency points.All spectrogram images were, first, resized to have 129 × 129 pixels and, then,*z*-normalized to have zero mean and standard deviation close to one.

(A) Wide-band spectrogram with 5 ms Hamming window;
(B) narrow-band spectrogram with 25 ms Hamming window.

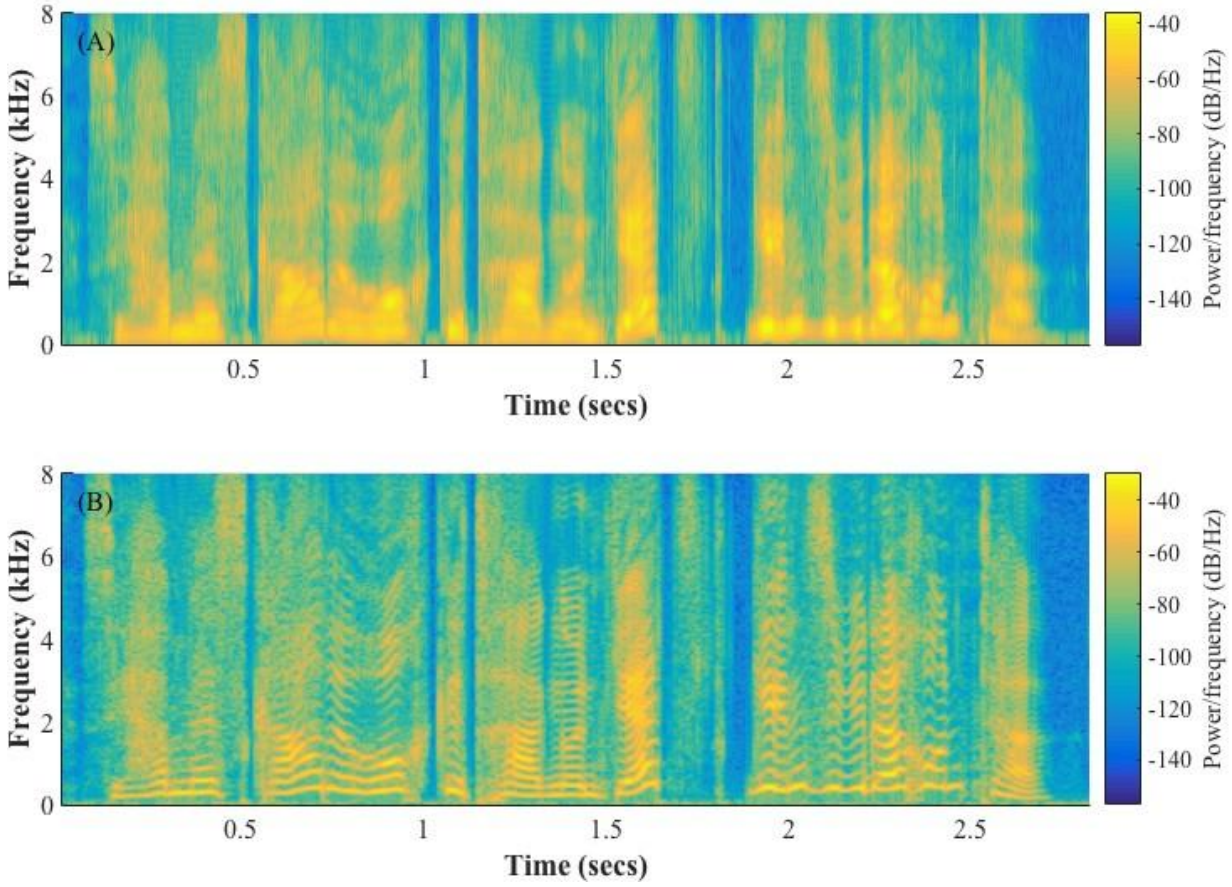


Figure 3.5 Wide band and Narrowband Spectogram

ALGORTIHM FLOW:

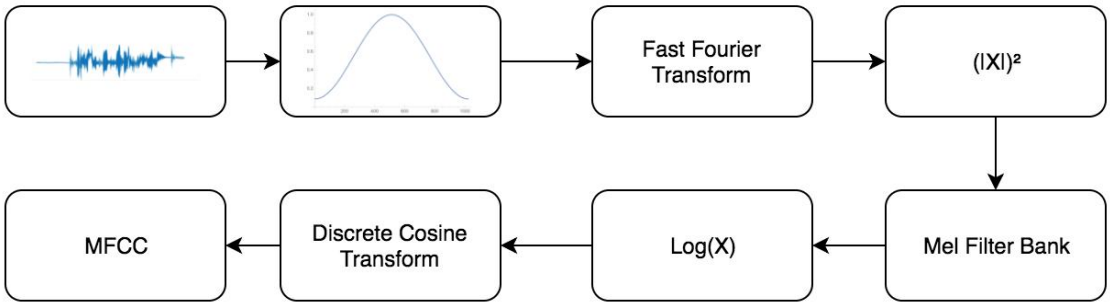


Figure 3.6 Algorithm flow

FLOW DIAGRAM OF FEATURE EXTRACTION:

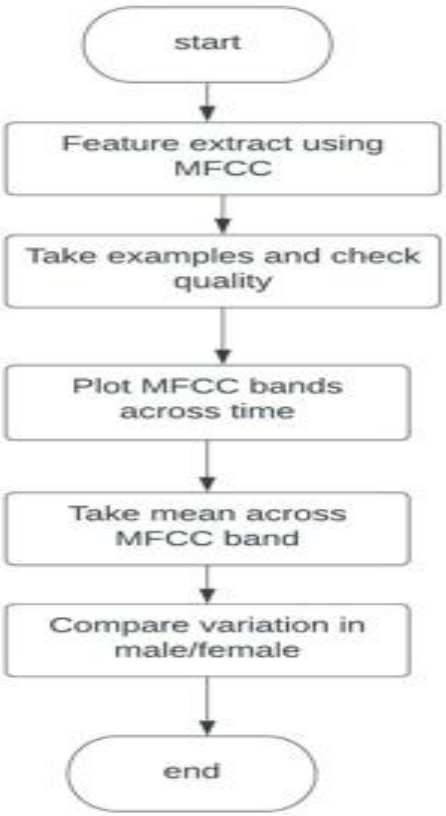


Figure 3.7 Flow diagram of feature extraction

Baseline model:

Within this module train the model for accuracy estimations. 1st, import necessary modules. Then pull the data set. Will receive the sampling rate value with librosa packages and mfcc function. Thereafter this value holds other variables. Now audio files and mfcc value hold a variable consequently it will add a list. Then zip the listand hold two variables x & y. Then have represented (x, y) shape values with the use of numpy package.

Speech represented in the form of image with 3 layers. While using CNN, do consider, 1st and 2nd derivatives of speech image with time and frequency. CNN can predict, analyze the speech data, CNN can learn from speeches and identify words or utterances.

FLOW DIAGRAM OF BASELINE MODEL:

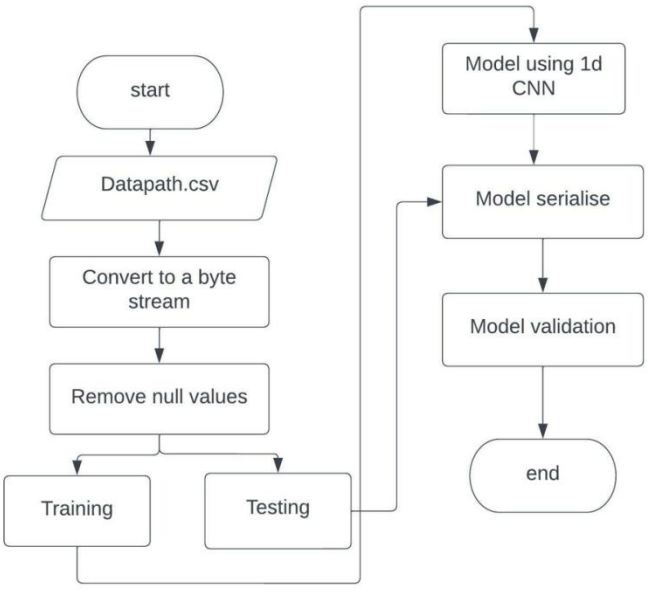


Figure3.8 Flow diagram of Baseline Model

DATA AUGMENTATION:

Static noise

This process involves addition of noise i.e. white noise to the sample. White noises are random samples distributed at regular intervals with mean of 0 and standard deviation of 1.

Time Shifting

Shift the wave by sample_rate/10 factor. This will move the wave to the right by given factor along time axis.

Stretch

This one is one of the more dramatic augmentation methods. The method literally stretches the audio. So the duration is longer, but the audio wave gets strecthed too. Thus introducing and effect that sounds like a slow motion sound. Look at the audio wave itself, you'll notice that compared to the orginal audio, the strected audio seems to hit a higher frequency note. Thus creating a more diverse data for augmentation.

Pitch

This method accentuates the high pitch notes, by... normalising it sort of.

FLOW DIAGRAM OF DATA AUGMENTAION:

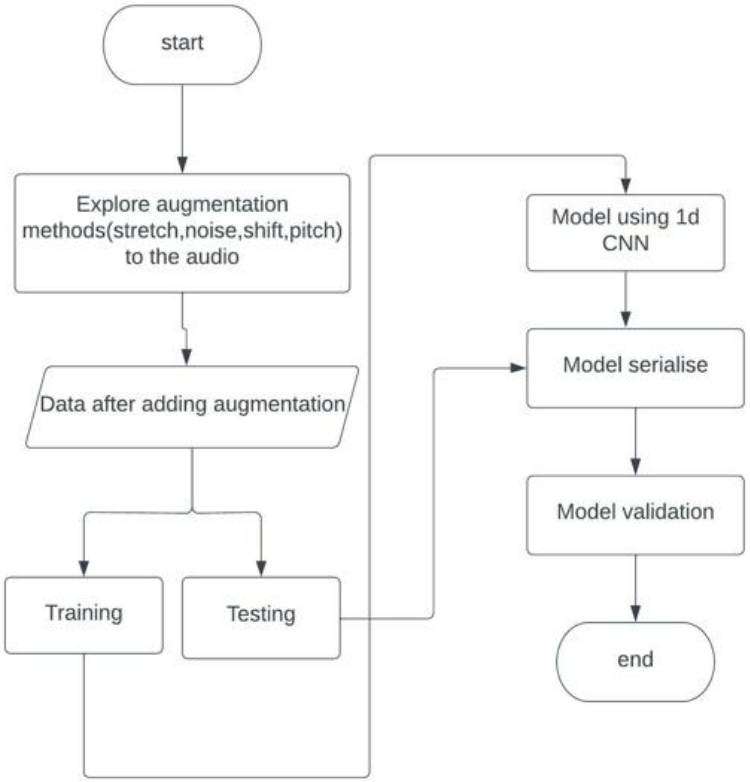


Figure 3.9 Flow Diagram of data augmentation

2D CNN CLASSIFICATION

Here will be adding the custom functions.2D CNN is a array of 30MFCC by 216 audio length as input data. Here will be finding various results using MFCC with augmentation and without augmentation and will also be using another Feature extraction method to compare the accuracy rate with MFCC the method are going to use here is log-melspectogram. Will be comparing all the inference with 1D CNN model and 2D CNN model with different feature extraction methods and the audio with

augmentation and without augmentation to predict the correct emotions. Below explains the architecture of 2D CNN.

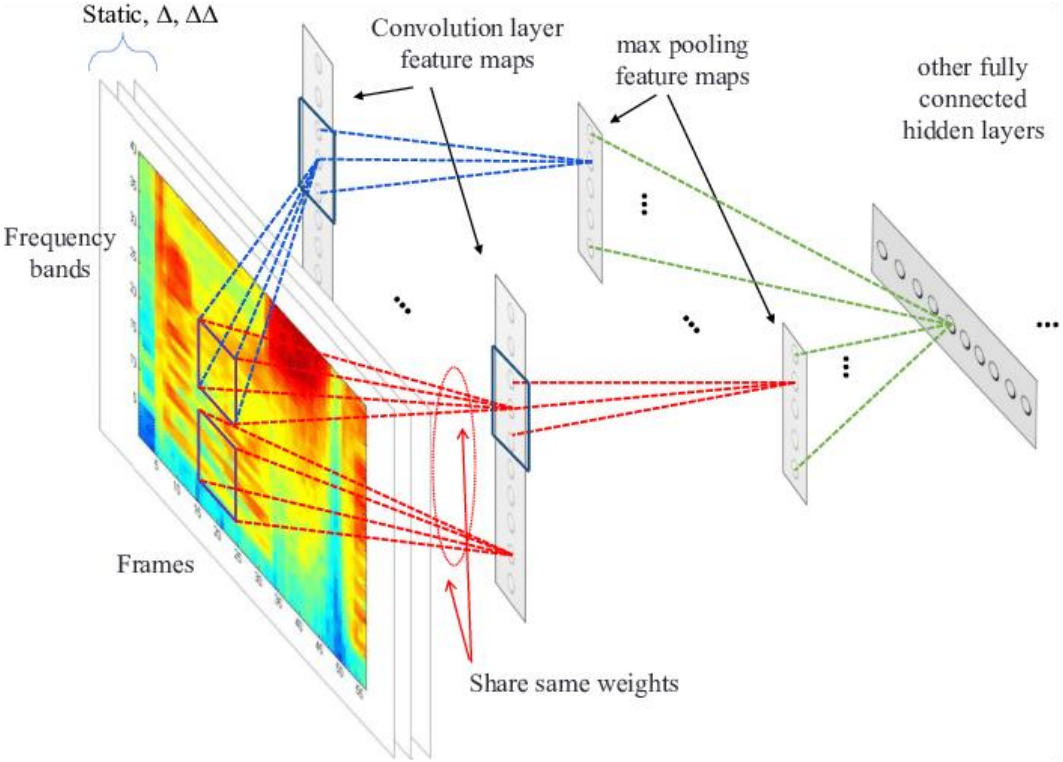


Figure 3.10 CNN using MFCC

FLOW DIAGRAM OF 2D CNN MODEL:

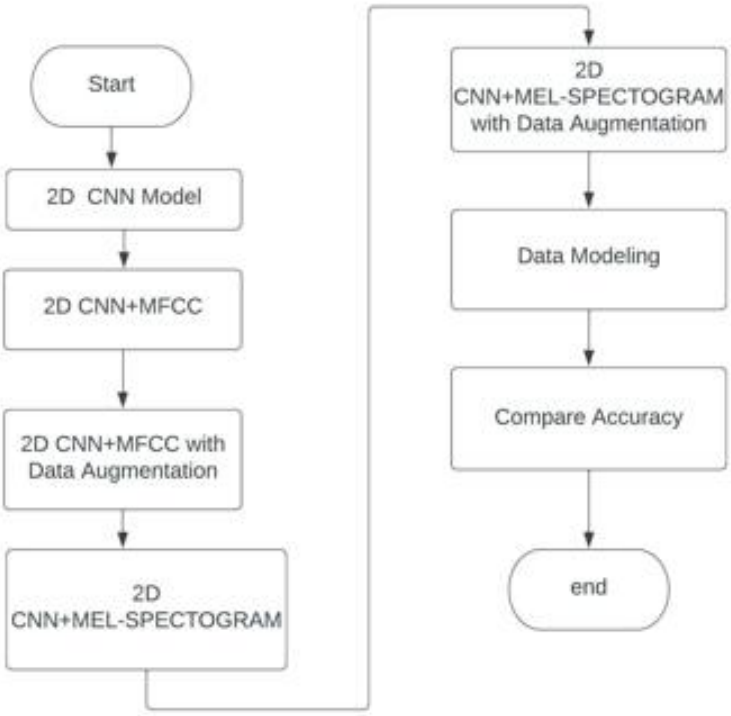


Figure 3.11 Flow diagram of 2D CNN model

CHAPTER 4
ALGORITHM AND IMPLEMENTATION

4.1 DATA PREPARATION:

ALGORITHM:

STEPS:

1. Load the data set from kaggle.
2. Assign variables for the data set.
3. Make the datasets into a list based on emotion ,gender,path.
4. Count the emotions based on the list which have created using file name.
5. Combine all into a csv file.

4.2 EXPLORE DATA:

ALGORITHM:

STEPS:

1. Read the raw audio.

2. Using librosa plot the time,amplitude graph.
3. Plot for all different data sets.
4. Check for silence in the image.
5. Analyse waveform for different emotion.
6. Confirm that it is a standard data.
7. Proceed for feature extract.

4.3 FEATURE EXTRACTION:

ALGORITHM:

STEPS:

1. Read the data set.
2. Use MFCC algorithm for feature extraction.
3. Find the statistical features.
4. Compare the variations between male and female voice.

MFCC ALGORITHM FLOW:

1. Pre emphasis

2. Framing

3. Windowing

4. Fast Fourier Transform

5. Mel filter Bank

6. Discrete cosine transform

4.4 BASELINE MODEL:

ALGORITHM:

STEPS:

1. Save Mfcc into list
2. Remove Null values
3. Split data set for training and testing
4. Normalize the data
5. Model using CNN
6. Model serialization
7. Model validation
8. Check accuracy

4.5 DATA AUGMENTATION:

ALGORITHM:

STEPS:

1. Explore augmentation methods for audio data
2. Add noise,pitch,stretch,shift to audio
3. Pre process data for modeling
4. Model using CNN for new audio file
5. Model serialization
6. Model validation
7. Check accuracy

4.6 2D CNN model:

ALGORITHM:

STEPS:

1. model using 2D CNN
2. Use 2D CNN+MFCC+augmented data
3. Use 2D CNN+MFCC+without augmented data
4. Try other feature extraction methods
5. Data modeling
6. Compare accuracy

OUTPUT SCREEN SHOTS

DATA PREPARATION

In [134]:
```
#loading data with diff key words
TESS = "C:/Users/mailt/Desktop/FP/TESS Toronto emotional speech set data/"
RAV = "C:/Users/mailt/Desktop/FP/audio_speech_actors_01-24/"
SAVEE ="C:/Users/mailt/Desktop/FP/ALL/"
CREMA = "C:/Users/mailt/Desktop/FP/AudioWAV/"

# Running one example
dir_list = os.listdir(SAVEE)
dir_list[:5]


```

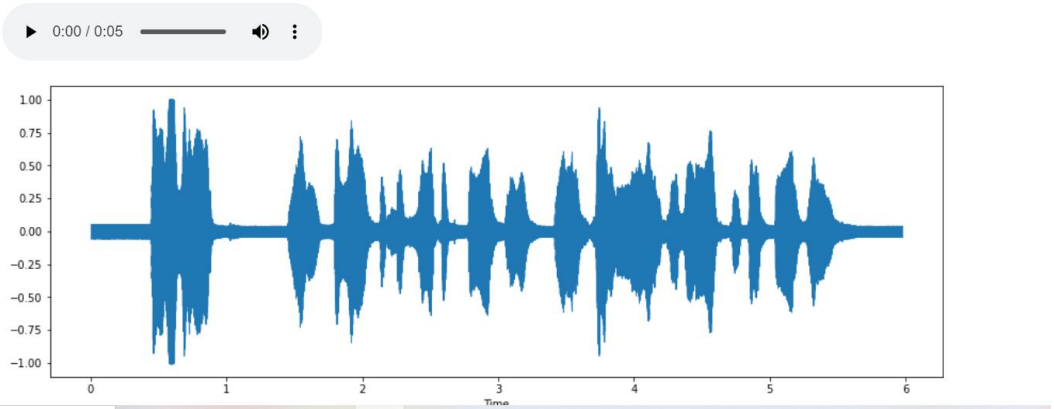
Out[134]:
```
['DC_a01.wav', 'DC_a02.wav', 'DC_a03.wav', 'DC_a04.wav', 'DC_a05.wav']
```

```
emotion=[]
path = []
for i in dir_list:
    if i[-8:-6]=='_a':
        emotion.append('male_angry')
    elif i[-8:-6]=='_d':
        emotion.append('male_disgust')
    elif i[-8:-6]=='_f':
        emotion.append('male_fear')
    elif i[-8:-6]=='_h':
        emotion.append('male_happy')
    elif i[-8:-6]=='_n':
        emotion.append('male_neutral')
    elif i[-8:-6]=='sa':
        emotion.append('male_sad')
    elif i[-8:-6]=='su':
        emotion.append('male_surprise')
    else:
        emotion.append('male_error')
    path.append(SAVEE + i)

# Now check out the label count distribution
SAVEE_df = pd.DataFrame(emotion, columns = ['labels'])
SAVEE_df['source'] = 'SAVEE'
SAVEE_df = pd.concat([SAVEE_df, pd.DataFrame(path, columns = ['path'])], axis = 1)
SAVEE_df.labels.value_counts()
```

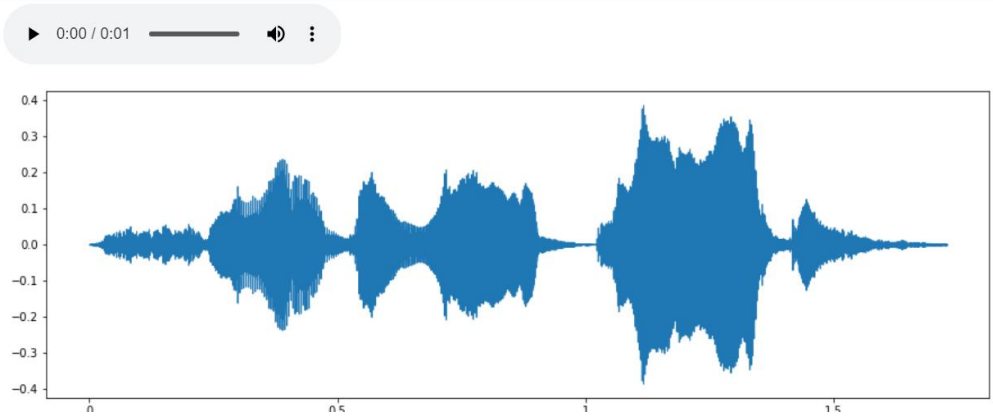
```
7]: male_neutral     120
    male_surprise     60
    male_sad          60
    male_disgust      60
    male_angry        60
```

EXPLORE DATA

```
# happy_speech_analyse
fname = SAVEE + 'DC_h11.wav'
data, sampling_rate = librosa.load(fname)
plt.figure(figsize=(15, 5))
librosa.display.waveshow(data, sr=sampling_rate)

# Lets play the audio
ipd.Audio(fname)
```



```
# fearful track
fname = TESS + 'YAF_fear/YAF_dog_fear.wav'

data, sampling_rate = librosa.load(fname)
plt.figure(figsize=(15, 5))
librosa.display.waveshow(data, sr=sampling_rate)

# Lets play the audio
ipd.Audio(fname)
```



FEATURE EXTRACT :

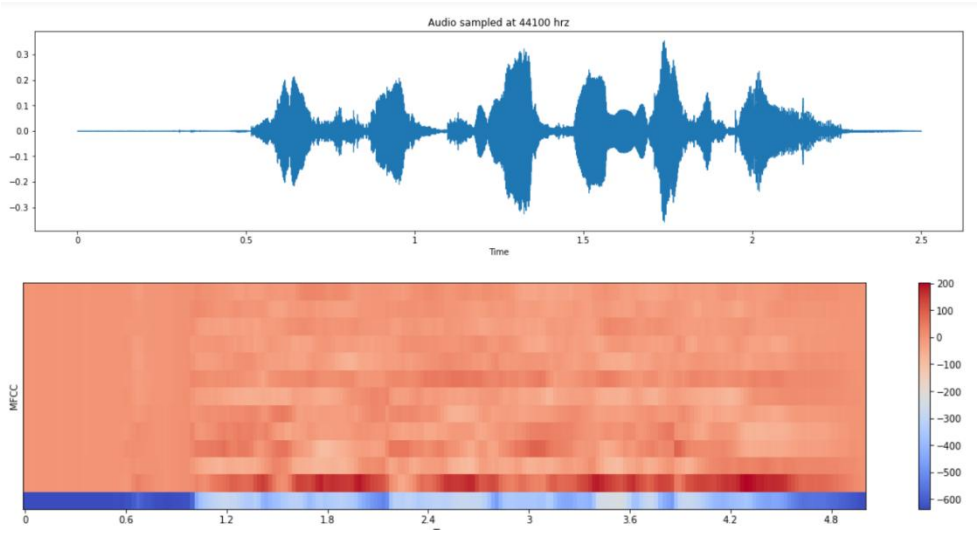
In [108]:

```
# lets pick up the meta-data that we got from our first part of the Kernel
ref = pd.read_csv("Data_path.csv")
ref.head(700)
```

Out[108]:

	labels	source	path
0	male_angry	SAVEE	C:/Users/mailt/Desktop/FP/ALL/DC_a01.wav
1	male_angry	SAVEE	C:/Users/mailt/Desktop/FP/ALL/DC_a02.wav
2	male_angry	SAVEE	C:/Users/mailt/Desktop/FP/ALL/DC_a03.wav
3	male_angry	SAVEE	C:/Users/mailt/Desktop/FP/ALL/DC_a04.wav
4	male_angry	SAVEE	C:/Users/mailt/Desktop/FP/ALL/DC_a05.wav
...	...	...	...
695	female_angry	RAVDESS	C:/Users/mailt/Desktop/FP/audio_speech_actors_...
696	female_fear	RAVDESS	C:/Users/mailt/Desktop/FP/audio_speech_actors_...
697	female_fear	RAVDESS	C:/Users/mailt/Desktop/FP/audio_speech_actors_...
698	female_fear	RAVDESS	C:/Users/mailt/Desktop/FP/audio_speech_actors_...
699	female_fear	RAVDESS	C:/Users/mailt/Desktop/FP/audio_speech_actors_...

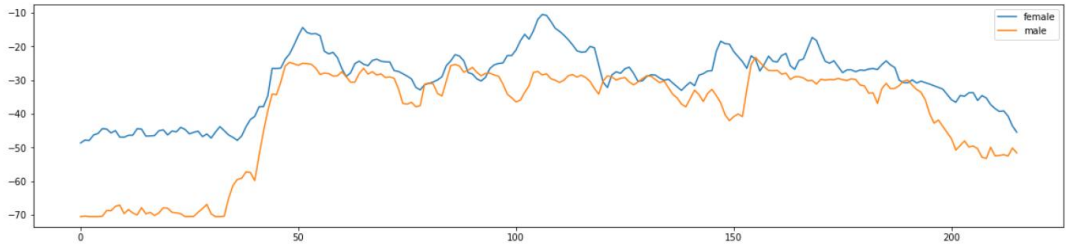
700 rows × 3 columns




```
# Source - RAVDESS; Gender - Female; Emotion - Angry
path = "C:/Users/mailt/Desktop/FP/audio_speech_actors_01-24/Actor_08/03-01-05-02-01-01-08.wav"
X, sample_rate = librosa.load(path, res_type='kaiser_fast',duration=2.5,sr=22050*2,offset=0.5)
female = librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=13)
female = np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=13), axis=0)
print(len(female))

# Source - RAVDESS; Gender - Male; Emotion - Angry
path = "C:/Users/mailt/Desktop/FP/audio_speech_actors_01-24/Actor_09/03-01-05-01-01-01-09.wav"
X, sample_rate = librosa.load(path, res_type='kaiser_fast',duration=2.5,sr=22050*2,offset=0.5)
male = librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=13)
male = np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=13), axis=0)
print(len(male))

# audio wave
plt.figure(figsize=(20, 15))
plt.subplot(3,1,1)
plt.plot(female, label='female')
plt.plot(male, label='male')
plt.legend()

```

```
216
216
```

```
<matplotlib.legend.Legend at 0x1bf3729c0d0>
```



```
import json
import os
import math
```

```
DATASET_PATH = "C:/Users/mailt/Desktop/FP/combine"
JSON_PATH = "data_10.json"
SAMPLE_RATE = 22050
TRACK_DURATION = 30 # measured in seconds
SAMPLES_PER_TRACK = SAMPLE_RATE * TRACK_DURATION
```

```
def save_mfcc(dataset_path, json_path, num_mfcc=13, n_fft=2048, hop_length=512, num_segments=5):
    """Extracts MFCCs from music dataset and saves them into a json file along with genre labels.
        """

    # dictionary to store mapping, labels, and MFCCs
    data = {
        "mapping": [],
        "labels": [],
        "mfcc": []
    }

    samples_per_segment = int(SAMPLES_PER_TRACK / num_segments)
    num_mfcc_vectors_per_segment = math.ceil(samples_per_segment / hop_length)

    # loop through all genre sub-folder
    for i, (dirpath, dirnames, filenames) in enumerate(os.walk(dataset_path)):

        # ensure we're processing a genre sub-folder level
        if dirpath is not dataset_path:
```

```
                signal, sample_rate = librosa.load(file_path, sr=SAMPLE_RATE)

                # process all segments of audio file
                for d in range(num_segments):

                    # calculate start and finish sample for current segment
                    start = samples_per_segment * d
                    finish = start + samples_per_segment

                    # extract mfcc
                    mfcc = librosa.feature.mfcc(signal[start:finish], sample_rate, n_mfcc=num_mfcc, n_fft=n_fft, hop_length=
                    mfcc = mfcc.T

                    # store only mfcc feature with expected number of vectors
                    if len(mfcc) == num_mfcc_vectors_per_segment:
                        data["mfcc"].append(mfcc.tolist())
                        data["labels"].append(i-1)
                        print("{}, segment:{}".format(file_path, d+1))

    # save MFCCs to json file
    with open(json_path, "w") as fp:
        json.dump(data, fp, indent=4)
```

```
save_mfcc(DATASET_PATH, JSON_PATH, num_segments=10)
```

```
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_ITH_SAD_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_ITS_ANG_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_ITS_NEU_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_ITS_SAD_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_IWL_DIS_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_IWL_NEU_XX.wav, segment:1
C:/Users/mailt/Desktop/FP/combine\AudioWAV\1006_IWL_SAD_XX.wav, segment:1
```

REFERENCES:

1. Y. Chen, Z. Lin, X. Zhao, S. Member, G. Wang, and Y. Gu, “Deep Learning-Based Classi fi cation of Hyperspectral Data,” pp. 1–14, 2014.
2. L. Chua and T. Roska, “The CNN Paradigm,” vol. 4, no. 9208, pp. 147–156, 1993.
3. X. Xu, J. Deng, E. Coutinho, C. Wu, and L. Zhao, “Connecting Subspace Learning and Extreme Learning Machine in Speech Emotion Recognition,” *IEEE*, vol. XX, no. XX, pp. 1–13, 2018.
4. Z. Huang, J. Epps, D. Joachim, and V. Sethu, “Natural Language Processing Methods for Acoustic and Landmark Event-based Features in Speech-based Depression Detection,” *IEEE J. Sel. Top. Signal Process.*, vol. PP, no. c, p. 1, 2019.

5. P. S. Member, “Transfer Linear Subspace Learning for Cross-corpus Speech Emotion Recognition,” vol. X, no. X, pp. 1–12, 2017.
6. J. Deng, X. Xu, Z. Zhang, and S. Member, “Semi-Supervised Autoencoders for Speech Emotion Recognition,” vol. XX, no. XX, pp. 1–13, 2017.
7. Y. Qin, S. Member, T. Lee, A. Pak, and H. Kong, “Automatic Assessment of Speech Impairment in Cantonese-speaking People with Aphasia,” *IEEE J. Sel. Top. Signal Process.*, vol. PP, no. c, p. 1, 2019.
8. M. D. Zeiler *et al.*, “ON RECTIFIED LINEAR UNITS FOR SPEECH PROCESSING New York University , USA Google Inc ., USA University of Toronto , Canada,” pp. 3–7.