F/SOP FYDP/09/00

UNDERGRADUATE FINAL YEAR PROJECT REPORT
Department of Computer & Information System Engineering
NED University of Engineering and Technology

Behavioral Monitoring Using Vocal Sentiment Analysis

Group Number: 03 **Batch: 2018**

Group Member Names:

Shehroz Waseem	CS-18121
Sukaina Asad	CS-18128
Abaad Murtaza	CS-18131
Aiman Nisar	CS-18140

Approved by

………………………………………………………………………………………………...

Dr. Urooj Ainuddin
Associate professor
Project Advisor

© NED University of Engineering & Technology. All Rights Reserved – March 2022

Authors Declaration

We declare that we are the sole authors of this project. It is the actual copy of the project that was accepted by our advisor(s) including any necessary revisions. We also grant NED University of Engineering and Technology permission to reproduce and distribute electronic or paper copies of this project.

Signature and Date	Signature and Date	Signature and Date	Signature and Date
....	....	....	....
Shehroz Waseem	SukainaAsad	Abaad Murtaza	Aiman Nisar
CS-18121	CS-18128	CS-18131	CS-18140
waseem4100057@cloud.neduet.edu.pk	asad4108018@cloud.neduet.edu.pk	murtaza4103292@cloud.neduet.edu.pk	nisar4106988@cloud.neduet.edu.pk

Statement of Contributions

Project Load was equally divided among the four group members.

The first Phase was Data Collection, Cleaning, and Pre Processing which was done by Aiman Nisar. She also contributed to Data Augmentation. Data Augmentation was to make our data compatible with the model. After Data Augmentation, Feature Extraction was yet another milestone to be achieved. Sukaina Asad was responsible for Feature Extraction alongside Data Augmentation. Model Development, its training, and testing were crucial tasks. Shehroz Waseem worked on Data Preparation, Model Research, and Development and analyzed the results very efficiently. Last but not the least, Abaad Murtaza also worked on Model Research and Development alongside Hyper Parameter Tuning and in Result Analysis.

After model development, our next task was web application development. For frontend development, Aiman Nisar played her part. API Development was another significant task that was carried out by Abaad Murtaza and Shehroz Waseem. Once the API was developed its integration into the web application was performed by Aiman and Shehroz. The project also maintains a database. All the tasks related to database management were carried out by Sukaina Asad. Final testing of the system was performed by Abaad Murtaza and Sukaina Asad. All group members were equally involved in the documentation.

Without each one’s contribution, this project would not have been a success.

Executive Summary

Human behavior has been assessed using Sentiment Analysis over the years. This is accomplished through a variety of methods. Text data, such as lexical features taken from the text, facial expressions (visual features), and audio data, can be used to track human emotions and behavior (acoustic features like pitch, tone, jitter, etc.). To understand a human's mindset/mood through a conversation, a computer must first understand who is speaking and what is being said, therefore we create a system that performs vocal sentiment analysis on speech data collected from users. The suggested methodology's goal is to extract sentiments from audio data to the time-consuming operation of judging human emotions. We want to make a simple mobile app that incorporates this model and provides us with detailed information on how people act in various situations. To train our model, we created our own dataset in Urdu Language. The data was pre-processed, and features were extracted to obtain a model with high accuracy. From our voice data, we extracted Mel Frequency Cepstral Coefficients (MFCC) characteristics. In addition, we employed data augmentation to make our data's shape suitable with the model. To extract sentiments from audio data, a convolutional neural network is employed in our project. We were inspired to make this project because the human voice can convey a wide spectrum of emotions, from joy to sorrow, misery to happiness, spontaneity to rigidity, a delicacy to harshness, laughter to tears etc.

Acknowledgments

Accomplishing a task is meaningless without appreciating the people who made it possible and whose constant support and assistance acted as an inspiration throughout the project. We'd want to take this opportunity to sincerely thank everyone who helped us.

This project has required a lot of effort from us. However, without the kind support and assistance of Dr. Urooj Ainuddin, our internal advisor, Department of Computer & Information Systems Engineering, it would not have been achievable. We would like to thank her for investing her time and efforts to help us achieve the goal. We all want to express gratitude to her.

Table of Contents

Contents

List of Figures

List of Tables

List of Abbreviations

MFCC	Mel Frequency Cepstral Coefficients
CNN	Convolutional Neural Network
EPBP	Economic Payback Period
HMM	Hidden Markov Model
FER	Facial Emotion Recognition
DCT	Discrete Cosine Transform
ANN	Artificial Neural Network
MLP	Multi-Layer Perceptron
LTU	Linear Threshold Unit
MSE	Mean Squared Error
EDA	Exploratory Data Analysis

United Nations Sustainable Development Goals

The Sustainable Development Goals (SDGs) are the blueprint to achieve a better and more sustainable future for all. They address the global challenges we face, including poverty, inequality, climate change, environmental degradation, peace and justice. Thereis a total of 17 SDGs as mentioned below. Check the appropriate SDGs related to the project.

□ No Poverty

□ Zero Hunger

□ Good Health and Well being

□ Quality Education

✓ Gender Equality

□ Clean Water and Sanitation

□ Affordable and Clean Energy

✓ Decent Work and Economic Growth

✓ Industry, Innovation and Infrastructure

□ Reduced Inequalities

□ Sustainable Cities and Communities

□ Responsible Consumption and Production

□ Climate Action

□ Life Below Water

□ Life on Land

✓ Peace and Justice and Strong Institutions

✓ Partnerships to Achieve the Goals

Similarity Index Report

Following students have compiled the final year report on the topic given below for partial fulfillment of the requirement for a Bachelor’s degree in Computer Systems Engineering.

Project Title **Behavioral Monitoring Using Vocal Sentiment Analysis**

S.No	Student Name	Seat Number
1	Shehroz Waseem	CS-18121
2	Sukaina Asad	CS-18128
3	Abaad Murtaza	CS-18131
4	Aiman Nisar	CS-18140

This is to certify that Plagiarism test was conducted on complete report,and overall similarity index was found to be less than 20%, with maximum 5% fromsingle source, as required.

Signature and Date

....
Dr.Urooj Ainuddin

Chapter 1
Introduction

1.1 Background Information

Speech signals are among the utmost innate and essential forms of human communication, and they have the advantage of being easily measured in real-time. They include the speaker's emotions as well as implicit paralinguistic information and linguistic content. It's crucial to integrateappropriate audio characteristics in speech emotion identification since feature selection impacts categorization performance.

Sentiment Analysis has been performed over the years to assess the behavior of individuals.There are several techniques used to achieve this. Human emotions or behavior can be monitored using the text data i.e. their lexical features extracted from the text, facial expressions (visual features), and vocal data (acoustic features like pitch, tone, jitter, etc.) One or more of these features can be used to solve the problem of voice emotionrecognition. If one wants to predict emotions from real-time audio, following the lexical features would necessitate a transcript of the speech, which would necessitate an additionalstep of text extraction from speech. Similarly, assessing visual aspects would necessitate access to video of the interactions, which may or may not be possible. The acoustic featureanalysis can be done in real-time while the discussion is going on because we only requirethe audio data to complete our assignment. As a result, we decided to focus on the acousticaspects of this project.

The human voice can communicate an ample number of feelings, from joy to misery, agony to contented, spontaneity to cruelty, a delicacy to brutality, wellbeing to sickness, giggling to crying, and justifies speech-based emotion recognition algorithms.

According to scientific evidence, all human's emotions cause psychological and physiological changes that affect the voice. Emotional speech processing technologies usecomputer analysis of speech features to determine the user's emotional state. Pattern recognition algorithms can be used to evaluate vocal variables and prosodic aspects suchas pitch variations and speech rate. Features extracted from human vocal data hold immensesignificance for sentiment analysis which gives insights into their emotional health.

Sentiment analysis is utilized in a variety of applications; in this case, we're using it to figure out what people are feeling based on their interactions with one another. To comprehend the mindset/mood of humans through a conversation, a computer must first know who is speaking and what is being said, so we develop a system that carries out a vocal sentiment analysis on speech data acquired from users.

1.2 Significance and Motivation

Emotion identification in human speech has recently gotten a lot of interest as a way to create a more intuitive human-computer connection. When considering alternative strategies for recognizing human emotions, speech is frequently the first thing that comes to mind. These applications are beneficial in situations where humans engage with automated technology, such as call centers and interactive movies.

With the advancement in technology, for instance of personal assistants like Google Assistant and Amazon Alexa, it's more critical than any other time to answer requests in a meaningful and significant manner in light of the client's state of mind. Product reviews are another area where human speech classification is useful. A huge amount of online speech reviews that would have previously required manual labeling maynow be examined, which is beneficial to product makers and sales.

Earlier studies aimed at automatic emotion recognition in speech relied heavily on prosody variables like pitch, force, and length, these variables are not difficult to manage but they just provide analysis on the arising aspect of emotion. However, there is emerging evidencethat using solely prosody elements in speech, it is difficult to discern between emotions likeanger and joy that have nearly identical levels in the arousal component of emotion.

In recent studies, the role of voice standard aspects in the valence part of feeling has been given importance. When disparate expressions occurred because of different phonation types, for example, breathy, creaky, harsh, etc. The voice quality features altered. Here we are going to extract both fundamental prosody elements and various voice quality features, and consolidate them to discover emotions like annoyance, misery, and happiness in this study.

1.3 Aims and Objectives

The objective of this proposed methodology is to extract sentiments from audio data to automate the time-consuming task of assessing human emotions.

We aim to create a simple mobile app that uses this model and gives us comprehensive information on how people behave in different situations.

1.4 Methodology

Initially we opted for vocal sentiment analysis in English language using datasets that were readily available over the internet and contained multiple audio files in .wav format. These four datasets are listed below:

i. Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)
ii. Surrey Audio-Visual Expressed Emotion (SAVEE)
iii. Toronto Emotional Speech Set (TESS)
iv. Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D)

These datasets included audio files with a variety of accents and voices, including both male and female voices. We trained and tested our model on the above-mentioned datasets, which in return provided us with considerable accuracies.

We then shifted our approach of sentiment analysis to **Urdu language**. Whose datasets were not easily available. Furthermore, we created the dataset in Urdu from scratch. The dataset included the audio files of two males and two females, speaking different utterances in anger and normal tone. Each individual recorded a total of 250 audios, which includes both emotions (125 audios for each emotion). Currently our dataset consists of 1000 rows.

1.5 Report Outline

The first chapter of this report is Introduction. It covers all the background aspects includingwhat motivated us for this project, our aims and objectives and beneficiaries of this project.The second Chapter is the Literature Review which provides a detailed description of the previous works related to vocal sentiment analysis. A comparison between all the possible approaches for accomplishing this task. The third chapter is Methodology which only focuses the methods, tools and techniques that we incorporated in our project.

Chapter 2

Software Requirements Specifications

2.1 Introduction

2.1.1 Purpose

The purpose of this application is to predict human emotions by analyzing their vocal data. This project aims to assist the supervisors when they want to assess the behavior of their employees with the customers. Primarily this application helps the supervisors of a call center where they want to check the behavior of their call agents that whether they are communicating in a considerate tone or not.

2.1.2 Document Conventions

This section of the report follows the format provided by IEEE for Software Requirements Specification. Significant details are made bold.

2.1.3 Targeted Audience

The project advisor, present and prospective developers, and the department of Computer & Information Systems Engineering are the target audiences for this document. The document offers information on the design choices made during the project's development. Later chapters contain study material on the implementation specifics.

2.1.4 Project Scope

The primary motivation behind this project is to automate the time-consuming task of supervisors. The model will be made available for further improvements to develop a complete Speech Emotion Recognition (SER) system that can be deployed in organizations where monitoring human behavior via their voice is necessary. The entire system can then be used as a stand-alone product or deployed online for increased accuracy while simultaneously gathering more data. The project's scope is not restricted to Pakistanis and can be scaled as necessary.

2.2 Overall Description

2.2.1 Project Perspective

The overall project workflow is divided into the following stages:

1. Dataset Preparation
2. Data Augmentation
3. Feature Extraction
4. Model Training and Testing
5. Web Application Development
6. API Integration

In the first stage, a voice dataset is prepared with equal proportions of male and female audios consisting of angry emotion and neutral emotion audios. In order to generate more samples and increase the dataset size, multiple data augmentation techniques are applied lime stretching and time shifting. For Feature Extraction, Mel Frequency Cepstral Coefficients are extracted from the vocal data. A Convolution Neural Network is trained. The model was accurate enough to predict angry emotions provided both English and Urdu audios.

2.2.2 Project Functionalities

The final Project has the following functionalities integrated:

1. Firstly a user can record an audio which is analyzed and the percentage of anger is predicted.
2. Analysis can also be performed on pre-recorded audios.
3. Supervisor can access the database where audios of the agents are saved. After accessing the database, supervisor can analyze the agent’s audios and monitor their behavior.

2.2.3 Operating Environment

Final Project is a Web Application. It’s developed on the FLASK Framework. For web development React is used and for database and authentication functionalities Firebase is incorporated.

2.2.4 User Documentation

Users of the web application have no trouble navigating its user interface, which is both straightforward and interactive. An appendix with user instructions and an API reference will be provided.

2.2.5 Assumptions and Dependencies

Adherence to these limits is a key underlying assumption because the project is developed with them in mind. The following are the software dependencies:

1. Flask (2.0 or above)
2. TensorFlow (2.0)
3. React (18.0 or above)
4. Firebase (9.0 or above)
5. Python 3.7

Accurate working of the final project will be solely dependent on the fulfillment of these dependencies.

2.3 External Interface Requirements

2.3.1 User Interfaces

The user interface is understated and stylish. Users will be able to select files from their PCs or record their own audio using a microphone using the UI. The third option is for the supervisor, who can log in and then access the database with an admin panel view to see the agents who report to him and their audio files. All of the information is dynamically pulled from the Firebase database.

2.3.2 Hardware Interfaces

The project is a web application therefore no hardware dependencies and constraints apply to it.

2.3.3 Software Interfaces

The software interfaces include a web-based application for recording and submitting audio files as well as displaying the model's forecast as a pie chart and, after a successful prediction, an audio player with the user's input audio will be displayed under the result.

2.4 System Features

2.4.1 Microphone Recording

2.4.1.1 Description and Priority

To obtain the recording from the user, the application needs authorization to utilize the device's microphone. This recorded audio can be used subsequently to determine file emotions by the user.

2.4.1.2 Response Sequences

The user must click the application's microphone button before the recording can begin. The audio can be paused by the user, and once it is finished, it can be uploaded for prediction.

2.4.1.3 Functional Requirements

A button for the microphone that records audio for the user. The user will make sure that the device gives him permission to utilize the microphone.

2.4.2 Audio File Submission

2.4.2.1 Description and Priority

The application can upload media files that the user chooses for emotion recognition.

2.4.2.2 Response Sequence

The user will select the .wav extension file from his computer and submit it to the application by selecting the submit button.

2.4.2.3 Functional Requirements

A module to allow users to submit audio files and then send them to the API end point for predictions.

2.4.3 Login Functionality

2.4.3.1 Description and Priority

A supervisor can log in to the application using the login screen. The supervisor and the agents working for him will be presented after a successful login to the authentication system, which was constructed using Firebase. The supervisor can listen to an agent's audio recordings and assess them by selecting them. .

2.4.3.2 Response Sequence

The user will select the .wav extension file from his computer and submit it to the application by selecting the submit button.

2.4.3.3 Functional Requirements

A module to allow users to submit audio files and store them in a database for generating transcription and further use. The user can also delete the file from the database.

2.4.4 Display Transcription

2.4.4.1 Description and Priority

After analyzing the audio speech, the application must present the predictions.

2.4.2.2 Response Sequence

The audio speech submitted by the user will be passed on to the trained model to identify the detected emotions and the computed result will be displayed.

2.4.2.3 Functional Requirements

The results must be stored and displayed on the screen of the application in a form of pie chart after it has been generated.

2.5 Non Functional Requirements

2.5.1 Performance Requirements

The application must be real-time and have a quick turnaround for output responses.

2.5.2 Safety Requirements

When general safety precautions are followed, the product is not known to harm anyone. No system warranty will be offered by the developers or their team. In the event that the system is tampered with or misused, the project team will not be held liable.

2.5.3 Security Requirements

The application won't gather any private data and will store submitted or recorded audio recordings anonymously. The system will use the provided data for additional training, and no third parties will ever see the provided data.

2.5.4 Software Quality Attributes

For the purpose of generating a more accurate outcome, we want to set a high accuracy following standard. The product will be simple to scale and maintain. Regarding performance in a noisy setting, robustness is a problem.

2.5.5 Other Attributes

Other requirements include time constraint. The project is to be developed in 10 months maximum with deliverables after 5 months. The project shall have complete documentation included with it so that any person who wishes to further enhance the project can do so without much problems

Chapter 3

Literature Review

3.1 Introduction

It has been the goal of researchers to create systems that can behave like humans since the advent of computer science. Numerous investigations on comprehending the human brain and creating machines that mimic human intelligence have been carried out during the past few decades. The intricate structure of the human brain has long served as a model for researchers studying AI. The human brain's neural networks are incredibly adept at picking up complex abstract ideas from simpler ones that are interpreted by the sensory periphery. Learning new languages, comprehending speech signals, and identifying faces are just a few of the multiple examples that demonstrate the incredible capability of the human brain to grasp complex ideas.

Making the system understand speech—a process that seems simple to humans but is incredibly challenging for machines—was one of the main difficulties in acknowledging the ideal of developing a machine that can function like a human. AI's main objective is to develop intelligent systems that can think rationally and act in ways comparable to human intellect and performance.

In the year 2000, researchers who were interested in human-computer interaction found that people interconnect with computers thinking they were real people, responding to feedback both positive and negative in exactly the same way they do from humans. High-performance personal computers are becoming increasingly common as theinformation society advances technologically. As a result, computer-human interactions areincreasingly becoming bidirectional. Due to this, a deeper understanding of human emotions is necessary. For a machine's intelligence, emotion-sensing is equally as importantas data-driven reasoning. The user experience would be enhanced overall, and machine performance would increase if computers possessed knowledge and understanding of emotions.

Even though research has been done in this field, work on voice-based sentiment analysis has shown less progress for two key reasons. The first is a lack of interest in this field, and the second, and most crucial, is a paucity of data that is essential for large-scale systems.

From the beginning to the present, the following sections will offer a quick outline of the evolution of Emotion Recognition Systems.

3.2 Emotions and Taxonomy of Emotions

Emotions are considered complex psychological states or the reactions that humans have in response to certain events or situations. These are the most reliable indicators generated by the human brain that helps in keeping track of how things are going in one’s life. Emotions can be positive or else negative and have a great impact as they play a significant role in human life.

In psychology, there are several theories about the classifications of emotions. In some theories, emotions are ordered, based on specific criterion, whereas in others, all emotions are viewed as equally significant. However, despite their conceptual differences, all of these theories emphasize the complexity of human emotional behavior.

Robert Plutchik, an American psychologist, is responsible for one of the most well-known classifications of emotions. There are eight primary emotions in his taxonomy (Table 1), which are divided into four sets of opposites. They all appear in varying levels of intensity, and when they combine and are known as secondary emotions.

joy	sadness
trust	disgust
fear	anger
anticipation	surprised

Table 1: The Fundamental Emotions

Plutchik graphically portrayed this intricate structure in a very intriguing chart that became known as 'Plutchik's wheel'. As can be seen in this diagram (Figure 1), basic emotions are linked to a wide range of emotions in varying degrees of intensity, with the emotions represented in the middle region of the wheel being the most extreme manifestations of these fundamental emotions.

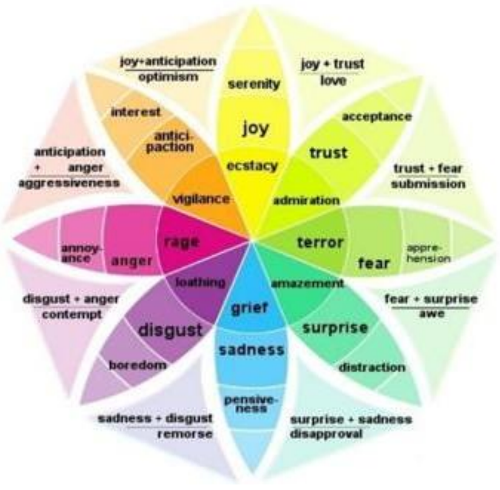


Figure 1 Plutchik's Wheel of Emotions

However, given the complexity of human emotions, there is general agreement that some of them are more relevant for computer communication and interaction. A simpler version of Plutchik's wheel (Figure 2) is generated and is considered as standard in emotion recognition.

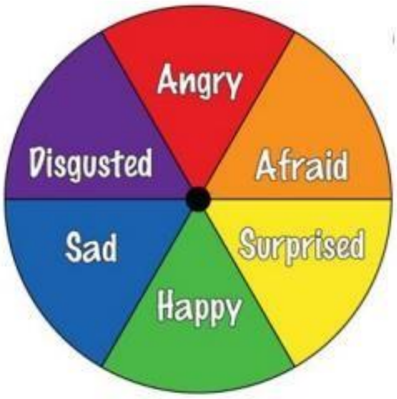


Figure 2 Plutchik's Wheel Simplified

Human emotions may typically be identified via face recognition, voice, non-verbal cues and actions, and bio-signals (physiological characteristics like heartbeat, skin conductivity, temperature, and others).

3.3 Sentiment Analysis

Sentiment Analysis is the translation and characterization of emotions (positive, negative, or neutral) within a set of data. It is possible to do so using content, sound, and video assessment methods. Sentiments or emotions are an important part of people's lives because they influence how they perceive or understand things. During the last 20 years, numerous techniques have been devised to facilitate sentiment analysis, ranging from manual methods such as questionnaires created by psychologists to computer-based methods.

After analyzing the responses of the human brain, there are multiple distinct kinds of sentiments that are expressed. Depending on the mood, the mind reacts, which is expressed in the action manner and it is a physiologic state related to nerves. If the emotions were intense, the person could react in a variety of ways. The sentiment analyzer attempts to analyze emotion based on text, voice, and facial expressions in this environment.

Sentiment analysis is widely used to monitor an individual’s behavior on social media and other platforms because it helps us to get a sense of how the general population feels about various issues.

Sentiment analysis is highly effective and has a variety of applications. Businesses all across the world are getting more and more interested in the ability to gain insights from social data. It has been demonstrated that shifts in social media sentiment coincide with movements in the stock market. Sentiment analysis also helps to strategize and prepare for the future when you can easily see the mood behind anything from forum postings to news items.

3.3.1 Sentiment Analysis based on Text Data

To extract critical interpretation, textual analysis was developed as a substitute to topic detection. Using a person's tweets or text from a certain document to determine if the individual is feeling good, negative, neutral, or more. This in addition helps to classify one’s opinions, sentiments in keeping with the depth of the speech written. Sentiment analysis can be termed as a procedure study that how a person’s opinions, attitudes, emotions, and views are expressed in language. This sort of analysis is termed text analysis. It plays an important role during this sort of prediction of emotions supported sentiment analysis. In Natural Language Processing (NLP), there are ways to do sentiment analysis covering both supervised and unsupervised methods, as well as the field's future goals and constraints. Using datasets that are annotated and automated learning, supervised sentiment analysis aims to develop predictive models for emotion.

The first attempt to supervise sentiment analysis involves categorizing texts as either subjective or objective using labels that don't reveal the direction it's pointing at but only the assessed content. Another method is a Keyword Spotting methodology, which uses a text file or document as input and generates output as an emotion class. The text data is translated into tokens, and the emotion words are then deduced and anticipated from these tokens in accordance, the model then predicts the emotion class in accordance with the results of this analysis of the strength of emotion words, which also includes checking the phrase for negation. There are several methods for performing sentiment analysis on text-based data, including the Lexical Affinity method, which assigns arbitrary words a probabilistic affinity for a specific emotion, learning-based methods, which uses trained classifiers to identify emotions, and SVM (Support Vector Machine) to help determine which emotion category the input text should fall under.

The examination of sentiment in relation to stock trading was one of the initial motivations for supervised approaches, which resulted in the development of a voting classifier with various techniques for sentiment retrieval from financial message boards. The majority of supervised sentiment analysis algorithms are developed using data from one particular area or communication context, such media or networking sites for instance socializing platforms. The most successful use of sentiment analysis at the moment is the categorization of the contrariety of tweets, which are concise public communications on Twitter.

The usage of emotion keywords is an easy way to find associated emotions with textual sentiment analysis. However, the definitions for these keywords may be many and unclear because most words can have several meanings based on usage and context. Recent developments have made sentiment analysis applications possible, even if the problem of text sentiment analysis is still distant from being resolved. The efficiency of the tools, however, varies substantially depending on the factors, formalism, and type of text getting studied, thus acknowledging the fact that there isn't currently a global explanation for the solution is necessary. Domain expertise is still a necessary component in the use of sentiment analysis, only an appropriate tools selection can ensure an accurate and legitimate method to assess sentiment from text

3.3.2 Sentiment Analysis based on Facial Expressions

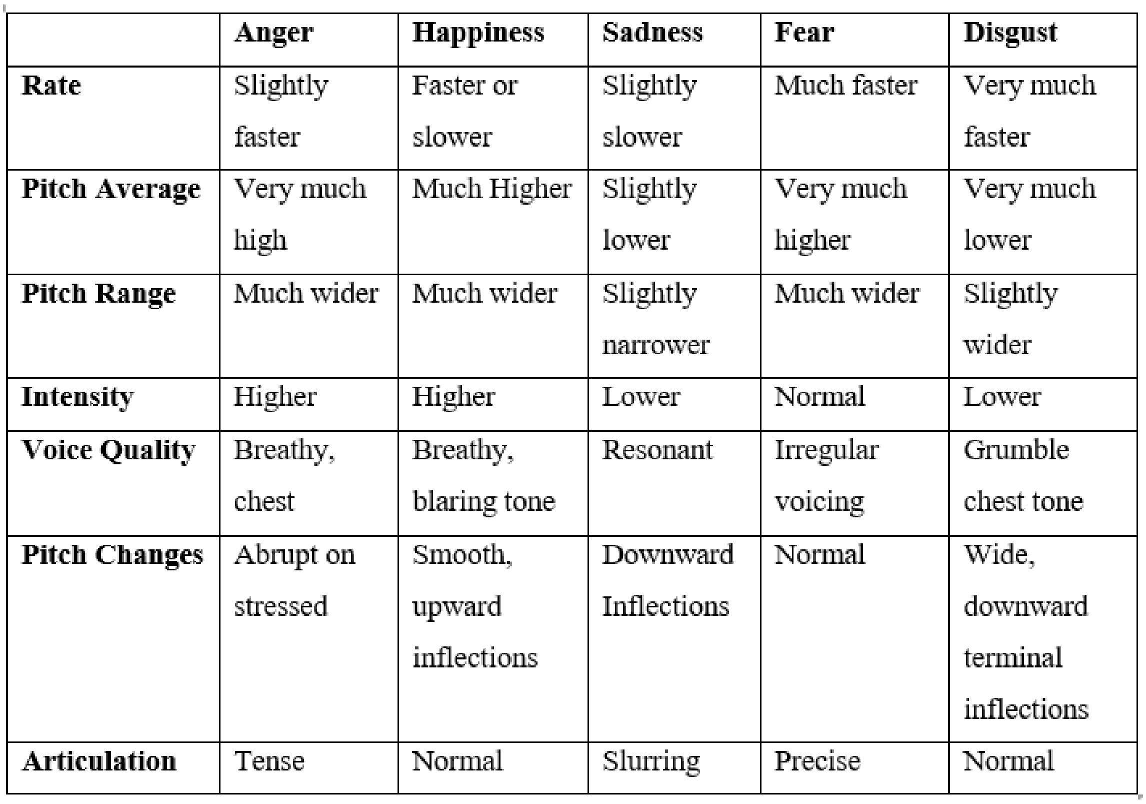
While there has been wide research in vision-based emotion recognition for some time, computer vision sentiment analysis is a relatively more recent field of study. A system called Facial Emotion Recognition (FER) analyses expressions in still images as well as videos to provide details about the individual's emotional state.

Recent advances in face detection, face tracking, and face recognition systems have sparked interest in facial expression analysis. Model-based methods, holistic methods, local methods, and motion extraction methods are some of the methodologies identified for facial expression analysis. They also took into account several obstacles in expression identification, such as stance, lighting, and occlusions. Based on the algorithm, emotions such as anger, sadness, surprise, joy and disgust, can be comprehended from facial expressions. Many algorithms are proposed for FER one of that is the HAAR classifier It uses features to encode the average image intensity difference between various sections of the image, which are black and white linked rectangles whose value is equal to the sum of the differences between the pixels in the black and white areas after that feature extraction is done from the collected data, In FER because of its great discriminative capacity, Gabor features are particularly popular. To increase the system's performance (problem solving or real-time), we execute Dimensionality Reduction using PCA on the high dimensional data generated via feature extraction. Facial expressions might vary somewhat across individuals, blend several emotions that occur at the same time (such as fright and rage, pleased and sadness), or fail to reflect any feeling at all, therefore analyzing emotions solely on facial expressions may not be accurate. Furthermore, even when emotions are accurately recognized, the findings may be used to make incorrect conclusions about someone, since FER fails to reveal the cause of emotions, which might relate to a memory of a recent or past event.

3.3.3 Vocal Sentiment Analysis

Speech is a signal information-rich signal, which includes both paralinguistic and linguistic information. A prime example of paralinguistic information is emotion, it is primarily expressed through speech. Developing interpretable machines that can understand non-linguistic information, for instance emotions, improve communication between humans and machines by making it more natural and understandable.

For years, researchers have been studying emotion recognition. Early framework of research on emotion recognition focused on identifying emotions from facial expressions and biological data like heartbeats. The conventional method of solving this issue was predicated on the correlation between acoustic characteristics and emotion. Acoustic and prosodic speech signal correlate, such as speaking rate, pronunciation, formant energy, tone, intensity, time, and spectral feature are used to encode emotion. Table 2 below provides an explanation of the key parameters to monitor in audio tapes during feature extraction process.



	Anger	**Happiness**	**Sadness**	**Fear**	**Disgust**
Rate	Slightly faster	Faster or slower	Slightly slower	Much faster	Very much faster
Pitch Average	Very much high	Much Higher	Slightly lower	Very much higher	Very much lower
Pitch Range	Much wider	Much wider	Slightly narrower	Much wider	Slightly wider
Intensity	Higher	Higher	Lower	Normal	Lower
Voice Quality	Breathy, chest	Breathy, blaring tone	Resonant	Irregular voicing	Grumble chest tone
Pitch Changes	Abrupt on stressed	Smooth, upward inflections	Downward Inflections	Normal	Wide, downward terminal inflections
Articulation	Tense	Normal	Slurring	Precise	Normal

Table 2 Emotions and Speech Parameters

A number of machine learning techniques have been studied to characterize emotions based on audio that correlates in spoken utterances. Artificial neural networks, nearest neighborhood classifiers, linear discriminant classifiers, Support Vector Machines (SVM) and Hidden Markov Models (HMM), are some types of techniques used to categorize emotions based on their relevant properties.

Speech analysis is a useful tool for determining affective state. According to some studies, the accuracy reported on average is 70 to 80 percent, this is inferior to other emotion recognition systems that evaluate facial expressions but superior to the average human accuracy (about 60%). Nevertheless, since some speech qualities are impacted by semantics or culture while others are not, voice analysis is an important part of research.

2.1 Mel Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficient is a popular and effective signal processing technique. When it comes to 1D signals, one of the most popular and efficient techniques for feature extraction is the use of MFCCs. We need to transform input voice audio intoa 1D signal, and then extract features. It produces a cepstral coefficient, which is then used as a feature vector in the classification procedure.

The frequencies and bandwidths that humans can perceive are the core basis of MFCC methodology. It is incapable of detecting frequencies greater than 1 kHz. MFCC accepts a voice or an audio signal as an input and processes it in a number of steps as mentioned below:

Step 01: Pre-Emphasis Filtering:

In order to have a steep roll-off in the high-frequency range, the vocal spectrum needs to be balanced initially. Low-frequency signals are given some more signal energy to achieve this balancing. Using equation 2.1 below, pre-emphasis filter can be calculated:

$$\boldsymbol{S}'(\boldsymbol{n}) = \boldsymbol{S}(\boldsymbol{n}) - a * \boldsymbol{S}(\boldsymbol{n} - \mathbf{1}) \qquad \textbf{(2.1)}$$

Here α is referred to as the pre-emphasis coefficient whose value lies in the range of 0.9-1.

Step 02: Frame Blocking

When the signal has passed the pre-emphasis step it is now ready to split into multiple frames. Speech must be studied over a sufficiently short period due to stable acoustic characteristics. As a result, speech analysis has to be performed always on segments of short length when the speech signal is presumed to be steady.

Step 03: Windowing

The windowing technique has to be applied to every frame to reduce signal discontinuity generated by the frame blocking process at both ends of each frame. The Hamming window is utilized, and the frame from the preceding operation is multiplied by it. The formula can be used to compute hamming Windowing:

$$\boldsymbol{w(n) = 0.54 - 0.46\cos\frac{2\pi n}{N-1}} \qquad \textbf{(2.2)}$$

Here n lies in the range of 0 till N-1.

Step 04: Fast Fourier Transform

Using Fast Fourier Transform every frame needs to be transformed from time domain to frequency domain. The frames are more comprehensible and simpler to understand if they are in the frequency domain. Equation 2.3 support this statement:

$$\boldsymbol{Y(w) = FFT[h(t) * x(t) = H(w) * X(w)} \qquad \textbf{(2.3)}$$

Step 05: Mel Frequency Wrapping

The voice signal comprises of different tones in different frequencies, that are calculated in Hertz (Hz). In the meanwhile, "Mel" units are used to express subjective pitch. The "Mel" frequency scale is linear for frequencies below 1 kHz and logarithmic for frequencies above 1 kHz. Equation 2.4 mentioned below is utilized to determine the Mel scale for a specific frequency in Hz.

$$\boldsymbol{fmel = 2595\log_{10}(1 + \frac{f}{700})} \qquad \textbf{(2.4)}$$

In this equation, f is a linear frequency.

Step 06: Discrete Cosine Transform (DCT)

DCT is now used to convert the log Mel spectrum back into the time domain. This step is responsible for producing the Mel Frequency Cepstral Coefficient. We refer to the set of coefficients as the acoustic vector. Equation 2.5 can be used to compute the result of this conversion:

$$\mathbf{c(n)} = \sum_{\boldsymbol{k-1}}^{\boldsymbol{k}} \mathbf{\log Sk} * \ \mathbf{\cos}(\boldsymbol{n}\,(\boldsymbol{k} - \frac{\mathbf{1}}{\mathbf{2}})\frac{\boldsymbol{\pi}}{\boldsymbol{K}}) \qquad \textbf{(2.5)}$$

Here n= 1, 2… K, whereas output is S_k at index k, and the expected coefficient is K.

2.2 Deep Artificial Neural Network

A subfield of artificial intelligence and computer science is machine learning. It mainly focuses on creating algorithms that can automatically learn new abilities and knowledge through repetition, like examining training data. Machine learning algorithms ought to be able to apply the knowledge they learned from such insights or observations to unseen data. As a result, many learning approaches such as supervised, and unsupervised learning are used.

In supervised learning, the training data include the predictable output, referred to as labels; that is, each observation is labeled. To correctly determine the label of each training/test instance is the goal of learning. One use of supervised learning is classification, in which the ML algorithm learns to divide input into more than two or just two groups. Based on the observed training data, it learns the discriminating traits (features) or attributes across distinct classes or categories. The classification of additional test data is subsequently done using these features. ANNs are trained using supervised learning algorithms, even though some ANNs can be taught via unsupervised learning. Biological neural networks have served as an inspiration for ANNs. In other words, they are composed of intricately linked units known as neurons. The core elements of deep learning, a potent contemporary machine learning method, are ANNs. ANNs with layer upon layer and neurons make up deep learning models. Classification is one machine learning application where deep learning models have succeeded. The underlying principle of deep learning models' effectiveness is their ability to derive complicated characteristics from sparse information.

2.2.1 Multi-Layer perceptron Network

The network design of Artificial Neural Networks (ANNs) is one of their distinguishing features. It establishes the connections between neurons. A well-known ANN architecture made up of neurons called Linear Threshold Units (LTUs) is the Multi-Layer Perceptron (MLP). A linearthreshold unit is shown in Figure 3:

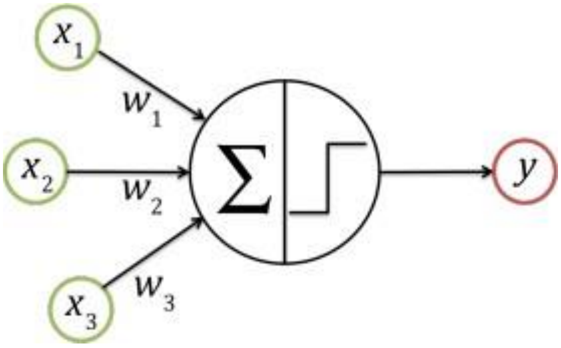


Figure 3 Linear Threshold Unit: It processes the inputs by computing the linear combination of inputs and applying a step function

Through the formula:

$$\mathbf{z = w1x1 + w2x2 + w3x + b},$$

Here b is the bias factor, LTU calculates the inputs after accepting weighted inputs from multiple neurons. With the help of a step function an output;

$$\mathbf{y = f\ (z)}$$

Here f(z) is the generated step function. If the weighted sum is higher than a threshold value, the bias value may have an impact on the output result that is generated.

One input layer, one or more hidden LTU layers, and one output layer make up an MLP in most cases. Information is passed through layers, which means to the output layer from the input layer. They are specifically referred to as "Feed Forward ANNs" for this purpose. The contents of a training audio data are represented by the input layer in many aspects. This could be the amplitude at various sampling points.

2.2.2 Convolutional Neural Network

CNNs or Convolutional Neural Networks have greater achievements in a variety of applications, including object detection, handwriting, face, and speech recognition, as well as natural language processing. “Convolution" refers to the mathematical operation that is being performed on two functions that results in a third function which describes how the shapes of the first and second are changed by the other. Convolutional, pooling, and fully connected layers are the three main building blocks of CNNs, and they together make up each of these three layers:

i. Convolutional Layer

The output of convolutional layers in CNNs is computed using convolution rather than multiplication. Not all convolutional layer neurons are connected to those in the layers preceding them.

The neurons are conditioned to react to stimuli unique to a certain region. As a result, convolutional neural networks have sparse connection and parameter sharing. It significantly decreases deep neural networks' parameter count. Figure 4 depicts the convolution of a kernel, which is a 2 x 2 matrix, with a one-channel 3 x 3 image. The output has a volume of 2 x 2 x 1. The output size, where **nh** is the input height, **nw** is the input width, and **nf** is the number of kernels, is typically **(nh f + 1) (nw f + 1) x nf.** The depth of the kernel is determined by the depth of the input.

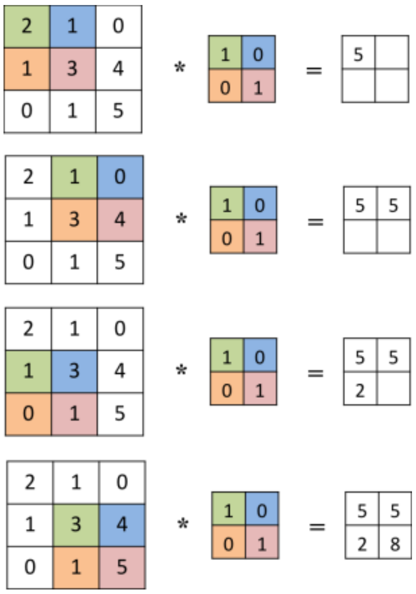


Figure 4 The convolution of a 3x3 image by a 2x2 kernel with a stride of 1

Convolutional layer local filtering enables the development of numerous feature maps and the detection of numerous features that can be of interest. The deep layers construct a high-level representation of the inputs using these feature maps.

ii. Pooling Layer

The second essential element is a pooling layer. This layer's purpose is to lessen the outcomes' sensitivity to small alterations in the inputs. When exact spatial features are not required, the pooling layer helps speed up the extraction of the desired characteristics. In order to avoid over fitting, pooling also assists in the reduction of dimensions and parameters. The subsamples are extracted from outputs in Pooling.

Similar to convolutional layers, pooling layers also use aggregation functions to summarize the output values of the neurons within the pooling kernel, such as average, weighted average, or maximum. Before creating a pooling layer in CNNs, we must first choose the pooling kernel's size, the number of steps to be shifted, and the padding amount. With a shift of 1 pixel (stride equals to 1) across the matrix and a pooling kernel size of 2 x 2, maximum pooling over a 3 x 3 matrix is shown in Figure 5.

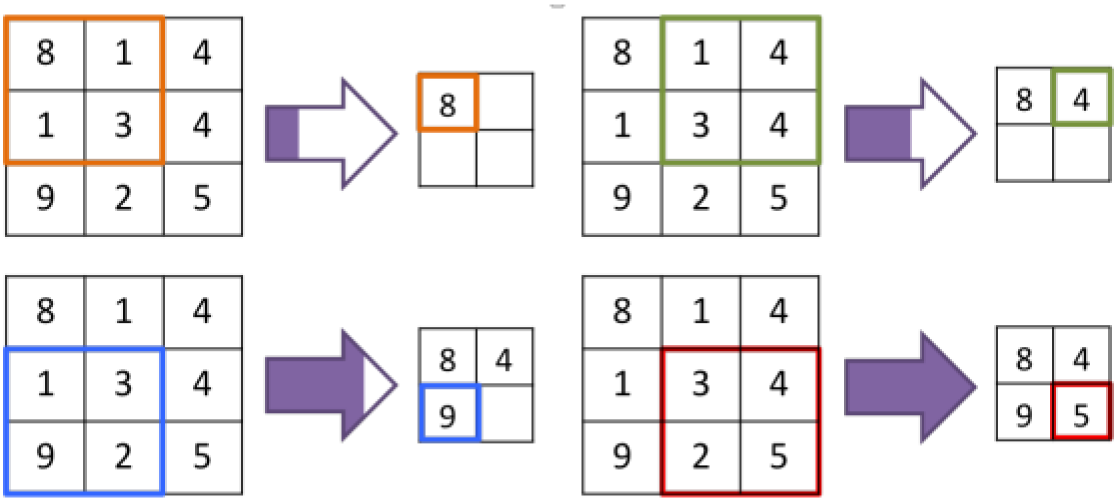


Figure 5 Pooling of a 3x3 image using a 2x2 kernel

iii. Fully Connected Layer

The last component of CNNs is a fully linked layer, which is essentially a standard MLP. A traditional CNN has several convolutional layers, each of which is preceded by a pooling layer and then these fully connected layers. By further processing the features, this component can either provide an abstraction of the inputs or classify them based on the traits (characteristics) that the preceding layers have retrieved.

iv. Softmax Unit

Typically, a softmax unit is the output of the entirely connected layer. The probability distribution of k classes is represented by a softmax unit, which uses the softmax function (normalized exponential).

$$\boldsymbol{\sigma\,(z) = \frac{1}{e^{-z}} = \frac{e^{z}}{1+e^{z}}} \qquad \textbf{(2.6)}$$

The sigmoid function displays the probability distribution of two separate classes, is generalized by the softmax function. The softmax function is shown in equation 2.7

$$\boldsymbol{softmax(\sim z)i = \frac{e^{zi}}{\sum_{j=1}^{k} e^{zi}}} \qquad \textbf{(2.7)}$$

The probability that the initial input instance belongs to class i is known as Softmax ($Z^{\rightarrow}$)i where zi is the ith element of the vector $Z^{\rightarrow}$, where Z is the result of the k-way softmax unit.

v. Rectified Linear Unit

Artificial neural networks have relied on activation functions for years (ANNs). That is, they include nonlinearity in ANNs, making them a powerful tool for learning complex models.

$$\boldsymbol{g(z) = \sigma(z) = \frac{1}{1+e^{-z}}} \qquad \textbf{(2.8)}$$

Two prominent activation functions, especially in classic ANNs, are the sigmoid function and the hyperbolic tangent function.

$$\boldsymbol{g(z) = \tanh(z) = 2\sigma(2z) - 1} \qquad \textbf{(2.9)}$$

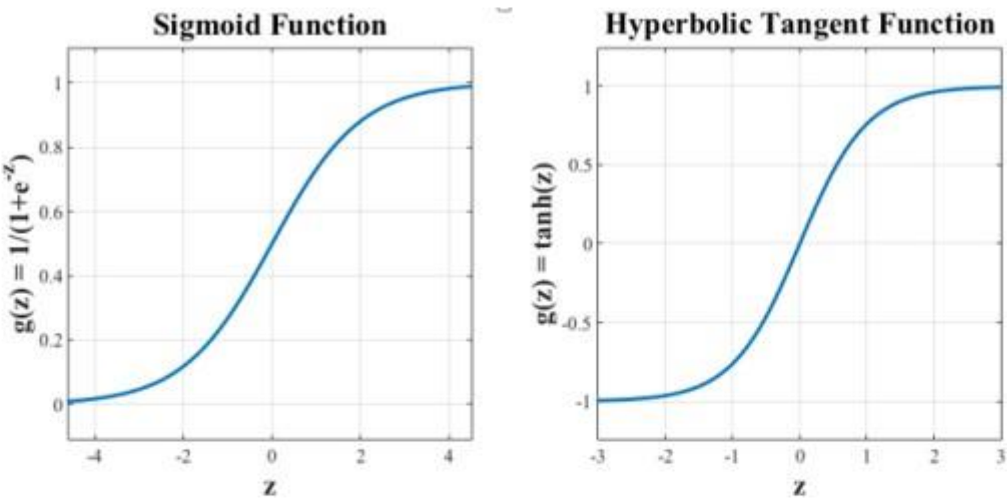


Figure 6 Sigmoid Activation Function and Hyperbolic Tangent activation functions

Gradient-based learning is hindered by the issue of these functions saturating for zs with high absolute values. The rectified linear unit (ReLU) g(z) = max(z, 0) can be used as an activation function to address this issue.

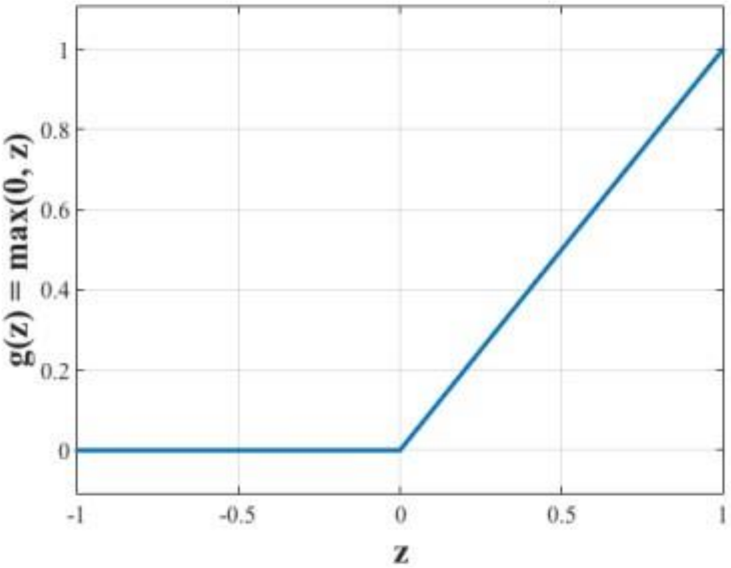


Figure 7 Rectified Linear Function

vi. Cross-Entropy

The loss function is measured between softmax function's generated outputs and training data's labels.

$$\boldsymbol{H}(\boldsymbol{l},\boldsymbol{s}) = \boldsymbol{H}(\boldsymbol{l}) + \boldsymbol{DKL}\ (\boldsymbol{l}\,||\boldsymbol{s}\ ) \tag{2.10}$$

Where *s* is the approximately label probability distribution of the softmax unit and *l* denotes the actual label probability distribution.

The cross-entropy is viewed as an improved loss measure than the Mean Squared Error (MSE) in CNNs where the softmax unit is used. This is primarily because MSE is effective in regression problems when the continuous value is provided as outputs.

vii. Mini Batch Learning

Deep learning models have a huge number of training instances, which slows learning and puts a strain on computational resources. To combat the problem of data drowning, the idea of using tiny batches of training set examples has been proposed. This means that the network is trained over multiple batches of data rather than using a single batch for each iteration. Instead of a single error function, the optimization problem occurs on numerous error sub-functions.

A balance between stochastic learning and batches is mini-batch learning. The fundamental benefit of these is the preservation of both time and computational resources. Size of mini-batches in deep learning models is frequently changed to sometimes 64, or 128, or 256, or 512, or 1024.

viii. Data Augmentation

Their full potential is increased when large data sets are utilized to train them, deep learning models are driven by data. Indeed, it has recently been demonstrated that expanding the training set's size lowers over fitting and enhances the deep learning models generalization. On the other hand, obtaining data is a costly process and consumes time too. Data augmentation, a regularization approach, is used to create training data and raise the quantity of training datasets in order to address the problem with data collecting. In recent years, data augmentation has significantly improved the performance of a variety of ML tasks, especially classification.

The quantity of training sets can be successfully increased without impacting instance labels, for example, by rotating images for an object identification task or by inserting background noise, shift, and increasing or decreasing speed of voice signals for a vocal sentiment detection task.

Chapter 4

Methodology

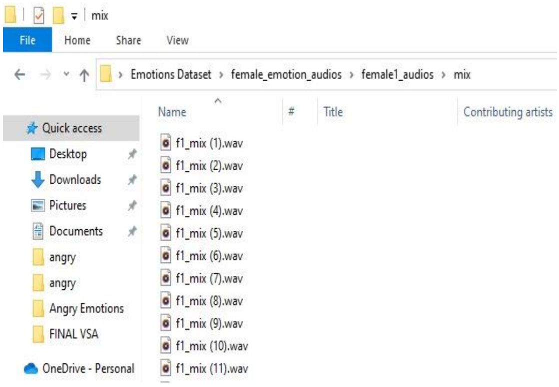
4.1 Datasets

Initially we opted for vocal sentiment analysis in English language using datasets that were readily available over the internet and contained multiple audio files in .wav format. These four datasets are listed below:

- Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)
- Surrey Audio-Visual Expressed Emotion (SAVEE)
- Toronto Emotional Speech Set (TESS)
- Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D)

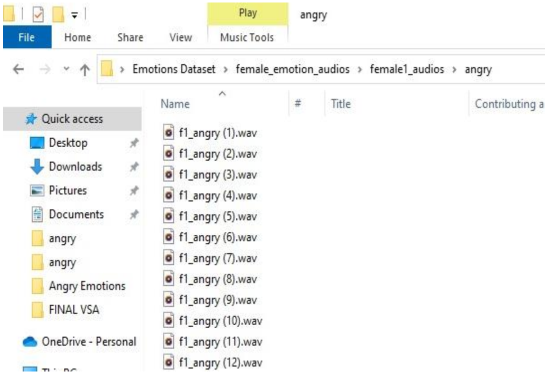
These datasets included audio files with a variety of accents and voices, including both male and female voices. We trained and tested our model on the above-mentioned datasets, which in return provided us with considerable accuracies.

We then shifted our approach of sentiment analysis to **Urdu language**. Whose datasets were not easily available. Furthermore, we created the dataset in Urdu from scratch. The dataset included the audio files of two males and two females, speaking different utterances in anger and normal tone. Each individual recorded a total of 250 audios, which includes both emotions (125 audios for each emotion). Currently our dataset consists of 1000 rows.



› Emotions Dataset › female_emotion_audios › female1_audios › mix

f1_mix (1).wav
f1_mix (2).wav
f1_mix (3).wav
f1_mix (4).wav
f1_mix (5).wav
f1_mix (6).wav
f1_mix (7).wav
f1_mix (8).wav
f1_mix (9).wav
f1_mix (10).wav
f1_mix (11).wav

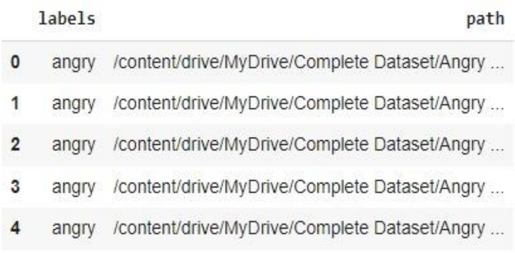


› Emotions Dataset › female_emotion_audios › female1_audios › angry

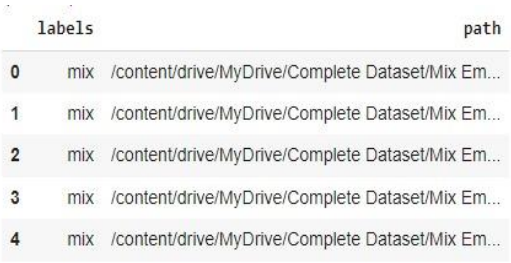
f1_angry (1).wav
f1_angry (2).wav
f1_angry (3).wav
f1_angry (4).wav
f1_angry (5).wav
f1_angry (6).wav
f1_angry (7).wav
f1_angry (8).wav
f1_angry (9).wav
f1_angry (10).wav
f1_angry (11).wav
f1_angry (12).wav

4.2 Data Preparation / Preprocessing

Now we will convert each of the datasets into their respective data frames in which there will be two columns one containing the label of the file from that dataset and that label will be indicating the emotion of that file as the file name indicates what emotion is in that vocal file, we will just interpret it according to the file name and the second column contains the path. For our dataset ‘f1’ represents female 1 and ‘m1’ represents male 1. The letters 'a' and ‘m’, represent 'anger' and ‘mix’, emotion classesrespectively.



	labels	path
0	angry	/content/drive/MyDrive/Complete Dataset/Angry ...
1	angry	/content/drive/MyDrive/Complete Dataset/Angry ...
2	angry	/content/drive/MyDrive/Complete Dataset/Angry ...
3	angry	/content/drive/MyDrive/Complete Dataset/Angry ...
4	angry	/content/drive/MyDrive/Complete Dataset/Angry ...



	labels	path
0	mix	/content/drive/MyDrive/Complete Dataset/Mix Em...
1	mix	/content/drive/MyDrive/Complete Dataset/Mix Em...
2	mix	/content/drive/MyDrive/Complete Dataset/Mix Em...
3	mix	/content/drive/MyDrive/Complete Dataset/Mix Em...
4	mix	/content/drive/MyDrive/Complete Dataset/Mix Em...

4.3 Data Visualization and Exploration

Firstly, we plotted a count plot for female and male voices separately and then combined them to make a mixed gender count plot which gave an equalnumber of audios for each emotion. All the datasets have equal number of audios for eachemotion and for each emotion there was equal number of audios for females and males.

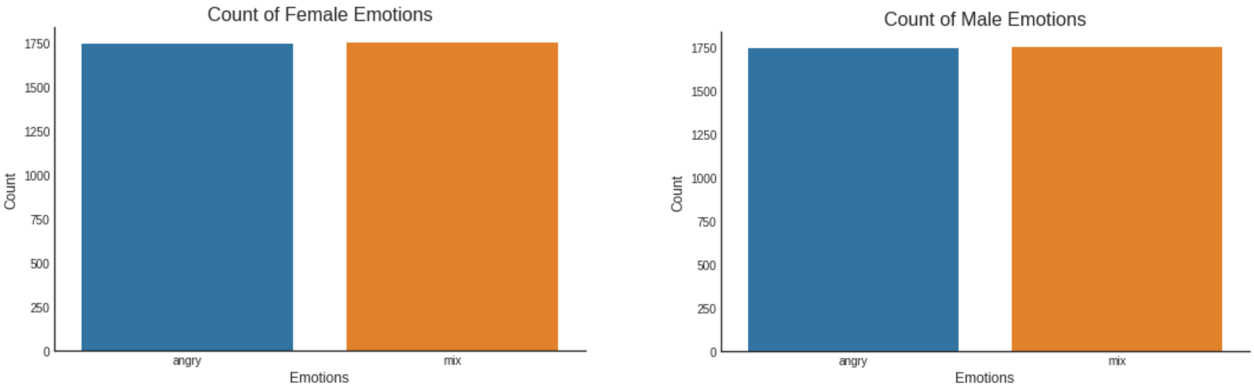


Figure 8 Count of Male and Female emotions

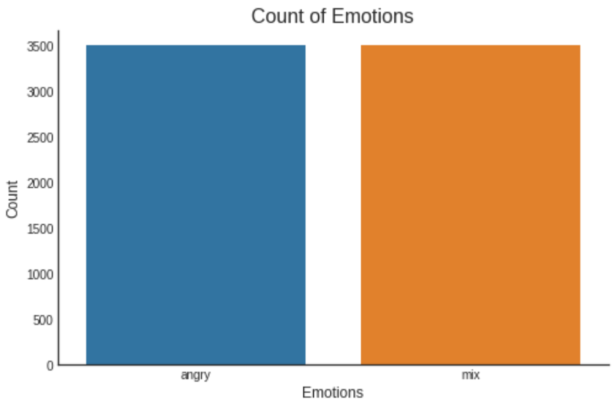


Figure 9 Count of Mix Gender Emotions

We plotted the waveform and spectrogram with various emotions to determine its properties. Wave plots show us how loud the audio is at any particular point in time. A spectrogram depicts the different frequencies that are playing at any one time, as well as their amplitude. Amplitude and frequency are important sound properties that are specific to each audio. Figure 10-11 are the wave plots and spectrograms of different emotions.

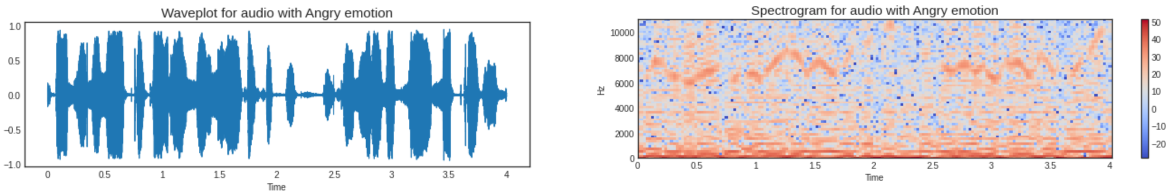


Figure 10 Wave plot and Spectrogram of Angry emotion

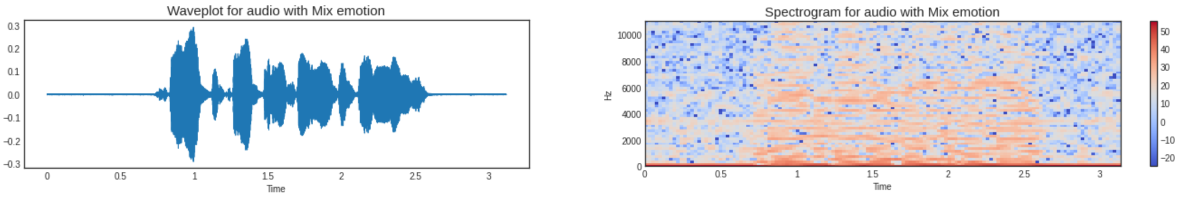


Figure 11 Wave plot and Spectrogram of mix emotion

On one axis, time is plotted, while on the other, frequency is plotted. At a certain frequency, dark patches represent relative energy. The spectrogram displays the produced wave's original sinusoidal components as horizontal bands.

4.4 Data Augmentation

Data augmentation is the process of creating further synthetic data samples from our initial training set by adding tiny perturbations. We can use noise injection, time shifts, pitch changes, and speed changes to provide syntactic data for audio. The goal is to make our model insensitive to those perturbations and improve its generalization ability. Adding perturbations must preserve the same label as the original training sample for this to operate.

We performed data augmentation in several ways on a neutral-based emotion audio from our dataset for which wave plots are shown in Figure 12.

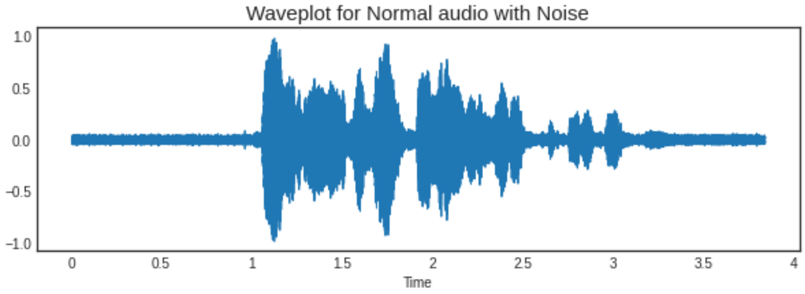


Figure 12 Wave plot for normal audio with noise

Using NumPy, it simply adds a random value to the data.

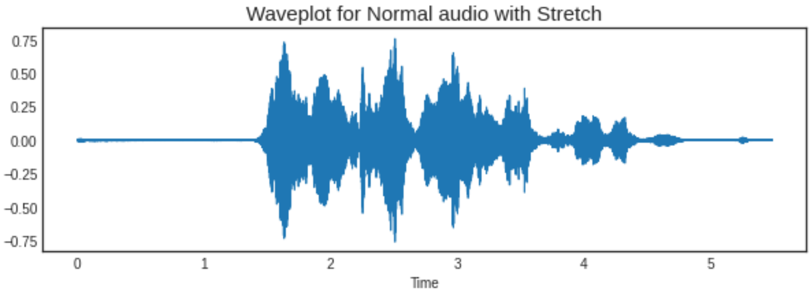


Figure 13 Wave plot for normal audio with stretch

This modifies an audio signal's speed or length without affecting its pitch

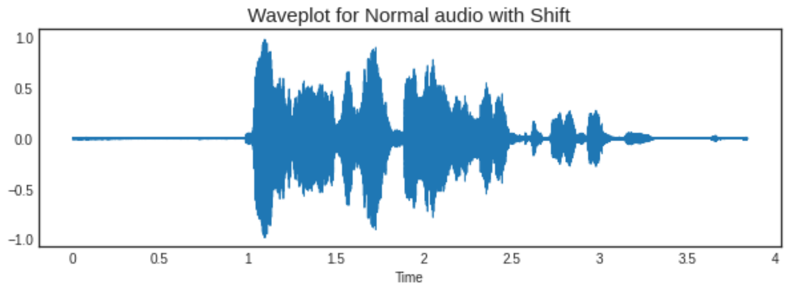


Figure 14 Wave plot for normal audio with shift

Shifting time is a relatively simple concept. It only randomly changes the audio for a split second. The initial x seconds of audio will be denoted as zero if you fast-forward it by x seconds (i.e., silence). The latest x seconds will be recorded as zero in case you shift the audio to the right (backward) for that many seconds (i.e., silence).

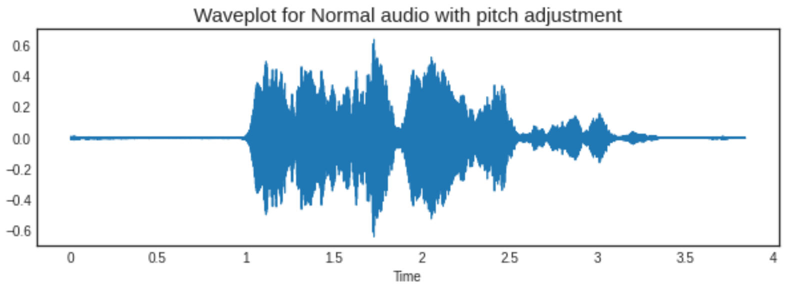


Figure 15 Wave plot for normal audio with pitch adjustment

This enhancement is a wrapper for the librosa function. It fluctuates in pitch at random.

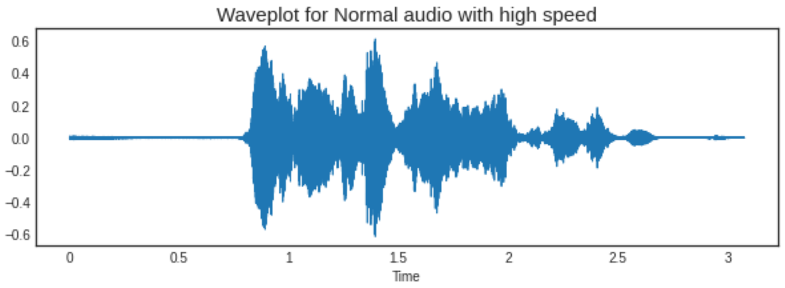


Figure 16 Wave plot for normal audio with high speed

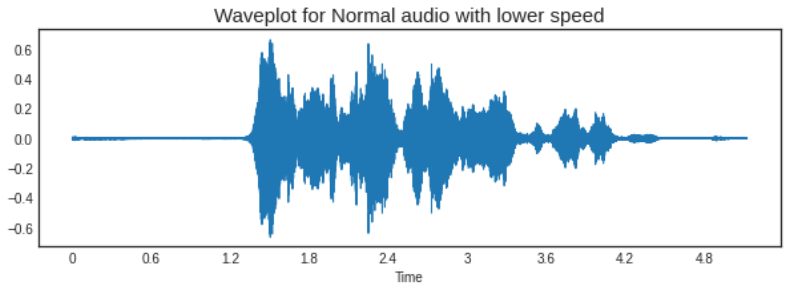
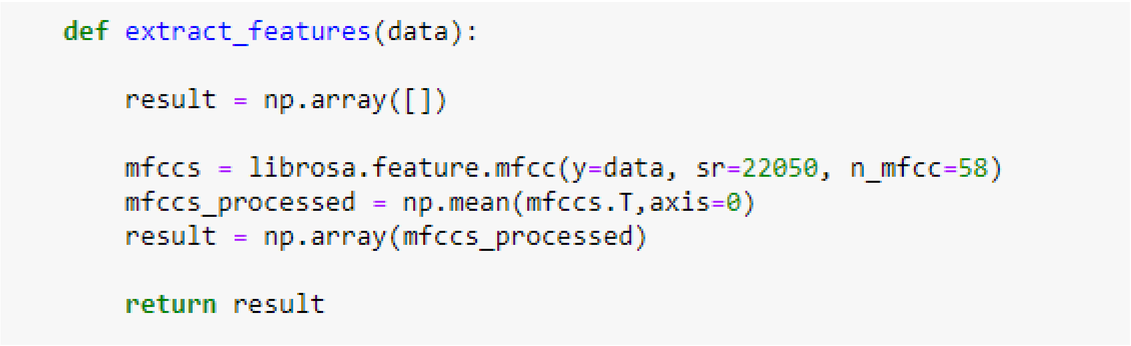


Figure 17 Wave plot for normal audio with low speed

4.5 Feature Extraction

The models are unable to directly understand the information provided by audio, therefore we must convert it into a format that the models can understand. The audio signal is a three-dimensional signal with time, amplitude, and frequency represented on three axes. Looking at the wave plots above, it becomes evident (based on aneye test) that the waveform itself does not always provide clear class defining information. In fact, they resemble one other quite closely.

Mel Frequency Cepstral Coefficients turns out to be one of the greatest tools for extracting features from audio waveforms (and digital signals in general) (MFCCs).

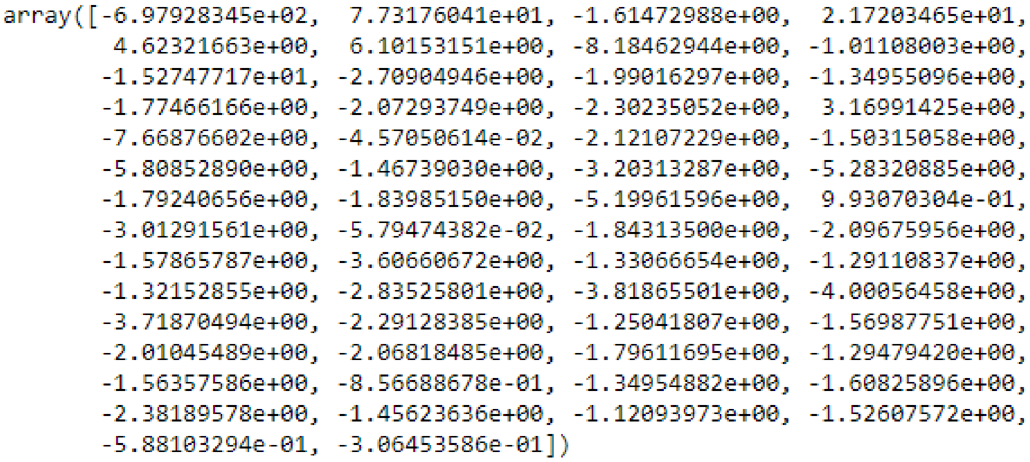


```
def extract_features(data):

    result = np.array([])

    mfccs = librosa.feature.mfcc(y=data, sr=22050, n_mfcc=58)
    mfccs_processed = np.mean(mfccs.T,axis=0)
    result = np.array(mfccs_processed)

    return result
```

Figure 18 Code for Feature Extraction

For extracting the MFCC feature from audio we have to define the sampling rate which is defined 22050 Hz here along with the number of mfccs to be extracted from the audio are 58.

For a given audio MFCCs extracted are shown in Figure 20

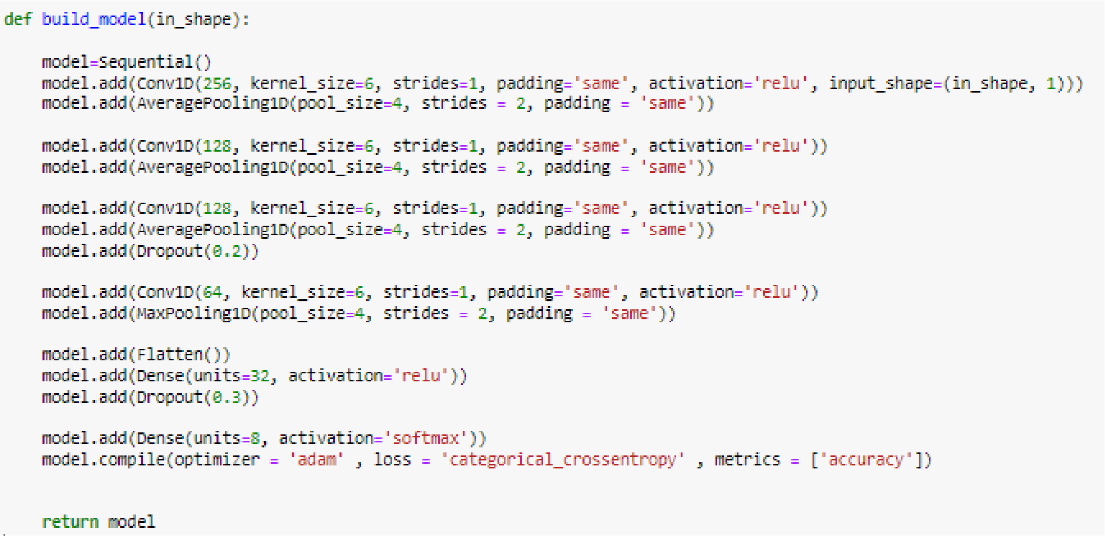


```
array([-6.97928345e+02,  7.73176041e+01, -1.61472988e+00,  2.17203465e+01,
        4.62321663e+00,  6.10153151e+00, -8.18462944e+00, -1.01108003e+00,
       -1.52747717e+01, -2.70904946e+00, -1.99016297e+00, -1.34955096e+00,
       -1.77466166e+00, -2.07293749e+00, -2.30235052e+00,  3.16991425e+00,
       -7.66876602e+00, -4.57050614e-02, -2.12107229e+00, -1.50315058e+00,
       -5.80852890e+00, -1.46739030e+00, -3.20313287e+00, -5.28320885e+00,
       -1.79240656e+00, -1.83985150e+00, -5.19961596e+00,  9.93070304e-01,
       -3.01291561e+00, -5.79474382e-02, -1.84313500e+00, -2.09675956e+00,
       -1.57865787e+00, -3.60660672e+00, -1.33066654e+00, -1.29110837e+00,
       -1.32152855e+00, -2.83525801e+00, -3.81865501e+00, -4.00056458e+00,
       -3.71870494e+00, -2.29128385e+00, -1.25041807e+00, -1.56987751e+00,
       -2.01045489e+00, -2.06818485e+00, -1.79611695e+00, -1.29479420e+00,
       -1.56357586e+00, -8.56688678e-01, -1.34954882e+00, -1.60825896e+00,
       -2.38189578e+00, -1.45623636e+00, -1.12093973e+00, -1.52607572e+00,
       -5.88103294e-01, -3.06453586e-01])
```

Figure 19 MFCC's of an Audio File

4.6 Model

We have used Convolution Neural Network (CNN) for training and testing our model because these deep neural networks have seen a lot of successin Speech Data Processing. The architecture of our CNN contains four convolution layers (Conv1D) each followed by a pooling layer out of which the starting three are Average Pooling layers and the last one is the Max pooling Layer. The foregoing processes produce a standardized 1D-vector (array), which is then sent to the convolution layer and subsequently to the pooling layers. The fully linked layer, which is essentially a conventional MLP, is the final component of CNNs. The CNN output is a 2D vector (array).

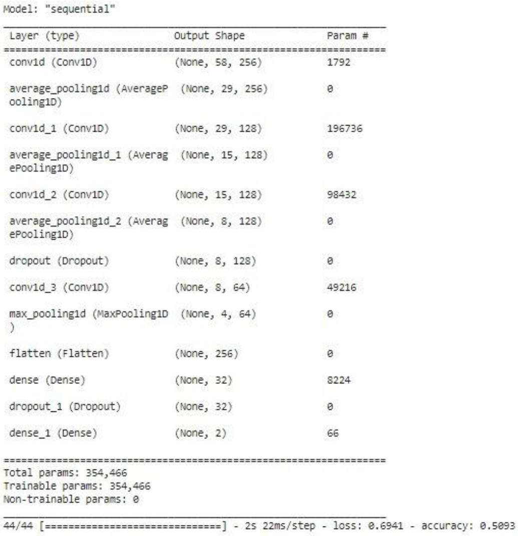


```
def build_model(in_shape):

    model=Sequential()
    model.add(Conv1D(256, kernel_size=6, strides=1, padding='same', activation='relu', input_shape=(in_shape, 1)))
    model.add(AveragePooling1D(pool_size=4, strides = 2, padding = 'same'))

    model.add(Conv1D(128, kernel_size=6, strides=1, padding='same', activation='relu'))
    model.add(AveragePooling1D(pool_size=4, strides = 2, padding = 'same'))

    model.add(Conv1D(128, kernel_size=6, strides=1, padding='same', activation='relu'))
    model.add(AveragePooling1D(pool_size=4, strides = 2, padding = 'same'))
    model.add(Dropout(0.2))

    model.add(Conv1D(64, kernel_size=6, strides=1, padding='same', activation='relu'))
    model.add(MaxPooling1D(pool_size=4, strides = 2, padding = 'same'))

    model.add(Flatten())
    model.add(Dense(units=32, activation='relu'))
    model.add(Dropout(0.3))

    model.add(Dense(units=8, activation='softmax'))
    model.compile(optimizer = 'adam' , loss = 'categorical_crossentropy' , metrics = ['accuracy'])


    return model
```

Figure 20 Fully Linked Layer

Every layer had ReLU (Rectified Linear Activation) used. The ReLU activation function takes in input and gives an output of y = x for all values greater than 1 of x and y = 0 for all values less than 1 of x. Because of its simplicity, it is widely utilized. The ReLU function is simple to compute, which is important for hidden layers because it speeds up the whole process. The last layer's function is 'SoftMax,' which returns y = 1 if x > 0.5 and 0 otherwise. This will help in the output of the one-hot vector. The next part is the optimizer used. The optimizer we used is Adam. Because the Adam optimizer improved the CNN abilities in classificationand segmentation with the highest accuracy.

The overall summary of our model is given in Figure 22:

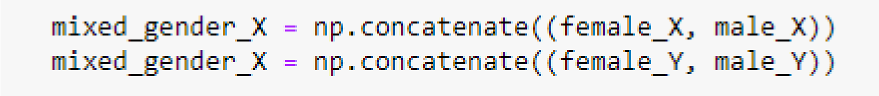


```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d (Conv1D)             (None, 58, 256)           1792

 average_pooling1d (AverageP  (None, 29, 256)          0
 ooling1D)

 conv1d_1 (Conv1D)           (None, 29, 128)           196736

 average_pooling1d_1 (Averag  (None, 15, 128)          0
 ePooling1D)

 conv1d_2 (Conv1D)           (None, 15, 128)           98432

 average_pooling1d_2 (Averag  (None, 8, 128)           0
 ePooling1D)

 dropout (Dropout)           (None, 8, 128)            0

 conv1d_3 (Conv1D)           (None, 8, 64)             49216

 max_pooling1d (MaxPooling1D  (None, 4, 64)            0
 )

 flatten (Flatten)           (None, 256)               0

 dense (Dense)               (None, 32)                8224

 dropout_1 (Dropout)         (None, 32)                0

 dense_1 (Dense)             (None, 2)                 66

=================================================================
Total params: 354,466
Trainable params: 354,466
Non-trainable params: 0
_________________________________________________________________
44/44 [==============================] - 2s 22ms/step - loss: 0.6941 - accuracy: 0.5093
```

Figure 21 Summary of Model

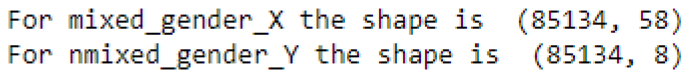
4.7 Training and Testing

As mentioned above for EDA we have split our dataset on the basis of gender i.e., male and female now for model training and testing we'll concatenate these two data frames so we have the full picture of our data



```
mixed_gender_X = np.concatenate((female_X, male_X))
mixed_gender_X = np.concatenate((female_Y, male_Y))
```

The total shape of this combined data frame is given in Figure 23:



```
For mixed_gender_X the shape is  (85134, 58)
For nmixed_gender_Y the shape is  (85134, 8)
```

Figure 22 Total Shape of Data

This complete dataset is now further divided into two subsets for training and testing of our model, while keeping the test size = 0.20 which means the model will be trained for 80% of the data and the rest 20% will be used in the testing phase for our model.

4.8 Results

During training, a Loss curve is one of the most commonly used curve to debug a neural network. It provides an overview of the training process as well as the network's learning trajectory. An Accuracy curve is another commonly used curve to understand the development of Neural Networks.

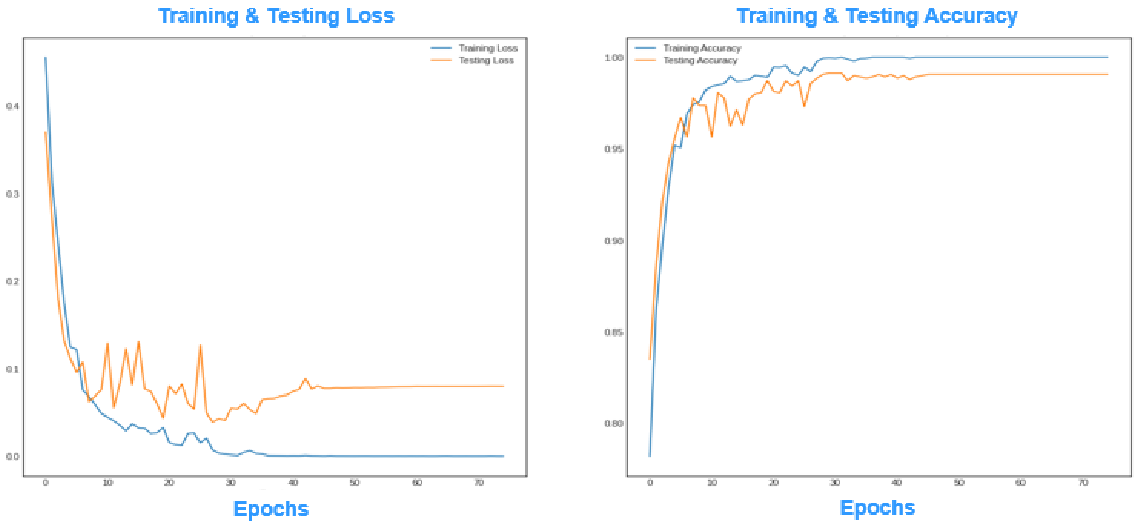


Figure 23 Loss and Accuracy Curves

Examining model learning curves during training helped us to diagnose learning issues, such as an under fitting or over fitting of our model, as well as whether the training and validation datasets are sufficiently representative.

The accuracy curve, which includes both training and validation accuracy, is a more essential curve. The training went smoothly for two hours approximately and after that the training accuracy came out to be 100.00%. The result for each is given below:

- Mixed Gender Emotion Training Accuracy: 100.00%
- Mixed Gender Emotion Testing Accuracy: 99.07%

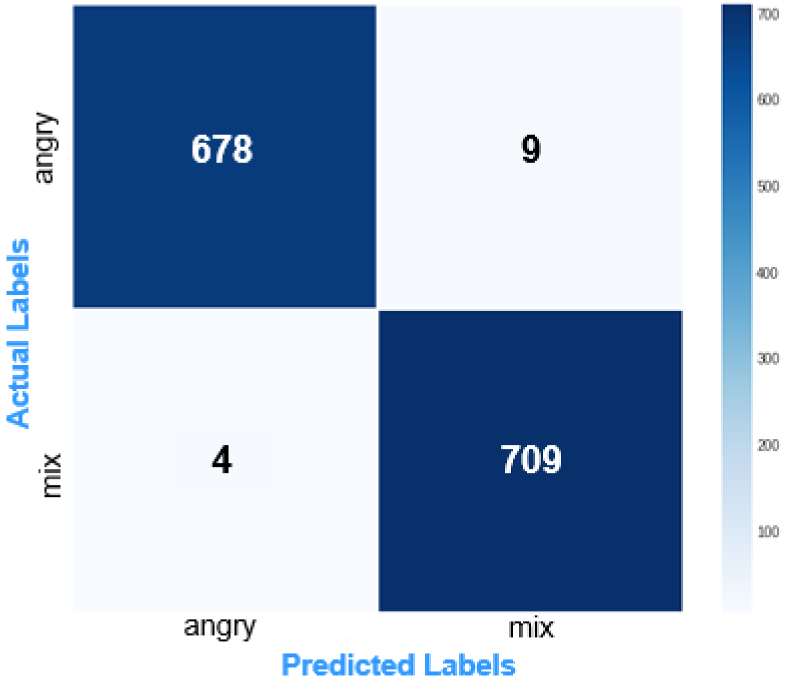


Figure 24 Confusion Matrix

We used Confusion Matrix to visualize the prediction of our True Positive results. As can be observed, the network mostly correctly categorized emotions most of the time. A total of 1400 audio samples were provided as an input to the model for testing. The confusion matrix elaborates that 1387 audios were correctly predicted.

Chapter 5 USER INTERFACE

5.1 Introduction about UI

The way a user interacts with a program or application is known as the user interface. It involves human-computer interaction. Desktop, mouse, screen, keyboard, microphone, speakers and the design of a system or application are all included. Several types of user interface include:

1. Graphical User Interface (GUI)
2. Command Line Interface (CLI)
3. Voice User Interface
4. Touch User Interface

5.2 Introduction about FLASK

For our project, the database was created using Flask. Python-based Flask is a web framework. It is utilized to create scalable and secure web applications. It is a micro framework; it is simple to utilize. It delivers practical features and tools that facilitate Python web application development. It allows ease to programmers and is a comparatively approachable platform for starters because you can easily create a web application with only a single Python file.

5.3 Tools and platform for flask

5.3.1 For Designing

For designing our web application, we have used Hyper-Text Markup Language (HTML), React, and Cascading Style Sheet (CSS). These files were created on Microsoft Visual Studio Code (VS Code).

5.3.2 For Database

We’ve used two types of databases a cloud storage and a real-time storage.

Firebase: Firebase being an application development platform, allows users to develop web applications, as well as mobile applications. The design of Firebase Cloud Storage takes into account mobile connectivity. When your application loses and regains connectivity, it automatically stops and resumes transfer.

Real-Time database in Firebase: Real-time Database in Firebase enables real-time data synchronization between different users. It allows for the global syncing, storing, and querying of app data.

5.3.3 User Authentication

Authentication process enables the exchange of information between human and the machine. This makes sure that an authentic user is trying to access the application. This procedure verifies a user's identity when they attempt to connect to a network or computing resource. For the purpose of authenticating users, Firebase authentication is utilized.

5.3.4 Agents Information

The real-time database contains information on agents who report to a supervisor. At any time, the supervisor can access to the agent's records. The storage is where the agent's recorded audios are kept.

5.4 Web Application Interface

5.4.1 Welcome page

This page consists of three options:

- Upload Audio
- Record Audio
- Access Database

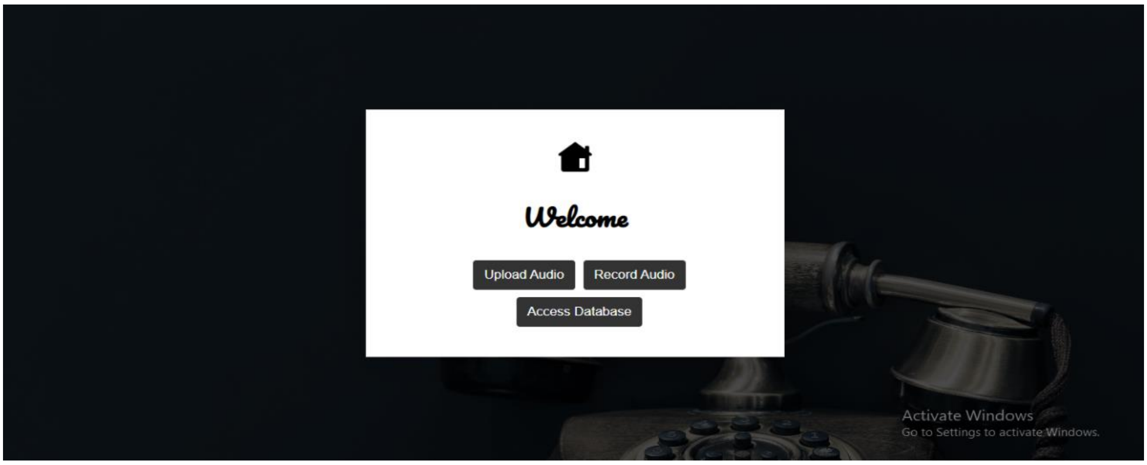


Figure 26 Welcome Page

Upload Audio: Upon clicking this button, a file modal window will pop up. The user is allowed to submit any type of audio in .wav format for sentiment analysis.

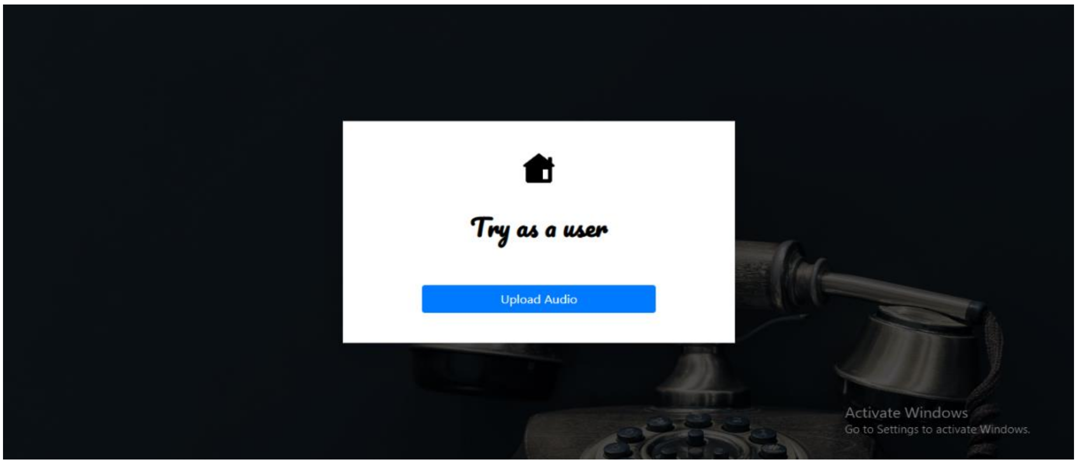
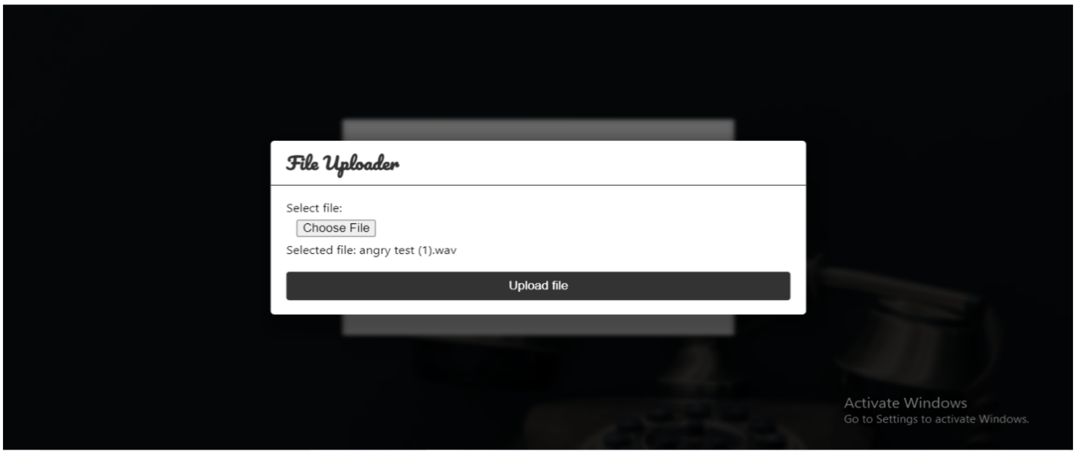


Figure 27 Upload Screen

File Modal

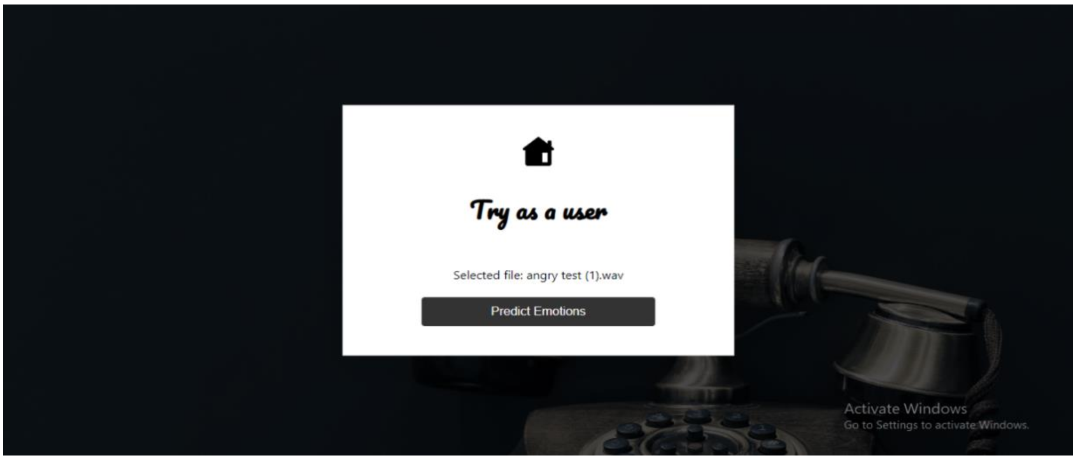


Select file:

Selected file: angry test (1).wav

Figure 28 File Modal

After that, the audio will be prepared for processing and emotion prediction.



Try as a user

Selected file: angry test (1).wav

Figure 29 Ready for prediction

Once the results are retrieved, an analysis of the uploaded audio will be displayed in the form of a pie chart along with the preview of the uploaded audio.

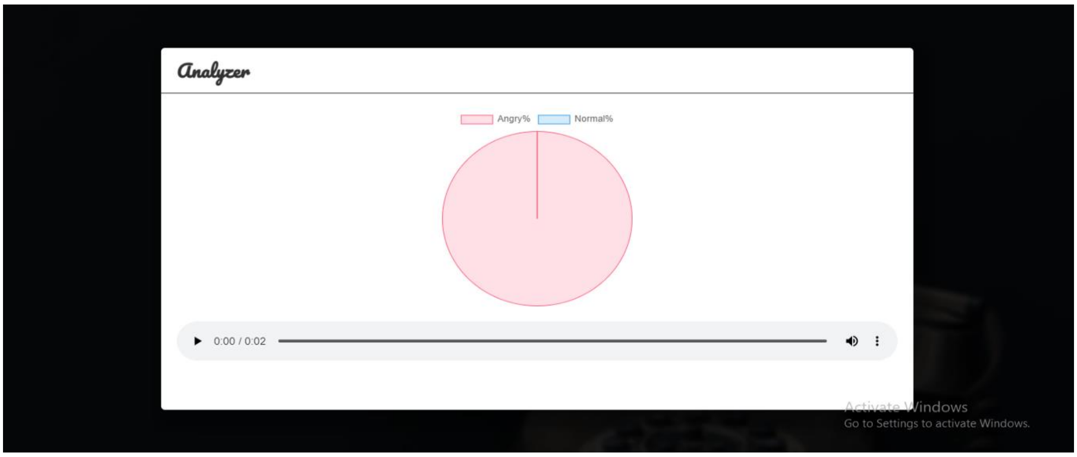


Figure 30 Analysis result

Record Audio:

Live Audio Recorder: The user can record real-time audios and predict sentiments using this feature.

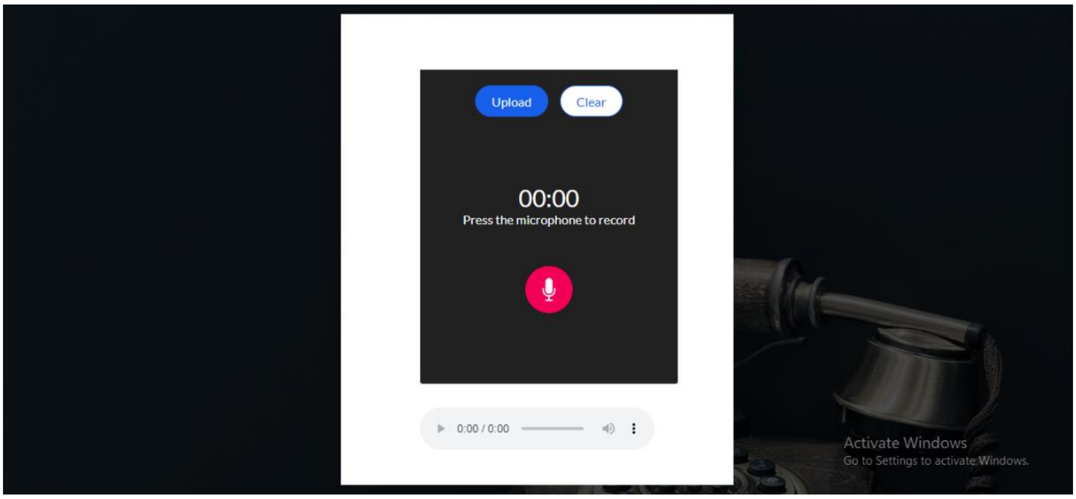

Upload
Clear
00:00
Press the microphone to record
0:00 / 0:00
Activate Windows
Go to Settings to activate Windows.

Figure 31 Recorder Screen

A counter will commence and start monitoring the duration of the audio as soon as the user clicks on the microphone.

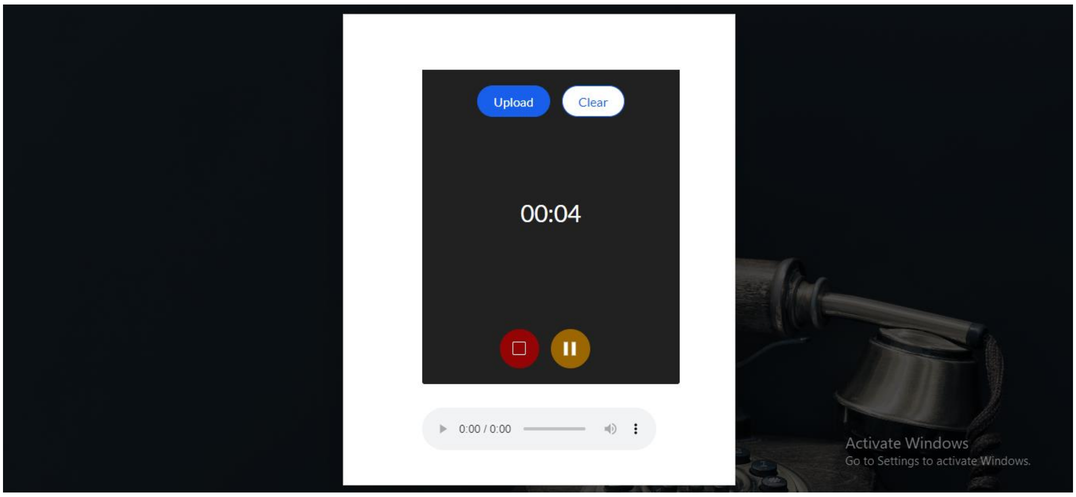


Figure 32 Recorder Screen

After recording, the user has choice of either uploading the audio for analysis or deleting it and starting over.

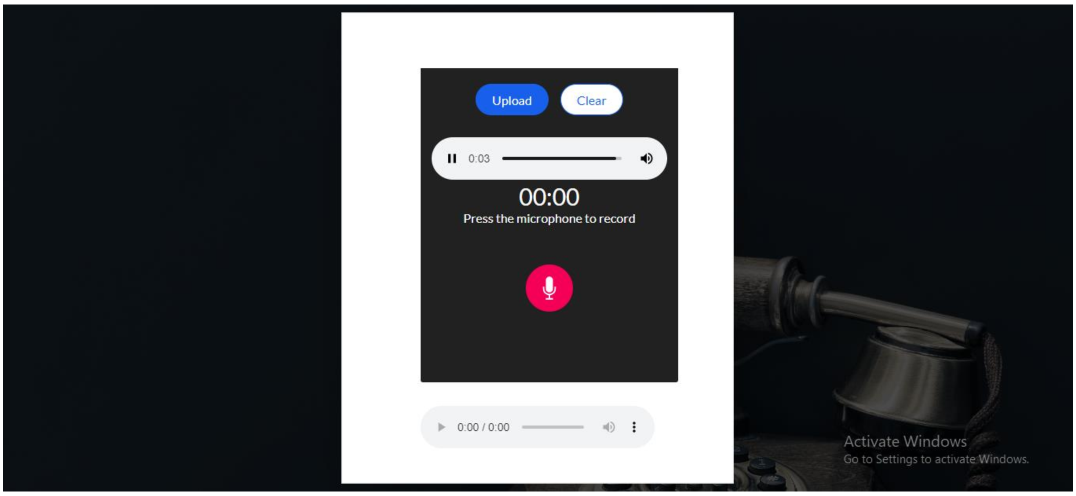

Upload
Clear
0:03
00:00
Press the microphone to record
0:00 / 0:00
Activate Windows
Go to Settings to activate Windows.

Figure 33 Recorder Screen

As a final result a pie chart will be generated with predicted emotions along with the preview of the recorded audio under it.

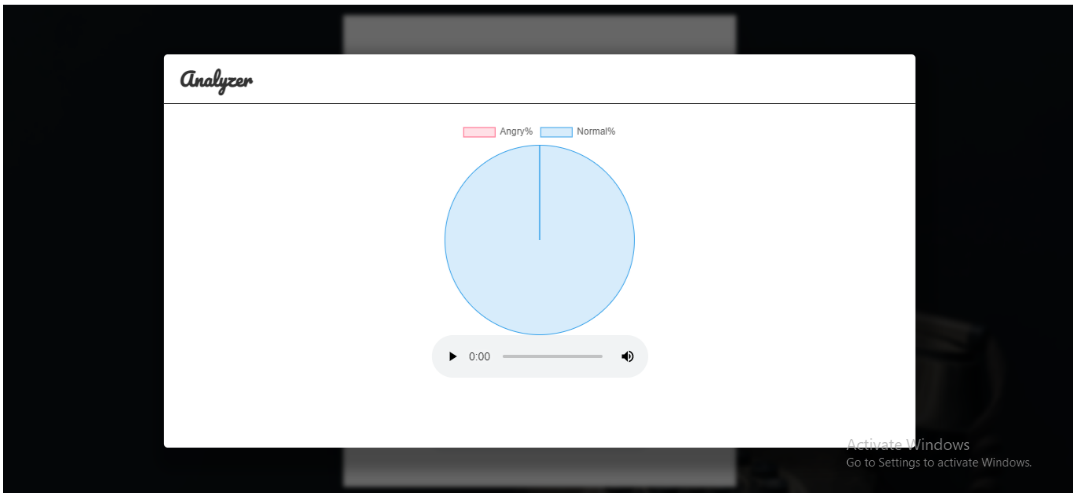


Figure 34 Live Audio Analysis

5.4.2 Supervisor’s login page

This page is particularly for the supervisors, the login information is stored in the firebase database for authenticated users i.e., supervisors.

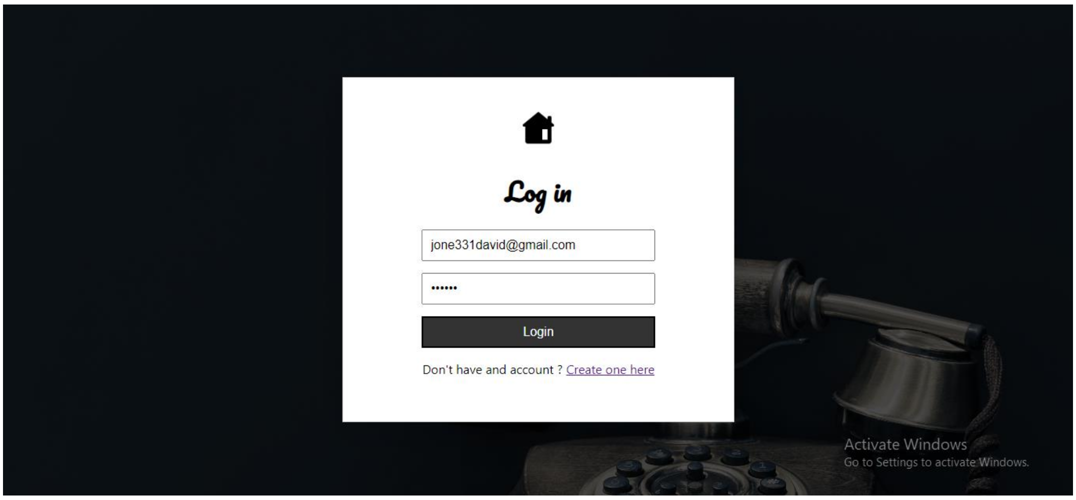
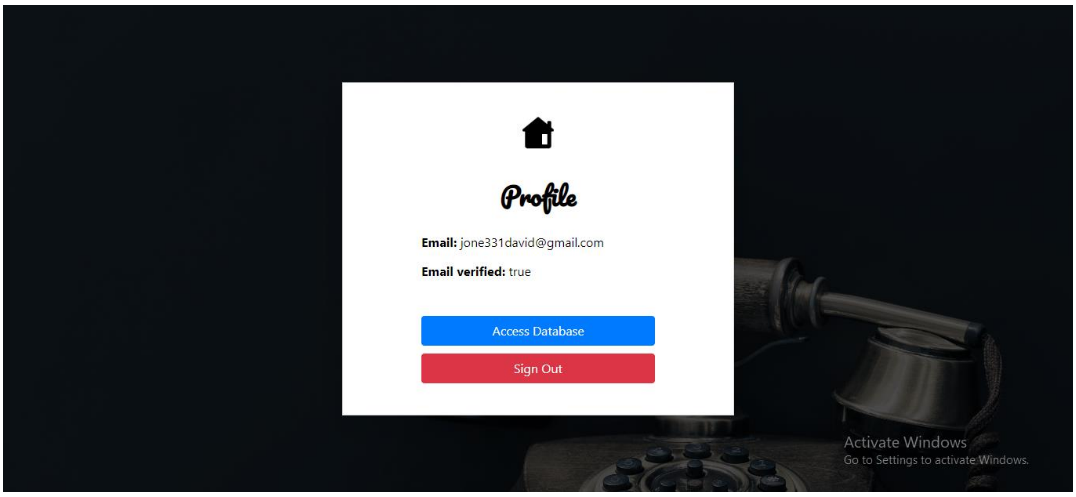

Log in
jone331david@gmail.com
••••••
Login
Don't have and account ? Create one here
Activate Windows
Go to Settings to activate Windows.

Figure 35 Supervisor-Login

Upon successful login the user will have an option to access the database or to sign out.

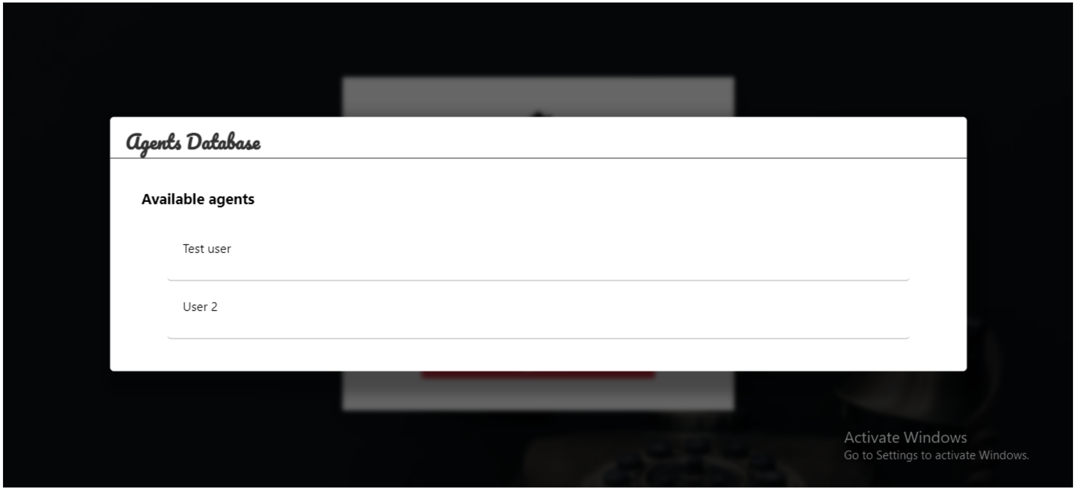


Email: jone331david@gmail.com

Email verified: true

Figure 36 Successful Login

Upon clicking the “Access Database” button, the supervisor will be directed to a modal with all the available agent’s information working under him/her. The agent’s information is being fetched from the firebase real-time database.



Agents Database

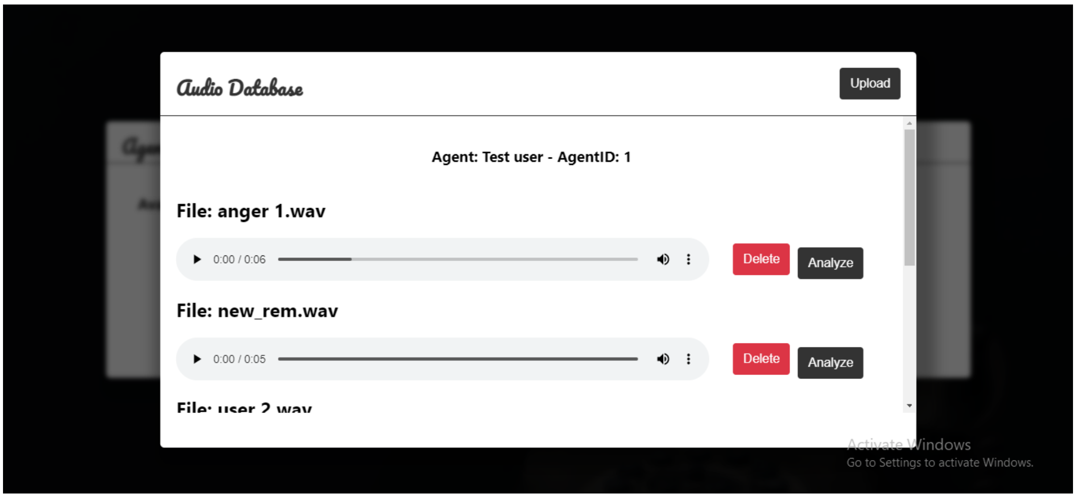
Available agents

Test user

User 2

Figure 37 Agents

Each agent's recorded audios will be saved in the database. The audios for one particular agent will be fetched after clicking on that agent's name. Supervisors can delete audios from an agent's records or upload any file related to that agent and store it in the database.



Audio Database

Agent: Test user - AgentID: 1

File: anger 1.wav

File: new_rem.wav

File: user 2 wav

Figure 38 Agent's Audios (Admin Panel)

The main work is done once the supervisor clicks on the analyze button, the API will be called, and the results will be predicted for that particular audio in the form of a pie chart.

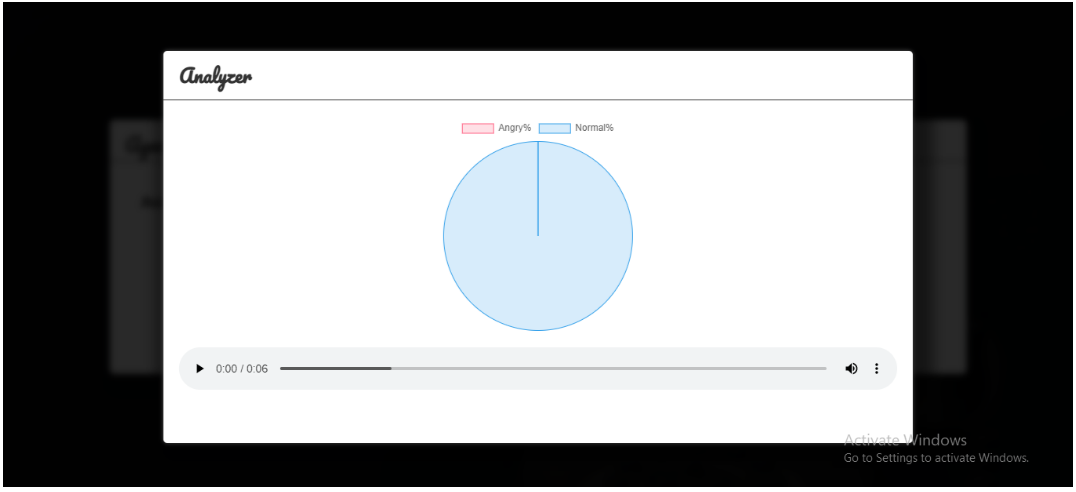


Figure 39 Agent's Audio Analysis Result

Chapter 6

Conclusion

6.1 Summary

For at least a couple of decades, sentiment analysis has existed in some form or another. It began as a rules-based system that searched for and counted the frequency of specific termsthat indicated the customer's mood. Getting the sentiment of the customers will improve theoutcome of the products. These technologies were rather basic, and they were mostly limitedto text-based information.

Natural language understanding and neural networks have improved over the time during pastfive years, and machine learning algorithms have emerged, that have radically transformed the game. These now consider not only the textual and facial data, but also vocal data providing a much more nuanced and more accurate picture of the sentiment behind audio data. Due to this, we made the decision to create a model that could manage a variety of AI-related applications by being able to recognize an individual's sentiments merely by their voice.

6.2 Results

CNNs had made a lot of success in Speech Data Processing. The CNN can be depicted as a conventional neural network. As opposed to traditional Neural Network techniques, which employ fully linked hidden layers, CNN offers a unique network made up of alternating convolution and pooling layers.

We identified the best CNN Model after rigorously implementing other models in order to perform emotion classification. With this model, we were able to obtain a training accuracy of 97 percent. If we had additional data to work with, our model would perform better. We can also see how the model anticipated the actual numbers in the graphs above.

We suggested a simple and small convolutional neural network (CNN) architecture having multiple layers alongside modified kernels and a pooling strategy to detect the sensitive cues on the basis of deep frequency characteristics from voice spectrograms, which are more selective and robust for identifying emotions from voice.

6.3 Recommendation for Future Work

The dataset can be expanded for both training and testing data in the future to improve overall application and outcomes. The model can currently distinguish angry emotions from other emotions like normal, sad, happy, etc. More research can be done to anticipate more than one emotion. Additionally, more research can be carried out to enhance the user experience and by adding new functionality, such as enabling several supervisors to use the application.

The entire sentiment monitoring application can be automated for supervisors by raising an alarm for every audio that contains a percentage of angry emotion higher than a specified threshold value. In this manner, only the audios that will trigger alarms would need to be examined by the supervisors.

References

[1] 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). (2006). *IEEE Transactions on Audio, Speech, and Language Processing*, *14*(5), 1886. https://doi.org/10.1109/tasl.2006.882560

[2] Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G., & Yu, D. (2014). Convolutional Neural Networks for Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *22*(10), 1533–1545. https://doi.org/10.1109/taslp.2014.2339736

[3] Byun, S. W., & Lee, S. P. (2021). A Study on a Speech Emotion Recognition System with Effective Acoustic Features Using Deep Learning Algorithms. *Applied Sciences*, *11*(4), 1890. https://doi.org/10.3390/app11041890

[4] Cho, H., Kim, Y., Lee, E., Choi, D., Lee, Y., & Rhee, W. (2020). Basic Enhancement Strategies When Using Bayesian Optimization for Hyperparameter Tuning of Deep Neural Networks. *IEEE Access*, *8*, 52588–52608. https://doi.org/10.1109/access.2020.2981072

[5] Chowhan, A., & Mathew, R. (2021). Study of Speech Emotion Recognition Using Neural Networks. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3769842

[6] Chul Min Lee, & Narayanan, S. (2005). Toward detecting emotions in spoken dialogs. *IEEE Transactions on Speech and Audio Processing*, *13*(2), 293–303. https://doi.org/10.1109/tsa.2004.838534

[7] Dasgupta, P. B. (2017). Detection and Analysis of Human Emotions through Voice and Speech Pattern Processing. *International Journal of Computer Trends and Technology*, *52*(1), 1–3. https://doi.org/10.14445/22312803/ijctt-v52p101

[8] el Ayadi, M., Kamel, M. S., & Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, *44*(3), 572–587. https://doi.org/10.1016/j.patcog.2010.09.020

[9] Gayathri, P., Priya, P. G., Sravani, L., Johnson, S., & Sampath, V. (2020). Convolutional Recurrent Neural Networks Based Speech Emotion Recognition. *Journal of Computational and Theoretical Nanoscience*, *17*(8), 3786–3789. https://doi.org/10.1166/jctn.2020.9321

[10] He, X., & Zhang, W. (2018). Emotion recognition by assisted learning with convolutional neural networks. *Neurocomputing*, *291*, 187–194. https://doi.org/10.1016/j.neucom.2018.02.073

[11] H.Mansour, A., Zen Alabdeen Salh, G., & A. Mohammed, K. (2015). Voice Recognition using Dynamic Time Warping and Mel-Frequency Cepstral Coefficients Algorithms. *International Journal of Computer Applications*, *116*(2), 34–41. https://doi.org/10.5120/20312-2362

[12] Hsieh, T. A., Wang, H. M., Lu, X., & Tsao, Y. (2020). WaveCRN: An Efficient Convolutional Recurrent Neural Network for End-to-End Speech Enhancement. *IEEE Signal Processing Letters*, *27*, 2149–2153. https://doi.org/10.1109/lsp.2020.3040693

[13] Huang, J. T., Li, J., & Gong, Y. (2015). An analysis of convolutional neural networks for speech recognition. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2015.7178920

[14] Huang, K. L., Duan, S. F., & Lyu, X. (2021). Affective Voice Interaction and Artificial Intelligence: A Research Study on the Acoustic Features of Gender and the Emotional States of the PAD Model. *Frontiers in Psychology*, *12*. https://doi.org/10.3389/fpsyg.2021.664925

[15] Huddar, M. G., Sannakki, S. S., & Rajpurohit, V. S. (2019). A Survey of Computational Approaches and Challenges in Multimodal Sentiment Analysis. *International Journal of Computer Sciences and Engineering*, *7*(1), 876–883. https://doi.org/10.26438/ijcse/v7i1.876883

[16] Jaiswal, S., & Nandi, G. C. (2019). Robust real-time emotion detection system using CNN architecture. *Neural Computing and Applications*, *32*(15), 11253–11262. https://doi.org/10.1007/s00521-019-04564-4

[17] Khalil, R. A., Jones, E., Babar, M. I., Jan, T., Zafar, M. H., & Alhussain, T. (2019). Speech Emotion Recognition Using Deep Learning Techniques: A Review. *IEEE Access*, *7*, 117327–117345. https://doi.org/10.1109/access.2019.2936124

[18] Kim, J., Bukhari, W., & Lee, M. (2017). Feature Analysis of Unsupervised Learning for Multi-task Classification Using Convolutional Neural Network. *Neural Processing Letters*, *47*(3), 783–797. https://doi.org/10.1007/s11063-017-9724-1

[19] Koduru, A., Valiveti, H. B., & Budati, A. K. (2020). Feature extraction algorithms to improve the speech emotion recognition rate. *International Journal of Speech Technology*, *23*(1), 45–55. https://doi.org/10.1007/s10772-020-09672-4

[20] Nagajyothi, D., & Siddaiah, P. (2018). Speech Recognition Using Convolutional Neural Networks. *International Journal of Engineering & Technology*, *7*(4.6), 133. https://doi.org/10.14419/ijet.v7i4.6.20449

[21] Nwe, T. L., Foo, S. W., & de Silva, L. C. (2003). Speech emotion recognition using hidden Markov models. *Speech Communication*, *41*(4), 603–623. https://doi.org/10.1016/s0167-6393(03)00099-2

[22] Pahwa, A., & Aggarwal, G. (2016). Speech Feature Extraction for Gender Recognition. *International Journal of Image, Graphics and Signal Processing*, *8*(9), 17–25. https://doi.org/10.5815/ijigsp.2016.09.03

[23] Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. *Interspeech 2019*. https://doi.org/10.21437/interspeech.2019-2680

[24] Tripathi, S., Acharya, S., Sharma, R., Mittal, S., & Bhattacharya, S. (2017). Using Deep and Convolutional Neural Networks for Accurate Emotion Classification on DEAP Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(2), 4746–4752. https://doi.org/10.1609/aaai.v31i2.19105

[25] Wu, S., Falk, T. H., & Chan, W. Y. (2011). Automatic speech emotion recognition using modulation spectral features. *Speech Communication*, *53*(5), 768–785. https://doi.org/10.1016/j.specom.2010.08.013

[26] Xiaodong Cui, Goel, V., & Kingsbury, B. (2015). Data Augmentation for Deep Neural Network Acoustic Modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *23*(9), 1469–1477. https://doi.org/10.1109/taslp.2015.2438544

[27] Zhang, S., Zhang, S., Huang, T., & Gao, W. (2018). Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching. *IEEE Transactions on Multimedia*, *20*(6), 1576–1590. https://doi.org/10.1109/tmm.2017.2766843