Contents

Classifying Emotions from Speech Using R for BAVED dataset.

Mrigank Shukla
19BCE1200
B.Tech Computer Science and Engineering (SCOPE)
VIT Chennai.
mrigank.shukla2019@vitstudent.ac.in

Abstract

Emotion recognition has evolved as a significant study subject that might provide useful information for a range of reasons. The ability to extract emotion from speech has a wide range of applications in fields such as technology, health, and business.My paper aims to extract features from speech data to train machine learning models such as SVM, Random Forest and LDA. These trained models can then be used to accurately classify a given emotion associated with the speech. The model has been trained using the BAVED (Aouf, 2019) dataset but a similar approach can be applied for larger dataset for any other language of choice. The best performing Random Forest model gives an accuracy of 100%.

1 Introduction

1.1 Objective and Goal of Project

In daily interpersonal human relationships, emotion plays a crucial role. This is necessary for both sensible and intelligent judgments. By expressing our sentiments and providing feedback to others, it helps us match and comprehend the feelings of others. Emotion has a significant impact in moulding human social interaction, according to research. Emotional displays provide a wealth of information about a person's mental state. This has spawned a new branch of study known as automated emotion recognition, whose primary purpose is to comprehend and recall desired feelings. Even during a class lecture as in VIT, the emotion of the lecturer helps deliver the lecture in a manner which is easily graspable by the students. Human-machine interaction (Erol et al., 2019) widely employed the concept of emotion recognition of a user, this can be through facial expressions, hand gestures (Ragot et al., 2017) and speech. When two people engage with one other, they may quickly detect the underlying emotion (Sun et al., 2020) in the other person's words. In the case of a machine the goal of an emotion recognition system is to emulate the mechanics of humans underlying sensory perception.

1.2 Problem Statement

Vocal characteristic have always been identified by humans in understanding the emotional state of the speaker. The slight variations in pitch, loudness or even the small jitter helps us understand the emotion behind the spoken words. These features if calibrated mathematically can be used to understand the emotions of the speaker using a computer or a machine. Understanding the emotion of the speaker can help in creating better human machine connection.The aim of my project will be to extract audio features that can be used to detect emotions in speech/voice. Train a Machine Learning model in R that can help identify the emotion of voice sample.

1.3 Motivation

The concept of emotion recognition from speech can be used in many applications such as robot interfaces, audio surveillance, internet E-learning, business applications, healthcare investigations, call centers (Han et al., 2020). Information regarding students' emotional states can help concentrate classroom reconfiguration or E-learning on improving teaching quality. For example while conduction our online classes in VIT if the responses of student can be recognized while answering a question or interacting with the professor. Then based on the emotion the teacher can change the methodology of delivering the lecture. If the emotion is sad or confused then the concept could be explained in much easier manner. This can help in giving feedback to the teacher making teaching more effective. Humans can quickly detect the speaker's emotion. Numerous generations of practise and observation are required to attain this. Humans assess several elements of a given speech before recog-

nising the speaker's emotion based on personal past experiences or observation. Energy, pitch, formant frequency, Linear Prediction Cepstrum Coefficients (LPCC), Mel-frequency cepstrum coefficients (MFCC), and modulation spectral features (MSFs) are the major speech features that include emotion information presented by many studies. Hence I will also use these features for gaining acoustic information about the collected speech. This will be then used to train my model to classify the emotions.

1.4 Challenges

The main hurdle in this project is to extract relevant features from the audio. As these features will become features for my machine learning model. If these features are able to capture the essence of whole speech then model will be able to perform good. Also if we want to deploy such model for all languages then we need to collect the voice samples from all relevant languages, which is very difficult if we consider the legal requirements. Hence as of now the model works very well with Arabic language or similar sounding languages.

2 Literature Survey

Mohammed and Aly (Mohamed and Aly, 2021) used Wav2vec2.0, Wav2vecU, WavBERT, and HuBERT that are very well formed multilayer deep learning frameworks that allow superior sentence representations and maximum information acquisition. Thousands of unlabeled data are used to train such paradigms, which are subsequently fine-tuned on a limited dataset for specific tasks. Regarding Arabic voice dialogues, this research proposed a deep learning designed emotions recognition system. They finally decided to used Wav2vec2.0 and HuBERT that are two state-of-the-art audio representations used in the built model. They employed Wav2vec2.0, a self-supervised voice encoding framework that employs the capabilities of transformers and contrastive filtering to capture the essential features of raw audios. The model is comprised of convolutional layers which analyze the raw waveforms inputs to produce a feature representations , followed by transformer layers that provide contextualised representation and linear translation to output. They employed a pretrained model (El-Geish, 2021) that was fine-tuned for Arabic utilising Common Voice and Arabic Speech Corpus training divisions. After doing this they fed these features to a multi layer perceptron model and Bidirectional - LSTM model. Marjanovic (Noroozi et al., 2017) presented a system built on video and auditory data processing for detecting adaptive emotions. They used Principal Component Analysis to extract some 87 features from the much used Mel Frequency Cepstral Coefficients and Filter Bank energies which were able to reduce the number of dimensional measurement that were previously derived from all the extracted features. Another researcher (Bandela and Kumar, 2017) delved deeper in integrating an auditory feature known as the MFCC with a prosodic features referred to as the Teager Energy Operator, he was able to detect five emotions that are described in the Berlin Emotional Speech database with an accuracy of 70%.

Table 1: Result of Mohammed and Aly (Mohamed and Aly, 2021)

Model	Audio Sample	Accuracy
Wav2Vec2.0	1945	89
HuBERT Base	1945	87
HuBERT Large	1945	84

3 Emotions

I would like to just give an detailed description of emotions that I will be referring in this paper. As many people have different interpretation of emotion. I wanted to set a base line for term "emotion" that I will be using in this project report .Emotion is amongst the most complex psychological concepts to articulate. In reality, many definitions of emotions have been proposed in the research literature. Emotion is defined as any comparatively short subjective experience marked by intense mental activity and a greater degree of comfort or dissatisfaction in common discourse. There is no unanimity on a term, and academic discourse has shifted to various definitions. Temperament, mood, personality, motivation, and disposition are all intertwined with emotion. Emotion is commonly characterized in psychology as a complicated state of feeling that causes physical and psychological changes. These changes have an impact on how people think and act. (Hoemann et al., 2020). In the context of my study the emotions expressed will be according on a scale instead of an absolute categorization of emotions such as anger, happiness, sad, disgust etc.

4 Requirement Specifications

4.1 Hardware Specification

The project has been tested on "intel Core i5 8th Gen" processor. The computationally intensive part of the project is manipulating audio. The time taken to process the audio for removal of silence can be very high for large audio files. This can be solved by using streaming audio processing. The extraction of audio features from the silenced removed audio files also requires cpu intensive operations, which can be handled by any modern multiprocessor CPU. Overall the project can be run easily on any laptop or PC that has a processor which is comparable to the processor I have mentioned initially.

4.2 Software Specification

The software used for this project is mainly R. Some external libraries are used in this project which can be easily downloaded from CRAN. Hence the software requirements are easily fulfilled. The project used R Shiny for deployment process which can easily be downloaded from CRAN as well.

5 BAVED Dataset

There are many researches being carried out for emotion recognition in English language. There is a lack of the same level of research being done for other languages especially from the region of middle east. One more reason for choosing this dataset in the present contest of my course CSE3506 was the nature of it being open source. This dataset is a collection is an organized collection of audio/wav recordings of Arabic words spoken in different emotional expressions.The dataset is composed of total recordings amounting to 1935 recordings in total. These are recorded by 61 speakers giving variety of speaker for good training. The gender distribution of speaker is not the porportionate, there are 45 male speakers and 16 female speakers. The scale for measuring emotion is different "0" means low emotion that is tired or exhausted, "1" for neutral emotion and "2" for high emotion which includes both happiness, joy or anger.

6 Methodology

The dataset contains the collections of voice recordings that I will use to train the model. It is not possible to directly use the voice sample as it can' t be understood by any machine learning model. Hence I would have to first extract the features from the audio samples and then create a feature matrix that can represent the actual features of the audio sample. We have listened to music, each person likes different types of music some people like classical, hip-hop, rap, electronic etc. We can differentiate between them based on parameters set by us, which we humans can describe and explain to each other. I am able to identify the sound of father's car approaching the parking out of the many cars in my building. This is possible because of our ears and the brain. The problem is how to explain these differences to a machine. How can we teach it to learn the differences between these sounds. The solution is a set of audio features, after years of ongoing research, researchers have been able to find audio features in sound that can help distinguish these sounds and characterize them to a machine. This a developing field as we are need of continuously more significant and better features to further increase the accuracy of our machines. This further establishes that this is a machine learning problem, as discussed in class, we can tell the computer that a round object with space in middle to hold things is a bowl. Then machine will tell all cups as bowls and so on. In similar manner based on audio features we are trying to capture all the different aspects of sound so that we can help train our machine to uniquely identify a person's voice out of many others. I am now describing the general process of extracting audio features. The general idea is that we have to first convert the voice to digital audio. This has already been done while recording, while digitizing we have to focus on sampling rate and quantization explained above. After doing this we have to use framing process as sampling rate is very high the actual time duration of a sample is $2.083 * 10^{-5} ms$ which cannot be perceived by our ears, as ear's time resolution is 10ms so take help of framing. Framing groups samples so that we have features 10 extraction that are similar to what our ears can perceive this is done to replicate human behavior in machine. Then we perform different computations to extract different features. Based on time-domain feature extraction or frequency domain feature extraction we have to introduce some extra mathematical functions to get accurate audio features. In frequency domain we have to apply windowing process so as to eliminate spectral leakage while using Fourier transform.

6.1 Basic Audio Features

A wave is created when the pressure of a vibrating item changes. The way sound travels is by way of a wave. As a result of the back-and-forth oscillation of the particles in the medium through which the sound wave travels, sound is a mechanical wave. The 'auditory' sense processes sound through our ears. As a result, sound is also known as audio. Audio processing is a well-researched field, with several excellent papers available. We will simply discuss extremely basic yet useful facts in this post in order to acquire an intuitive understanding.

Amplitude of a sound dictates how loud or quiet it will be. The strength of sound waves is measured by loudness. The quantity of energy emitted by a sound is measured in intensity. When you hear the same sound in a smaller space, it becomes more intense. We term sounds with a higher intensity louder in general.As a result, amplitude is a measure of energy. The bigger the amplitude of a wave, the more energy it possesses. Intensity rises in lockstep with amplitude.

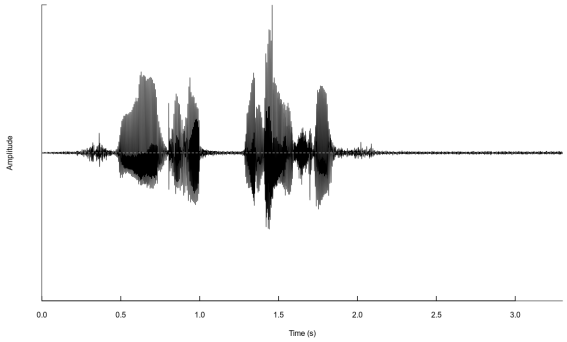

Fig.1 Amplitude time graph of low level emotion.

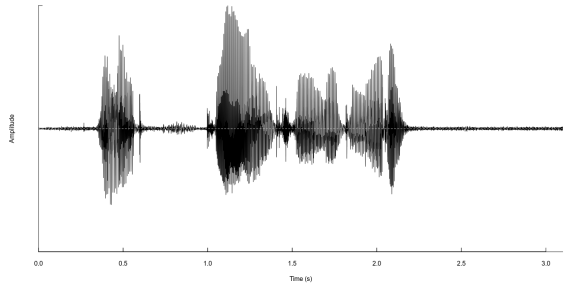

Fig.2 Amplitude time graph of neutral level emotion.

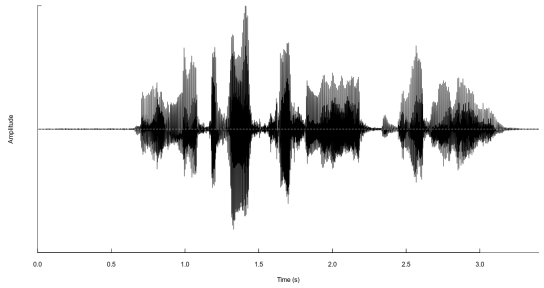

Fig.3 Amplitude time graph of high level emotion.

Frequency, often known as rate, is a metric that represents how many times an event (n) occurs within a certain time interval (t). The frequency of a sine wave is the number of times an amplitude cycle is repeated in one second. The frequency of a wave on the time axis is known as the conventional frequency or, rather generally, the frequency (f). Rather of calculating the number of cycles generated every second, it is more common to measure the length of one cycle and then compute the inverse of that length, or the period's reverse (T)

Zero Crossing Rate The amount of sign changes c of x(n) per unit of time (usually one second) is described by the Zero-Crossing Rate (ZCR). A signal with a high ZCR or MCR has a lot of high-frequency material in it. The zero crossing rate of typical harmonic signals is low, and it is proportional to the signal's fundamental frequency. A single pure sine has a zero crossing rate twice its frequency, for example.

6.2 Advanced Audio Features

Mel-Frequency Cepstral Coefficients Computing Mel-Frequency Cepstral Coefficients is the most common and widely used approach for extracting spectral characteristics (MFCC). MFCCs are one of the most widely used time frequency feature extraction methods in emotion classification techniques like that I am implementing in this project, depending on the Mel scale, which is based on the human ear scale (Tsuji et al., 2021). Frequency domain characteristics, such as MFCCs, are far more precise than time domain features.

Mel-Frequency Cepstral Coefficients (MFCC) are a characterization of a windowed brief signal's actual cepstral produced from the signal's Fast Fourier Transform (FFT). The use of a nonlinear frequency scale, that estimate the behaviour of the auditory cortex, distinguishes it from the true cepstral. Furthermore, these coefficients are accurate and robust in the face of different variants in speakers and

recording circumstances. MFCC is an audio feature extraction method that derives parameters from speech that are similar to those utilized by people when hearing speech (Paul et al., 2021) while de emphasizing all other data. Initially, the voice signal is separated into time frames, each of which contains an arbitrary number of samples. In most applications, frame overlapping is employed to create a seamless transition from one frame to the next. To avoid discontinuities at the edges, every time frame would then be windowed with something like a Hamming window.

$$W(n) = 0.54 - 0.46\cos(\tfrac{2*\pi*n}{N-1}), 0 \leq n \leq N-1s$$

The numeric variable N represents the total number of sample and n is current sample. Following windowing, the Fast Fourier Transformation (FFT) is used to extract frequency components of a signal in the time domain for each frame. For speeding up the whole process, FFT is employed. The Fourier processed frame is passed through the logarithmic Mel-Scaled filter bank. To one 1 kHz, this scale is roughly linear, and at higher frequencies, it becomes logarithmic.

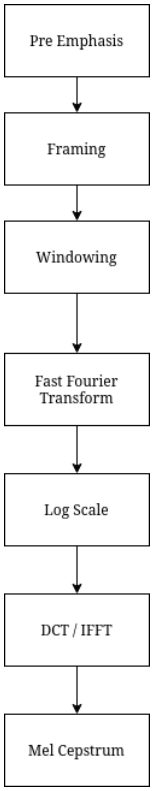

Fig.4 The process of Creation of MFCC feature.

The next step is to compute the Discrete Cosine Transformation (DCT) of the filter bank's outputs. The zeroth component is removed since it is untrustworthy, hence DCT ranges coefficients according to relevance. A collection of MFCC calculated for every speech frame. An acoustical vector is a collection of coefficients that reflects the phonetically relevant aspects of speech and is extremely valuable for any further processing and analyzing in emotion categorization.

```
path = rootPath
num_files = 0
for (i in listFiles){
len = str_length(i)
if(len-2>0) {
format_file = substr(i, len-2, len)
if(format_file == "wav") {
num_files = num_files+1 }}
}
print(num_files)
number_of_cep_features = 20
col_names <- vector(mode = "character",
length = number_of_cep_features)

for (i in 1:number_of_cep_features){
col_names[i]
=
paste("mfcc",as.character(i),sep = "")
}

col_names
[number_of_cep_features+1]="class"
matrix_features = matrix(nrow=num_files,
ncol=number_of_cep_features+1)
colnames(matrix_features)=col_names

for (i in 1:num_files){
audio = readWave(paste(rootPath
,listFiles[i],sep ="/"))
pos = str_locate_all(pattern = "_",
listFiles[i])
class = substr(listFiles[i],
pos[[1]][4]+1,
pos[[1]][4]+1)
class = as.integer(class)
```

```
mfcc = melfcc(audio, sr =
 ↪  audio@samp.rate,
wintime = 0.025,
hoptime = 0.005,
numcep = 20,
sumpower = TRUE,
nbands = 40,
bwidth = 1,
preemph = 0.60,
)
for (j in
 ↪  1:length(col_names)-1){
matrix_features[i,j]=
mean(mfcc[,j],
na.rm = TRUE)
}
matrix_features
[i,
number_of_cep_features+1]=class
}

write.csv( matrix_features,
"MFCC_FEATURES_EXTRACTED.csv")
```

Code Snippet MFCC Feature Extraction

Linear Predictive Cepstral Coefficients(LPC)

Linear Predictive Coding (LPC) is the most widely used method for simulating human voice output. It works well in clean environments but not very well in chaotic ones. Formant, Short Time Spectrum, Pitch Period, Speech Frame Energy, bandwidth are the variables for audio signal. LPC is a very essential feature utilised in auditory and speech signal processing that extracts speech properties such as spectra and pitch factors. LPC is also referred as the temporal method since it was created to be equal to the resonant parts of the human vocal cords, which formed the subsequent sound. Each speech sample is first analysed by running it through a filtration system with the purpose of spectrally flattening the signal and making it less susceptible to precision effect (Ancilin and Milton, 2021).

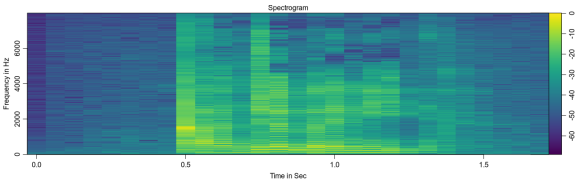

Fig.5 Spectogram Obtained for finding MFCC features.

Filter coefficients should range from 0.9 to 1. Following the pre-emphasis stage, the resultant speech is separated into frames, each of which has M samples lasting 20 to 40 seconds. To assure stillness between frames, there is a customary overlap of 10ms between two consecutive frames. The geometry of the vocal tract determines the nature of the sound produced, according to LPCC. Formant, Pitch Period and Speech Frame Energy are all frequent speech signal components, and LPCC has been one of the most essential methods for estimating them. The objective of feature is to highlight spoken signal using a limited number of signal metrics. We may obtain the LPCC coefficients with the use of LPC, which are then translated into cepstral coefficients. This method works on the principle that one voice sample at any given time can be interpreted as a linear series of previous speech samples (Palo et al., 2021). Utilizing a first order high filter, the input audio signal is pre emphasised during the first phase, which includes widowing and framing. Because lower frequencies carry greater energy than higher frequencies. Pre emphasis of the resultant signal is done to augment the energy at higher frequencies.

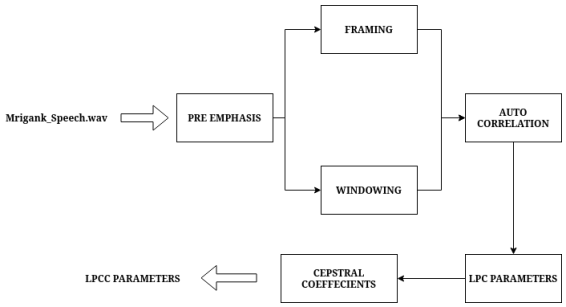

Fig.6 The process of Creation of LPCC feature.

Extracting the audio features of all voice samples in dataset gives us a way to represent each audio in matrix form of features. Using wrassp (Bombien et al., 2021) , tuneR (Ligges et al., 2018) and signal (Signal developers, 2014) packages available in CRAN, audio is read and sampled, then the above mentioned algorithm is applied to get the resultant features of MFCC and LPCC. I use the top 20 features of LPCC and top 20 features of MFCC, this gives us a matrix consisting of rows which represent audio samples and column which represent the audio features. A final class column is added to dataset for the target feature of level of emotion.

	lpc1	lpc2	lpc3	lpc4	lpc5	lpc6	
1	-1.13773817268385	1.0776790980291	-0.762141528135331	0.651205267990103	-0.771721761332468	0.94984534	
2	-1.26453539837971	1.82238546051665	-1.55861484267238	1.61573365358032	-1.50342831505977	1.5000167	
3	-1.45613129794977	2.27801375200942	-3.04116858005643	3.05791618165495	-2.83598250525784	2.489161	
4	0.319873489236586	-0.025750862919318	-0.992002664028766	-0.329572850283697	0.195816028953237	0.75283954	
5	-0.49655272334557	0.381743631987142	-0.636897189499902	0.124919649153696	-0.437420259525852	0.39621070	
6	-0.442773340119096	0.665267258573638	-0.560087750717366	0.85684879271166	-0.840627850609722	0.86526334	
7	-0.066353747149179	1.40521123928867	-0.348524955192205	1.35203848522952	-0.922964556494985	1.3342333	
8	0.360349253274583	0.478975292076512	0.119062596372691	0.328875356891249	-0.332410711347882	0.1479911	
9	-0.209053182460656	0.189241001418879	-0.442084531615587	0.393149489487628	-0.325698070500659	0.55609133	
10	-0.468645209721844	0.258389862093657	-0.689333033671749	0.665865599013273	-0.523846510117578	0.62617933	
11	0.081373292319855 4	0.356264346397696	-0.195298771770981	0.270675407752094	-0.280575962071271	0.06298932	
12	-0.414250547213136	0.309456990432917	-0.306041985415024	0.285717741909194	-0.020986243009223 9	-0.07057785	
13	0.258745243473195	0.266465223613114	-0.367352469975864	0.14380299759569	-0.3405032586519	0.02187091	
14	-1.60416555103384	2.09418786584077	-2.65167140541461	2.92475496057971	-3.19873530555466	3.45130759	
15	-2.09410847295419	3.23077281348329	-3.87471496727465	4.46591268967236	-4.88743327848707	5.38808144	
16	-1.91760888137172	2.92000432617814	-3.61120052283282	4.09225542462474	-4.39892452683251	4.65578164	
17	-0.388407842917 34	-0.005242019143873 26	-0.269920479255858	0.093518920416 76	-0.245241166969286	0.22087988	
18	-0.701504422173747	0.141062831009766	0.130655585448 46	-0.848332993909005	0.385826950455272	0.21111989	
19	0.384061948232792	0.156360058809664	-0.118249725430398	-0.084350345442770 5	-0.10103098949681	0.10691009	

Fig.7 Subset of final dataset that has all the features.

The next step is to use this feature dataset to train various machine learning models, to accurately classify the class of emotion.

Support Vector Machine:

This is perhaps the most modern methods for supervised machine learning. Support Vector Machine (SVM) models are similar to multilayer perceptron neural networks in that they use support vectors. The concept of a margin on either side of a hyperplane that separate the two data into two classes is important to SVMs. It has been demonstrated that increasing the margin and therefore generating the biggest possible space between the separating hyperplane and the instances on either side of it reduces the expected generalization error.

Random Forest:

Random forest is a supervised classification algorithm that is commonly used to solve regression and classification problems. It creates decision trees from various samples, using the simple majority for categorization and the mean for regression. Among the most essential characteristics of the Random Forest Algorithm is that one can accommodate data sets with both categorical and continuous, as in classification and regression problems.

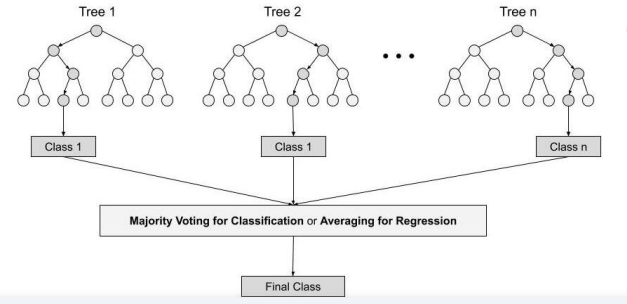

Fig.8 Random Forest Classification and Regression Algorithm. (E R, 2021)

When it comes to categorization difficulties, it outperforms the competition. In Random Forest, 'M' randomized records are selected at random from a data set with 'K' records. For every sample, a unique decision tree is built. Every decision tree would produce a result (Pal, 2005). For regression and classification, the final result is dependent on simple majority or finding the mean of all the results. In this way we can handle both classification problems and in case of variables that are continuous we can also accommodate regression problems.

Linear Discriminant Analysis (LDA) :

LDA is a technique used to separate data points by understanding associations among high-dimensional data points as well as the learner vector. It takes high-dimensional data and converts it to linear-dimensional data. It can also be used for data visualisation, categorization, and dimension reduction. Due to its simple structure, the LDA approach frequently yields reliable, good, and understandable classification results. Linear Discriminant Analysis (LDA) is an approach for categorising binary and non-binary features by establishing the connection between the dependent and independent parameters using a linear algorithm (AbuZeina and Al-Anzi, 2018). It employs the Fischer formula to minimize the data's dimensionality and fit it into a linear dimension. LDA is a classification algorithm, dimensionality reducer, and can also be used as a data visualizer all in one method. The goal of LDA is to reduce inter-class variance by grouping as many comparable data points as feasible into a single class. There will be reduced misclassifications as a result of this. It also seeks to optimise the distance between the mean of classes, with the mean positioned as far away as feasible to achieve high predictions confidence.

7 Results

7.1 SVM Radial Kernel

The performance of SVM is best with radial kernel as it is able to represent the multidimensional feature of data using radial kernel technique. Although the accuracy of model is not that high. This is due to the fact that most of data points can not be separated that easily by a single maximal margin hyperplane. Hence there is a lot of miss-classifications error in this. The model was able to get a maximum accuracy of 69.98%.

7.2 Linear Discriminant Analysis

The performance of LDA is not very good and it perform worst compared to other model that I tried. This is due to the fact that number of features in my datasets are very high. Even a single feature can act as a separating factor for the classes. The approach

Table 2: Evaluation metrics of SVM model with Radial Kernel

Evaluation Metr.	Class:0	Class:1	Class:2
Sensitivity	0.6027	0.7321	0.7515
Specificity	0.8783	0.8063	0.8631
Pos Pred Value	0.6822	0.6685	0.7471
Neg Pred Value	0.8362	0.8495	0.8658
Prevalence	0.3023	0.3478	0.3499
Detection Rate	0.1822	0.2547	0.2629
Detection Preval.	0.2671	0.3810	0.3520
Balanced Acc.	0.7405	0.7692	0.8073

Table 3: Results of SVM model with Radial Kernel

Metrics	Value
Accuracy	0.6998
95% CI	(0.6567, 0.7404)
No Information Rate	0.3499
P-Value [Acc > NIR]	2e-16
Kappa	0.5476
Mcnemar's Test P-Value	0.06448

of transforming all these features into a single line for better separability is not a good idea. As model is not able to capture information of all the feature when it transforms them to a single dimension.

Table 4: Evaluation metrics of Linear Discriminant Analysis model.

Evaluation Metr.	Class:0	Class:1	Class:2
Sensitivity	0.5137	0.6429	0.4852
Specificity	0.8338	0.6540	0.8312
Pos Pred Value	0.5725	0.4977	0.6074
Neg Pred Value	0.7983	0.7744	0.7500
Prevalence	0.3023	0.3478	0.3499
Detection Rate	0.1553	0.2236	0.1698
Detection Prev.	0.2712	0.4493	0.2795
Balanced Acc.	0.6738	0.6484	0.6582

7.3 Random Forest

Random forest performs the best among all the models. The model is able to get an accuracy of 100%. The reason for its good performance is rule based approach that is able to select the best feature which provide the highest information gain at each node split. As it is an ensemble model as well, we are able to use many decision trees that use different subset of features for building. Then when majority algorithm is applied for final class

Table 5: Results of Linear Discriminant Analysis model.

Metrics	Value
Accuracy	0.5487
95% CI	(0.5031, 0.5937)
No Information Rate	0.3499
P-Value [Acc > NIR]	2.2e-16
Kappa	0.3202
Mcnemar's Test P-Value	0.001641

value selection, it is able to provide highly accurate results. A sample image of the model when it is

Table 6: Evaluation metrics of Random Forest model.

Evaluation Met.	Class:0	Class:1	Class:2
Sensitivity	1.000	1.000	1.000
Specificity	1.000	1.000	1.000
Pos Pred Value	1.000	1.000	1.000
Neg Pred Value	1.000	1.000	1.000
Prevalence	0.302	0.348	0.349
Detection Rate	0.302	0.348	0.349
Detection Preval.	0.302	0.348	0.349
Balanced Acc.	1.000	1.000	1.000

Table 7: Results of Random Forest model.

Metrics	Value
Accuracy	1
95% CI	(0.9924, 1)
No Information Rate	0.3499
P-Value [Acc > NIR]	2.2e-16
Kappa	1
Mcnemar's Test P-Value	NA

deployed using R Shiny is provided below. The used can upload the audio samples using the GUI provided, the app will then print the emotion of the audio samples.

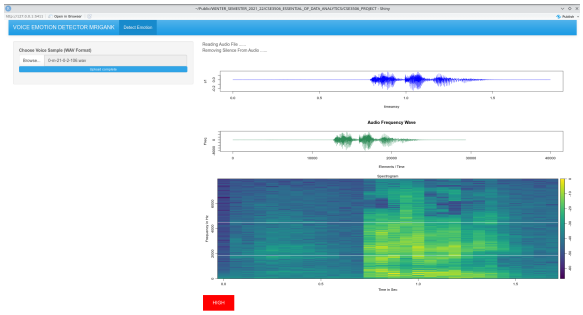

Fig 9 Emotion Classification app deployed on R Shiny.

8 Conclusion and Future Work

I was able to get an accuracy of 100% using random forest model, which is very good when compared with existing models for BAVED dataset. The immense potential of machine learning and deep learning can take emotion classification to a new level. I have used only a single language and the audio samples are not super long. A similar model can be applied for large audio files, or continuous audio streaming, where each chunk of audio can be labeled as a one emotion type. We can also gather a larger dataset which has a noisy recordings as well, so that we can deploy this model in real time systems as well. Audio samples from different languages can also be utilized which will make this model truly robust and global in approach.

References

Dia AbuZeina and Fawaz S Al-Anzi. 2018. Employing fisher discriminant analysis for arabic text classification. *Computers & Electrical Engineering*, 66:474–486.

J Ancilin and A Milton. 2021. Improved speech emotion recognition with mel frequency magnitude coefficient. *Applied Acoustics*, 179:108046.

A. Aouf. 2019. Basic arabic vocal emotions dataset (baved).

Surekha Reddy Bandela and T Kishore Kumar. 2017. Stressed speech emotion recognition using feature fusion of teager energy operator and mfcc. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5. IEEE.

Lasse Bombien, Raphael Winkelmann, and Michel Scheffers. 2021. *wrassp: an R wrapper to the ASSP Library*. R package version 1.0.1.

Shruthi E R. 2021. Random forest — introduction to random forest algorithm.

Mohamed El-Geish. 2021. Wav2vec2-large-xlsr-53-arabic hugging face.

Berat A Erol, Abhijit Majumdar, Patrick Benavidez, Paul Rad, Kim-Kwang Raymond Choo, and Mo Jamshidi. 2019. Toward artificial emotional intelligence for cooperative social human–machine interaction. *IEEE Transactions on Computational Social Systems*, 7(1):234–246.

Wenjing Han, Tao Jiang, Yan Li, Björn Schuller, and Huabin Ruan. 2020. Ordinal learning for emotion recognition in customer service calls. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6494–6498. IEEE.

Katie Hoemann, Rachel Wu, Vanessa LoBue, Lisa M Oakes, Fei Xu, and Lisa Feldman Barrett. 2020. Developing an understanding of emotion categories: Lessons from objects. *Trends in Cognitive Sciences*, 24(1):39–51.

Uwe Ligges, Sebastian Krey, Olaf Mersmann, and Sarah Schnackenberg. 2018. *tuneR: Analysis of Music and Speech*.

Omar Mohamed and Salah A Aly. 2021. Arabic speech emotion recognition employing wav2vec2. 0 and hubert based on baved dataset. *arXiv preprint arXiv:2110.04425*.

Fatemeh Noroozi, Marina Marjanovic, Angelina Njegus, Sergio Escalera, and Gholamreza Anbarjafari. 2017. Audio-visual emotion recognition in video clips. *IEEE Transactions on Affective Computing*, 10(1):60–75.

Mahesh Pal. 2005. Random forest classifier for remote sensing classification. *International journal of remote sensing*, 26(1):217–222.

Hemanta Kumar Palo, Niharika Pattanaik, Bibhu Prasad Mohanty, and Laxmi Prasad Mishra. 2021. Effect of feature dimension on classification of speech emotions. In *2021 1st Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON)*, pages 1–5. IEEE.

Bachchu Paul, Somnath Bera, Rakesh Paul, and Santanu Phadikar. 2021. Bengali spoken numerals recognition by mfcc and gmm technique. In *Advances in Electronics, Communication and Computing*, pages 85–96. Springer.

Martin Ragot, Nicolas Martin, Sonia Em, Nico Pallamin, and Jean-Marc Diverrez. 2017. Emotion recognition using physiological signals: laboratory vs. wearable sensors. In *International Conference on Applied Human Factors and Ergonomics*, pages 15–22. Springer.

Signal developers. 2014. *signal: Signal processing*.

Jessie Sun, H Andrew Schwartz, Youngseo Son, Margaret L Kern, and Simine Vazire. 2020. The language of well-being: Tracking fluctuations in emotion experience through everyday speech. *Journal of Personality and Social Psychology*, 118(2):364.

Toshiaki Tsuji, Koyo Sato, and Sho Sakaino. 2021. Contact feature recognition based on mfcc of force signals. *IEEE Robotics and Automation Letters*, 6(3):5153–5158.