Emotion Recognition through Speech Analysis using various Deep Learning Algorithms

Aayushi Arora[1], Harshada Jadhav[2], Prachi Palkar[3]

[1,2,3] Usha Mittal Institute of Technology SNDT Women's University, Mumbai
Guide:- Prof. Mohan Bonde, Usha Mittal Institute of Technology SNDT Women's University, Mumbai

Abstract:

Emotions are reactions or the feeling that everyone has and it plays an important role in human life. Emotions are reflected from speech, hand and gestures of the body and through facial expressions. But now-a-days understanding the emotions has become a challenge, dictating the voice or speech and finding the feeling of the person whether he is happy, sad or angry is important to have a healthy communication between human and machine. In order to build an intelligent machine, it's necessary to understand the emotion. To overcome this problem and to develop a strong interaction between human and machine we are introducing a speech emotion recognition system which will recognise the emotions beside the voice through speech analysis using various algorithms. Also, we will compare the accuracy of these two algorithms. In this project, we have considered seven emotions such as Neutral, Happy, Sad, Angry, Fearful, Disgusted, and Surprised.

Keywords: *Healthy Communication, RNN, SVC, Emotion Recognition, Speech Analysis, CNN, Feature extraction, Confusion Matrix, RandomForest.*

Introduction:

The fact that voice often reflects the underlying emotion through tone and pitch can surely be capitalized. This is the same way that animals understand humans. It is basically a technology that extracts emotional features from speech signals. There are few universal emotions that mostly include happiness, sadness, anger, neutral etc. in which any intelligent system with a finite number of resources can identify or synthesize as per requirement. Emotion recognition in speech is a topic on which little research has been done till date. In this project, we discuss why emotion recognition using speech is a significant and applicable project topic, and present a system for emotion recognition using CNN algorithm and diarization technique. We have tested seven emotions which are 'Neutral', 'Happy', 'Sad', 'Angry', 'Fearful', 'Disgusted', 'Surprised'. Recently, the researchers have introduced a various number of deep neural networks (DNNs) techniques to model the emotions recognition in speech.

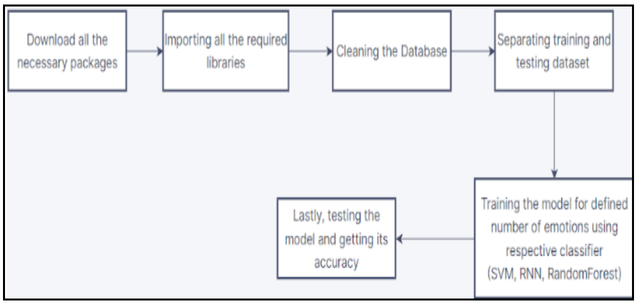
Related Work:

1. Chenchen Huang, Wei Gong, Wenlong Fu, and Dongyu Feng presented A Research of Speech Emotion Recognition
2. Chengwei Huang,1Ruiyu Liang, Qingyun Wang, Ji Xi, Cheng Zha, and Li Zhao proposed Practical Speech Emotion Recognition Based on Online Learning: From Acted Data to Elicited Data
3. Han et al. worked on Speech Emotion Recognition Using Deep Neural Network And extreme machine learning algorithms
4. Huang et al. speech emotion recognition using CNN

Abbreviations:

CNN- Convonutional Neural Network
RNN- Recurrent Neural Network
SVC- Support Vector Classifier
MFCC- Mel Frequency Cepstral Coefficient
Conv2D Convolutional 2D
CRF- Conditional Random Field

Methodology:


Download all the necessary packages
Importing all the required libraries
Cleaning the Database
Separating training and testing dataset
Training the model for defined number of emotions using respective classifier (SVM, RNN, RandomForest)
Lastly, testing the model and getting its accuracy

Step 1: Downloading necessary packages. For this project import the following packages such as librosa, numpy, pandas, soundfile, wave, sklearn, tqdm, matplotlib, libasound2- dev, portaudio19-dev libportaudio2, libportaudiocpp0 ffmpeg, Pyaudio, tensorflow.

Step 2: Importing the required libraries Importing all necessary libraries from all downloaded packages.

Step 3: Cleaning the dataset

We download and convert the dataset to be suited for extraction.

The process is about loading the dataset in Python which involves extracting audio features, such asobtaining different features such aspower, pitch, and vocal tract configuration from the speech signal, we will use librosa library to do that.

Step 4: Separating the training and testing datasets. We split the modelling dataset into training and testing sets is to assign 2/3 data points to the former and the remaining one-third to the latter. Therefore, we train the model using the training set and then apply the model to the test set. In this way, we can evaluate the performance of our model.

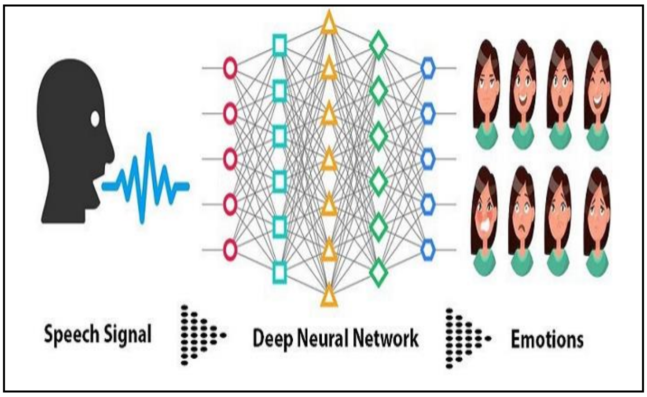
- Train Dataset: Used to fit the machine learning model.
- Test Dataset: Used to evaluate the fit machine learning model.

Step 5: Training the model for defined number of emotions using respective classifier (SVM and RNN). Here we are going to train the model for the given datasets for defined number of emotions for a particular model and will get the accuracy score for training the dataset

Step 6: Lastly, Testing the model and getting the accuracy. We finally test the model using the given dataset for a particular classifier and getting the accuracy score for recognising the emotions of speech.We are even going to compare the accuracy of each model to get best one.

Algorithms Used:

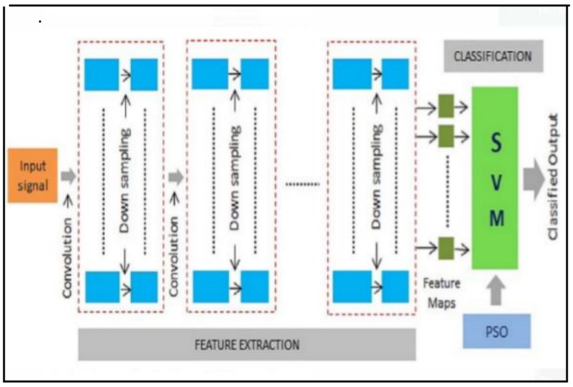
(1) Convolutional Neural Network (CNN): Convolutional Neural Network is used to perform the feature learning and classification, and CRFs are used for the decoding stage.



A convolutional neural network (CNN) contains one or more convolutional layers, pooling or fully connected, and uses a variation of multi-layer perceptrons discussed above. Convolutional layers use a convolution operation to the input passing the result to the next layer. This operation allows the network to be deeper with much fewer parameters.

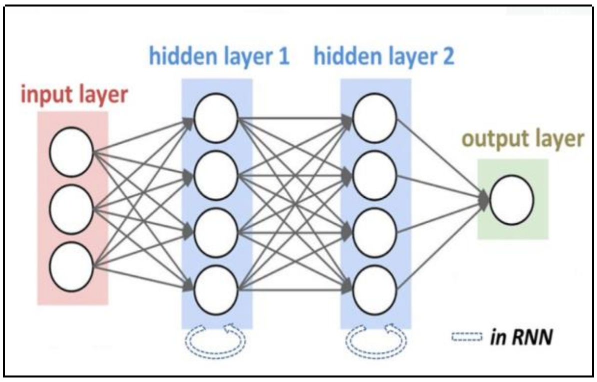
(2) Support Vector Machine (SVM):

SVM algorithms separate the training data in feature space by a hyperplane defined by the type of kernel function used. They find the hyperplane of maximal margin, defined as the sum of the distances of the hyperplane from the nearest data point of each of the two classes. Statistical learning theory shows that generalization performance of a hyperplane depends only on its margin (which bounds the VC-dimension of the hyperplane), not on the dimensionality of the embedding space.



(1) Recurrent Neural Network(RNN):

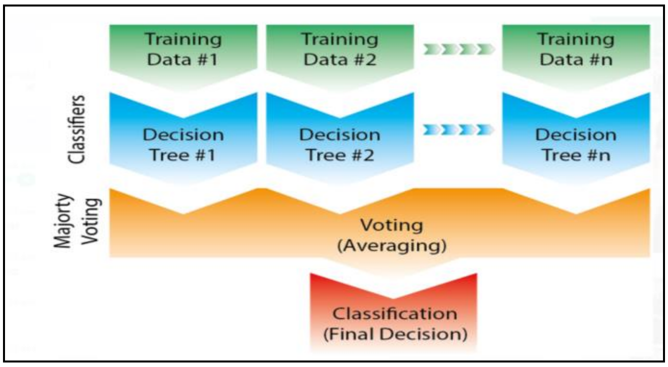
Recurrent neural networks (RNN) are suitable for learning time series data, and it has shown improved performance for classification task.While RNN models are effective at learning temporal correlations, they suffer from the vanishing gradient problem which increases with the length of the training sequences.



To resolve this problem, long short-term memory (LSTM) RNNs were proposed by Hochreiter et al.it uses memory cells to store information so that it can exploit long-range dependencies in the data.

(4) RandomForest Classifier:

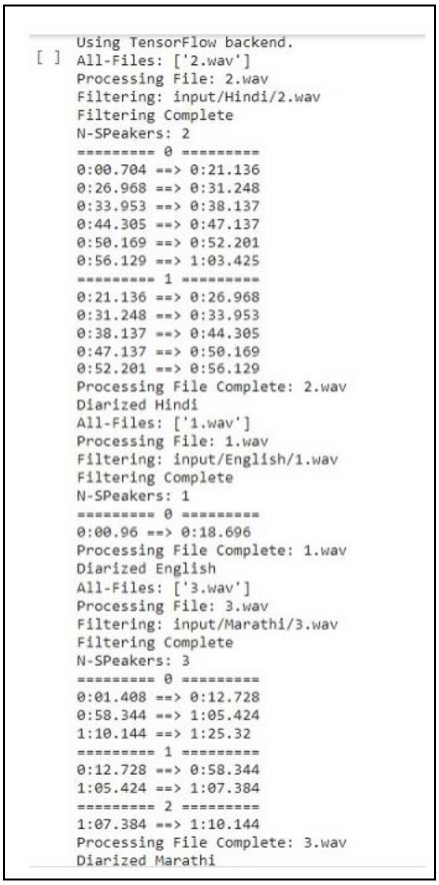
Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.



One of the most important features of the RandomForest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification. It performs better results for classification problems.

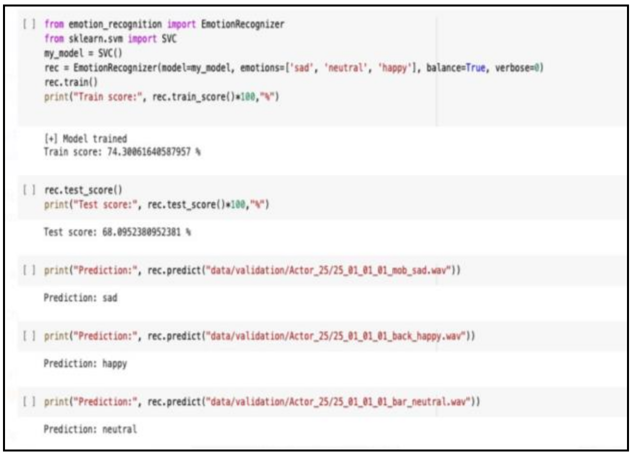
Result and Discussion:

The output for CNN Algorithm is shown below.
The number of speakers and the duration of each speaker is also shown with predicted emotion.



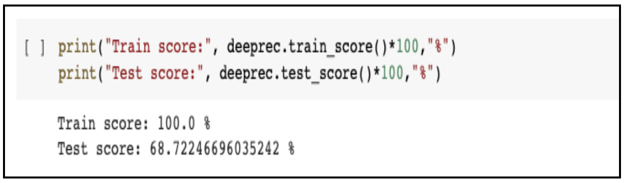
```
    Using TensorFlow backend.
[ ] All-Files: ['2.wav']
    Processing File: 2.wav
    Filtering: input/Hindi/2.wav
    Filtering Complete
    N-SPeakers: 2
    ========= 0 =========
    0:00.704 ==> 0:21.136
    0:26.968 ==> 0:31.248
    0:33.953 ==> 0:38.137
    0:44.305 ==> 0:47.137
    0:50.169 ==> 0:52.201
    0:56.129 ==> 1:03.425
    ========= 1 =========
    0:21.136 ==> 0:26.968
    0:31.248 ==> 0:33.953
    0:38.137 ==> 0:44.305
    0:47.137 ==> 0:50.169
    0:52.201 ==> 0:56.129
    Processing File Complete: 2.wav
    Diarized Hindi
    All-Files: ['1.wav']
    Processing File: 1.wav
    Filtering: input/English/1.wav
    Filtering Complete
    N-SPeakers: 1
    ========= 0 =========
    0:00.96 ==> 0:18.696
    Processing File Complete: 1.wav
    Diarized English
    All-Files: ['3.wav']
    Processing File: 3.wav
    Filtering: input/Marathi/3.wav
    Filtering Complete
    N-SPeakers: 3
    ========= 0 =========
    0:01.408 ==> 0:12.728
    0:58.344 ==> 1:05.424
    1:10.144 ==> 1:25.32
    ========= 1 =========
    0:12.728 ==> 0:58.344
    1:05.424 ==> 1:07.384
    ========= 2 =========
    1:07.384 ==> 1:10.144
    Processing File Complete: 3.wav
    Diarized Marathi
```

Our SVC model achieves 64.28% of training score and 75.31% of testing score. Below figure shows the accuracy along with the type of emotions being recognised.



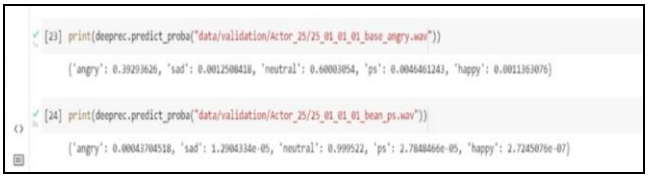
```
[ ] from emotion_recognition import EmotionRecognizer
    from sklearn.svm import SVC
    my_model = SVC()
    rec = EmotionRecognizer(model=my_model, emotions=['sad', 'neutral', 'happy'], balance=True, verbose=0)
    rec.train()
    print("Train score:", rec.train_score()*100,"%")

    [+] Model trained
    Train score: 74.30061640587957 %

[ ] rec.test_score()
    print("Test score:", rec.test_score()*100,"%")

    Test score: 68.0952380952381 %

[ ] print("Prediction:", rec.predict("data/validation/Actor_25/25_01_01_01_mob_sad.wav"))

    Prediction: sad

[ ] print("Prediction:", rec.predict("data/validation/Actor_25/25_01_01_01_back_happy.wav"))

    Prediction: happy

[ ] print("Prediction:", rec.predict("data/validation/Actor_25/25_01_01_01_bar_neutral.wav"))

    Prediction: neutral
```

Below figure shows the testing and training score of RNN Algorithm



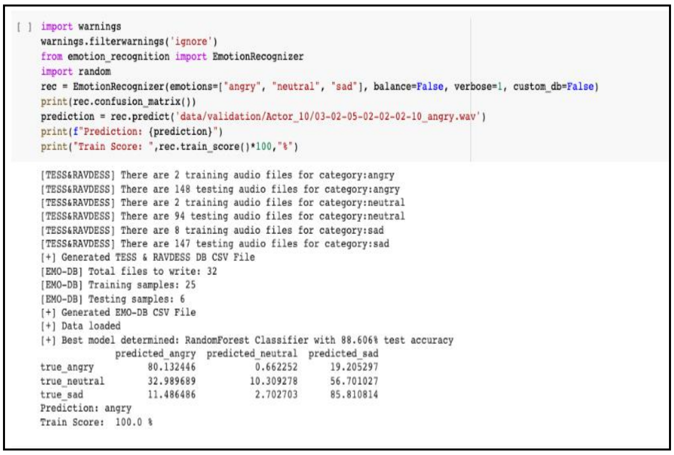
```
[ ] print("Train score:", deeprec.train_score()*100,"%")
    print("Test score:", deeprec.test_score()*100,"%")

    Train score: 100.0 %
    Test score: 68.72246696035242 %
```

Following figure shows the probability of every emotion using RNN algorithm



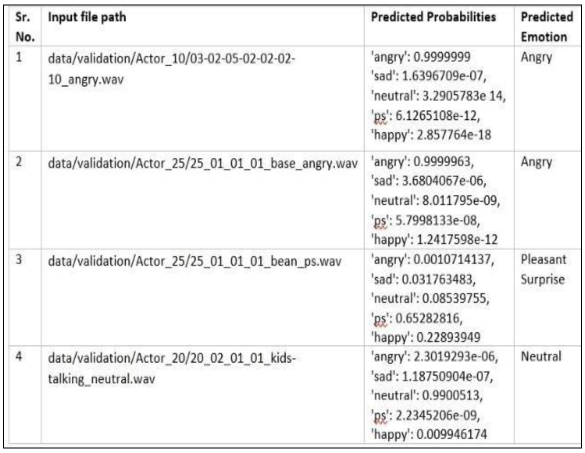
```
[23] print(deeprec.predict_proba("data/validation/Actor_25/25_01_01_01_base_angry.wav"))

     {'angry': 0.39293626, 'sad': 0.0012508418, 'neutral': 0.60003054, 'ps': 0.0046461243, 'happy': 0.0011363076}

[24] print(deeprec.predict_proba("data/validation/Actor_25/25_01_01_01_bean_ps.wav"))

     {'angry': 0.00043704518, 'sad': 1.2904334e-05, 'neutral': 0.999522, 'ps': 2.7848466e-05, 'happy': 2.7245076e-07}
```

The below figure shows the scored for RandomForest Classifier



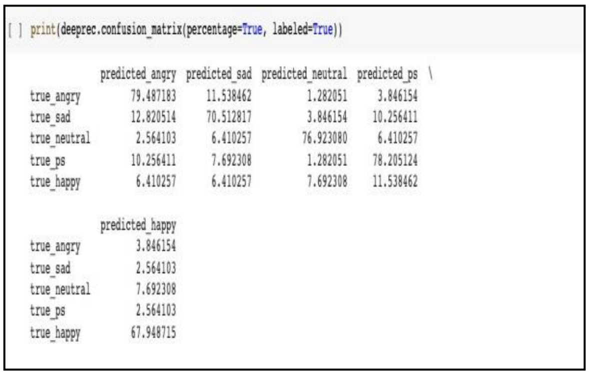
```
[ ] import warnings
    warnings.filterwarnings('ignore')
    from emotion_recognition import EmotionRecognizer
    import random
    rec = EmotionRecognizer(emotions=["angry", "neutral", "sad"], balance=False, verbose=1, custom_db=False)
    print(rec.confusion_matrix())
    prediction = rec.predict('data/validation/Actor_10/03-02-05-02-02-02-10_angry.wav')
    print(f"Prediction: {prediction}")
    print("Train Score: ",rec.train_score()*100,"%")

    [TESS&RAVDESS] There are 2 training audio files for category:angry
    [TESS&RAVDESS] There are 148 testing audio files for category:angry
    [TESS&RAVDESS] There are 2 training audio files for category:neutral
    [TESS&RAVDESS] There are 94 testing audio files for category:neutral
    [TESS&RAVDESS] There are 8 training audio files for category:sad
    [TESS&RAVDESS] There are 147 testing audio files for category:sad
    [+] Generated TESS & RAVDESS DB CSV File
    [EMO-DB] Total files to write: 32
    [EMO-DB] Training samples: 25
    [EMO-DB] Testing samples: 6
    [+] Generated EMO-DB CSV File
    [+] Data loaded
    [+] Best model determined: RandomForest Classifier with 88.606% test accuracy
                  predicted_angry  predicted_neutral  predicted_sad
    true_angry          80.132446           0.662252      19.205297
    true_neutral        32.989689          10.309278      56.701027
    true_sad            11.486486           2.702703      85.810814
    Prediction: angry
    Train Score:  100.0 %
```

Below table shows the probability of different emotions in the respective audio file



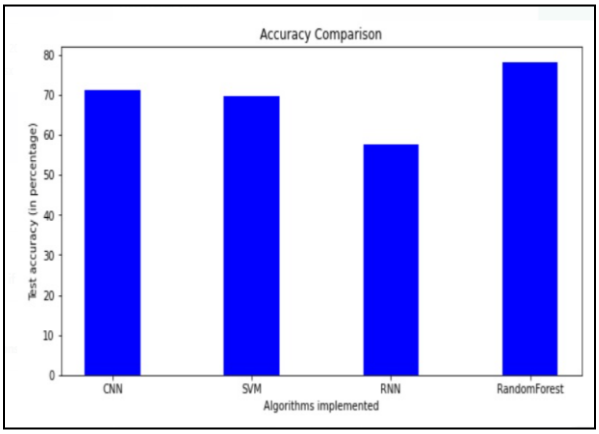
Sr. No.	Input file path	Predicted Probabilities	Predicted Emotion
1	data/validation/Actor_10/03-02-05-02-02-02-10_angry.wav	'angry': 0.9999999 'sad': 1.6396709e-07, 'neutral': 3.2905783e 14, 'ps': 6.1265108e-12, 'happy': 2.857764e-18	Angry
2	data/validation/Actor_25/25_01_01_01_base_angry.wav	'angry': 0.9999963, 'sad': 3.6804067e-06, 'neutral': 8.011795e-09, 'ps': 5.7998133e-08, 'happy': 1.2417598e-12	Angry
3	data/validation/Actor_25/25_01_01_01_bean_ps.wav	'angry': 0.0010714137, 'sad': 0.031763483, 'neutral': 0.08539755, 'ps': 0.65282816, 'happy': 0.22893949	Pleasant Surprise
4	data/validation/Actor_20/20_02_01_01_kids-talking_neutral.wav	'angry': 2.3019293e-06, 'sad': 1.18750904e-07, 'neutral': 0.9900513, 'ps': 2.2345206e-09, 'happy': 0.009946174	Neutral

Confusion Matrix



```
[ ] print(deeprec.confusion_matrix(percentage=True, labeled=True))

             predicted_angry  predicted_sad  predicted_neutral  predicted_ps  \
true_angry         79.487183      11.538462           1.282051      3.846154
true_sad           12.820514      70.512817           3.846154     10.256411
true_neutral        2.564103       6.410257          76.923080      6.410257
true_ps            10.256411       7.692308           1.282051     78.205124
true_happy          6.410257       6.410257           7.692308     11.538462

             predicted_happy
true_angry          3.846154
true_sad            2.564103
true_neutral        7.692308
true_ps             2.564103
true_happy         67.948715
```

The below graph is the comparison of the accuracies of the algorithms implemented



Applications:

1. Improves Human computer interaction:

The emotion recognition system should be applied in different kinds of the Human computer interaction systems, such as dialogue systems, automatic answering systems and human robots etc. A system that is based on the user's emotion, makes human computer interaction synchronized.

2. Call centre:

The voice call centre is a tool that helps operators and supervisors to visualize emotional content of voice messages provided by users which will help to know the emotions of customer and in turn improve the services that are served by the company.

3. Student's voice review:

Emotion recognition model can be used in education field as to understand whether students enjoy the learning process or not. Teacher can ask students to provide their voice reviews which then can be analysed by the system. This helps teacher to know where the improvement is needed and accordingly plan their future lectures.

4. Helps to build healthy relationship:

In social media most of the people fears to express their emotions. Here the emotion recognition system helps to find their hidden emotions through their own posted videos which in turn detects emotions behind their voice.

Conclusion:

Understanding different emotions in today's world has become an important aspect of human life in order to recognize what a person is going through. Sometimes people don't like to share with the world what they are going through. To overcome this problem, they can record their voice personally which is then analysed through our model in order to identify the emotions behind their voice and accordingly solution can be found based on the emotion of a person. We have trained the model using CNN, SVC, RNN, RandomForest and also tested it against the dataset. We were able to achieve some good amount of accuracy. We were also able to find confusion matrix for the same.

We can conclude that Random Forest is one of the best techniques with high performance which is widely used in various industries for its efficiency. It can handle binary, continuous, and categorical data. Random forest is a great choice if anyone wants to build the model fast and efficiently as one of the best things about the random forest is it can handle missing values. Overall, random forest is a fast, simple, flexible, and robust model.

Future Scope:

We are planning to add real-time emotion detection to our project and get the number of samples detected for different emotions for training and testing.

References:

1. Chenchen Huang[2014, August 12] A Research of Speech Emotion Recognition
2. Reza Chu[2019, June 1] Speech Emotion Recognition with Con-volutional Network Teddy Surya Gunawan, Muhammad
3. Fahreza Alghifari, Malik Arman Morshidi, Mira Kartiwi, A Review on Emotion RecognitionAlgorithms using Speech Analysis
4. Tiya Maria Joshy, Dr. Anjana S Chandran [3 March 2020]
5. International journal of creative research thoughts
6. Adrian Rosebrock [December 31, 2018] pyimagesearch V.Priya, Nimisha, Komal [2020] Emotion recognition from text

Acknowledgement:

We have a great pleasure to express our gratitude to all those who have contributed and motivated during our project work. We would like to extend our sincere gratitude towards Prof. Mohan Bonde for guiding us throughout the project. We are grateful toProf. Anita Morey for helping us.

Authors:

Aayushi Arora

Prachi Palkar

Harshada Jadhav