

Speech Emotion Recognizer Using Machine Learning

Madhura Mohod¹, Nisarga Mhaisane¹,

Aditi Thakare¹, Sakshi Nichat¹

47, Samarpan Colony, Near Pathyapustak Mandal, V. M.

Area, Amravati, Maharashtra 444604.

Prof. Vaishali R. Thakare²

Sai Nagar, Amravati

¹Student, Computer Science and Engineering, P. R. Pote Patil College of Engineering and Management, Amravati, India.

²Assistant Professor, Computer Science and Engineering, P. R. Pote Patil College of Engineering and Management, Amravati, India

Abstract

In this project, a deep learning approach for emotion classification using speech data from different modalities is performed. A convolutional neural network(CNN) that captures discriminative information from audio features is trained to correctly classify the emotion labels. Further experiments also include experimenting with different network architectures of each individual model, using regularization strategies such as dropout, etc. To ensure that the trained model performs well on an unknown audio sample, different from the samples used for training and testing, audio samples from a completely separate dataset are collected and tested using the trained model. To ensure that the results obtained are indeed true, the Kaggle kernel1 used for training the models is also made public. Emotion recognition from speech signals is an important but challenging component of Human-Computer Interaction (HCI). In the literature of speech emotion recognition (SER), many techniques have been utilized to extract emotions from signals, including many well-established speech analysis and classification techniques. Deep Learning techniques have been recently proposed as an alternative to traditional techniques in SER. This paper presents an overview of Deep Learning techniques and discusses some recent literature where these methods are utilized for speech-based emotion recognition. The review covers databases used, emotions extracted, contributions made toward speech emotion recognition and limitations related to it. The expression of emotions in human communication plays a very important role in the information that needs to be conveyed to the partner. The forms of expression of human emotions are very rich. It could be body language, facial expressions, eye contact, laughter, and tone of voice. The languages of the world's peoples are different, but even without understanding a language in communication, people can almost understand part of the message that the other partner wants to convey with emotional expressions as mentioned. Among the forms of human emotional expression, the expression of emotions through voice is perhaps the most studied.

Keywords— Speech Emotion Recognizer, CNN, Machine Learning, Python, Support Vector Machine(SVM).

Introduction

This project is an effort in the above direction. Neural Networks which learn to automatically classify emotions from spectral features are made use of. Accuracies reported on the test set for various neural network architectures, show that they perform very well on the test set, thereby correctly predicting emotions from audio samples. The rest of the report is organized as follows, Section 2 talks about the dataset, Section 3 talks about the modifications to the initial proposal, Section 4 discusses the features and models used, Section 5 talks about the hyper parameters for each model, Section 6 talks about various setbacks encountered over the course of the project and their solutions, Section 7 discusses the results, Section 8 discusses if the model actually works, Section 9 provides instructions on testing the trained networks, Section 10 lists some of the conclusions of the project.

Human emotions help individuals respond to different situations such as dealing with others in the best possible way and thereby avoiding conflicts or confrontations. Thus, understanding human emotions plays an important role in improved interaction amongst people. Automating this task, may help learners improve their social interaction skills and help them achieve their goals. This project is an effort in the above direction. Neural Networks which learn to automatically classify emotions from spectral features are made use of. Accuracies reported on the test set for various neural network architectures, show that they perform very well on the test set, thereby correctly predicting emotions from audio samples. The rest of the report is organized as follows, Section 2 talks about the dataset, Section 3 talks about the modifications to the initial proposal, Section 4 discusses the features and models used, Section 5 talks about the hyper parameters for each model, Section 6 talks about various setbacks encountered over the course of the project and their solutions, Section 7 discusses the results, Section 8 discusses if the model actually works, Section 9 provides instructions on testing the trained networks, Section 10 lists some of the conclusions of the project.

Deep Learning techniques for SER have several advantages over traditional methods, including their capability to detect the complex structure and features without the need for manual feature extraction and tuning; tendency toward extraction of low-level features from the given raw data, and ability to deal with unlabeled data. Deep Neural Networks (DNNs) are based

on feed-forward structures comprised of one or more underlying hidden layers between inputs and outputs. The feed-forward architectures such as Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs) provides efficient results for image and video processing. On the other hand, recurrent architectures such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) are much effective in speech-based classification such as natural language processing (NLP) and SER. Apart from their effective way of classification these models do have some limitations. For instance, the positive aspect of CNNs is to learn features from high-dimensional input data, but on the other hand, it also learns features from small variations and distortion occurrence and hence, requires large storage capability. Similarly, LSTM-based RNNs are able to handle variable input data and model long-range sequential text data.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g.” Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

Related Work

Complete review on the speech emotion recognition is explained in which reviews properties of dataset, speech emotion recognition study classifier choice. Various acoustic features of speech are investigated and some of the classifier methods are analyzed in which is helpful in the further investigation of modern methods of emotion recognition. This paper investigated the prediction of the next reactions from emotional vocal signals based on the recognition of emotions, using different categories of classifiers. Some of the classification algorithms like K-NN, Random Forest are used in to classify emotion accordingly. Recurrent Neural network arises enormously which tries to solve many problems in the filed of data science. Deep RNN like LSTM, Bi-directional LSTM trained for acoustic features are used in. Various range of CNN are being implemented and trained for speech emotion recognition are evaluated in. Emotion is inferred from speech signals using filter banks and Deep CNN which shows high accuracy rate which gives an inference that deep learning can also be used for emotion detection. Speech emotion recognition can be also performed using image spectrograms with deep convolutional networks which is implemented in. Here, Xinzhou Xu et al generalized the Spectral Regression model exploiting the joins of Extreme Learning Machines (ELMs) and Subspace Learning (SL) was expected for overlooking the disadvantages of spectral regression based Graph Embedding (GE) and ELM. Using the GSR model, in the execution of Speech Emotion Recognition (SER) we had to precisely represent theses relations among data. These multiple embedded graphs were constructed for the same. Demonstration over 4 Speech Emotional **Corpora** determined that the impact and feasibility of the techniques compared to prior methods that includes ELM and Subspace Learning

(SL) techniques. The system output can be improved by exploring embedded graphs at more precise levels. Only Least-Square Regression along with l2-norm minimization was considered in the regression stage. Zhaocheng Huang et al uses a heterogeneous token-used system to detect the speech depression. Abrupt Changes and acoustic areas are solely and collectively figured out in joins among different embedding methods. Contributions towards the detection of depression were used and probably various health problems that would affects vocal generation. Landmarks are used to pull out the information particular to individual type of articulation at a time. This is a hybrid

Sr No.	Name of Paper	Year	Method used	Summary
1]	Human Speech Emotion Recognition by using Machine Learning	2021	MFCC, CNN	Accuracy: 79.4% Precision: 60.6%
2]	Speech Emotion Detection using Machine Learning Techniques	2018	SVM, MFCC	Accuracy: 70% Precision: 65%
3]	Human Speech Emotion Recognition	2016	Prosodic Features, Basis Function Network	Accuracy: 75%

Proposed Methodology

I. Dataset:

The RAVDESS Audio-Visual Database of Emotional Speech and Song (RAVDESS) consists of 7356 files, totaling in 24.8 Gb in size. 24 actors record two versions of three different modalities(audio-only, audio-video, video-only) in speech and song formats. Each file was rated 10 times on emotional validity, intensity, and genuineness. Ratings were provided by 247 individuals who were characteristic of untrained adult research participants from North America. A further set of 72 participants provided test-retest data. High levels of emotional validity, interrater reliability, and test-retest intrarater reliability were reported. The distribution of the dataset is reported in the following table:-

Modality	Number of actors	Number of trials per actor	Number of files
Speech(Audio)	24	60	1440
Song(Audio)	23	44	1012
Speech(Audio-Video + Video-only)	24	60	2880
Song(Audio-Video + Video-only)	23	44	2024

II. System Architecture / Design

A typical speech recognition system is developed with major components that include acoustic front-end, acoustic model, lexicon, language model and decoder as shown in figure 1. Acoustic front-end takes care of converting the speech signal into appropriate features which provides useful information for recognition. The input audio waveform from a microphone is converted into a sequence of fixed-size acoustic vectors is a process called feature extraction. The parameters of word / phone models are estimated from the acoustic vectors of training data.

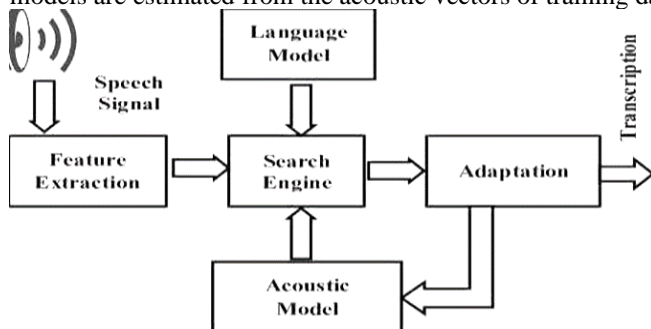


fig.1

III. Working Of Proposed System

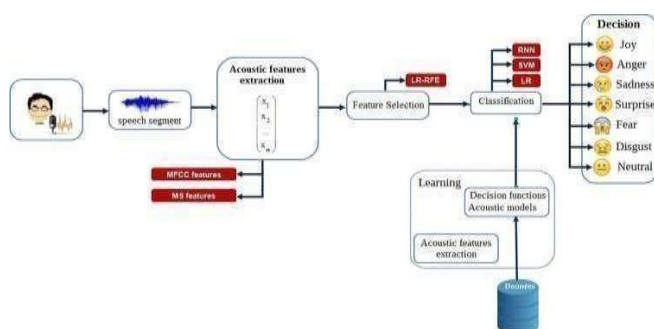


fig.2

IV. System Implementation and Testing

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is the number of features you have) with the value of each feature being the value of a particular coordinate. Data can be scaled before applying to an SVM classifier to avoid attributes in greater numeric ranges while processing it. Scaling also serves the purpose of avoiding some numerical difficulties during the calculation.

For this task, the dataset is built using 5252 samples from: Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset

Toronto emotional speech set (TESS) dataset

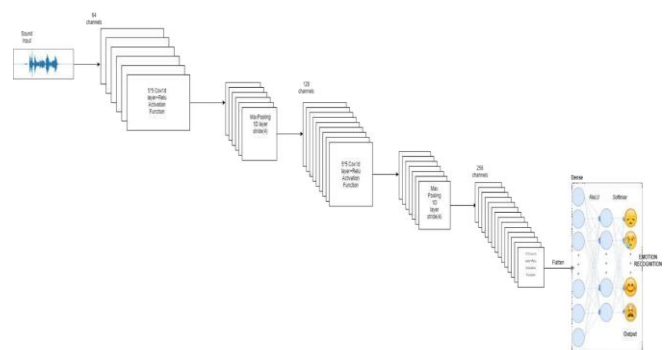


fig.3(a)

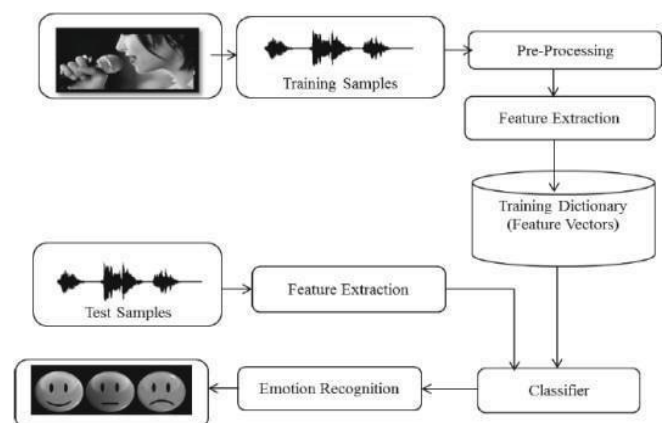


fig.3(b)

Libraries

OS module

Python OS module provides the facility to establish the interaction between the user and the operating system. It offers many useful OS functions that are used to perform OS-based tasks and get related information about operating systems. The OS comes under Python's standard utility modules. This module offers a portable way of using operating system dependent functionality.

os.name()

This function provides the name of the operating system module that it imports.

Currently, it registers 'posix', 'nt', 'os2', 'ce', 'java' and 'riscos'.

`os.mkdir()`

The `os.mkdir()` function is used to create a new directory.

`os.getcwd()`

It returns the current working directory(CWD) of the file.

tkinter

The tkinter package (“Tk interface”) is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, including macOS, as well as on Windows systems. Running `python -m tkinter` from the command line should open a window demonstrating a simple Tk interface, letting you know that tkinter is properly installed on your system, and also showing what version of Tcl/Tk is installed, so you can read the Tcl/Tk documentation specific to that version.

Tkinter supports a range of Tcl/Tk versions, built either with or without thread support. The official Python binary release bundles Tcl/Tk 8.6 threaded. See the source code for the `_tkinter` module for more information about supported versions.

Tkinter is not a thin wrapper, but adds a fair amount of its own logic to make the experience more pythonic. This documentation will concentrate on these additions and changes, and refer to the official Tcl/Tk documentation for details that are unchanged.

PIL

Python Imaging Library (expansion of PIL) is the de facto image processing package for Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images. Support for Python Imaging Library got discontinued in 2011, but a project named pillow forked the original PIL project and added Python 3.x support to it. Pillow was announced as a replacement for PIL for future usage. Pillow supports a large number of image file formats including BMP, PNG, JPEG, and TIFF. The library encourages adding support for newer formats in the library by creating new file decoders.

Pandas

Pandas is a Python library for data analysis. Started by Wes McKinney in 2008 out of a need for a powerful and flexible quantitative analysis tool, pandas has grown into one of the most popular Python libraries. It has an extremely active community of contributors.

Pandas is built on top of two core Python libraries—matplotlib for data visualisation and NumPy for mathematical operations. Pandas acts as a wrapper over these libraries, allowing you to access many of matplotlib's and NumPy's methods with less code. For instance, `pandas'.plot()`

combines multiple matplotlib methods into a single method, enabling you to plot a chart in a few lines.

Before pandas, most analysts used Python for data munging and preparation, and then switched to a more domain specific language like R for the rest of their workflow.

Experimental Results and Discussion

The The deep neural network(CNN) designed for the classification task is reported operationally. The network is able to work on vectors of 40 features for each audio file provided as input. The 40 values represent the compact numerical form of the audio frame of 2s length. Consequently, we provide as input a of size $< \text{number of training files} > \times 40 \times 1$ on which we performed one round of a 1D CNN with a ReLu activation function, dropout of 20% and a max-pooling function 2×2 .

The rectified linear unit (ReLu) can be formalized as $g(z) = \max\{0, z\}$, and it allows us to obtain a large value in case of activation by applying this function as a good choice to represent hidden units. We have run the process described once more by changing the kernel size. Following, we have applied another dropout and then flatten the output to make it compatible with the next layers. Finally, we applied one Dense layer (fully connected layer) with a soft max activation function, varying the output Size from 640 elements to 8 and estimating the probability distribution of each of the classes properly encoded(0=Neutral; 1= Calm; 2= Happy; Sad=3; Angry=4; Fearful= 5; Disgust=6; Surprised=7).

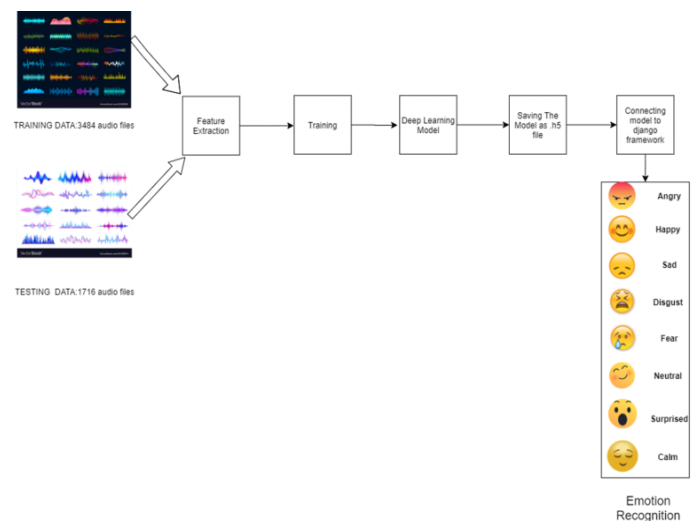


fig.4

Conclusion

In this project we have tried to analyze some samples of speech using the deep learning technique. Firstly we loaded the datasets then we visualized the different human emotions using our functions wave show and spectrogram using the Librosa Library. Then we extracted the acoustic features of all our samples using the MFCC method and arranged the sequential data obtained in the 3D array from as accepted by the LSTM model then we build the LSTM model and after training the model we visualized the data into the graphical from using matplotlib library and after some repeated testing using different values the average accuracy of the model is found to be 90%. Our Project can be Extended to integrate with the robot to have a better conversation as well as it can be integrated with various music. Applications to recommend songs to its users according to his / her emotions, It can also be used in various Online Shopping Applications such as Amazon to improve the product recommendation for its users.

Results Accuracy

In this Python project, we learned to recognize emotions from speech. We used an MLPClassifier for this and made use of the sound file library to read the sound file, and the librosa library to extract features from it. As you'll see, the model delivered an accuracy of 90.00%. That's good enough for yet. **Accuracy Score: 90%**

Result Analysis

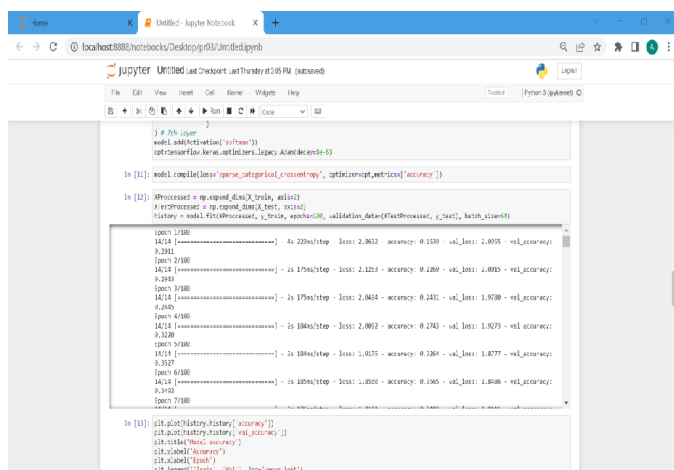


fig.5

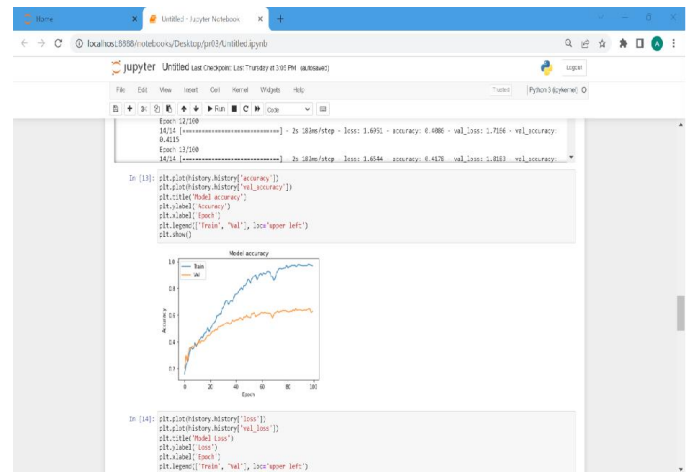


fig.6

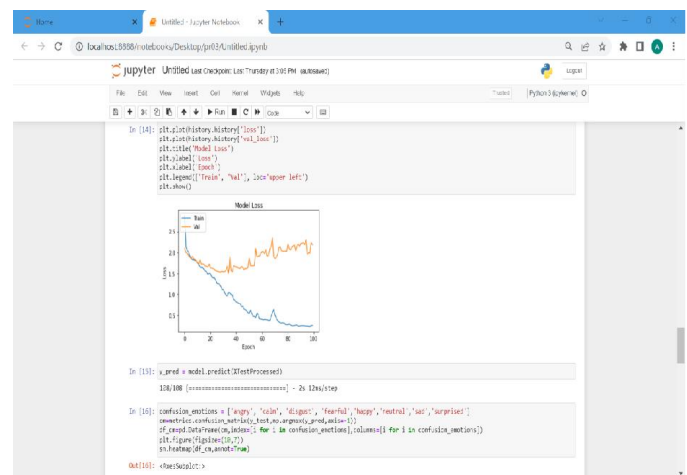


fig.7

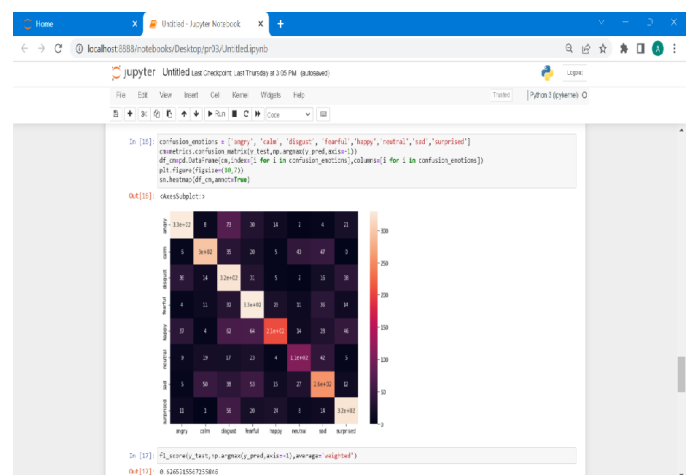
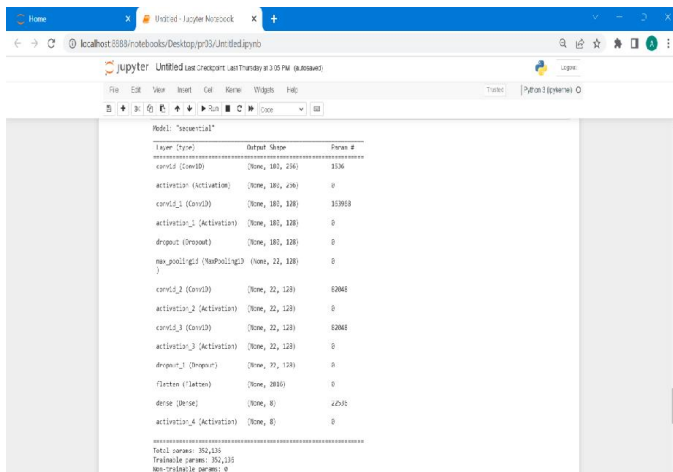


fig.8



Layer (Type)	Output Shape	Param #
conv1d (Conv2D)	(None, 160, 160)	3136
activation (activation)	(None, 160, 160)	0
conv1d_1 (Conv2D)	(None, 160, 160)	31360
activation_1 (activation)	(None, 160, 160)	0
dropout (Dropout)	(None, 160, 160)	0
max_pooling2d (MaxPooling2D)	(None, 22, 22)	0
conv1d_2 (Conv2D)	(None, 22, 22)	82048
activation_2 (activation)	(None, 22, 22)	0
conv1d_3 (Conv2D)	(None, 22, 22)	82048
activation_3 (activation)	(None, 22, 22)	0
dropout_1 (Dropout)	(None, 22, 22)	0
flatten (Flatten)	(None, 2880)	0
dense (dense)	(None, 8)	2304
activation_4 (activation)	(None, 8)	0

Total params: 162,144
 Trainable params: 162,144
 Non-trainable params: 0

fig.9

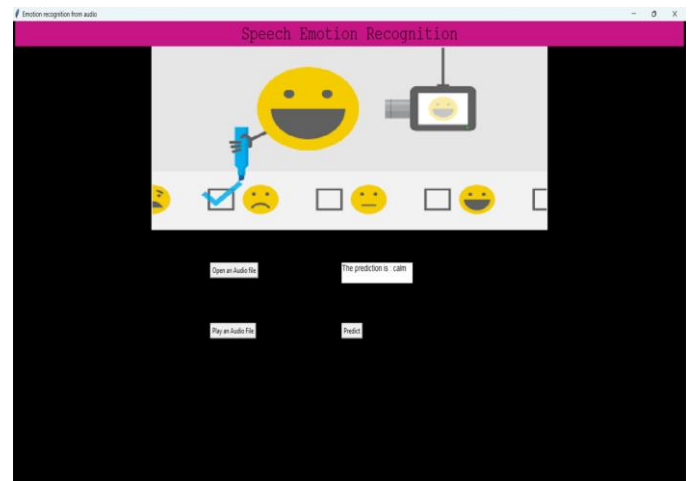


fig.12

Output

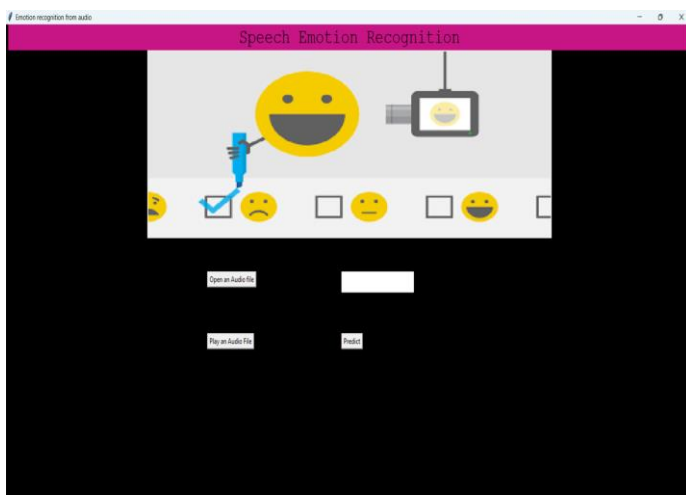


fig.10

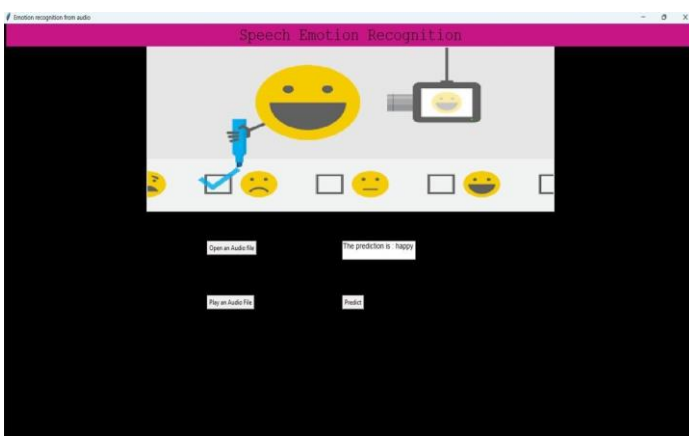


fig.11

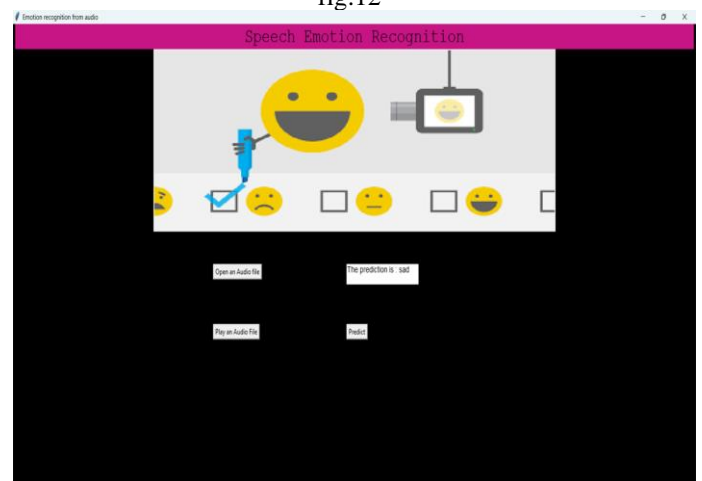


fig.13

References

1. H. Cao, R. Verma, and A. Nenkova, "Speaker-sensitive emotion recognition via ranking: Studies on acted and spontaneous speech," Comput. Speech Lang., vol. 28, no. 1, pp. 186–202, Jan. 2015.
2. L. Chen, X. Mao, Y. Xue, and L. L. Cheng, "Speech emotion recognition: Features and classification models," Digit. Signal Process., vol. 22, no. 6, pp. 1154–1160, Dec. 2012.
3. T. L. Nwe, S. W. Foo, and L. C. De Silva, "Speech emotion recognition using hidden Markov models," Speech Commun., vol. 41, no. 4, pp. 603–623, Nov. 2003.
4. S. Wu, T. H. Falk, and W.-Y. Chan, "Automatic speech emotion recognition using modulation spectral features," Speech Commun., vol. 53, no. 5, pp. 768–785, May 2011.
5. J. Rong, G. Li, and Y.-P. P. Chen, "Acoustic feature selection for automatic emotion recognition from speech," Inf. Process. Manag., vol. 45, no. 3, pp. 315–328, May 2009.
6. C.-H. Wu and W.-B. Liang, "Emotion Recognition of Affective Speech Based on Multiple Classifiers Using

Acoustic-Prosodic Information and Semantic Labels,” IEEE Trans. Affect. Comput., vol. 2, no. 1, pp. 10–21, Jan. 2011.

7. S. S. Narayanan, “Toward detecting emotions in spoken dialogs,” IEEE Trans. Speech AudioProcess., vol. 13, no. 2, pp. 293–303, Mar. 2005.
8. B. Yang and M. Lugger, “Emotion recognition from speech signals using new harmonyfeatures,” Signal Processing, vol. 90, no. 5, pp. 1415–1423, May 2010
9. E. M. Albornoz, D. H. Milone, and H. L. Rufiner, “Spoken emotion recognition using hierarchical classifiers,” Comput. Speech Lang., vol. 25, no. 3, pp. 556–570, Jul. 2011.
10. C.-C. Lee, E. Mower, C. Busso, S. Lee, and S. Narayanan, “Emotion recognition using a hierarchical binary decision tree approach,” Speech Commun., vol. 53, no. 9–10, pp. 1162– 1171, Nov. 2011.