



# Python Mini Project – Speech Emotion Recognition with librosa

## Python Mini Project

Speech emotion recognition, the best ever python mini project. The best example of it can be seen at call centers. If you ever noticed, call centers employees never talk in the same manner, their way of pitching/talking to the customers changes with customers. Now, this does happen with common people too, but how is this relevant to call centers? Here is your answer, the employees recognize customers' emotions from speech, so they can improve their service and convert more people. In this way, they are using speech emotion recognition. So, let's discuss this project in detail.

Speech emotion recognition is a simple Python mini-project, which you are going to practice with DataFlair. Before, I explain to you the terms related to this mini python project, make sure you bookmarked the ***complete list of Python Projects***.

1. [Fake News Detection Python Project](#)
2. [Parkinson's Disease Detection Python Project](#)
3. [Color Detection Python Project](#)
4. [Speech Emotion Recognition Python Project](#)
5. [Breast Cancer Classification Python Project](#)
6. [Age and Gender Detection Python Project](#)
7. [Handwritten Digit Recognition Python Project](#)
8. [Chatbot Python Project](#)
9. [Driver Drowsiness Detection Python Project](#)
10. [Traffic Signs Recognition Python Project](#)
11. [Image Caption Generator Python Project](#)

*Stay updated with latest technology trends*

[Join DataFlair on Telegram!!](#)

## What is Speech Emotion Recognition?

Speech Emotion Recognition, abbreviated as SER, is the act of attempting to recognize human emotion and affective states from speech. This is capitalizing on the fact that voice often reflects underlying emotion through tone and pitch. This is also the phenomenon that animals like dogs and horses employ to be able to understand human emotion.

SER is tough because emotions are subjective and annotating audio is challenging.

## What is librosa?

librosa is a [Python library](#) for analyzing audio and music. It has a flatter package layout, standardizes interfaces and names, backwards compatibility, modular functions, and readable code. Further, in this Python mini-project, we demonstrate how to install it (and a few other packages) with pip.

## What is JupyterLab?

JupyterLab is an open-source, web-based UI for Project Jupyter and it has all basic functionalities of the Jupyter Notebook, like notebooks, terminals, text editors, file browsers, rich outputs, and more. However, it also provides improved support for third party extensions.

To run code in the JupyterLab, you'll first need to run it with the command prompt:

```
1. C:\Users\DataFlair>jupyter lab
```

This will open for you a new session in your browser. Create a new Console and start typing in your code. JupyterLab can execute multiple lines of code at once; pressing enter will not execute your code, you'll need to press Shift+Enter for the same.

## Speech Emotion Recognition – Objective

To build a model to recognize emotion from speech using the librosa and sklearn libraries and the RAVDESS dataset.

## Speech Emotion Recognition – About the Python Mini Project

In this Python mini project, we will use the libraries librosa, soundfile, and sklearn (among others) to build a model using an MLPClassifier. This will be able to recognize emotion from sound files. We will load the data, extract features from it, then split the dataset into training and testing sets. Then, we'll initialize an MLPClassifier and train the model. Finally, we'll calculate the accuracy of our model.

# The Dataset

For this Python mini project, we'll use the RAVDESS dataset; this is the Ryerson Audio-Visual Database of Emotional Speech and Song dataset, and is free to download. This dataset has 7356 files rated by 247 individuals 10 times on emotional validity, intensity, and genuineness. The entire dataset is 24.8GB from 24 actors, but we've lowered the sample rate on all the files, and you can [download it here](#).

## Prerequisites

You'll need to install the following libraries with pip:

```
1. pip install librosa soundfile numpy sklearn pyaudio
```

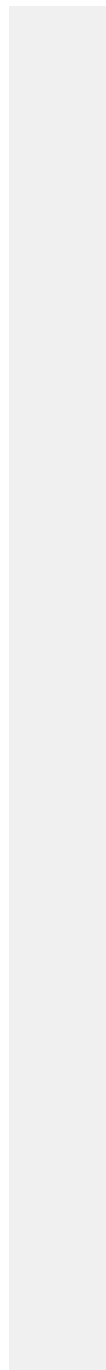
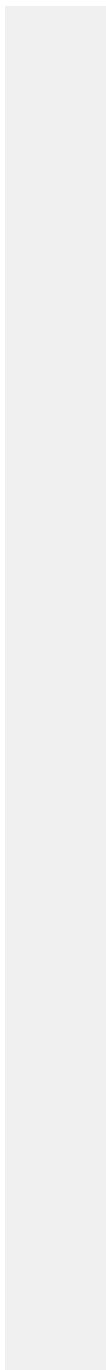
If you run into issues installing librosa with pip, you can try it with conda.

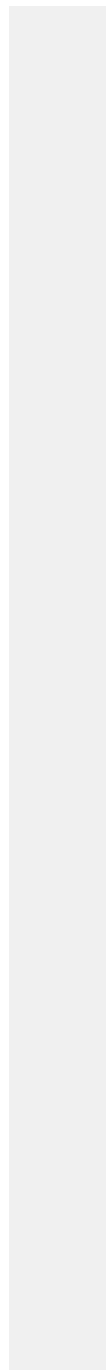
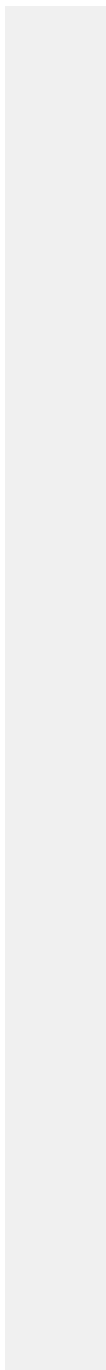
## Steps for speech emotion recognition python projects

1. Make the necessary imports:

```
1. import librosa
2. import soundfile
3. import os, glob, pickle
4. import numpy as np
5. from sklearn.model_selection import train_test_split
6. from sklearn.neural_network import MLPClassifier
7. from sklearn.metrics import accuracy_score
```

**Screenshot:**





2. Define a function `extract_feature` to extract the mfcc, chroma, and mel features from a sound file. This function takes 4 parameters- the file name and three Boolean parameters for the three features:

- **mfcc:** Mel Frequency Cepstral Coefficient, represents the short-term power spectrum of a sound
- **chroma:** Pertains to the 12 different pitch classes
- **mel:** Mel Spectrogram Frequency

*Learn more about [Python Sets and Booleans](#)*

Open the sound file with `soundfile.SoundFile` using `with-as` so it's automatically closed once we're done. Read from it and call it `X`. Also, get the sample rate. If `chroma` is `True`, get the Short-Time Fourier Transform of `X`.

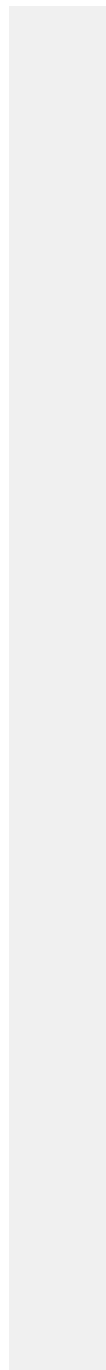
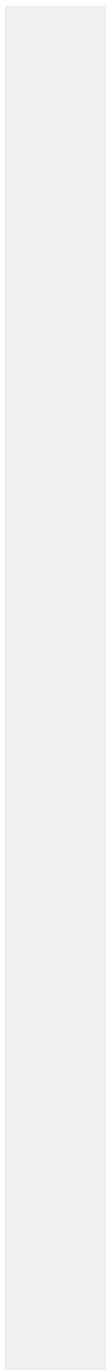
Let `result` be an empty numpy array. Now, for each feature of the three, if it exists, make a call to the corresponding function from `librosa.feature` (eg- `librosa.feature.mfcc` for `mfcc`), and get the mean value. Call the function `hstack()` from `numpy` with `result` and the feature value, and store this in `result`. `hstack()` stacks arrays in sequence horizontally (in a columnar fashion). Then, return the result.

```
1. #DataFlair - Extract features (mfcc, chroma, mel) from a sound file
```



```
2. def extract_feature(file_name, mfcc, chroma, mel):
3.     with soundfile.SoundFile(file_name) as sound_file:
4.         X = sound_file.read(dtype="float32")
5.         sample_rate=sound_file.samplerate
6.         if chroma:
7.             stft=np.abs(librosa.stft(X))
8.             result=np.array([])
9.             if mfcc:
10.                 mfccs=np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=40).T, axis=0)
11.                 result=np.hstack((result, mfccs))
12.             if chroma:
13.                 chroma=np.mean(librosa.feature.chroma_stft(S=stft, sr=sample_rate).T,axis=0)
14.                 result=np.hstack((result, chroma))
15.         if mel:
16.             mel=np.mean(librosa.feature.melspectrogram(X, sr=sample_rate).T,axis=0)
17.             result=np.hstack((result, mel))
18.     return result
```

**Screenshot:**



3. Now, let's define a [dictionary](#) to hold numbers and the emotions available in the RAVDESS dataset, and a list to hold those we want- calm, happy, fearful, disgust.

```
1. #DataFlair - Emotions in the RAVDESS dataset
2. emotions={
3.     '01':'neutral',
4.     '02':'calm',
5.     '03':'happy',
6.     '04':'sad',
7.     '05':'angry',
8.     '06':'fearful',
9.     '07':'disgust',
10.    '08':'surprised'
11. }
12.
13. #DataFlair - Emotions to observe
14. observed_emotions=['calm', 'happy', 'fearful', 'disgust']
```

**Screenshot:**

**Facing Failure in Interview?**

**Prepare with DataFlair – [Frequently Asked Python Interview Questions](#)**

4. Now, let's load the data with a function `load_data()` – this takes in the relative size of the test set as parameter. `x` and `y` are empty lists; we'll use the `glob()` function from the `glob` module to get all the pathnames for the sound files in our dataset. The pattern we use for this is: “`D:\\DataFlair\\ravdess data\\Actor_*\\*.wav`”. This is because our dataset looks like this:

**Screenshot:**

So, for each such path, get the basename of the file, the emotion by splitting the name around '-' and extracting the third value:

**Screenshot:**

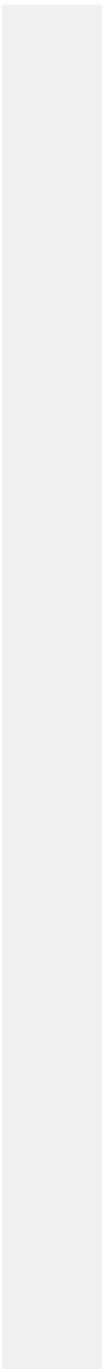
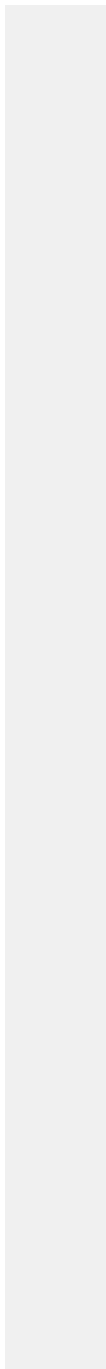
Using our emotions dictionary, this number is turned into an emotion, and our function checks whether this emotion is in our list of observed\_emotions; if not, it continues to the next file. It makes a call to extract\_feature and stores what is returned in 'feature'. Then, it appends the feature to x and the emotion to y. So, the list x holds the features and y holds the emotions. We call the function train\_test\_split with these, the test size, and a random state value, and return that.

```
1. #DataFlair - Load the data and extract features for each sound file
2. def load_data(test_size=0.2):
3.     x,y=[],[]
4.     for file in glob.glob("D:\\DataFlair\\ravdess data\\Actor_*\\*.wav"):
5.         file_name=os.path.basename(file)
6.         emotion=emotions[file_name.split("-")[2]]
```

```
7.         if emotion not in observed_emotions:
8.             continue
9.         feature=extract_feature(file, mfcc=True, chroma=True, mel=True)
10.        x.append(feature)
11.        y.append(emotion)
12.    return train_test_split(np.array(x), y, test_size=test_size, random_state=9)
```

**Screenshot:**





5. Time to split the dataset into training and testing sets! Let's keep the test set 25% of everything and use the `load_data` function for this.

```
1. #DataFlair - Split the dataset
2. x_train,x_test,y_train,y_test=load_data(test_size=0.25)
```

**Screenshot:**

6. Observe the shape of the training and testing datasets:

```
1. #DataFlair - Get the shape of the training and testing datasets
2. print((x_train.shape[0], x_test.shape[0]))
```

## Screenshot:

7. And get the number of features extracted.

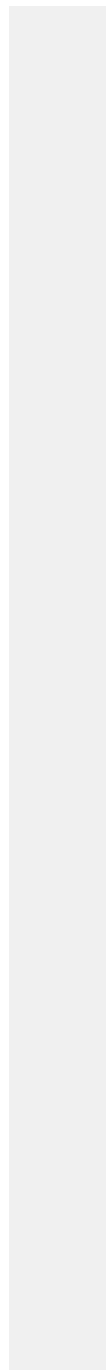
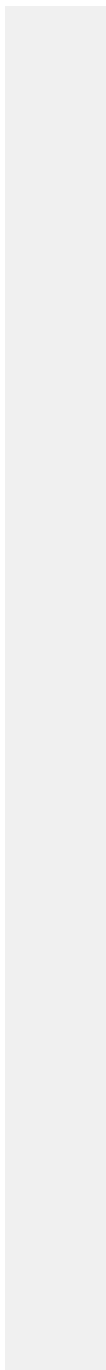
```
1. #DataFlair - Get the number of features extracted
2. print(f'Features extracted: {x_train.shape[1]}')
```

### Output Screenshot:

8. Now, let's initialize an MLPClassifier. This is a Multi-layer Perceptron Classifier; it optimizes the log-loss function using LBFGS or stochastic gradient descent. Unlike SVM or [Naive Bayes](#), the MLPClassifier has an internal neural network for the purpose of classification. This is a feedforward ANN model.

```
1. #DataFlair - Initialize the Multi Layer Perceptron Classifier
2. model=MLPClassifier(alpha=0.01, batch_size=256, epsilon=1e-08, hidden_layer_sizes=(300,),
   learning_rate='adaptive', max_iter=500)
```

**Screenshot:**



9. Fit/train the model.

```
1. #DataFlair - Train the model
2. model.fit(x_train,y_train)
```

**Output Screenshot:**



10. Let's predict the values for the test set. This gives us `y_pred` (the predicted emotions for the features in the test set).

```
1. #DataFlair - Predict for the test set
2. y_pred=model.predict(x_test)
```

**Screenshot:**

11. To calculate the accuracy of our model, we'll call up the `accuracy_score()` function we imported from [sklearn](#). Finally, we'll round the accuracy to 2 decimal places and print it out.

```
1. #DataFlair - Calculate the accuracy of our model
2. accuracy=accuracy_score(y_true=y_test, y_pred=y_pred)
3.
4. #DataFlair - Print the accuracy
5. print("Accuracy: {:.2f}%".format(accuracy*100))
```

**Output Screenshot:**

## Summary

In this Python mini project, we learned to recognize emotions from speech. We used an MLPClassifier for this and made use of the soundfile 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 72.4%. That's good enough for us yet.

Hope you enjoyed the mini python project.

**Want to become next Python Developer??**

**Enroll for [Online Python Course](#) at DataFlair NOW!!**

***Reference – [Zenodo](#)***

**Your 15 seconds will encourage us to work even harder**

Please share your happy experience on [Google](#) | [Facebook](#)

Tags:

Python data science projects

python machine learning projects

Python mini project

Python Projects

Simple Python project

speech emotion recognition

## 116 RESPONSES

 **Comments** **33**

 **Pingbacks** **0**

---

**Brijesh Gujar**  March 8, 2021 at 11:12 pm

Did you figure this out? If yes could you help me.

Reply

**Anamika**  March 18, 2021 at 9:54 pm

where can I find link to this source code?

[Reply](#)

**Fonarik**  [March 28, 2021 at 11:38 pm](#)

If someone who successfully ran this code, please help)

[Reply](#)

**Debasree Das**  [March 30, 2021 at 4:25 pm](#)

Run the code step by step in jupyter or spider. Its running fine and producing a test accuracy of 74%.

[Reply](#)

**Debasree Das**  [March 30, 2021 at 4:24 pm](#)

The code is running fine.

[Reply](#)

**Divyansh**  [May 9, 2021 at 7:02 pm](#)

from sklearn.model\_selection import train\_test\_split this is now working for me can you please help me to resolve this issue

[Reply](#)

**Adnan**  [May 5, 2021 at 1:10 pm](#)

ValueError: With n\_samples=0, test\_size=0.2 and train\_size=None, the resulting train set will be empty. Adjust any of the aforementioned parameters.

help me out with this

[Reply](#)

**Adnan**  [May 6, 2021 at 3:40 pm](#)

how do i input the test data and test for the output

[Reply](#)

**Abdussami Qamroddin Bangi**  [May 17, 2021 at 8:48 pm](#)

yes plz, tell us how can we input a sound file and get output from this trained model.

[Reply](#)

**Satyasai**  [May 10, 2021 at 5:08 pm](#)

ValueError: With n\_samples=0, test\_size=0.2 and train\_size=None, the resulting train set will be empty. Adjust any of the aforementioned parameters.

help me out with this

[Reply](#)

**Kunal Parihar**  [June 6, 2021 at 11:23 am](#)

Hi, did you get the answer?

I am stuck on the same question.

[Reply](#)

**Lorenzo Valenti**  [August 13, 2021 at 8:14 pm](#)

It's because you might forget something on importing the directory of the sound-file.

probably on: for file in glob.glob("//Users//Apple//Desktop//Program//coding//GitHub//speech-emotion-recognition-ravdess-data//Actor\_\*//\*.wav"): You might forget something here, but some common mistake is the: ...//Actor\_\*//\*.wav"):



I am a 16 years old student learning machine learning.

Hope this helped you, it has been passed a lot of time. How the things are going?

[Reply](#)

**Lorenzo Valenti**  [August 13, 2021 at 8:14 pm](#)

It's because you might forget something on importing the directory of the sound-file.

probably on: for file in glob.glob("//Users//Apple//Desktop//Program//coding//GitHub//speech-emotion-recognition-ravdess-data//Actor\_\*/\*.wav"): You might forget something here, but some common mistake is the: ...//Actor\_\*/\*.wav"):

I am a 16 years old student learning machine learning.

Hope this helped you, it has been passed a lot of time. How the things are going?

[Reply](#)

**Satyasai**  [May 10, 2021 at 5:11 pm](#)

The project is about recognising the speech,so the output should be like sad or,happy.but why the output is accuracy score.can you tell me

[Reply](#)

**Shivansh Noob**  [May 17, 2021 at 2:10 pm](#)

same question bro

[Reply](#)

**Ritesh Pudasaini**  [May 17, 2021 at 12:41 pm](#)

Invalid shape for monophonic audio: ndim=2, shape=(166566, 2)

I am getting this error in 5th instruction while splitting the dataset. Can anyone help me?

[Reply](#)

**Kunal Parihar**  [June 6, 2021 at 11:23 am](#)

Hi, did you get the answer?

I am stuck on the same question.

[Reply](#)

**PADALA LIKHITH RISHI**  [June 10, 2021 at 12:18 pm](#)

yes i got the same error did u solved it

[Reply](#)

**Abdussami Qamroddin Bangi**  [May 17, 2021 at 8:48 pm](#)

yes plz tell us how can we input a sound file and get output from this trained model.

[Reply](#)

**Manjunath**  [May 22, 2021 at 10:27 am](#)

name 'mel' is not defined

This is the error I am getting while executing the second part

[Reply](#)

**ram**  [May 24, 2021 at 9:49 pm](#)

hi, could you please send me the complete executed source code. because I am currently working on this project. it is really so important to me. my emailid:ramkumarpoluri@gmail.com

[Reply](#)

**ram**  [May 26, 2021 at 3:20 pm](#)

hi can you please send me the complete executed source code, because currently, I am working on this project so it is so important to me.

my emailid:ramkumarpoluri@gmail.com

[Reply](#)

**Sai teja**  [July 24, 2021 at 7:27 pm](#)

Do you have the code

[Reply](#)

**ram**  [May 26, 2021 at 3:28 pm](#)

hi sameera, could you please send me the complete executed source code, because currently, I am working on this project it is so important to me. my emailid:ramkumarpoluri@gmail.com

[Reply](#)

**rishi**  [May 31, 2021 at 9:42 pm](#)

Could you please add accuracy and loss plots for better visualisation of training model

[Reply](#)

**PADALA LIKHITH RISHI**  [June 10, 2021 at 12:17 pm](#)

Invalid shape for monophonic audio: ndim=2, shape=(172972, 2)

I am getting this error in 5th instruction while splitting the dataset. Can anyone help me?

[Reply](#)

**Alex777**  [June 14, 2021 at 6:42 pm](#)

Calculate a piece of code many times and get different results

```
model=MLPClassifier(alpha=0.01, batch_size=256, epsilon=1e-08, hidden_layer_sizes=(300,), learning_rate='adaptive',
max_iter=500)
model.fit(x_train,y_train)
y_pred=model.predict(x_test)
accuracy=accuracy_score(y_true=y_test, y_pred=y_pred)
print("Accuracy: {:.2f}%".format(accuracy*100))
```

I counted 3 times and here are the results

Accuracy: 59.38%

Accuracy: 73.96%

Accuracy: 61.98%

[Reply](#)

**zaid**  [June 17, 2021 at 3:06 am](#)

this is really helpfull.thanks!

[Reply](#)

**hoo**  [June 21, 2021 at 9:37 am](#)

ValueError: With n\_samples=0, test\_size=0.2 and train\_size=None, the resulting train set will be empty. Adjust any of the aforementioned parameters.

[Reply](#)

**Lorenzo Valenti**  [August 13, 2021 at 8:18 pm](#)

It's because you might forget something on importing the directory of the sound-file.  
probably on: `for file in glob.glob("//Users//Apple//Desktop//Program//coding//GitHub//speech-emotion-recognition-ravdess-data//Actor_*//*.wav")`: You might forget something here, but some common mistake is the:  
`...//Actor_*//*.wav`):

I am a 16 years old student learning machine learning. Feel free to network with me and chat some deep learning, AI, ML and something else: [instagram: \\_\\_lore.valenti\\_\\_](#)

Hope this helped you, it has been passed a lot of time. How the things are going?

[Reply](#)

**hust**  [June 21, 2021 at 9:38 am](#)

ValueError: With `n_samples=0`, `test_size=0.2` and `train_size=None`, the resulting train set will be empty. Adjust any of the aforementioned parameters.

[Reply](#)

**Lorenzo Valenti**  [August 13, 2021 at 8:16 pm](#)

It's because you might forget something on importing the directory of the sound-file.  
probably on: `for file in glob.glob("//Users//Apple//Desktop//Program//coding//GitHub//speech-emotion-recognition-ravdess-data//Actor_*//*.wav")`: You might forget something here, but some common mistake is the:  
`...//Actor_*//*.wav`):

I am a 16 years old student learning machine learning. Feel free to network with me and chat some deep learning, AI, ML and something else: [instagram: \\_\\_lore.valenti\\_\\_](#)

Hope this helped you, it has been passed a lot of time. How the things are going?

[Reply](#)

**Andy Sheelavantar**  [September 2, 2021 at 1:00 am](#)

this is path error.. set the path properly

[Reply](#)

[« Older Comments](#)

## LEAVE A REPLY

**Comment**

**Name \***

This  
site  
is

**Email \***

protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

**Post Comment**

[Home](#) [About us](#) [Contact us](#) [Terms and Conditions](#) [Privacy Policy](#) [Disclaimer](#) [Write For Us](#) [Success Stories](#)

