EXPRE-ME

Mini Project Report

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

A.PRASANNA TULASI	**20L31A5402**
L.LOKESWAR RAO	**20L31A5438**
P.HARSHA PRIYA	**20L31A5451**
P.V.VENKATA KALYAN	**20L31A5448**
T.PRAMOD KUMAR	**21L35A5406**

Under the guidance of

Mrs.I.Mani kumari ,M tech.

Assistant Professor

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY

(Autonomous) Affiliated to JNTU Kakinada & Approved by AICTE, New Delhi

Re-Accredited by NBA & NAAC (GPA of 3.41/ 4.00)

ISO 9001:2008, ISO 14001:2004, OHSAS 18001:2007 Certified Institution

VISAKHAPATNAM – 530 039

2020-2024

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY

Department of Artificial Intelligence and Data Science

CERTIFICATE

This is to certify that the project report entitled "**EXPRE-ME**" is the Bonafide record of project work carried out under my supervision by A.PRASANNA TULASI(20L31A5402), L.LOKESHWAR RAO (20L31A5438), P.HARSHA PRIYA(20L31A5451), P.V.VENKATA KALYAN(20L31A5448) and T.PRAMOD KUMAR(21L35A5406), during the academic year 2021-2022, in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in A of Jawaharlal Nehru Technological University, Kakinada. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Project Guide

Mrs.I.Manikumari

Head of the Department

Dr.T.V.Madhusudhan

DECLARATION

We hereby declare that the project report entitled "**EXPRE-ME**" has been written by us and has not been submitted either in part or whole for the award of any degree, diploma or any other similar title to this or any other university.

A.PRASANNA TULASI	**20L31A5402**
L.LOKESWAR RAO	**20L31A5438**
P.HARSHA PRIYA	**20L31A5451**
P.V.VENKATA KALYAN	**20L31A5448**
T.PRAMOD KUMAR	**21L35A5406**

Date:

Place:

ACKNOWLEDGEMENT

We feel great pleasure in submitting the project report on "**EXPRE-ME**". Every orientation work has the imprint of many people and this work is no different. This work gives us an opportunity to explore the same. While preparing the project report. We received endless help from several people. This report would be incomplete if we don't convey my sincere thanks to all those who were involved. We are highly thankful for the cooperation extended by Mrs.I.Mani kumari, Assistant Professor Department of Artificial Intelligence and Data Science for giving comments for improvement of work, encouragement, and help during the completion of the project. We would like to personally thank Prof. R.V.L.S.N.Sastry for providing us with all the necessary facilities for the successful completion of the project. Last but not the least; I should say thanks from the bottom of my heart to my family and friends for their never-ending love, help, and support in so many ways through all this time.

We also take the opportunity to acknowledge the contribution of Prof**. Dr. T.V. Madhusudan Rao,** Head, Department of Artificial Intelligence and Data Science, for his full support and assistance during the development of the project.

ABSTRACT

It is often confusing for a person to decide which music he/she have to listen from a massive collection of existing options. There have been several suggestion frameworks available for issues like music, dining, and shopping depending upon the mood of user. The main objective of our music recommendation system is to provide suggestions to the users that fit the user's preferences. The analysis of the facial expression/user emotion may lead to understanding the current emotional or mental state of the user. Music and videos are one region where there is a significant chance to prescribe abundant choices to clients in light of their inclinations and also recorded information. It is well known that humans make use of facial expressions to express more clearly what they want to say and the context in which they meant their words. More than 60 percent of the users believe that at a certain point of time the number of songs present in their songs library is so large that they are unable to figure out the song which they have to play. By developing a recommendation system, it could assist a user to make a decision regarding which music one should listen to helping the user to reduce his/her stress levels. The user would not have to waste any time in searching or to look up for songs and the best track matching the user's mood is detected, and songs would be shown to the user according to his/her mood. The image of the user is captured with the help of a webcam. The user's picture is taken and then as per the mood/emotion of the user an appropriate song from the playlist of the user is shown matching the user's requirement.

CONTENTS

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

Majority of people of all ages like music, and we believe that music players should be capable of much more than merely playing songs and allowing users to create playlists. As a result, listening to music to unwind after work can improve one's health. For this, a music player should be clever and respond to the choices of the user. A music player should assist users in automatically organizing and playing songs without requiring much effort in song selection and reorganization. An Emotion-Based Music Player gives all music listeners a better platform and ensures song selection automation.

An Emotion-Based Music Player provides a better platform for all music listeners by automating song selection and updating playlists regularly based on the user's identified emotion. These assist users in organizing and playing songs based on their mood, making them less stressed. So, the goal is to design a recommender system on the music and emotion domain. The system will be composed of server-side components and client-side components. The server-side component will manage the database operations and algorithms that produce recommendation results. The client-side components will be graphical interfaces that are integrated into corresponding larger systems

1.2 IMPORTANCE OF EXPRE-ME

Music can be categorized into several genres, such as pop, rock, jazz, blues, folk etc. Listening to music in the digital age is easier because of the features on the smartphone that can play music offline and online. Nowadays, the availability of digital music is very abundant compared to the previous era, so to sort out all this digital music is very time consuming and causes information fatigue. Therefore, it is very useful to develop a music recommender system that can search music libraries automatically and suggest songs that are suitable for users. Music streaming applications like Spotify and Pandora have features to recommend music to users. These features can help to get a list of appropriate music from the popular music libraries based on music that has been heard previously. This makes the recommender system play an important role in maintaining the streaming music business. Music recommendations are done by looking for similarities from one music to another or by giving preference from one user to another The challenge of music recommender system is to create a system that can continually find attractive new music which understand the users' preferences in music This requires that the music personalized recommender system should effectively reflect the personal preferences. It needs adjustments to achieve personalized recommendations for the needs of different audiences. Therefore, the music personalized recommender system is a more complicated than the general recommender system. It is necessary to consider user needs comprehensively and combines the music feature recognition and audio processing technologies to extract the music features. A music player that helps users play tracks automatically

without requiring much effort in searching songs. An emotion-based music player improves the listening experience for all music listeners and automates song selection.

CHAPTER 2

TECHNOLOGIES USED

2.1 Python

Python is a general purpose, dynamic,high-level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.

Python is easy to learn yet powerful and versatile scripting language, which makes it attractive for Application Development.

Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development.

Python supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.

Python is not intended to work in a particular area, such as web programming. That is why it is known as multipurpose programming language because it can be used with web, enterprise, 3D CAD, etc.

We don't need to use data types to declare variable because it is dynamically typed so we can write a=10 to assign an integer value in an integer variable.

Python makes the development and debugging fast because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

Python has wide range of libraries and frameworks widely used in various fields such as machine learning, artificial intelligence, web applications, etc. We define some popular frameworks and libraries of Python as follows.

- **Web development (Server-side) -** Django Flask, Pyramid, CherryPy
- **GUIs based applications -** Tk, PyGTK, PyQt, PyJs, etc.
- **Machine Learning -** TensorFlow, PyTorch, **Scikit-learn**, Matplotlib, SciPy, etc.
- **Mathematics -** NumPy, Pandas, etc.

2.2 NumPy

NumPy stands for numeric python which is a python package for the computation and processing of the multidimensional and single dimensional array elements.

It is an extension module of Python which is mostly written in C. It provides various functions which are capable of performing the numeric computations with a high speed.

NumPy provides various powerful data structures, implementing multi-dimensional arrays and matrices. These data structures are used for the optimal computations regarding arrays and matrices.

NumPy may be a library for the Python programing language, adding support for giant, multi-dimensional arrays and matrices, alongside an outsized collection of high-level mathematical functions to work on these arrays.

The need of NumPy

With the revolution of data science, data analysis libraries like NumPy, SciPy, Pandas, etc. have seen a lot of growth. With a much easier syntax than other programming languages, python is the first choice language for the data scientist.

NumPy provides a convenient and efficient way to handle the vast amount of data. NumPy is also very convenient with Matrix multiplication and data reshaping. NumPy is fast which makes it reasonable to work with a large set of data.

There are the following advantages of using NumPy for data analysis.

1. NumPy performs array-oriented computing.
2. It efficiently implements the multidimensional arrays.
3. It performs scientific computations.
4. It is capable of performing Fourier Transform and reshaping the data stored in multidimensional arrays.
5. NumPy provides the in-built functions for linear algebra and random number generation.

Nowadays, NumPy in combination with SciPy and Mat-plotlib is used as the replacement to MATLAB as Python is more complete and easier programming language than MATLAB.

2.3 Pandas

Pandas is defined as an open-source library that provides high-performance data manipulation in Python. The name of Pandas is derived from the word **Panel Data**, which means **an Econometrics from Multidimensional data.**

Pandas is a Python library used for working with data sets.

It has functions for analyzing, cleaning, exploring, and manipulating data.

The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis"

Pandas allows us to analyze big data and make conclusions based on statistical theories.

Pandas can clean messy data sets, and make them readable and relevant.

Pandas is built on top of the **Numpy** package, means **Numpy** is required for operating the Pandas.

Before Pandas, Python was capable for data preparation, but it only provided limited support for data analysis. So, Pandas came into the picture and enhanced the capabilities of data analysis. It can perform five significant steps required for processing and analysis of data irrespective of the origin of the data, i.e., **load, manipulate, prepare, model, and analyze**.

Key Features of Pandas

- It has a fast and efficient DataFrame object with the default and customized indexing.
- Used for reshaping and pivoting of the data sets.
- Group by data for aggregations and transformations.
- It is used for data alignment and integration of the missing data.
- Provide the functionality of Time Series.
- Process a variety of data sets in different formats like matrix data, tabular heterogeneous, time series.
- Handle multiple operations of the data sets such as subsetting, slicing, filtering, groupBy, re-ordering, and re-shaping.
- It integrates with the other libraries such as SciPy, and scikit-learn.
- Provides fast performance, and If you want to speed it, even more, you can use the **Cython**.

Benefits of Pandas

The benefits of pandas over using other language are as follows:

- **Data Representation:** It represents the data in a form that is suited for data analysis through its DataFrame and Series.
- **Clear code:** The clear API of the Pandas allows you to focus on the core part of the code. So, it provides clear and concise code for the user

Python Pandas DataFrame

It is a widely used data structure of pandas and works with a two-dimensional array with labeled axes (rows and columns). DataFrame is defined as a standard way to store data and has two different indexes, i.e., row index and column index. It consists of the following properties:

- The columns can be heterogeneous types like int, bool, and so on.
- It can be seen as a dictionary of Series structure where both the rows and columns are indexed. It is denoted as "columns" in case of columns and "index" in case of rows.

2.4 TensorFlow

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions.TensorFlow may be a free and open-source software library for machine learning and AI. It is often used

The official website of TensorFlow is mentioned below −

www.tensorflow.org

- It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.
- It includes a programming support of deep neural networks and machine learning techniques.
- It includes a high scalable feature of computation with various data sets.
- TensorFlow uses GPU computing, automating management. It also includes a unique feature of optimization of same memory and the data used.

TensorFlow is so popular

TensorFlow is well-documented and includes plenty of machine learning libraries. It offers a few important functionalities and methods for the same.

TensorFlow is also called a “Google” product. It includes a variety of machine learning and deep learning algorithms. TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

Tensor Data Structure

Tensors are used as the basic data structures in TensorFlow language. Tensors represent the connecting edges in any flow diagram called the Data Flow Graph. Tensors are defined as a multidimensional array or list.

Tensors are identified by the following three parameters −

Rank

Unit of dimensionality described within tensor is called rank. It identifies the number of dimensions of the tensor. A rank of a tensor can be described as the order or n-dimensions of a tensor defined.

Shape

The number of rows and columns together define the shape of Tensor.

Type

Type describes the data type assigned to Tensor's elements.

A user needs to consider the following activities for building a Tensor −

- Build an n-dimensional array
- Convert the n-dimensional array.

2.5 Keras

Keras is an open-source high-level Neural Network library, which is written in Python and is capable enough to run on Theano, TensorFlow, or CNTK. It was developed by one of the Google engineers, Francois Chollet. It is made user-friendly, extensible, and modular for facilitating faster experimentation with deep neural networks. It not only supports Convolutional Networks and Recurrent Networks individually but also their combination.

It cannot handle low-level computations, so it makes use of the **Backend** library to resolve it. The backend library act as a high-level API wrapper for the low-level API, which lets it run on TensorFlow, CNTK, or Theano.

Initially, it had over 4800 contributors during its launch, which now has gone up to 250,000 developers. It has a 2X growth ever since every year it has grown. Big companies like Microsoft, Google, NVIDIA, and Amazon have actively contributed to the development of Keras. It has an amazing industry interaction, and it is used in the development of popular firms likes Netflix, Uber, Google, Expedia, etc.

Specialities of Keras

- Focus on user experience has always been a major part of Keras.
- Large adoption in the industry.
- It is a multi backend and supports multi-platform, which helps all the encoders come together for coding.
- Research community present for Keras works amazingly with the production community.
- Easy to grasp all concepts.
- It supports fast prototyping.
- It seamlessly runs on CPU as well as GPU.
- It provides the freedom to design any architecture, which then later is utilized as an API for the project.
- It is really very simple to get started with.
- Easy production of models actually makes Keras special.

Keras user experience

1. **Keras is an API designed for humans**

   Best practices are followed by Keras to decrease cognitive load, ensures that the models are consistent, and the corresponding APIs are simple.

2. **Not designed for machines**

   Keras provides clear feedback upon the occurrence of any error that minimizes the number of user actions for the majority of the common use cases.

3. **Easy to learn and use.**

4. **Highly Flexible**

   Keras provide high flexibility to all of its developers by integrating low-level deep learning languages such as TensorFlow or Theano, which ensures that anything written in the base language can be implemented in Keras.

Support of Keras for the claim of being multi-backend and multi-platform

Keras can be developed in R as well as Python, such that the code can be run with TensorFlow, Theano, CNTK, or MXNet as per the requirement. Keras can be run on CPU, NVIDIA GPU, AMD GPU, TPU, etc. It ensures that producing models with Keras is really simple as it totally supports to run with TensorFlow serving, GPU acceleration (WebKeras, Keras.js), Android (TF, TF Lite), iOS (Native CoreML) and Raspberry Pi.

Keras

Keras being a model-level library helps in developing deep learning models by offering high-level building blocks. All the low-level computations such as products of Tensor, convolutions, etc. are not handled by Keras itself, rather they depend on a specialized tensor manipulation library that is well optimized to serve as a backend engine. Keras has managed it so perfectly that instead of incorporating one single library of tensor and performing operations related to that particular library, it offers plugging of different backend engines into Keras.Keras is that the most used deep learning framework among top-5 winning teams on Kaggle. Because Keras makes it easier to run new experiments, it empowers you to undertake more ideas than your competition, faster. And this is how you win.

Keras consist of three backend engines, which are as follows:

1. TensorFlow
2. Theano
3. CNTK

CHAPTER-3

LITERATURE REVIEW

3.1 Literature Review

Reference	Emotions Analyzed	Visual Features	Decision Methods	Database
Real-time mobile [22]	Seven emotions	• Active shape model fitting landmarks • Displacement between landmarks	SVM	CK+ [10]
Ghimire and Lee [23]	Seven emotions	• Displacement between landmarks in continuous frames	Multi-class AdaBoost, SVM	CK+ [10]
Global Feature [24]	Six emotions	• Local binary pattern (LBP) histogram of a face image	Principal component analysis (PCA)	Self-generated
Local region specific feature [33]	Seven emotions	• Appearance of LBP features from specific local regions • Geometric normalized central moment features from specific local regions.	SVM	CK+ [10]
InfraFace [34]	Seven emotions, 17 AUs detected	• Histogram of gradients (HoG)	A linear SVM	CK+ [10]
3D facial expression [39]	Six prototypical emotions	• 3D curve shape and 3D patch shape by analyzing shapes of curves to the shapes of patches	Multiboosting and SVM	BU-3DFE [40]
Stepwise approach [31]	Six prototypical emotions	• Stepwise linear discriminant analysis (SWLDA) used to select the localized features from the expression	Hidden conditional random fields (HCRFs)	CK+ [10], JAFFE [41], B+ [42], MMI [43]

This work, introduces Emotion Based Music Recommendation System, an affective cross-platform music player that recommends music based on the user's current mood. Emotion Module, Music Classification Module, and Recommendation Module are the three components that make up the music player. The Emotion Module uses a photo of the user's face as input and uses deep learning algorithms to accurately detect their mood with a 90.23 percent accuracy rate. The Music Classification Module uses audio features to categorize songs into four different mood classes and reach a stunning result of 97.69 percent. The Recommendation Module recommends music to the user by mapping their feelings to the song's mood type and taking into account the user's preferences.

Emotions Classification from Facial Expressions part of the report is one the most major and important parts of the project. Because of this, publications of research papers, whitepapers, and previous results are near to my study. According to some scientists for this department. Since researchers tried to build a new app with a much more efficient and also computationally effective app, they started by classifying emotions into four major emotions of Happiness, Sadness, Anger, Neutral, Fear, Surprise, and Disgust. While other emotions are the fusion of emotions like grief is of Sadness and Anger. Thus, this proposed project can utilize Convolutional Neural Network(CNN) for the identification of facial expressions, as CNN can give good accuracy and precision in a reasonable amount of time.

Renuka R Londhe et al. [1] proposed a paper which focused on the study of changes in the curvatures of the face and the intensities of the corresponding pixels. The author used Artificial Neural Networks(ANN), which was used to classify the emotions. The author also proposed various approaches

for a playlist. Zheng et al. [2]proposed two significant categories for facial feature extraction, which included Appearance-based feature extraction and geometric based feature extraction, which included extraction of some essential points of the face such as mouth, eyes, and eyebrows.Nikhil et al. [3].determines the mindset of the user by using facial expression Humans often express their feeling by their expressions, hand gestures, and by raising the voice of tone but mostly humans express their feelings by their face. Emotion-based music player reduces the time complexity of the user. Generally, people have a large number of songs on their playlist. Playing songs randomly does not satisfy the mood of the user. This system helps user to play songs automatically according to their mood. The image of the user is captured by the web camera, and the images are saved. The images are first converted from RGB to binary format. This process of representing the data is called a feature-point detection method. This process can also be done by using Haar Cascade technology provided by Open CV. The music player is developed by using a java program. It manages the database and plays the song according to the mood of the user.Z. Zeng et al.[5] researched various advances in human affect recognition. He focused on various approaches that can handle audio and/or visual recordings of displays of affective states. The paper provides a detailed review of audio/visual computing methods. The effect is described as a prototype of emotion categories which include happiness, sadness, fear, anger, disgust, and surprise. This paper focused on discussing the challenges in computing methods for the development of automatic, spontaneous affect recognizer, which helped in emotion detection. It also identified some problems that have been missed or avoided in uni-modal posed emotion recognition. Parul Tambe et al. [7] proposed an idea which automated the interactions between the users and music player, which learned all the preferences, emotions and activities of a user and gave song selection as a result. The various facial expressions of users were recorded by the device to determine the emotion of the user to predict the genre of the music.

CHAPTER-4

PROPOSED WORK

Proposed work

This project has been divided into three phases.

(i) Reading the facial expression

(ii) Picking the playlist based on the expression

(iii) Playing a song from the playlist

4.1 READING THE FACIAL EXPRESSION:

A human face has significant and distinguishing characteristics that aid in the recognition of facial expressions. FER is defined as a change in facial expression caused by an individual's internal emotional state. It is used in a wide range of human-computer interaction (HCI) applications, such as face image processing, facial video surveillance, and facial animation, as well as in the fields of computer vision, digital image processing, and artificial intelligence. Automatic facial expression recognition is a difficult topic that has piqued the interest of many researchers in recent years. In FER, the stage of feature extraction is critical. Alek et al. demonstrated in the literature that facial expression accounts for 55% of total transmission while vocal and spoken communication contributes 38% and 7%, respectively.

There are two primary techniques for designing a FER system. As an initial step, some systems employ a sequence of images ranging from a neutral face to the peak level of emotions. In comparison, some systems use a single image of the face to recognize related emotions, and because they have access to less information, they often perform worse than leading approaches. Apart from the approach type modeled by a FER system, another classification is based on the type of features employed in the recognition process, with a FER system utilizing one or both of these feature categories. The first set of traits is obtained from the facial organs' posture and the skin's texture. The second type of feature is geometric features, which hold information about various positions and points on the face and are used to analyze a static image or a sequence of photos by utilizing the movement of the positions and points within the sequence. Using face landmarks as a starting point for extracting geometric features is one way. Landmarks are significant places on the face that provide useful information for facial analysis. Numerous studies have been undertaken on the subject of facial landmark identification; however, they are outside the scope of this work.

Now to achieve these landmarks there are many ways but the efficient and effective way is to train a model based on the data prebuilt in the history. So we have used a few Deep learning concepts to train our model using python language and few libraries in python. Such as tensorflow, pandas, numpy, etc.

OBJECTIVES WHILE BUILDING THE MODEL:

1) Choosing the dataset

2) Exploratory data analysis of the dataset
3) Training the model

Choosing the dataset:

There is a dataset built in 2013 named fer2013, a dataset made based upon the human expressions, In which it has covered various human expressions, mainly focusing on the most used and predictable expressions named as:

1) Happy
2) Sad
3) Angry
4) Neutral
5) Disgust
6) Surprise
7) Fear

Exploratory data analysis of the dataset

So after choosing the dataset, some exploratory data analysis has been done on the dataset as the chosen data was totally in the form of an excel sheet. The following changes were to be done on the dataset:

(i) Shuffling the dataset.
(ii) Perfectly reforming the pixels for the images.
(iii) Test and train data split.

To accomplish these goals a library in python named numpy and pandas were used.

Training the model

For training the model , a library in python named tensorflow has been used , TensorFlow is **an end-to-end open source platform for machine learning**. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

The below figure shows the creation of the model which has to be trained:

```
model = Sequential([
    layers.Conv2D(64,(5,5),input_shape =(48,48,1),activation='relu'),
    layers.MaxPool2D(pool_size=(5,5),strides=(2,2)),


    layers.Conv2D(64,(3,3),activation='relu'),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.AveragePooling2D(pool_size=(3,3),strides=(2,2)),

    layers.Conv2D(128,(3,3),activation='relu'),
    layers.Conv2D(128,(3,3),activation='relu'),
    layers.AveragePooling2D(pool_size=(3,3),strides=(2,2)),

    layers.Flatten(),


    layers.Dense(1024,activation = 'relu'),
    layers.Dropout(0.2),
    layers.Dense(1024,activation = 'relu'),
    layers.Dropout(0.2),
    layers.Dense(7,activation='softmax')

])
```

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 44, 44, 64)        1664

max_pooling2d (MaxPooling2D  (None, 20, 20, 64)        0
)

conv2d_1 (Conv2D)            (None, 18, 18, 64)        36928

conv2d_2 (Conv2D)            (None, 16, 16, 64)        36928

average_pooling2d (AverageP  (None, 7, 7, 64)          0
ooling2D)

conv2d_3 (Conv2D)            (None, 5, 5, 128)         73856

conv2d_4 (Conv2D)            (None, 3, 3, 128)         147584

average_pooling2d_1 (Averag  (None, 1, 1, 128)         0
ePooling2D)

flatten (Flatten)            (None, 128)               0

...
Trainable params: 1,485,831
Non-trainable params: 0
_________________________________________________________________
None
```

After training the model we have achieved an accuracy of 93.71%, predicting all the facial expressions correctly , the alpha testers of expre-me have given excellent remarks while testing, which brings us to the phase which is retrieving a playlist from Spotify api.

4.2 PICKING THE PLAYLIST BASED ON THE EXPRESSION:

Now we enter into our second phase of selection of a playlist based on the expression predicted, so to achieve this we are using spotify api to extract playlist contents for the expression. As there are totally seven expressions , our team has chosen a few key-words for each expression. in which our code is written in such a way that it will select a random playlist keyword for each song, each time we change the song, which increases the diversity of selection of songs. Another benefit of having this feature is to explore a wide range of genres each time a user clicks or taps next in Spotify. even the user gets bored if he/she get to listen to the same songs again and again, so our code also covers this aspect of the problem, which is repetitive playing of the same songs.

refer this link to get to know the procedure to access the playlists info from the API using spotipy:

1) https://medium.com/ryanjang-devnotes/handy-guide-to-spotify-api-spotipy-for-starters-5e4e05e1b0c7
2) https://medium.com/@maxtingle/getting-started-with-spotifys-api-spotipy-197c3dc6353b

We have used library called “spotipy” to access the data from spotify API. Spotipy is a lightweight Python library for the Spotify Web API. With Spotipy you get full access to all of the music data provided by the Spotify platform.

4.3 PLAYING A SONG FROM THE SELECTED PLAYLIST:

Our developers have written the code in such a way that it’ll first take the number of songs from the playlist and then play a song randomly from the playlist.

CHAPTER-5

IMPLEMENTATION

5.1 EXPREME CODE SNIPPETS:

main.py

```
import cv2

import time

import numpy as np

from PIL import Image

import streamlit as st

import spotify_client as xy

from tensorflow import keras

from keras.models import load_model

from keras.preprocessing.image import img_to_array

from streamlit_webrtc import webrtc_streamer, VideoTransformerBase


emotion_dict = ["Angry", "Disgust", "Fear",

                "Happy", "Neutral", "Sad", "Surprise"]

classifier = load_model('expreme_engine.h5')

classifier.load_weights("expreme_engine.h5")

# load face

try:

    face_cascade =
cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

except Exception:

    st.write("Error loading cascade classifiers")
```

```
global cnt
cnt = 0


class VideoTransformer(VideoTransformerBase):
    def __init__(self):
        self.cnt = 0
    def transform(self, frame):
        img = frame.to_ndarray(format="bgr24")
        img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces = face_cascade.detectMultiScale(
            image=img_gray, scaleFactor=1.3, minNeighbors=5)
        cnt = 0
        for (x, y, w, h) in faces:
            cv2.rectangle(img=img, pt1=(x, y), pt2=(
                x + w, y + h), color=(136, 6, 138), thickness=2)
            roi_gray = img_gray[y:y + h, x:x + w]
            roi_gray = cv2.resize(roi_gray, (48, 48),
                                  interpolation=cv2.INTER_AREA)
            if np.sum([roi_gray]) != 0:
                roi = roi_gray.astype('float') / 255.0
                roi = img_to_array(roi)
                roi = np.expand_dims(roi, axis=0)
                prediction = classifier.predict(roi)[0]
```

```
                maxindex = int(np.argmax(prediction))
                finalout = emotion_dict[maxindex]
                output = str(finalout)
            label_position = (x, y)
            cv2.putText(img, output, label_position,

                        cv2.FONT_ITALIC, 1, (28, 204, 248), 2)
         self.cnt += 1
         if(self.cnt == 40):
        xy.playsong(output)
         self.cnt =0
          return img
         return img
def main():
   st.title("EXPRE-ME")
    activiteis = ["EXPRE-ME", "About"]
    choice = st.sidebar.selectbox("Select Activity", activiteis)
    st.sidebar.markdown(
        """ Built by Lokeswarlakhi
            Mail me on lokeshwarlakhi@gmail.com""")
    if choice == "EXPRE-ME":
        img = Image.open('./expre-me-logo.png')
        st.image(img, width=700)
```

```
        st.header("Click on start and watch the magic happen!!")
        # st.write("Click on start to start the magic")
        cpq = webrtc_streamer(key="example",
video_transformer_factory=VideoTransformer)
        # del(cpq)
        # cnt+=1
    elif choice == "About":
        img = Image.open('./expre-me-logo.png')


        st.image(img, width=700)
        st.subheader("About EXPREME")
        html_temp_about1 = """
        <div style="background-color:#cb1854;padding:10px;font-family:
Verdana;">
            <h4 style="color:white;text-align:center;">
            A music player that helps users play tracks automatically without
requiring much effort in searching songs . An emotion-based music player
improves the listening experience for all music listeners and automates song
selection.
            </h4>
        </div>
        </br>
        """
```

```
        st.markdown(html_temp_about1, unsafe_allow_html=True)
        html_temp4 = """
                <div style="background-color:#cb1854;padding:10px">
                <h4 style="color:white;text-align:center;"> </h4>
                <h4 style="color:white;text-align:center;">Thanks for Visiting</h4>
                </div>
                <br></br>
                """
        st.markdown(html_temp4, unsafe_allow_html=True)
    else:
        pass
# if __name__ == "__main__":#     main()
```

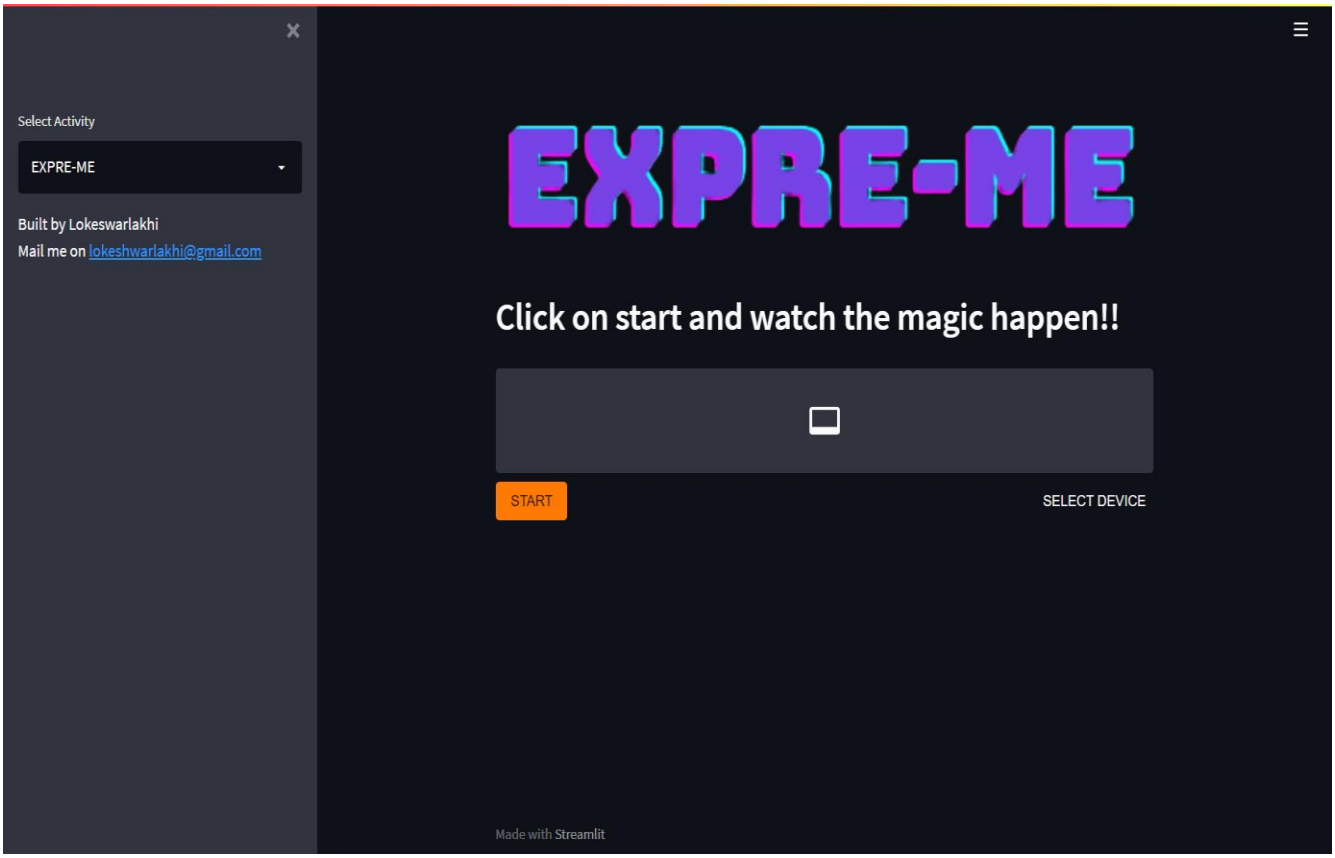
spotify client.py

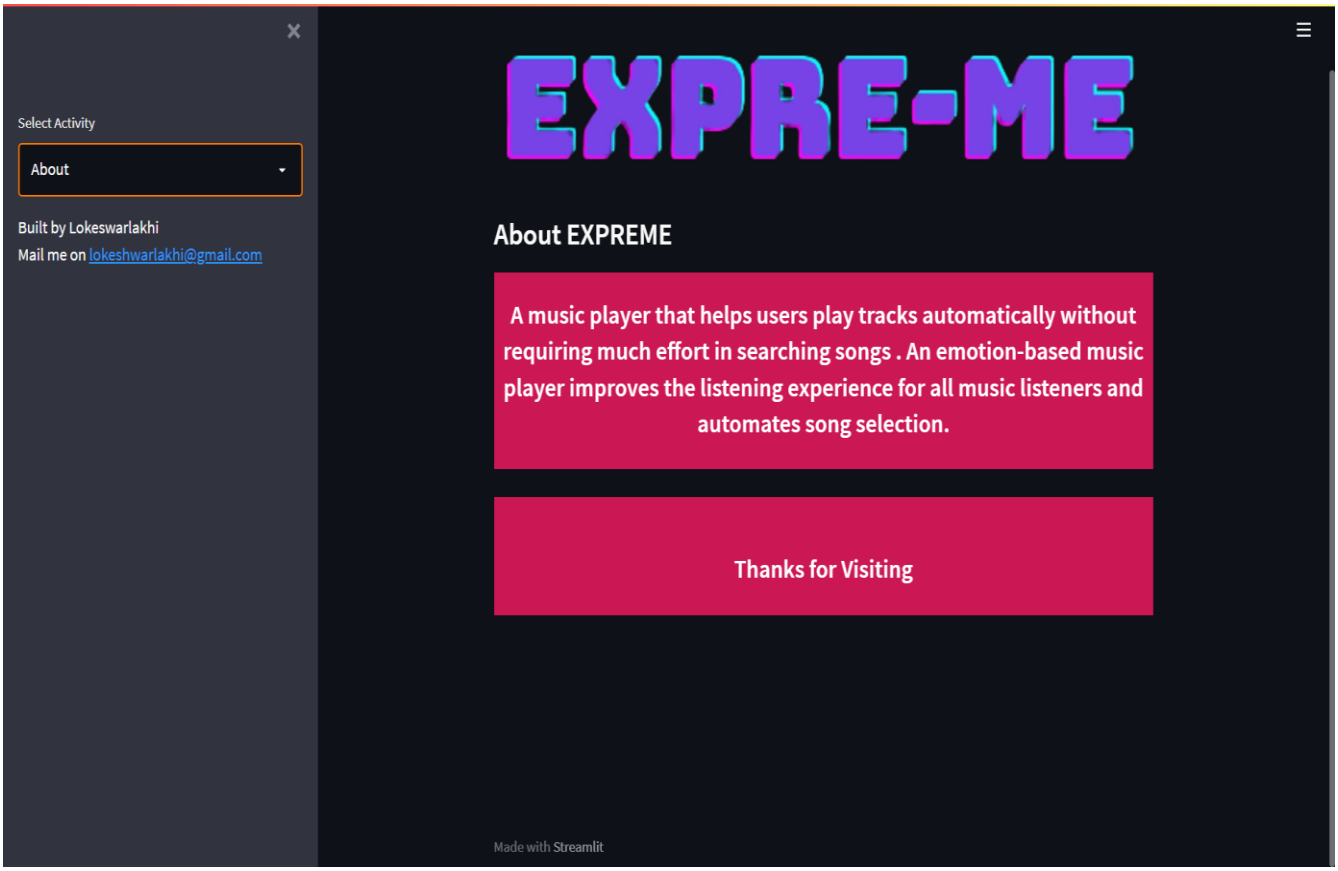
```
import spotipy
import json
import webbrowser
import creds
import random
import time

username = 'lokidoki'
client_id = creds.client_id
client_secret = creds.client_secrect
redirect_uri = 'http://google.com/'
cache = '.spotipyoauthcache'
SCOPE = 'user-library-read'


oauth_obj = spotipy.SpotifyOAuth(client_id, client_secret, redirect_uri)
token_dict = oauth_obj.get_access_token()
token = token_dict['access_token']

spotify_obj = spotipy.Spotify(auth=token)
```

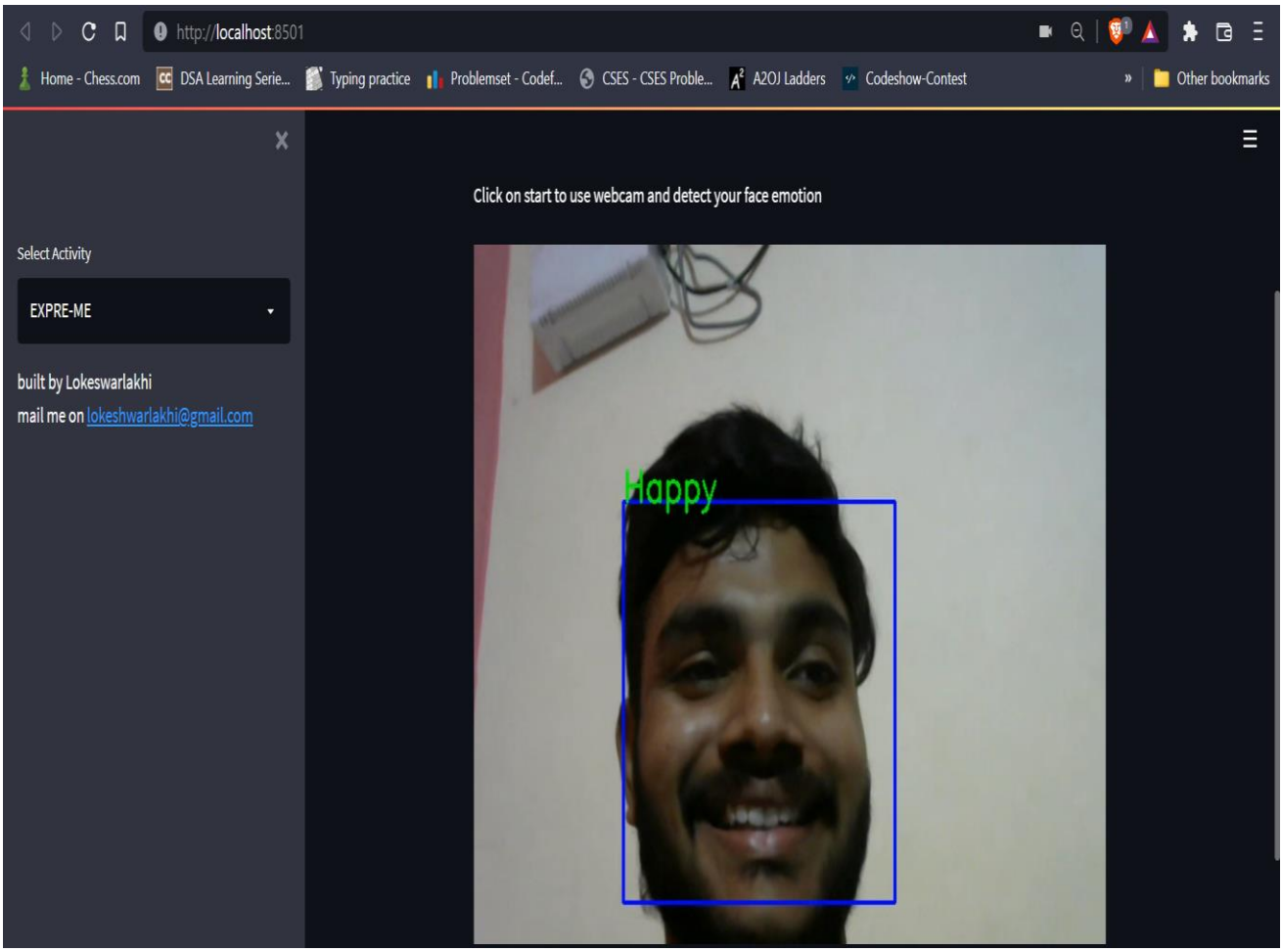
```
user = spotify_obj.current_user()

json.dumps(user)



def playsong(fang):

    # cnt = 0

 recommendation_dict = {

        "Angry"    : ['top metal songs', 'metal mix', 'rage workout',
'beast  mode',’angry telugu’,’river’],

        "Sad"      : ['english breakup songs', 'sad hindi songs',’sad songs hindi mood
off’,’sad melodies telugu’],

        "Neutral" : ['all time hit international songs', 'pop hits 2022', 'all time pop
hits'],

        "Happy"    : ['harry styles', 'elvis presely', 'telugu happy',’happy english
songs’,’happy mood’,’happy vibes’,’happy hits’,’happy favourites’],

        "Disgust" : [‘disgusting’,’disgusting life’,’dissapointed but not surprised’,’Am
disgusted with myself’],

        "Surprise": ['alan walker', 'martin garix',’Surprise me!’,No surprises’,’full of
surprises’],

        "Fear"     : [‘fear’s play,’fear garden playlists’,’fearless soul songs’,Fear]


    song_data = recommendation_dict[fang][random.randint(

        0, len(recommendation_dict[fang])-1)]
```

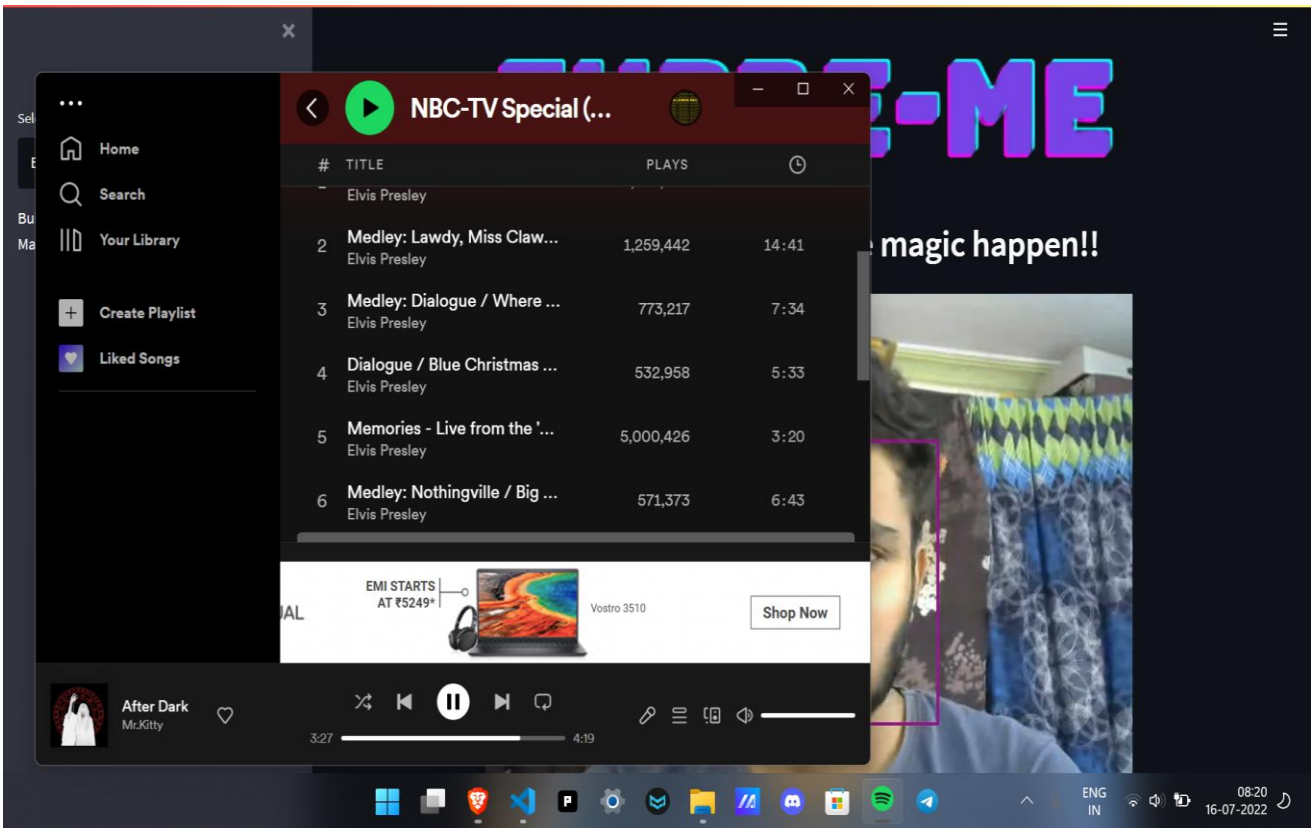
```
search_res = spotify_obj.search(song_data, 4, 0, 'playlist')
playlist_dict = search_res['playlists']

# webbrowser.open(sample_playlist)

# track_id, track_name = [], []
while(1):
    sample_playlist = playlist_dict['items'][random.randint(0, 3)]['uri']
    plst_id = sample_playlist.split(':')[2]
    plst = spotify_obj.user_playlist_tracks('spotify', plst_id)['items']
    ran_track = random.randint(0, len(plst)-1)
    track_id = plst[ran_track]['track']['id']
    track_name = plst[ran_track]['track']['id']
    sng_drtn = spotify_obj.audio_features(track_id)[0]['duration_ms']
    webbrowser.open(f'spotify:track:{track_id}')
    print(sng_drtn)
    time.sleep(sng_drtn/1000)
```

5.2 EXPREME UI:






http://localhost:8501
Home - Chess.com
DSA Learning Serie...
Typing practice
Problemset - Codef...
CSES - CSES Proble...
A2OJ Ladders
Codeshow-Contest
Other bookmarks
Click on start to use webcam and detect your face emotion
Select Activity
EXPRE-ME
built by Lokeswarlakhi
mail me on lokeshwarlakhi@gmail.com
Happy


magic happen!!
NBC-TV Special (...
Home
Search
Your Library
Create Playlist
Liked Songs
TITLE PLAYS
Elvis Presley
2 Medley: Lawdy, Miss Claw... 1,259,442 14:41
Elvis Presley
3 Medley: Dialogue / Where ... 773,217 7:34
Elvis Presley
4 Dialogue / Blue Christmas ... 532,958 5:33
Elvis Presley
5 Memories - Live from the '... 5,000,426 3:20
Elvis Presley
6 Medley: Nothingville / Big ... 571,373 6:43
Elvis Presley
EMI STARTS AT ₹5249*
Vostro 3510
Shop Now
After Dark
Mr.Kitty
3:27
4:19
ENG IN
08:20
16-07-2022

CHAPTER-6

SYSTEM ANALYSIS

6.1 Problem Statement

In the present-day environment, there is a lot of stress and fatigue, which leads to more despair and ill health. Medical studies demonstrate that interactive and high-quality games and shows, as well as music tailored to one's mood, can lift one's spirits. By reducing stress hormones, can aid in the treatment of depression along with helping in physical aches and pains induced by the high level of stress. However, additional research shows that listening to random music that is unrelated to one's mood can have a stressful effect on people. As a result, listening to music (premium ad-free or with advertisements) to unwind after work can improve one's health. A music player that helps users play tracks automatically without requiring much effort in searching songs. An emotion-based music player improves the listening experience for all music listeners and automates song selection by the facial expression of a human.

6.1.1 Existing System

Spotify is a digital music streaming service that gives access to million of songs podcasts and videos from artists all over the world.Majority of people of all ages like music, and we believe that music players should be capable of much more than merely playing songs and allowing users to create playlists. As a result, listening to music to unwind after work can improve one's health. For this, a music player should be clever and respond to the choices of the user. A music player should assist users in automatically organizing and playing songs without requiring much effort in song selection and reorganization. An Emotion-Based Music Player gives all music listeners a better platform and ensures song selection automation.

An Emotion-Based Music Player provides a better platform for all music listeners by automating song selection and updating playlists regularly based on the user's identified emotion. These assist users in organizing and playing songs based on their mood, making them less stressed. So, the goal is to design a recommender system on the music and emotion domain. The system will be composed of server-side components and client-side components. The server-side component will manage the database operations and algorithms that produce recommendation results. The client-side components will be graphical interfaces that are integrated into corresponding larger systems.

6.1.1.1 Disadvantages Of Existing System

1. Lack of Lyrical Features

Spotify removed its own feature for the lyrics. It is very difficult for people who are fond of reading the lyrics while listening to the songs. When you want to understand the meaning of the lyrics, you can use Genius' Behind the Lyrics, and you can also use Sound Hound to read your favourite album's lyrics.

2. Expensive

If you switch to the paid plan, you'll have to pay every month irrespective of whether or not you're using the service. At $120 per year for the Premium service, that's about 10 to 12 albums that you could buy from a vendor and actually own, rather than pushing money into Spotify's service, from which you basically just rent your music unless you decide to buy tracks from the firm.

3. Sound Quality

Weak quality of sound for free users. If you're a Spotify Free user, you need to note that when listening, the sound quality is poor and the rate is 160kbps, whereas the Premium users enjoy 320kbps of music streaming.

4. Not Available in all Countries

Available juts within a small num ber of countries. One of the biggest flaws in many video and music streaming services is the fact that a limited number of countries can only have access to them, same goes with Spotify.

5. Ads

Spotify users have to listen to audio advertisements between their favourite songs and podcast shows if they use the free account. Plus, you can't stream individual songs on demand, and you can skip just 6 tracks every hour. This is an extremely bad experience for users, especially when you hear things you don't like.

6. Limitations

By using a free account mobile app, you can listen to any songs in any order with unlimited track skips as long as they appear on one of the 15 personalized playlists selected by Spotify's machine learning algorithms for you.

You can stream a whole playlist or only 3,333 songs for offline use after you've upgraded to the Premium services, but there's no way to download one single song. When your subscription expires, you won't have access to any of the content you downloaded offline.

6.1.2 Proposed system

Music can be categorized into several genres, such as pop, rock, jazz, blues, folk etc. Listening to music in the digital age is easier because of the features on the smartphone that can play music offline and online. Nowadays, the availability of digital music is very abundant compared to the previous era, so to sort out all this digital music is very time consuming and causes information fatigue. Therefore, it is very useful to develop a music recommender system that can search music libraries automatically and suggest songs that are suitable for users. Music streaming applications like Spotify and Pandora have features to recommend music to users. These features can help to get a list of appropriate music from the popular music libraries based on music that has been heard previously. This makes the recommender system play an important role in maintaining the streaming music business. Music recommendations are done by looking for similarities from one music to another or by giving preference from one user to another The challenge of music recommender system is to create a system that can continually find attractive new music which understand the users' preferences in music This requires that the music personalized recommender system should effectively reflect the personal preferences. It needs adjustments to achieve personalized recommendations for the needs of different audiences. Therefore, the music personalized recommender system is a more complicated than the general recommender system. It is necessary to consider user needs comprehensively and combines the music feature recognition and audio processing technologies to extract the music features. A music player that helps users play tracks automatically without requiring much effort in searching songs. An emotion-based music player improves the listening experience for all music listeners and automates song selection by our facial expression.This was the project of system design about "Music Recommendation System using Countenance Recognition" supported python. Development of this technique takes tons of effort as different technologies are needed to be integrated with the software. this technique can provide a lot of satisfaction to the music lovers and therefore the users. However, not every task is claimed to be perfect during this development field. Even more improvements could also be possible during this system, but the most motto of this project is to play songs consistent with the emotion of the person and it's somehow satisfying as of now.

6.1.2.1 Advantages of Proposed System

1. A music player assisting users in automatically orgaizing and playing songs without requiring much effort in song selection and recongnization. An emotion based music player gives all music listeners a better platform and ensure song selection automation.

6.2 Objectives

The main goal of the project is to reduce the human efforts. A music player that helps users play tracks automatically without requiring much effort in searching songs. An emotion-based music player improves the listening experience for all music listeners and automates song selection by our facial expression.

The main objective of the web development are as follows:

1. Reduced operational time
2. Increased accuracy and reliability
3. Reduced the human efforts
4. Easy maintains of data

6.3 Feasibility of study

Feasibility study is used to check whether the particular system is practical in the real world. It studies the influence of the system in the organization during the development stage of the system. It also checks whether the system efficiently uses its resources. If the impact of the system in organization is acceptable, then the system can be considered as feasible system.Student Grievance Redressal System uses these ways of feasibility study such as Technical feasibility, Economic feasibility, Legal feasibility, Operational feasibility and Scheduling feasibility.

6.3.1 Technical Feasibility

The system should be technically feasible. The system should be designed and developed by the person with good technical knowledge. Expre-Me meets all technical aspects need for the system.

A. KERAS: Keras is that the most used deep learning framework among top-5 winning teams on Kaggle. Because Keras makes it easier to run new experiments, it empowers you to undertake more ideas than your competition, faster. And this is how you win.

B. NUMPY: NumPy may be a library for the Python programing language, adding support for giant, multi-dimensional arrays and matrices, alongside an outsized collection of high-level mathematical functions to work on these arrays.

C.REQUESTS: Requests may be a HTTP library for the Python programing language. The goal of the project is to form HTTP requests simpler and more human-friendly.

D. OPENCV :OpenCV may be a library of programming functions mainly aimed toward real-time computer vision.firstly developed by Intel, it had been latterly supported by Willow Garage also Itseez. The library is cross-platform and free to be used under the open-source Apache 2 License.

E. TENSORFLOW: TensorFlow may be a free and open-source software library for machine learning and AI. It is often used across a variety of tasks but features a particular specialise in training and inference of deep neural networks.

6.3.2 Economic feasibility

The system should be economically feasible. The cost for the development of the system is low for the organization. The development process is achieved by using the needed resources, so organizations need not invest more and so, the system can be considered as economically feasible. Expre-Me system is a low cost application that is developed for the purpose of public users. It is useful for the citizens.

6.3.3 Legal Feasibility

The system is legally feasible. It checks whether the proposed system conflicts the legal requirements. The security level of the application is checked.

CHAPTER-7

REQUIREMENTS AND SPECIFICATION

7.1 System specifications

7.1.1 Software requirements

1. Operating system
2. Installed PIP packages
3. TENSORFLOW
4. Keras
5. NumPy
6. Pandas

7.1.2 Hardware requirements

1. 16 GB RAM
2. 512 GB SSD

7.2 Development Tools

1. We made our model using DeepLearning
2. Languages used : Python
3. We used Visualstudio as editor

7.3 Features

1. Inexpensive
2. User friendly
3. Reduced operational time
4. Increased accuracy and reliability
5. Reduced the human efforts
6. Easy maintains of data

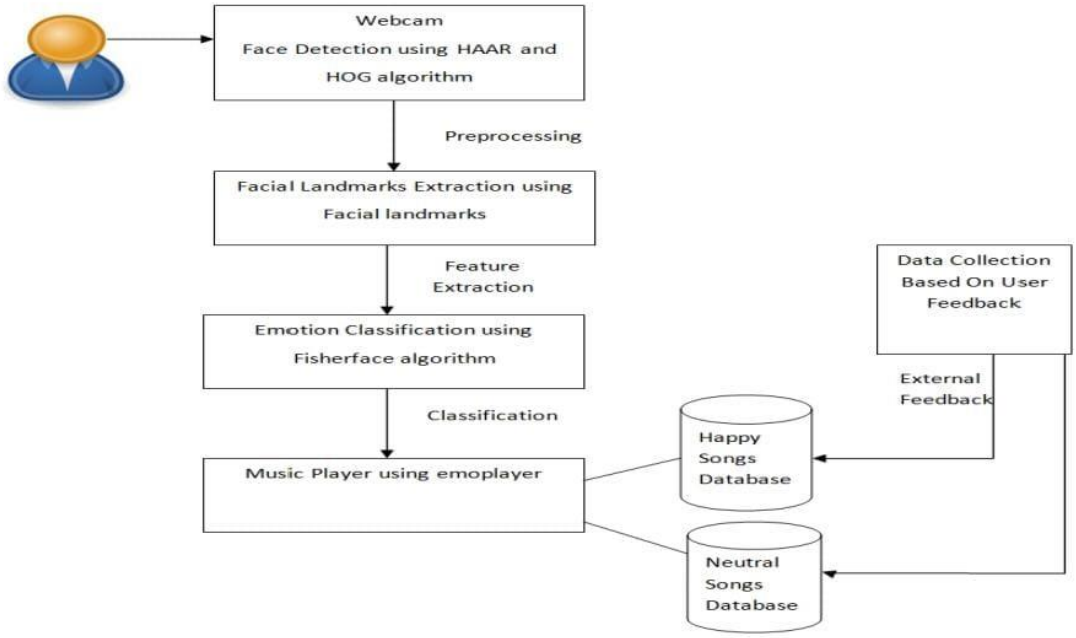
7.4 Proposed Architecture

This project has been divided into three phases.

(i) Reading the facial expression

(ii) Picking the playlist based on the expression

(iii) Playing a song from the playlist



The proposed system can detect the facial expressions of the user and based on his/her facial expressions extract the facial landmarks, which would then be classified to get a particular emotion of the user. Once the emotion has been classified the songs matching the user's emotions would be shown to the user.

CHAPTER-8

CONCLUSION

8.1 Conclusion

We have proposed a design for successfully implementing a platform that uses a deep learning algorithm (Convolutional Neural Network) to recognize emotions and play music accordingly. The Convolutional Neural Network technique works well to create a face recognition module to showcase accurate detection results.Convolutional neural networks remain a growing area of research in automated song selection.The main goal of the project is to reduce human efforts. A music player that helps users play tracks automatically without requiring much effort in searching songs. An emotion-based music player improves the listening experience for all music listeners and automates song selection by our facial expression.

8.2 Future Scope

- Mobile application can be developed for this system.
- This system can be upgraded by including more features like language facility.
- Reduce the time required to train the classifier.
- Use of EEG signals to make the software even more optimized and to detect the exact mood /emotion of the user.

CHAPTER-9

REFERENCES

9.1 References

[1] Londhe RR and Pawar DV 2012 Analysis of facial expression and recognition based on statistical approach International Journal of Soft Computing and Engineering 2

[2] Kabani H, Khan S, Khan O and Tadvi S 2015 Emotion based music player International Journal of Engineering Research and General Science 3 750-6

[3] Gupte A, Naganarayanan A and Krishnan M Emotion Based Music Player-XBeats International Journal of Advanced Engineering Research and Science 3 236854

[4] Hadid A, Pietikäinen M and Li SZ 2007 Learning personal specific facial dynamics for face recognition from videos International Workshop on Analysis and Modeling of Faces and Gestures pp1-15 Springer Berlin Heidelberg

[5] Zeng Z, Pantic M, Roisman GI and Huang TS 2008 A survey of affect recognition methods Audio, visual, and spontaneous expressions IEEE transactions on pattern analysis and machine intelligence 31 39-58

[6] Patel AR, Vollal A, Kadam PB, Yadav S and Samant RM 2016 MoodyPlayer a mood based music player Int. J. Comput. Appl. 141 0975-8887

[7] ParulTambe, YashBagadia, Taher Khalil and Noor UlAin Shaikh 2015 Advanced Music Player with Integrated Face Recognition Mechanism International Journal of Advanced Research in Computer Science and Software Engineering 5

[8] Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z and Matthews I 2010 The extended cohnkanade dataset (ck+) A complete dataset for action unit and emotion-specified expression In 2010 ieee computer society conference on computer vision and pattern recognitionworkshops 94-101 IEEE

[9] Kanade T, Cohn JF and Tian Y 2000 Comprehensive database for facial expression analysis InProceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition 46-53 IEEE

[10] Luoh L, Huang CC and Liu HY 2010 Image processing based emotion recognition In2010 International Conference on System Science and Engineering 491-494 IEEE

[11] Vivek JD, Gokilavani A, Kavitha S, Lakshmanan S and Karthik S 2017 A novel emotion recognition based mind and soul-relaxing system In2017 International Conference on Innovations in Information, Embedded and Communication Systems 1-5 IEEE

[12] Jyoti Rani and Kanwal Garg 2014 Emotion Detection Using Facial Expressions A Review International Journal of Advanced Research in Computer Science and Software Engineering 4

[13] Joshi A and Kaur R 2013 A Study of speech emotion recognition methods Int. J. Comput. Sci. Mob. Comput. 2 28-31

[14] Shoaib M, Hussain I, Mirza HT and Tayyab M 2017 The role of information and innovative technology for rehabilitation of children with autism a systematic literature review In2017 17th International Conference on Computational Science and Its Applications 1-10 IEEE

[15] Dubey M and Singh L 2016Automatic emotion recognition using facial expression a review International Research Journal of Engineering and Technology (IRJET) 3 488-92