MUSIC GENRE CLASSIFICATION USING MACHINE LEARNING

A UG PHASE-1 PROJECT REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In Partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

MUPPA VARUN: 19UK1A05N3

MUDDU AVANTHIKA: 19UK1A05N6

NEELAM KRANTHI KUMAR: 19UK1A05M7

PUNNAM SAI GANESH: 20UK15A0521

Under the esteemed guidance of

MR.CH SHIVA SAI PRASAD

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

(Affiliated to JNTUH, Hyderabad)

Bollikunta, Warangal – 506005

(2019-2023)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

BOLLIKUNTA, WARANGAL – 506005
2019 – 2023

CERTIFICATE

This is to certify that the UG ProjectPhase-1entitled **"MUSIC GENRE CLASSIFICATION USING MACHINE LEARNING"** is being submitted by **M.VARUN (H.NO:19UK1A05N3), M.AVANTHIKA (H.NO:19UK1A05N6), N.KRANTHIKUMAR(H.NO:19UK1A05M7), P.SAIGANESH (H.NO:20UK5A0521).** In partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering to Jawaharlal Nehru Technological University Hyderabad** during the academic year 2022-23, is a record of work carried out by them under the guidance and supervision**.**

Project Guide	Head of the Department
MR .CH.SHIVA SAI PRASAD	**Dr. R. Naveen Kumar**
(Assistant Professor)	(Professor)

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO**, Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this MAJOR Project in the institute.

We extend our heartfelt thanks to **Dr.R.NAVEEN KUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and there by giving us freedom to carry out the MAJOR Project.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the UG Project Phase-1 and for their support in completing the MAJOR Project .

We express heartfelt thanks to the guide, **MR.CH.SIVA SAI PRASAD** (Assistant professor), Department of CSE for her constant support and giving necessary guidance for completion of this MAJOR Project .

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

M.VARUN	**(19UK1A05N3)**
M.AVANTHIKA	**(19UK51A0N6)**
N. KRANTHI KUMAR	**(19UK1A05M7)**
P.SAI GANESH	**(20UK5A0521)**

ABSTRACT

The aim of this work is to predict the genres of songs by using machine learning techniques. For this purpose, feature extraction is done by using signal processing techniques, then machine learning algorithms are applied with those features to do a multiclass classification for music genres. Music information retrieval (MIR) has been an active field of research for the past decade, especially in retrieving music genre information as genre classification for music files during distribution are still reliant on manual annotations. The purpose of this research is to determine the best set of hyper parameters for the best performing machine learning models in classifying music genres based on Mel Frequency Cepstral Coefficient (MFCC) audio features. The paper also explores the best feature selection techniques to be used in tandem with the best performing machine learning models. From the results, it was found that the Extra Trees ensemble model yielded the highest accuracy among the 8 models implemented in this paper with an accuracy score of 68.5%. The greedy feature selection method provided the best accuracy improvements for the K-Nearest neighbors model, but reduced accuracy rates for Random Forest and Extra Trees models.

TABLE OF CONTENTS

1. Introduction

    1.1 Overview……………………………………. 7

    1.2 Purpose……………………………………… 8

2. Literature Survey

    2.1 Existing Problems………………………………. 9

    2.2 Proposed Solution………………………………. 9-10

    2.2.1 K-Nearest Neighbour Algorithm (KNN)……... 10-13

3. Theoretical Analysis

    3.1 Block Diagram………………………………….. 14

    3.2 Technical Architecture………………………….. 14

    3.3 Hardware Requirements………………………… 15

    3.4 Software Requirements………………………….. 15

4. Experimental Investigations

    4.1 Problems We Faced……………………………… 16

5. Flow Chart……………………………………… 17

06. Result

    6.1 Final findings (Output) of the Project…………... 18-23

7. Advantages and Disadvantages…………………... 24

8. Applications………………………………………. 25

9. Conclusion……………………………………….. 26

10. Future Scope……………………………………… 26

11. Bibliography………………………………………. 27-28

1 INTRODUCTION

1.1 OVERVIEW:

The proposed system uses three different types of media feature extraction techniques. These include pectral centroid features. In addition, support vector spectral centroid features. In addition, support vector machines (SVM), k-nearest neighbor classifiers, and a multilayer perceptron were used as the base learners in order to perform automatic music genre classification.Each method was tested on a set of 100 songs (each song is represented by a 10-second segment). For each song, three classifiers were trained, based on different data sets: training data with 1000 samples (1000 songs),training data with 5000 samples (5000 songs) and test data with 3000 samples (3000 songs). Accuracy was calculated using confusion matrices. The results show that the accuracy of the multilayer perceptron is higher than other methods; therefore the chosen method is Multilayer Perceptron (MLP).Th e results show that the proposed technique out performed other methods, achieving a classification accuracy of 91.7%

1.2 PURPOSE:

Digital Music Services like Spotify, Apple Music, etc., offers streaming music from more than 50 million tracks, uses a recommendation engine based on machine learning to help users discover new music. Based on user data, the company uses machine learning algorithms to learn what kinds of music people listen toand then recommends similar artists and songs to them. A common method of classifying musical genres is based on song attributes. These attributes include instruments used, chord progressions, and rhythm patterns [1]. In order to determine how well a particular genre fits into a certain category, it is a must to first understand what makes up a genre, and then to identify the most important attributes that defi ne the genre. Once this is accomplished, the data can be used to traina machine-learning algorithm to predict the genre of new songs. Music streaming companies could use such models to automatically classify and recommend songs based on user preferences

machines (SVM), k-nearest neighbor classifiers, and a multilayer perceptron were used as the base learners in order to perform automatic music genre classification

2 LITERATURE SURVEY

2.1 EXISTING PROBLEM :

Music plays a very important role in people’s lives. Music bring likeminded people together and is the glue that holds communities together. Communities can be recognized by the type of songs that they compose, or even listen to. Different communities and groups listen to different kinds of music. One main feature that separates one kind of music from another is the genre of the music.

The aim of this project is:

- To build a machine learning model which classifies music into its respective genre.

- To compare the accuracies of this machine learning model and the pre-existing models, and draw the necessary conclusions.

2.2 PROPOSED SOLUTION:

Music genre classification forms a basic step for building a strong recommendation system.

The idea behind this project is to see how to handle sound files in python, compute sound and audio features from them, run Machine Learning Algorithms on them, and see the results.

In a more systematic way, the main aim is to create a machine learning model, which classifies music samples into different genres. It aims to predict the genre using an audio signal as its input. The objective of automating the music classification is to make the selection of songs quick and less cumbersome. If one has to manually classify the songs or music, one has to listen to a whole lot of songs and then select the genre. This is not only time-consuming but also difficult. Automating music classification can help to find valuable data such as trends, popular genres, and artists easily. Determining music genres is the very first step towards this direction.

2.2.1 K-Nearest Neighbour Algorithm (KNN)

One of them, K-Nearest Neighbour (KNN), is a technique that has been reportedly successful in categorizing music into different genres. Let us find out how.

A supervised machine learning algorithm, the K-Nearest Neighbour technique is used to find solutions for classification and regression problems. Relying on labelled input data to process unlabelled data in the future, this ML technique is used in music genre classification.

" The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. " KNN Algorithm - ML

The KNN algorithm, when implemented in music genre classification, looks at similar songs and assumes that they belong to the same category because they seem to be near to each other. Among various other techniques that prevail in this concept, the best results have been procured out of this technique.

One of the simplest ML techniques, the KNN algorithm interprets data in a manner such that when the new data is fed, the machine automatically recognizes it and categorizes it as per the similarity of features.

What's more is that a particular set of traits of a particular music genre make it different from others which, in turn, helps machines to readily classify new data inputs.

Here we will understand the step-by-step process of music genre recognition. In the process of music genre classification, any technique of Machine Learning constitutes 5 steps. These are as follows:-

- **Prerequisite Data**

The prerequisite data involves past datasets that are required by machines to analyse past information and build analysis on that basis. The prerequisite for machine learning, in this the most necessary step involved in preparing machines for music genre classification using Machine Learning Algorithm.

- **Theoretical Foundation**

The theoretical foundation of this concept implies that different techniques of Machine Learning can be incorporated to classify music into different genres based on pre-data sets and their subsequent analysis. It is important for us to understand the theory of music genre classification before proceeding to the next step. The KNN algorithm, which is considered to be the most successful algorithm, in this case, is involved in this theory.

- **Data Preprocessing Analysis**

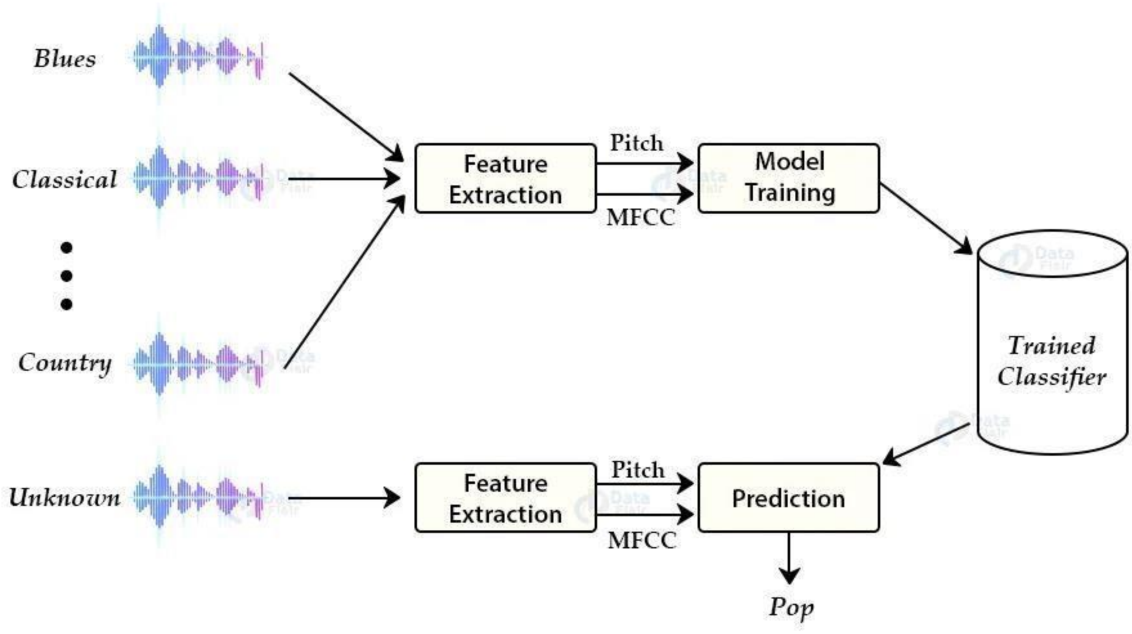
The training dataset involves feeding the machine with new data that will test the efficacy of the algorithm. The music genre classification dataset can be procured in any manner since it only requires random songs that can be classified by the algorithm into different genres.

- **Testing New Inputs**

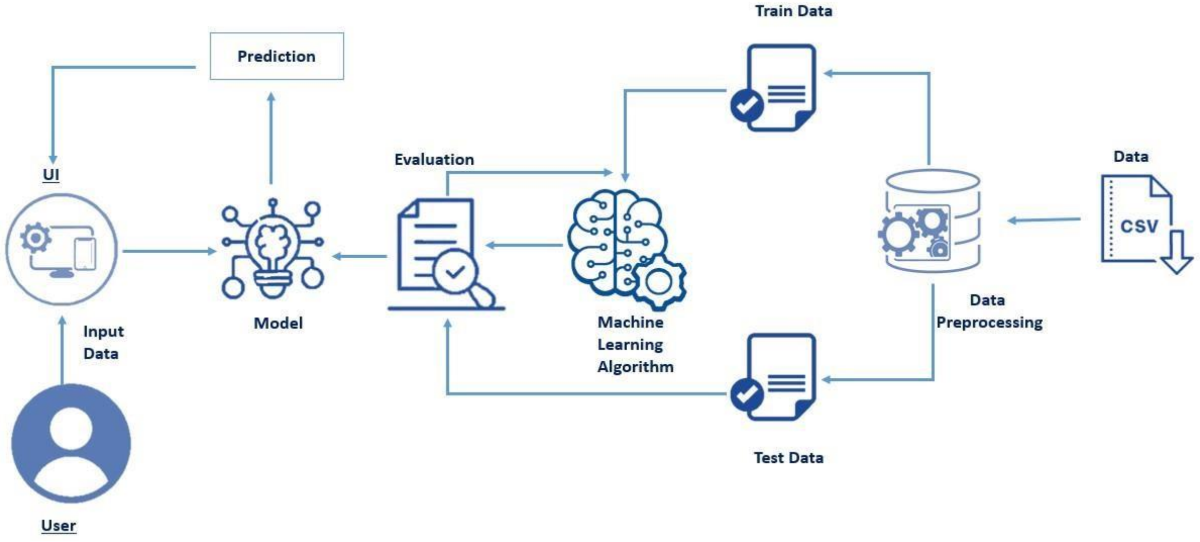
The last step, testing new inputs, refers to the machine working on the training data and testing data so that the algorithm can now filter music and carry out the task of music genre recognition. This step will provide results that will ultimately reveal the success of this algorithm.

3 THEORETICAL ANALYSIS

3.1 BLOCK DIAGRAM:



3.2 TECHINACAL ARCHITECTURE:



Hardware and Software Requirements:

Requirements are the minimal configurations of a device and software required for the model to work properly and efficiently.

3.3Hardware requirements:

- Graphics Processing Unit (GPU).
- Intel Core i3 processor or above.

3.4 Software requirements:

- Windows 7 or above / Linux.
- Python 2.7 or above. Jupyter Notebook.

4 EXPERIMENTAL EVALUATION

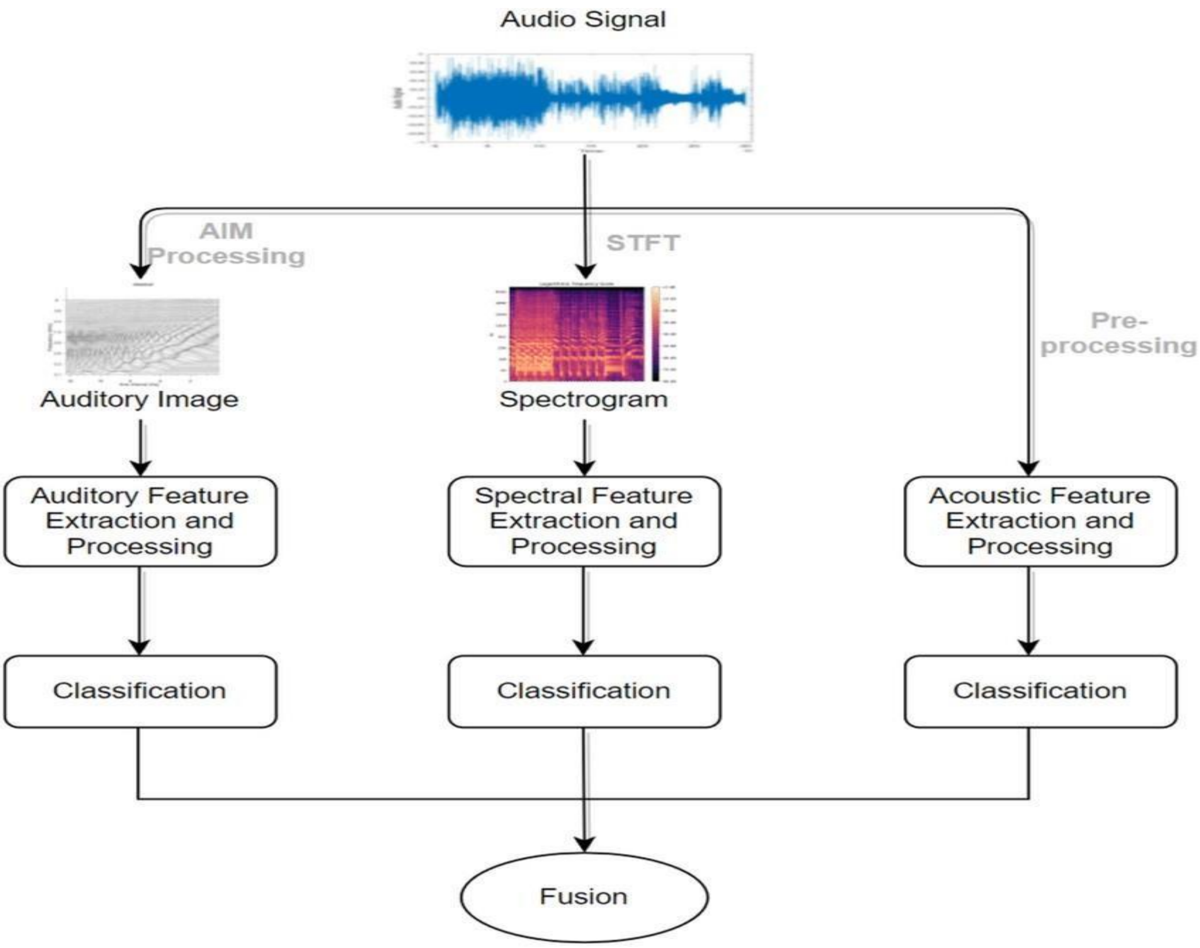
Machine learning models are evaluated using these metrics:

Confusion Matrix: Confusion matrices help us understand how good our models are at classifying new examples. In this visualization it is clear how well our model classifies items into one category (positive/negative) from another. Accuracy: It is the percentage of correct classifications made by an algorithm for a given dataset. 3Repeated, 10-Fold Validation Accuracy: After repeating the experiment three times, it gives an average value for each fold (i.e., one run). Then take the average of these values across the folds. To improve the reliability of our classifications, it has to be ensured that there isn't any bias introduced by splitting the dataset into test in gand training sets.• Training Time: It is the time taken for fitting the training set into a model. it is either measured in milli seconds or seconds

4.1 PROBLEMS WE FACED

- Inadequate Training Data
- Poor Quality of Data
- Overfitting and Underfitting
- Slow Implementations and result
- Problem in recognition of music

5.FLOW CHART


Audio Signal
AIM Processing
STFT
Pre-processing
Auditory Image
Spectrogram
Auditory Feature Extraction and Processing
Spectral Feature Extraction and Processing
Acoustic Feature Extraction and Processing
Classification
Classification
Classification
Fusion

6.RESULT

6.1 Final findings (Output) of the project:

Libraries Installation

```
!pip install python_speech_features

!pip install scipy
```

Step-1) Import required libraries

```
  import numpy as np

import pandas as pd

import scipy.io.wavfile as wav from
python_speech_features import mfcc from
tempfile import TemporaryFile import os import
math import pickle import random import
operator
```

Step-2) Define a function to calculate distance between feature vectors, and to find neighbours.

Step-3) Identify the class of nearest neighbours

```
#define a function to get distance between feature vectors and find
neighbors def getNeighbors(trainingset, instance, k):
    distances = []     for x in
range(len(trainingset)):
        dist = distance(trainingset[x], instance, k) +
distance(instance,trainingset[x],k)
```

```
        distances.append((trainingset[x][2], dist))
distances.sort(key=operator.itemgetter(1))
neighbors = []     for x in range(k):
neighbors.append(distances[x][0])     return
neighbors
```

```
#function to identify the nearest neighbors def
nearestclass(neighbors):      classVote = {}
for x in range(len(neighbors)):
response = neighbors[x]         if response in
classVote:
classVote[response] += 1         else:
classVote[response] = 1

    sorter = sorted(classVote.items(), key=operator.itemgetter(1),
reverse=True)     return sorter[0][0]
```

Step-4) Model Evaluation

```
def  getAccuracy(testSet,  prediction):
correct   =   0            for   x   in
```

```
range(len(testSet)):                 if
testSet[x][-1]     ==     prediction[x]:
correct += 1     return 1.0 * correct /
len(testSet)
```

Step-5) Feature Extraction

```
directory = '../input/gtzan-dataset-
musicgenreclassification/Data/genres_original'
```

```
f = open("mydataset.dat", "wb") i = 0
for folder in os.listdir(directory):
    #print(folder)     i += 1     if i == 11:
break     for file in
os.listdir(directory+"/"+folder):
#print(file)         try:
            (rate, sig) = wav.read(directory+"/"+folder+"/"+file)
mfcc_feat = mfcc(sig, rate, winlen = 0.020, appendEnergy=False)
covariance = np.cov(np.matrix.transpose(mfcc_feat))       mean_matrix
= mfcc_feat.mean(0)             feature =
(mean_matrix, covariance, i)                pickle.dump(feature, f)
except Exception as e:  print("Got an exception: ", e, 'in folder: ',
folder, ' filename: ', file)
f.close()
```

Step-6) Train-test split the dataset

```
dataset = []
 def loadDataset(filename, split, trset,
teset):     with open('my.dat','rb') as f:
```

```
while True:              try:
dataset.append(pickle.load(f))
except EOFError:
                f.close()                    break
```

```
    for x in range(len(dataset)):         if
random.random() < split:
trset.append(dataset[x])
else:
teset.append(dataset[x])
  trainingSet = []
testSet
= []
loadDataset('my.dat', 0.68, trainingSet, testSet)
```

Step-7) Calculate the distance between two instance

```
def distance(instance1, instance2,
k):      distance = 0     mm1 =
instance1[0]     cm1 = instance1[1]
mm2 = instance2[0]     cm2 =
instance2[1]
    distance = np.trace(np.dot(np.linalg.inv(cm2), cm1))
distance    +=    (np.dot(np.dot((mm2-mm1).transpose(),
np.linalg.inv(cm2)),  mm2-mm1))            distance  +=
np.log(np.linalg.det(cm2))  -  np.log(np.linalg.det(cm1))
distance
-= k     return distance
```

Step-8) Training the Model and making predictions

```
# Make the prediction using KNN(K nearest Neighbors)

length = len(testSet) predictions = [] for x in

range(length):

    predictions.append(nearestclass(getNeighbors(trainingSet,
testSet[x], 5)))


accuracy1 = getAccuracy(testSet, predictions)

print(accuracy1)
```

Step-9) Test the Classifier with the new Audio File

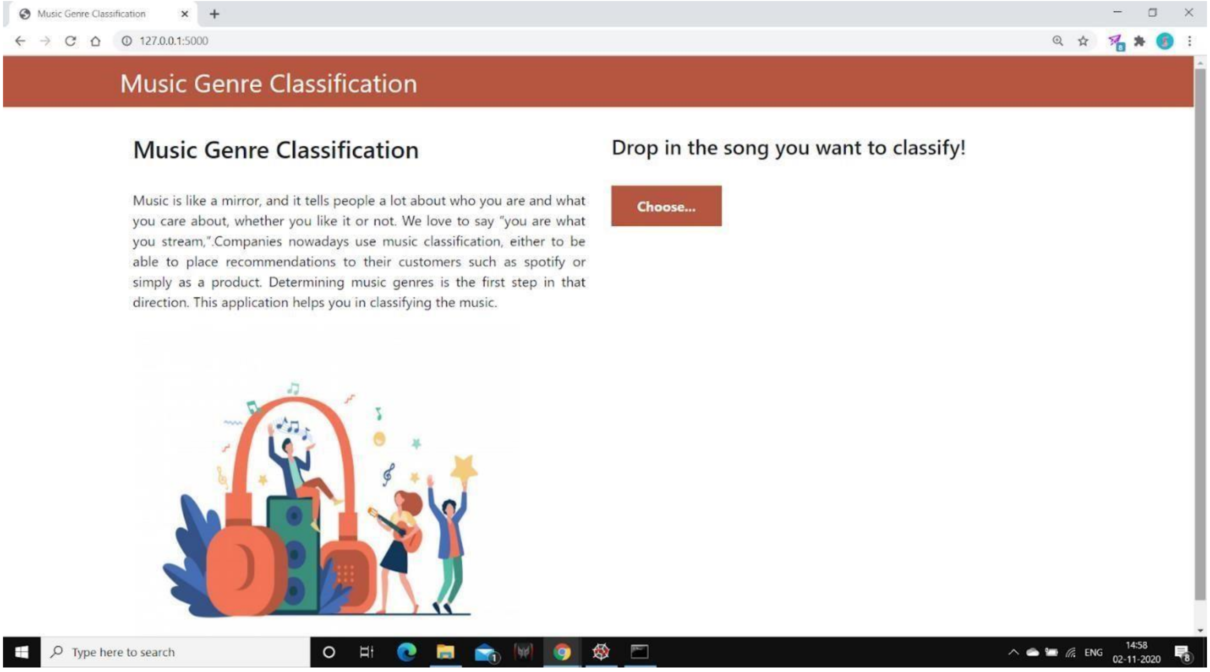
```
from collections import defaultdict  results

= defaultdict(int)

  directory = "../input/gtzan-dataset-
musicgenreclassification/Data/genres_original"
i = 1 for folder in os.listdir(directory):
    results[i] = folder

i += 1
```

Now we can predict a new audio file and receive a label for it and print the name of the category using the result dictionary.



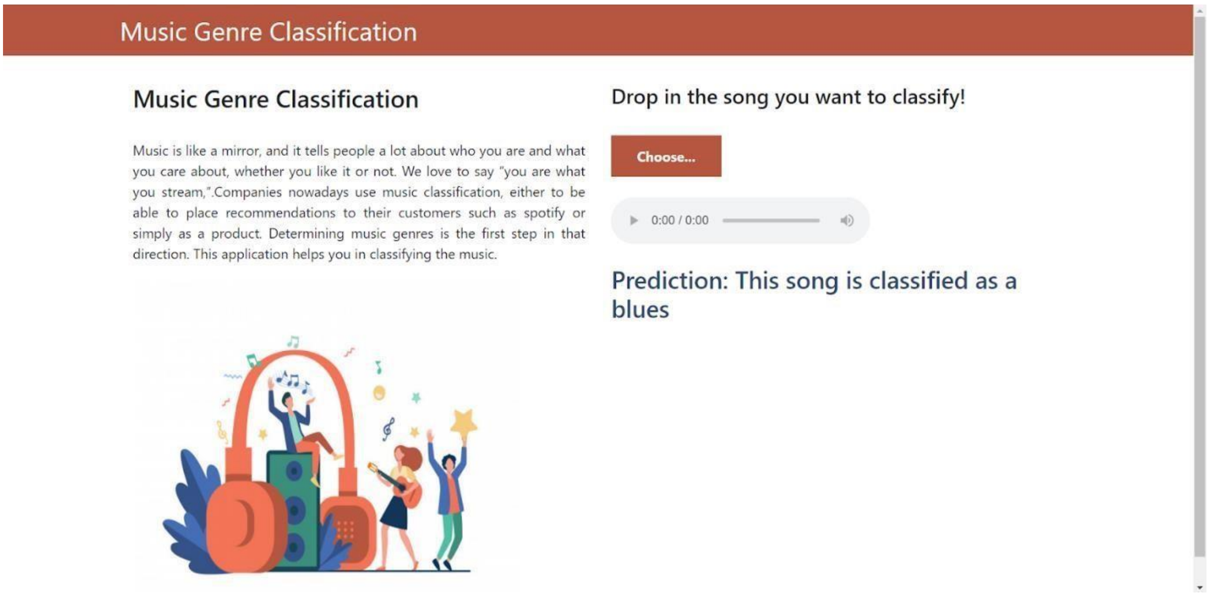


Fig. Images of the web pages

7 ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- It proposes a method for automatically classifying music by assigning tags to each song

- It investigated how audio signals are categorised into a music genre hierarchy automatically

- Create the most accurate music genre classification model possible and extract musical patterns from the audio file.

- When using Audio set data, the performance of various models and music genres varies greatly.

DISADVANTAGES:

- The proposed feature set is useless for query-by-example retrieval of musical signal similarities and audio thumb-nailin

- There is no effective way to use audio content analysis to aid in video content analysis and indexing.

- The model is insufficiently robust to apply the training results to previously unknown musical data.

- Some audio characteristics derived from the raw audio signals were left out.

8 APPLICATIONS

Malls:

Music is played continuously in the malls, and selection of right music to be played is hectic as well as time consuming work. So here, our system helps to choose the song according to any occasion or event.

Restaurants:

In a restaurant, choosing the right music is an important task when it comes to various occasions as per customer’s demand; our system will help to choose a particular genre song for the same.

Airports:

Music is played in the airports for the entertainment of people as they wait for hours due to various reasons, so our system will help to choose the song as per the requirements.

9 CONCLUSION

In this paper, a decent accuracy rate of almost 70% was achieved in classifying genres for 1000 songs based on musical features such as tempo, beat, MFCCs etc. in a dataset consisting of 10 music genres. This was done through the use of various preprocessing techniques such as scaling, categorical mapping, feature selection and outlier detection, as well as hyper-parameter optimizations. However, the various feature selection methods implemented were not able to improve upon the accuracy scores after hyperparameter optimization. LASSO and Ridge based feature selection methods yield the most consistent and gave the least accuracy reduction scores from the hyperparameter optimization phase.

10 FUTURE SCOPE

A deeper exploration of feature selection techniques should definitely be done in the future, particularly combining the use of filter feature selection techniques in wrapper based search of optimal feature subsets. Besides that, future exploration of more audio features such as chord progressions and voice timbres could be incorporated along with MFCC for possible improvements in music genre recognition and classification.

11 BIBILIOGRAPHY

[1] ’The Echo Nest - Powering Intelligent Digital Music Applications.’ The Intelligent Music Application Platform. Web. 07 May 2012. http://the.echonest.com/

[2] ’Last.fm.’ Last.fm. Web. 07 May 2012. http://www.last.fm/

[3] Category-based intrinsic motivation,Proceedings of the Ninth International Conference on Epigenetic Robotics,Rachel Lee, Ryan Walker, Lisa Meeden, and James Marshall (2009)

[4] A growing neural gas learns topologies Advances in Neural Information Processing Systems 7 , Bernd Fritzke (1995)

[5] ‘Musical Genre Classification of Audio Signals’,George Tzanetakis and Perry Cook,IEEE Transactions on Speech and Audio Processing, 10(5), July 2002

[6] ‘Scanning the Dial: The Rapid Recognition of Music Genres’, Robert o. Gjerdingen and David Perrot Journal of New Music Research, Vol. 37, No. 2. (2008), pp. 93-100

[7] ‘Running with Data.’ Danceability and Energy: Introducing Echo Nest Attributes.Web.07 May 2012. http://runningwithdata.com/post/1321504427/danceabilityandenergy.

[8] ‘Overview Network X 1.6 Documentation.’ Overview Network X 1.6 Documentation. Web. 09 May 2012. http://networkx.lanl.gov/. 11 [9] ‘Intro.’ Matplotlib: Python Plotting Matplotlib V1.1.0 Documentation. Web. 09 May 2012. http://matplotlib.sourceforge.net/.

[9] Musical Genre Classification of Audio Signal’,George Tzanetakis and Perry Cook,IEEE Transactions on Speech and Audio Processing, 10(5), July 2002.

[10]https://docs.google.com/presentation/d/1Y024bJxOXtvH163T e59dOHUYZXpMY3 nDPI2lb48RKo/htmlpresent

[11]https://www.analyticsvidhya.com/blog/2022/03/musicgenreclassificationproject-using-machine-learning-techniques/

[12]https://www.researchgate.net/publication/324218667_Music_Genre_Classification_using_Machine_Learning_Techniques

[13]https://www.researchgate.net/publication/329396097_Music_Genre_Classification_and_Recommendation_by_Using_Machine_Learning_Techniques.