CS354N
Project Report

Mir Razee Mohideen (200001045)
Nishit Singh (200001056)

Music Genre Classification

Table of contents

01

Introduction

Description of the problem

02

Data

Collection of data and preprocessing

03

Algorithm

The model architecture and evaluation of the same

04

Results

Performance of the model, Conclusion and Further work

01
Introduction

Introduction

- Music genre classification is the task of automatically categorizing a piece of music into one or more predefined genres, such as rock, pop, jazz, classical, etc.
- It involves using machine learning algorithms to analyze the acoustic features of a song, such as its rhythm, melody, harmony, and timbre, and using those features to make predictions about which genres the song belongs to.

Importance of Music Genre Classification

- Music streaming platforms like Spotify, Apple Music use genre classification to create personalized playlists and recommendations for their users
- It helps music critics and journalists to write more informed reviews and analyses of music.
- It is used by record labels and producers to understand the market for their music and to target their market efforts towards specific audiences.

Importance of Music Genre Classification Cont.

- Music genre classification is used to help identify instances of copyright infringement and ensure that artists are properly compensated for their work.
- It is used by musicologists and other researchers to study trends and patterns in different genres of music over time

02
Data Processing

Data Collection

Large number of datasets are available for music genre classification namely:

- GTZAN Dataset (https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification)
- MSD-1 Dataset (https://zenodo.org/record/1240485#.Y_XEenZBw2w)
- CAL-500 Dataset

We will be using the GTZAN dataset.

GTZAN Dataset

- The dataset consists of 1000 audio tracks.
- Each audio file is 30 seconds long.
- It has 10 classes, each represented by 100 tracks.

Data Preprocessing and Visualization

We did the following in Data Preprocessing and Visualization:

- GTZAN dataset was first checked for corrupted audio files
- Only **one corrupted audio file** was found in the dataset and it was removed.
- The features dataframe was checked for NA values but none were found.
- Plotted Spectral Roll-off, Amplitude, Linear-frequency power spectrogram to get brief idea of the dataset.

03
Algorithm and testing

Algorithms

For solving the music genre classification problem many algorithms have been tested:

Support Vector Machine(SVM), Random Forests and Decision Trees:

G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," in IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, July 2002, doi: 10.1109/TSA.2002.800560.

CNN:

S. Prince, J. J. Thomas, S. J. J, K. P. Priya and J. J. Daniel, "Music Genre Classification using Deep learning - A review," 2022 6th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/CSITSS57437.2022.10026394.

Algorithm - CNN

Our approach to solve this problem is by using convolutional neural networks (CNN) to extract features from audio data, and then feed those features into a fully connected neural network (FCNN) to make the final classification.
The audio data is usually preprocessed by converting it into a spectrogram, which is a visual representation of the frequency components of the audio signal.
The CNN is trained on a dataset of audio samples labeled with their respective genres, and the weights of the network are adjusted during training so that it can accurately classify new audio samples based on their features.

BaseLine Models

We have implemented base models which include:

1. Linear Regression
2. KNeighborsClassifier
3. DecisionTreeClassifier

Linear Regression

Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. The goal of linear regression is to find the best fit line that minimizes the distance between the predicted values and the actual values of the dependent variable.

Accuracy: 26% (for GTZAN dataset)

DecisionTreeClassifier

DecisionTreeClassifier is a classification algorithm that builds a tree-like model by recursively splitting the data based on the values of input features. The algorithm selects the best feature to split the data based on some criterion, such as information gain or Gini impurity. The tree is built until a stopping criterion is met, and each leaf node is assigned a class label based on the majority class of the samples in that node.

Accuracy: 64% (for GTZAN dataset)

KNeighborsClassifier

KNeighborsClassifier is a classification algorithm that assigns a label to a data point based on the class labels of its nearest neighbors in the training dataset. The algorithm works by calculating the distance between the new data point and each point in the training dataset, and then selecting the k-nearest neighbors. The class label assigned to the new data point is then determined by a majority vote of the labels of the k-nearest neighbors.

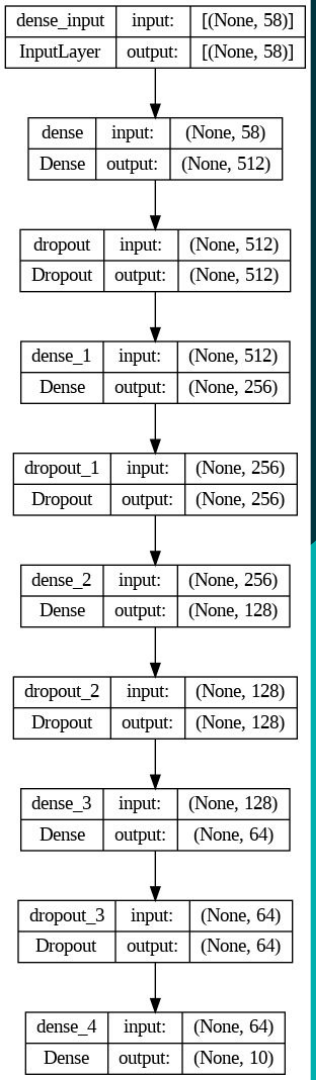
Accuracy: 85.5% (for GTZAN dataset and n_neighbors=5)

CNN Architecture

We use 5 dense layers with decreasing number of neurons each layers ReLU

Activated, with 4 dropout layers to prevent overfitting.

The final dense layer has a softmax activation to predict the classes.



Optimiser

We also use an **Adam**(Adaptive Moment Estimation) optimiser to adapt the learning rate based on the mean and variance of the gradient.

This results in faster convergence and better generalization performance, while handling noisy and sparse data, and avoiding overfitting.

Loss function

There are multiple methods to evaluate our model. We will use the **Sparse Categorical cross entropy loss function**.

It takes the true integer labels and the predicted class probabilities as input and calculates the cross-entropy loss.

The following is the formula for the function;

$$L = - \sum[y * \log(p)]$$

where y=true label

p=probability of corresponding class

Performance Metrics

There are a lot of metrics to judge a machine learning algorithm like F1 score, Recall, Precision, Accuracy, ROC curve,etc.

For our specific case we have chosen **Accuracy** as our main performance metric, besides some other metrics like Recall, Precision and F1 score

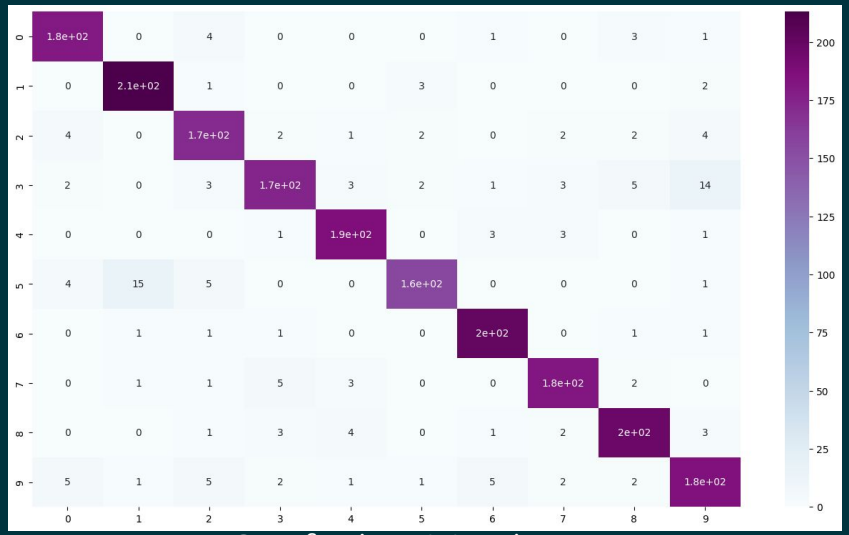
04
Results

Results of CNN Model

We have tested for model by varying batch-sizes and here are the results:

Batch Size	Accuracy
64	90.7%
128	91.3%
256	90.5%

Results for best batch size (128)



Confusion Matrix

References

1. https://scikit-learn.org/stable/modules/tree.html#classification
2. https://ieeexplore.ieee.org/document/10026394
3. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
4. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

Learning

We had a great experience in completing this project starting from reading research papers for music genre classification to finding datasets, cleaning it, designing the final CNN model.

We would like to Thank Aruna Ma’am and all the TA’s for guiding us through the ups and downs of the project

Thank You

Nishit Singh
Mir Razee Mohideen