A Project Report
on
Music Genre Classifier

Submitted by

ANIKET ROUTRAY (RA2211026030018)
BALADITYA YADAV (RA2211026030019)
ARIN KULSHRESTHA (RA2211026030009)
ASHUTOSH JAISWAL (RA2211026030042)

Under the Supervision of
Ms. Deepinder Kaur
(Assistant Professor, Department of Computer Science and Engineering)

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
DELHI-NCR CAMPUS, GHAZIABAD (U.P)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING & TECHNOLOGY
SRM INSTITUTE OF SCIENCE & TECHNOLOGY
DELHI-NCR CAMPUS, MODINAGAR
SIKRI KALAN, DELHI MEERUT ROAD,
DIST. – GHAZIABAD - 201204

Odd Semester- 2024-25

SRM
INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
DELHI-NCR CAMPUS, GHAZIABAD (U.P)

BONAFIDE CERTIFICATE

Certified to be the bonafide record of work done by Aniket Routray (RA2211026030018), Arin Kulshreshtha (RA2211026030009), Baladitya Yadav (RA2211026030019) & Ashutosh Jaiswal (RA2211026030042) Of 5th semester 3rd year B.TECH degree course in SRM INSTITUTE OF SCIENCE & TECHNOLOGY, DELHI-NCR Campus for the Department of Computer Science & Engineering, in Machine Learning LAB (21CSC305P) during the academic year 2024-25.

Lab In charges **Head of the department**

Submitted for end semester examination held on __/___/___at SRM INSTITUTE OF SCIENCE & TECHNOLOGY, DELHI-NCR Campus.

Abstract

- The **music genre classifier** project aims to automate the identification of music genres by leveraging **machine learning techniques**. This project involves extracting audio features such as **Mel-Frequency Cepstral Coefficients (MFCCs), chroma features, and spectrograms** to capture essential patterns in the music. Various machine learning models, including **Support Vector Machines (SVMs), K-Nearest Neighbors (KNNs)**, and **deep learning architectures like Convolutional Neural Networks (CNNs)**, are explored to classify audio tracks into predefined genres.
- This project has practical applications in **music streaming services, audio library management, and personalized recommendations**. Future enhancements include real-time classification and expanding the model to identify **subgenres** for finer categorization, contributing to improved music discovery and organization.

Introduction

- Music is the art of arranging sound and noise together to create harmony, melody, rhythm, and expressive content. It is organized so that humans and sometimes other living organisms can express their current emotions with it. We all have our own playlist, which we listen to while traveling, studying, dancing, etc. So here today, we will learn how can we implement the task of genre classification using Machine Learning in Python.
- Music plays a significant role in human culture, with genres acting as a way to categorize and identify different musical styles. With the increasing volume of music being produced and streamed globally, there is a growing need for **automated classification of music genres**. Traditionally, music genres were manually labeled by experts, but this process can be subjective, time-consuming, and prone to inconsistencies. **Machine Learning (ML) offers a powerful solution** by automating genre classification with high efficiency and accuracy.
- A **music genre classifier using machine learning** is a system that analyzes the features of audio tracks and predicts their genre, such as **pop, jazz, rock, or classical**. By leveraging techniques from **signal processing** and **machine learning algorithms**, the classifier extracts meaningful features from audio data and identifies patterns unique to each genre.

Objectives of the Project

•The primary goals of this project are:

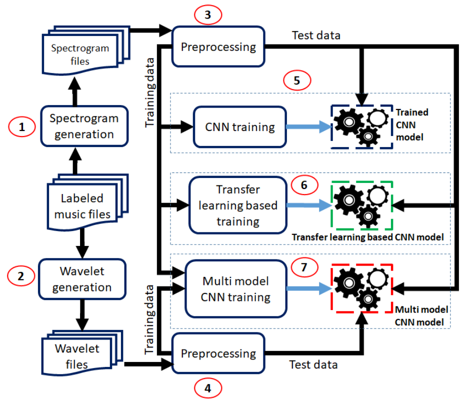
- To develop an automated system for **music genre classification** using machine learning.
- To explore different **audio features** and determine which ones provide the most informative patterns for classification.
- To build and compare the performance of **traditional ML algorithms** (e.g., KNN, SVM) with **deep learning models** (e.g., CNNs, RNNs).

Scope of the Project

- The scope of this project covers the following aspects:

1. **Data Preprocessing:** Handling various audio formats, resampling, and trimming silence.
2. **Feature Extraction:** Extracting meaningful features like **MFCCs, chroma features, and mel spectrograms** to capture essential musical characteristics.
3. **Model Building:** Developing and training machine learning models using frameworks such as **Scikit-Learn, TensorFlow, or PyTorch**
4. **User Interaction:** Providing users with an intuitive way to upload an audio file and receive a genre prediction.

Existing Problems



- **Data Quality and Quantity**: High-quality labelled datasets are crucial for training effective models. Many available datasets might be imbalanced (e.g., fewer samples for some genres) or contain noisy data (e.g., mislabelled tracks).
- **Feature Extraction**: Choosing the right features is essential. Common approaches use audio signals, but different features (like spectrograms, MFCCs, or raw waveforms) can lead to varying results. The choice of features can significantly affect the classifier's performance.
- **Genre Overlap**: Many genres have overlapping characteristics, making it difficult to distinguish between them. For instance, songs in the pop and rock genres may share similar instrumentation and structure.
- **Subjectivity of Genres**: Genre classification can be subjective. What one person considers jazz might overlap with another's definition of blues or soul. This variability can lead to inconsistent labeling in training datasets.
- **Contextual Factors**: The cultural and contextual background of music can affect its genre classification. Songs from different regions or eras may defy typical genre boundaries.

Proposed Solution

The proposed solution aims to develop a robust machine learning-based classifier that identifies the genre of music tracks by analyzing their acoustic characteristics. This system will leverage audio preprocessing techniques to extract features such as Mel-Frequency Cepstral Coefficients (MFCCs), chroma features, and spectrograms, which provide insights into timbre, rhythm, and tonal content.

A Convolutional Neural Network (CNN) will serve as the core architecture,given its ability to detect spatial patterns in spectrogram images.For capturing sequential dependencies in music, Recurrent Neural Networks (RNNs) or LSTMs can be integrated into the model.These networks will help handle the temporal structure inherent in musical compositions. Hybrid models combining CNNs and RNNs are proposed for improved performance.

To address challenges such as genre overlap and dataset imbalance,data augmentation (e.g., pitch shifting or time stretching) will be applied.The classifier will be trained on publicly available datasets like GTZAN to ensure reliable performance.Metrics such as accuracy, F1-score, and confusion matrix will be used for evaluation.

The final system is intended to work in real-world applications,such as music streaming platforms and audio library management, facilitating better personalized recommendations and efficient music categorization.

Technologies Used

- **Data Collection and Storage**

This phase involves sourcing, organizing, and storing the audio data required for training the classifier.

Datasets:

- **GTZAN Dataset:** A popular dataset with 10 music genres.
- **FMA (Free Music Archive):** Open-licensed music tracks.
- **Million Song Dataset (MSD):** Large-scale metadata and audio features.

- **File Handling Libraries:**
  - **OS Module (Python):** For local file management.
  - **PyDub:** For audio format conversion and manipulation.
  - **FFmpeg:** Command-line tool for handling and converting multimedia files.

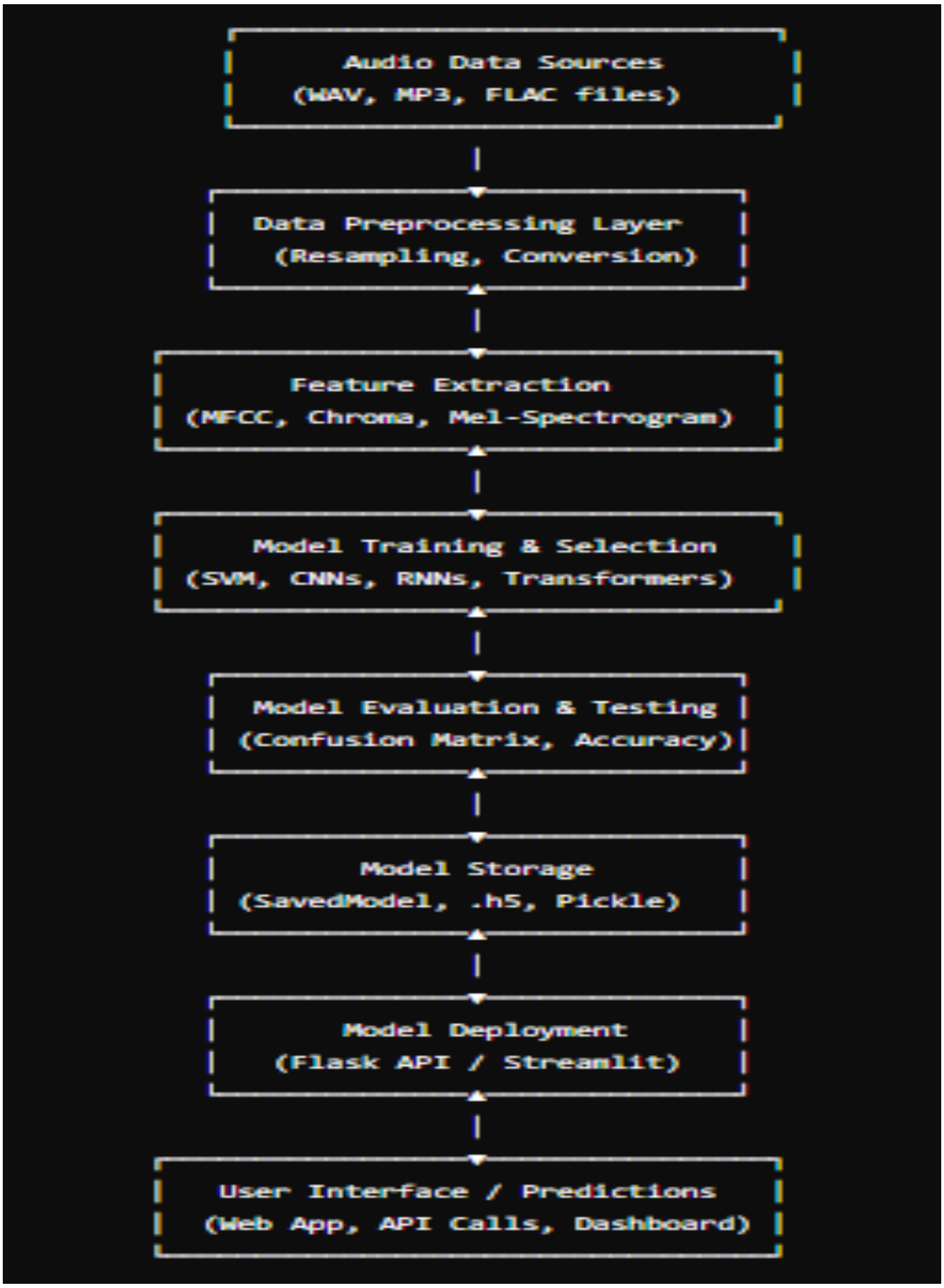
- **Data Preprocessing**

Data preprocessing ensures that the audio files are ready for analysis by converting formats, resampling, and normalizing data.

- **NumPy:** For handling numerical data and arrays.
- **SciPy:** For signal processing operations.

Phase	Technologies
Data Collection	GTZAN, Google Drive
Preprocessing	LibROSA, PyDub, FFmpeg, NumPy, SciPy
Feature Extraction	LibROSA
Machine Learning Models	Scikit-Learn, KNN
Evaluation	Scikit-Learn, Matplotlib
Model Storage	Pickle
Project Management	Google Colab, GitHub

Architectural Diagram



Methodology

1. Data Collection and Preparation

Data is the foundation of any machine learning project. This step ensures that the raw audio data is collected, organized, and preprocessed for further analysis.

- **Data Sources:**
  - Datasets like **GTZAN**, **FMA**, and **Million Song Dataset** provide labeled audio files.
- **Data Augmentation (Optional):**
  - **Audiomentations Library**: Adds variations by applying noise, pitch shift, or time stretch to increase the diversity of the dataset.
- **Handling Imbalanced Data:**
  - Oversampling rare genres using **SMOTE (Synthetic Minority Oversampling Technique)**.
  - Alternatively, **undersampling** the majority classes.

2.Data Preprocessing

Preprocessing ensures that the audio data is uniform and ready for feature extraction.

- **Standardization of Audio Files:**
  - **Format Conversion:** Convert audio files into WAV or MP3 using **FFmpeg**.
  - **Resampling:** Resample all audio files to a common sample rate (e.g., 22.05 kHz).
  - **Mono Conversion:** Convert stereo tracks to mono to reduce complexity.
- **Silence Removal and Trimming:**
  - Tools like **LibROSA** help detect and remove silent sections in the audio.

3.Feature Extraction

- This step extracts relevant features from the audio signal, which serve as inputs to the machine learning models.
- **Time-Domain Features:**
  - **Zero-Crossing Rate (ZCR):** Measures how frequently the signal changes from positive to negative. Useful for detecting percussive sounds.
- **Frequency-Domain Features:**
  - **MFCCs (Mel-Frequency Cepstral Coefficients):** Capture the timbre of the music, making them crucial for genre classification.
  - **Chroma Features:** Analyze harmonic content by mapping pitches to a 12-note chromatic scale.
- **Time-Frequency Representations:**
  - **Mel Spectrograms:** Visualize frequency vs. time, suitable for CNNs.

4.Model Selection and Training

- **Traditional Machine Learning Models:**
- These models work well on smaller datasets with pre-extracted features like MFCCs.

Support Vector Machines (SVM): Effective for high-dimensional data.

Random Forests / Decision Trees: Useful for interpretable models.

- **Deep Learning Models:**
- These models can learn from raw data or spectrograms, extracting spatial and temporal patterns.

Convolutional Neural Networks (CNNs):

- Use **spectrograms** as input to detect spatial patterns.

Recurrent Neural Networks (RNNs) / LSTMs:

- Capture **temporal relationships** between successive frames in audio.
- Suitable for tracking changes in rhythm, melody, etc.

Result

```
The predicted genre for the music clip is: hiphop
```

Conclusion

- In conclusion, the music genre classifier project successfully demonstrates how **machine learning models** can automate genre identification with a high degree of accuracy. From **feature extraction** to **model deployment**, each phase of the project plays a crucial role in building a robust, scalable system. While challenges like overlapping genres and subjectivity remain, the combination of **traditional machine learning** and **deep learning techniques** provides a solid foundation for future development.
- This project not only showcases the potential of **ML in music analytics** but also opens up new avenues for exploration, such as **real-time classification, personalized recommendations, and trend analysis**. With continuous advancements in machine learning and access to larger datasets, the music genre classifier holds great promise for transforming the way we experience and interact with music.

References

Below are key references for building a **music genre classifier:**

- https://www.geeksforgeeks.org/music-genre-classifier-using-machine-learning/
- https://github.com/ABSounds/MusicGenreClassification
- https://ar5iv.labs.arxiv.org/html/1804.01149
- https://medium.com/@baraq/classifying-song-genres-with-machine-learning-37258250504a
- https://www.kaggle.com/datasets/purumalgi/music-genre-classification