Differentiation of music genre from an audio file using Neural Networks

Pushker Jain[1], Vishal Singh[2], Tanupriya Chowdhury[3], Ayan Sar[4] and Ketan Kotecha[5]

[1] Informatics Cluster, School of Computer Science University of Petroleum and Energy Studies
(UPES) Dehradun 2480007, Uttarakhand, India.
pushker0101jain@gmail.com
[2] Informatics Cluster, School of Computer Science University of Petroleum and Energy Studies
(UPES) Dehradun 2480007, Uttarakhand, India.
vishalsingh08052003@gmail.com
[3] Professor, CSE Dept., Symbiosis Institute of Technology, Symbiosis International University,
Lavale Campus, Pune, Maharashtra,412115, India.
Ex-Professor, SoCS, University of Petroleum and Energy Studies (UPES), Dehradun,
Uttarakhand 248007, India
tanupriya.choudhury@sitpune.edu.in
tanupriya@ddn.upes.ac.in
[4] Informatics Cluster, School of Computer Science University of Petroleum and Energy Studies
(UPES) Dehradun 2480007, Uttarakhand, India.
ayan.sarbwn@gmail.com
[5] Symbiosis Centre for Applied Artificial Intelligence
Symbiosis Institute of Technology, Symbiosis International University, Pune, 411045, India
director@sitpune.edu.in

Abstract. This research paper aims to explore deep learning and machine learning techniques specifically CNN and KNN for differentiating music genres from audio files. In this paper, Mel Frequency Cepstral Coefficients (MFCC) were used as the main methods for classifications. The reason this paper focuses on MFCC features is because today's models have mostly focused on computer vision technique which involves classification of genre based on spectrogram images of different types of genres. This not only takes time but also a lot of computational power. Using MFCC features tends to take less time and less computational resources.

Keywords: Audio classification, Neural networks, Music Genre classification

1 Introduction

Music classification is a field in the field of Music Recovery (MIR) and sound signal processing research. Neural Network is a modern way of classifying music.

The classification of music using Neural Networks (NNs) has been very successful in recent years. Various song libraries, machine learning technologies, input formats, and the use of Neural Networks are used for the classification of genre in a particular song. The Deep Learning Approach is used for system training. We have used the Mel Frequency Cepstral Coefficient (MFCC) as our input features for our CNN model.

2 Problem Identification

Who doesn't love music, everyone loves music and everyone has different and unique tastes. Over the last few years, the music industry has experienced significant transformations, departing from its conventional form of existence. These changes are not only evident in how music is created but also in the way it is consumed. With the continuous expansion of the customer base, the market for various music styles has also grown substantially. People's preferences have diversified, leading to increased demand for a wide range of musical genres. As technology advances and the music landscape evolves, the industry continues to adapt to meet the ever-changing needs of music enthusiasts worldwide. In music recommendation systems like Spotify, classifying music by genre is vital for enhancing the user experience. It allows the app to curate personalized playlists and suggest songs that match each user's preferences, leading to a more enjoyable music discovery journey. The problem of differentiating music genres from audio files encompasses the challenge of automatically classifying and categorizing music based on its sonic characteristics. While humans can easily distinguish between various music genres, developing automated methods to accurately and reliably classify music based on its audio content remains a complex task. Several key challenges and problem areas within this domain can be identified:

2.1 Feature Extraction and Representation

Selection of all the features which can help in the differentiation of music into different categories. Extraction of these features as well as preprocessing the features was identified as a problem.

2.2 Genre Ambiguity and Subjectivity

Sometimes, it is hard to put music into specific boxes because different songs can have a mix of different styles or they don’t fit into any specific category. This makes it difficult to say exactly what genre a song belongs to because sometimes music genres vary from person to person. With the advancement and introduction of new technology, many new features are being added and subtracted.

2.3 Variability and Robustness

Accounting for the inherent variability and diversity within music genres, songs within the same genre can exhibit significant differences in terms of instrumentation, tempo, mood, and production quality. Ensuring the robustness of genre classification algorithms to handle variations in audio quality, noise, and recording conditions.

2.4 Data Availability and Annotation

Finding large datasets of different types of music having a wide variety of songs and also having specified features of songs. Overcoming the challenges related to dataset bias and lack of standardization in genre annotations. Pre-processing of datasets that do not have specific features or have some different songs in different categories.

2.5 Computational Efficiency

Development of algorithms that can process the music in real time and classify the music quickly and correctly and also get trained on newly released songs.

3 Literature Review

From [5], we came to know about the survey conducted by K. Meenakshi, the researchers aimed to classify music into different genres using a Convolutional Neural Network (CNN). They pre-processed the music database, extracting two types of feature vectors: Mel Spectrum with 128 coefficients and MFCC coefficients using the Python "librosa" package. The CNN model consisted of Convolutional, Pooling, and Fully Connected layers. The evaluation was based on a dataset with 10 arrays representing various music genres, and they inputted 1000 labeled songs for testing. The results showed a learning accuracy of 76% for the Mel Spectrum feature vector and 47% for the MFCC feature vector. This research contributes valuable insights into automated music genre classification and its potential applications in music recommendation systems and related domains.From [6], we came to know about the research conducted by Nirmal M. R employed spectrograms, which are visual representations of a signal's frequency spectrum over time. They used the Short Time Fourier Transform (STFT) to generate spectrograms, graphically representing data in both time and frequency domains. Two models were implemented for the Convolutional Neural Network (CNN) – a user-defined sequential CNN model and a pre-trained ConvNet (MobileNet). The classification accuracy of the user-defined CNN model was 40%, whereas MobileNet achieved a higher accuracy of 67%. The research was carried out using Python 3 on a LINUX operating system, with the deep learning model built-in Python Keras framework. Over the past few years, researchers have employed machine-learning techniques to classify music genres with remarkable success. In 2002, G. Tzanetakis and P. Cook [1] made significant achievements in this area by utilizing the mixture of the Gaussian model and k-nearest neighbors. They combined these techniques with three sets of features that represented timbral texture, rhythmic content, and pitch content of the music tracks. Their approach was comprehensive, as they carefully hand-extracted the features to capture essential characteristics of each genre. By using a combination of advanced statistical methods like the mixture of Gaussians and the k-nearest neighbors' algorithm, they were able to effectively classify music into different genres. Their findings were efficient, as they achieved an accuracy of 61% in their music genre classification task.

4 Existing System Issues

No model is ever perfect. There are always some flows or limitations in features or data. No doubt there are many different models available that help in differentiating music based on genre but the problem is that today's models have mostly focused on computer vision technique which involves the classification of genre based on spectrogram images of different types of genres. They take spectrogram images of that song as the input and then use that to classify the genre of that music. In computer vision, a lot of computational power is used for processing images. That is why we will only be using key features from audio files which can be converted to numeric values and can be easily fed to our CNN model.

5 Proposed System Design

5.1 Dataset Preparation

We obtained all of our musical data from the public GTZAN dataset. This dataset contained 100 songs from each genre, and there are a total of 10 genres in the dataset overall making 1000 songs in the dataset.

5.2 Feature Extraction

The specific audio features that were extracted from the songs depend on the chosen signal processing technique. The audio feature we used for music genre classification is Mel-frequency Cepstral Coefficients (MFCCs)
The process of extracting MFCCs involves several steps:
Pre-emphasis: Enhances higher frequencies in the audio signal to improve the signal-to-noise ratio.
Frame Segmentation: Divides the audio signal into short frames to capture the temporal characteristics of the signal.
Windowing: Applies a window function (e.g., Hamming window) to each frame to reduce spectral leakage.
Fast Fourier Transform (FFT): Computes the magnitude spectrum of each frame.
Mel Filterbank: Applies a bank of triangular filters on the magnitude spectrum to obtain Filterbank energies. The filters are spaced based on the Mel scale, which approximates human perception of frequency.
Logarithmic Compression: Applies a logarithmic compression to the Filterbank energies to approximate the non-linear human perception of loudness.
Discrete Cosine Transform (DCT): Computes the DCT of the log-compressed energies to obtain the MFCC coefficients.

5.3 Neural Network Modelling

We used the CNN model for training and testing of our model. The model included convolutional layers to capture local patterns in the audio features and pooling layers to reduce dimensionality. We chose ReLU as the activation function and Adams optimization algorithm for training the model.

6 Dataset and Algorithms discussed

GTZAN Dataset is one of the most widely used datasets for music genre classification. It is a public dataset collected in 2000-2001. It comprises 10 genres and each genre has 100 audio files. The audio files are in .wav format and each audio file is 30 seconds in length. It is a collection of 2 csv files and 2 folders namely genres original and images original. The size of the whole dataset is 1.2 GB.

S.No.	**Genre**	**Count**
1.	Blues	100
2.	Classical	100
3.	Country	100
4.	Disco	100
5.	Hip-hop	100
6.	Jazz	100
7.	Metal	100
8.	Pop	100
9.	Reggae	100
10.	Rock	100
	Total	**1000**

6.1 Convolutional Neural Network (CNN)

In the Algorithms section, we present the key approach used in our research for music genre classification: the Convolutional Neural Network (CNN). CNNs are a powerful class of deep learning algorithms widely employed for image and audio processing tasks. In our study, we leverage the capabilities of CNNs to effectively classify music genres based on audio features. CNNs are a type of deep learning algorithm commonly used for image and audio processing tasks. Our CNN model consists of three convolutional layers. Each convolutional layer is followed by a max pooling layer for downsampling and a dropout layer for regularisation. This is the structure of CNN we used in our model as depicted in [Fig.1].

- Convolutional Layer 1: 32 filters, kernel size 3x3, ReLU activation
- Max Pooling Layer 1: Pool size 2x2
- Dropout Layer 1: Dropout rate 0.25
- Convolutional Layer 2: 64 filters, kernel size 3x3, ReLU activation
- Max Pooling Layer 2: Pool size 2x2
- Dropout Layer 2: Dropout rate 0.25
- Convolutional Layer 3: 128 filters, kernel size 3x3, ReLU activation
- Max Pooling Layer 3: Pool size 2x2
- Dropout Layer 3: Dropout rate 0.4

After the convolutional layers, the output is flattened and fed into a fully connected (dense) layer with 128 units and a ReLU activation function. Finally, a dropout layer with a dropout rate of 0.3 is applied to the output of the dense layer. The last layer is a dense layer with the number of units equal to the number of classes (genres) in the dataset, followed by a softmax activation function to produce the probability distribution over the genres.

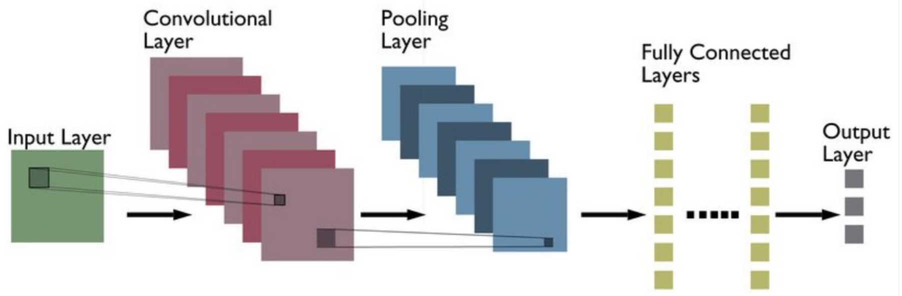

Fig 1: Full CNN Layering Diagram

6.2 K-Nearest Neighbor (KNN)

The other algorithm used in our machine learning model for music genre classification is the K-Nearest Neighbor classifier. It is one of the main machine learning algorithms and belongs to the supervised learning domain. It is used for both classification and regression problems but it is extensively used for classification problems.

In this model, the K-Nearest Neighbor classifier is used for music genre classification in the GTZAN dataset. It is imported from the sklearn python library. We imported "train_test_split" from the scikit python library in which we divided the dataset into x_train, y_train, x_test, and y_test. The test size of the model is 24% of the total data of the dataset. The values X and Y used for computing in the model are first transformed using a standard scaler. The target variable (Y) i.e., genre is transformed using a label encoder. The genres that were in categorical values are converted into a numerical format for our model.

Here, the value of K i.e., hyperparameter is set to K=5 which means it will consider 5 nearest data points when making predictions for the data point.

K- Fold cross-validation is also used in this model to find out the optimum value of K (hyperparameter) which will provide the best test accuracy of the model. It is used for hyperparameter tuning. GridSearchCV is used to find the best value of a parameter from 1 to 17(odd values). It takes metrics like “Euclidean” distance and “Manhattan” distance. GridSearchCV is then used to evaluate the model using 5-fold cross-validation to find the optimum desired result.

7 Results and Discussions

The main quantitative metric that we used to judge our CNN model is accuracy (that is, the percentage of predicted labels that matched their true labels), and our way of visualizing the performance of our best model is through the confusion matrices
The hyperparameters used in the music genre classification system may vary depending on the specific implementation and model architecture. However, here are some commonly used hyperparameters in a CNN-based music genre classification system:
segment_duration: The duration of each audio segment used for classification in our case was 30 sec.
hop_length: The number of audio samples between consecutive frames (default: 512).
n_mfcc: The number of Mel-frequency cepstral coefficients (MFCCs) we extracted was 13.
n_fft: The length of the FFT window (default: 2048).
padding_duration: The duration of silence padding added to the beginning and each segment was 10 seconds respectively.
model: The architecture of the CNN model used for music genre classification.
batch_size: The number of samples per gradient update during training, (default: 32).
epochs: The number of times to iterate over the entire training dataset ws finally set to 50 because accuracy maxed at 98% [Table 1] and became constant for every epoch after 30. So, epoch = 30 would be the best case.

Table 1: Evaluation metrics of CNN after training

Train Accuracy	Train loss	Val accuracy	Val Loss
0.9843	0.05	0.6937	val_loss: 2.5314

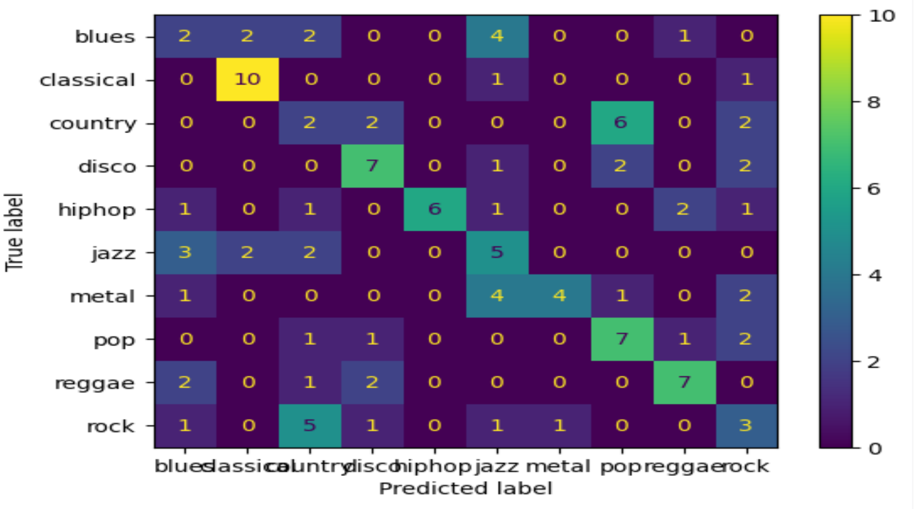

Fig 2: Quantitative metric for KNN model

The quantitative metric used to judge the performance of our KNN model is the accuracy and confusion matrix.

Accuracy is defined as the percentage of the number of correct predictions to the total number of predictions.

Confusion Matrix is a tabular or matrix representation that helps us to evaluate the performance of a model. It takes true labels and predicted labels as input and arranges them into a matrix form to analyze the performance of the model.

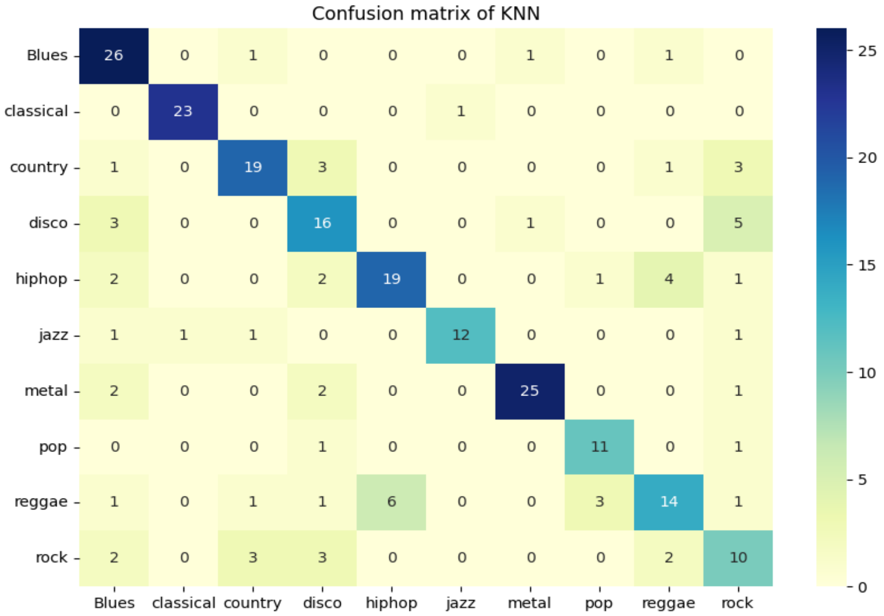

Fig 3: Confusion Matrix for KNN model

The hyperparameter used in KNN is K which is defined as the number of neighbours we have to consider for making predictions.

Table 2: Testing accuracy of KNN

Test Accuracy	K (Hyperparameter)
0.71	5

After k-fold cross-validation (hyperparameter tuning), where CV=5.

Table 3: After k-fold cross-validation, the accuracy of KNN

Train Set Accuracy	Test Set Accuracy	K (Best n neighbours)
0.81	0.73	5

8 Conclusion

In conclusion, our research paper focused on the task of differentiating music genres from audio files using deep learning and machine learning techniques, specifically convolutional neural networks (CNNs) and (KNNs). We identified key challenges in this domain, including feature extraction and representation, genre ambiguity and subjectivity, variability and robustness, data availability and annotation, and computational efficiency.

To address these challenges, we proposed a system design that involved dataset preparation, feature extraction using Mel-frequency Cepstral Coefficients (MFCCs), and the use of a CNN model and KNN Model for genre classification. We used the GTZAN dataset.

Through our experiments and evaluations, we achieved promising results, we received an accuracy of 98% on the training set in CNN, while on the validation set, we achieved an accuracy of 70%. We also received an accuracy of 81% on training set in KNN and accuracy of 73% in test set where K=5. We visualized the performance of our best model using confusion matrices and discussed the hyperparameters used in our system.

Overall, our research contributes to the field of music genre classification by demonstrating the effectiveness of deep learning techniques, specifically CNNs, in accurately differentiating music genres from audio files. This research opens up opportunities for further exploration and improvements in automated music genre classification systems, with potential applications in music recommendation systems, music streaming platforms, and other music-related domains.

References

1. G. Tzanetakis and P. Cook. Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing,
2. Mingwen Dong. Convolutional neural network achieves human-level accuracy in music genre classification. CoRR,
3. Hareesh Bahuleyan. Music genre classification using machine learning techniques. CoRR, abs/1804.01149, 2018
4. Music Genre Classification Techniques by Gautam Chettiar Kalaivani S SENSE Department of Communication Engineering, SENSE Vellore Institute of Technology Vellore Institute of Technology Vellore, Tamil Nadu, India Vellore, Tamil Nadu, India.
5. S. Vishnupriya and K. Meenakshi, "Automatic Music Genre Classification using Convolution Neural Network," 2018 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2018, pp. 1-4, doi: 10.1109/ICCCI.2018.8441340.
6. N. M R and S. Mohan B S, "Music Genre Classification using Spectrograms," 2020 International Conference on Power, Instrumentation, Control and Computing (PICC), Thrissur, India, 2020, pp. 1-5, doi: 10.1109/PICC51425.2020.9362364.