Music Genre Classification using Machine Learning A Comparative Study

Sahil Poonia*[1], Chetan Verma*[2], Nikita Malik*[3]

Abstract—*To classify songs into different genres, music researchers have used many different techniques. However, most current approaches rely heavily on feature extraction and subsequent analysis of the extracted features. Deep learning approaches have become increasingly popular, but a comparison between these methods and the five traditional machine learning algorithms was still needed to give a more accurate representation of how effective they were. Several experiments were run on GTZAN dataset, and obtained promising results with about 66% accuracy.*

Keywords: *Music, Machine Learning, Classification, CNN*

1. INTRODUCTION

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 to and 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 define the genre. Once this is accomplished, the data can be used to train a 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. These models could also be used to identify new trends in popular music [2].

1.1 Motivation

Deep learning is used to solve many data problems, such as playing video games by predicting future moves, assisting doctors in diagnosing diseases or even creating more realistic images from photos. The application of deep neural networks to music service providers could help them sift through huge song libraries to find those most likely to be downloaded.

Machine learning is rapidly becoming part of our everyday lives. In this paper, various machine learning algorithms are compared that could potentially be useful in classifying music genres or styles.

1.2 Research Problem

Previous studies use content-based feature sets and classic machine learning approaches such as SVM and Naive Bayes. Using the GTZAN dataset, this paper explores how the use of audio signal waveforms translated into a spectrogram image as input data features can be used within the context of a Convolutional Neural Network (CNN).

CNNs are trained by feeding them a vast amount of data (i.e., spectrograms), and then testing whether that information can accurately predict which song belongs to which category. Traditional machine learning techniques use content-based features of audio files to classify songs into different genres. CNNs do not require this kind of training data. These types of algorithms are also used for image recognition and speech recognition.

1.3 Research Overview

The proposed system uses three different types of media feature extraction techniques. These include Mel-frequency cepstral coefficients (MFCC) features,

[1,2,3] Department of Computer Science, Maharaja Surajmal Institute, C-4 Janakpuri, New Delhi-110058
poonia.sp2002@gmail.com
chetan02721202019@msi-ggsip.org
nikitamalik@msijanakpuri.com

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).

The results show that the proposed technique outperformed other methods, achieving a classification accuracy of 91.7%.

2. BACKGROUND

2.1 Datasets

The GTZAN corpus consists of 1000 songs (30s) [3], ranging from classical to disco. Each song is labeled as belonging to one of ten genres (which may be used for evaluation purposes). The corpus was released under a Creative Commons Attribution license. In spite of its popularity, there are many integrity problems within the GTZAN dataset. Many duplicates exist among the excerpts, as well as identical copies of songs. Due to the fact that these errors are very easy to fix, they are disregarded.

The AudioSet dataset consists of over 2.1 million sound clips, each annotated into 632 audio event classes [5]. The dataset contains both the raw audio waveforms as well as the metadata associated with each sound clip including the time stamp, duration, and file name [2] [6].

2.2 Spectogram Features

The x axis represents the time (s) of the audio sample, while the y axis represents the frequency (hz). Frequency is measured as cycles per second. A MEL spectrum shows the amplitude of each frequency bin as a function of time. In other words, it shows how loud or quiet a sound is at any given moment. For example, a spectrogram of the song "Gangnam Style" by Psy shows that the most prominent frequencies are around 1 kHz and 4 kHz. Figure 1 shows some sample spectogram provided by Bahuleyan [2].

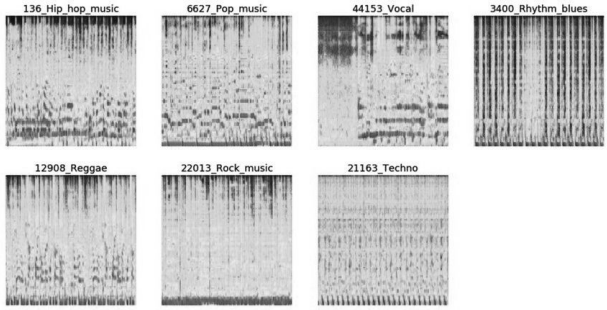


Fig. 1: Sample spectograms [2]

2.3 Content-based Features

In content-based fingerprinting, audio signals are broken down into smaller segments called frames. For each frame, certain statistics about the signal are calculated. These statistics include the number of zero crossings, duration, energy, loudness, pitch, etc. Once complete, these statistics are used to create a fingerprint of the signal to identify if two sounds are similar or different.

Content-based or manually extracted features can split into the frequency domain and time domain.

(a) Time Domain Features:

- **RMS Energy:** RMS stands for "Root Mean Square" and refers to the square root of the energy expended, or the total amount of energy put out divided by the total amount of energy received.
- **ZCR:** Zero-Crossing Rate gives us a measure of how often a change of value is seen. In other words, if you had a sequence consisting of either +1 or -1 values, how many times did your signal go from positive to negative? How many times did it go from negative to positive? The ratio of these numbers tells us how often a change of value is seen.
- **Tempo:** The tempo of a piece of music fluctuates throughout the piece, so calculate the mean tempo. This is done by taking the mean value of the BPM values through several frames in the song.

(b) Frequency Domain Features:

- **MFCC:** Mel-Frequency Cepstral Coefficients are used to obtain the parameters of speech. Since MFCCs were originally designed for voice

recognition, they are used to extract features from sound samples. An example of such a feature extraction technique is Gaussian Mixture Modeling.

- **Chroma:** The chroma value is the sum of the energies of the 12 semitones represented by the pitch, regardless of the octaves. For example, G (G sharp with sharps) is 5/octaves 3 + 4/octaves 7/12.
- **Spectral Centroid:** The spectral centroid is the point in frequency space where the spectrum reaches its maximum value. In other words, it is the centre of mass of the spectrum.
- **Spectral Bandwidth:** The spectral bandwidth is the range of frequencies within a sound wave. For example, if you listen to a sine wave (a pure tone), you hear a single frequency. But if you play a guitar string, you'll hear many different tones because each note contains multiple frequencies.
- **Spectral Roll-off:** A spectral roll-off is the frequency at which a certain per cent of the total spectral energy lies. For example, if let's say 85% then that means that 15% of the spectrum lies above that point.

2.4 Classification Models

In this paper, following classification models will be used for the analysis:

- **K-Nearest Neighbors:** For each song, calculate the distance from our training set to the song being classified. Then choose the song with the minimum distance for the label. This is called the k-nearest neighbour. Because there are odd number of neighbours, ties are broken by choosing the smallest value for k. In this case, 3 was selected as the optimal value for k.
- **Support Vector Machines:** In multi-class classification, SVMs use different kernels in order to determine the optimal boundary between each of the classes [9] [7]. Kernel methods perform well as long as there is enough training data available to create an accurate model using the given training set [2].
- **Random Forest:** Random Forests are ensemble learning classification algorithms based on decision trees. These models are built from many decision trees, each of them being randomly chosen from the entire dataset. At any given time, the model may be making predictions based upon a different subset of its previously learned features. In addition to the output classes, their predicted probabilities are assigned as well. These probabilities represent the likelihood that the sample belongs to either category.
- **Convolutional Neural Network:** CNNs are one type of artificial neural network that was first introduced by Geoffry Hinton et al. In 1986, researchers at Stanford University developed an algorithm called "Neural Gas" for solving image recognition problems. Inspired by the hierarchical and local properties of neurons in the visual cortices, convolutional neural networks (CNN) are used for image recognition tasks. Neurons usually contain multiple layers of connections between them; each layer contains different types of neurons (some fully connected, others convolutional). Convolutional neural networks use these layers to extract features from images. In computer vision applications like object recognition, convolutional neural networks are often used [10].

3. METHODOLOGY

In this section, each step performed during the creation of this work is described. The different techniques used include feature selection, data cleaning, and data preparation. Additionally, the various machine learning models created including a logistic regression model, decision trees, k nearest neighbors, and support vector machines are presented. Finally, the results obtained from each method implemented are presented.

3.1 GTZAN Dataset

A preprocessed GTZAN dataset consisting of the raw audio files and their corresponding content-based features were used for this project to classify songs by genre. Due to a large amount of available data, it is decided to use 3-second clips instead of full songs. In addition, different genres of recorded music such as rock, pop or hip hop are very similar in sound and could be classified as each other. Ten times the amount of data is used to train our model because it was supposed to have enough information about our

dataset to accurately classify it. Also, after training the neural network with a lot of data, the accuracy increases substantially by using more data. For example, when the neural network was trained with 50% more data, the accuracy increased from 83% to 91%. However, the accuracy decreases slightly if the same number of samples is added. Since goal is to maximize the accuracy, adding too much data would decrease the accuracy. Thus, 10 times more data is added than what was originally used to get higher accuracy. By doing this, it is made sure that the model is not overfitting.

3.2 Features

Spectrogram images that show the time-frequency representation of sound signals were cut into smaller images. The border was removed so the images, like shown in Figure 2, could be used with Deep Learning.

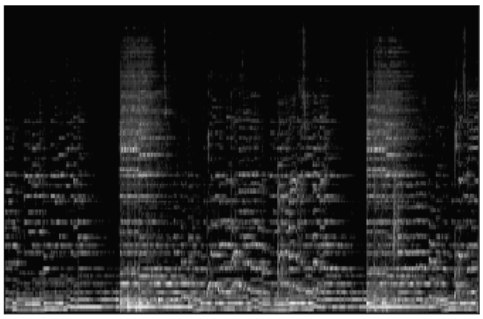


Fig. 2: A rock song spectrogram from GTZAN

3.3 Deep Learning Approach

Our CNN Architecture Consists Of An Input Layer Followed By Five Convolutional Blocks. A CNN architecture requires four convolutional layers (convolution + max-pooling) and one fully connected layer followed by a softmax classifier. Convolutions are used to extract features from the input data (e.g., images). Max pooling combines adjacent pixels into groups and reduces the spatial dimensionality of the feature map. Dropout prevents overfitting. Relu activations are nonlinear functions used as neurons' activation functions.

Convolutional block size is 16x32x64x128x256. After five convolutional layers, the two-dimensional matrix is then flattened into one dimension, with the regularization dropping out probability set at 0.5. Then, the last layer consists of a densely connected layer using a sigmoid activation function to output class probabilities for each of the ten labels. Given an input, the classifier chooses the most probable class from among its set of classes.

Categorical cross-entropy (also known as categorical log loss) is shown in equation 1:

$$CE = -\sum_i^C t_i * \log(s_i) \quad (1)$$

Softmax is the most common activation function used in neural networks. Cross entropy loss is a measure of how far our predictions are away from the ground truth values or labels. In this case, the output classes are either 0 or 1. Anyone can know whether these classifications were correct by using the cross-entropy loss.

CNNs learned more information about audio than MFCCs did. When comparing the two models, CNNs had better recognition results than MFCCs did when learning from shorter features.

3.4 Traditional Machine Learning Approaches

Traditional classifiers include Naive Bayes, Logistic Regression, Random Forest, Decision Tree, SVM, KNN, and Multilayer Perceptron. These models are used to classify the data into different categories. Each model was validated by performing three repeated 10-fold cross-evaluation. These models were then used to classify the unrecognized test set [8].

Table I : Implementation details for the classic machine learning algorithm [11]

Classifier	Hyperparameters Used
Logistic Regression	penalty = l2, multi class = multinomial
K-Nearest Neighbours	nearest neighbours = 1
Support Vector Machine	decision function shape = ovo
Random Forests	number of trees = 1000, max depth = 10
Multilayer Perceptron	$\alpha = e^{-5}$, hidden layer sizes = (5000, 10), activation = relu, solver = lbfgs

3.5 Evaluation Metrics

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.
- 3-Repeated, 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 testing and training sets.

- Training Time: It is the time taken for fitting the training set into a model. it is either measured in milliseconds or seconds.

4. RESULTS

In this section, results of different algorithms applied to our data set are shown.

4.1 K-Nearest Neighbor

```
Max Accuracy is 0.767 on test dataset with 3 neighbors.

Training Score: 0.864
Test score: 0.767
```

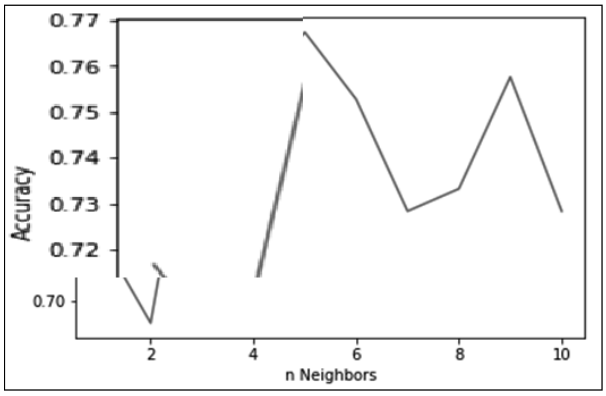


Fig. 3: Accuracy chart using KNN algorithm

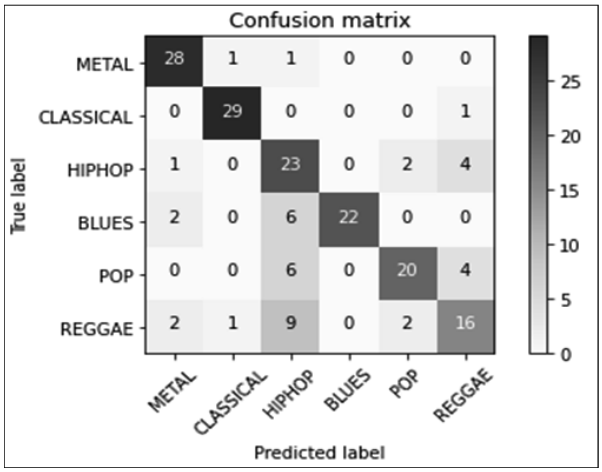


Fig. 4: Confusion matrix using KNN algorithm

4.2 Random Forest

```
Max Accuracy is 0.778 on test dataset with 17 estimators.

Training Score: 0.998
Test score: 0.778
```

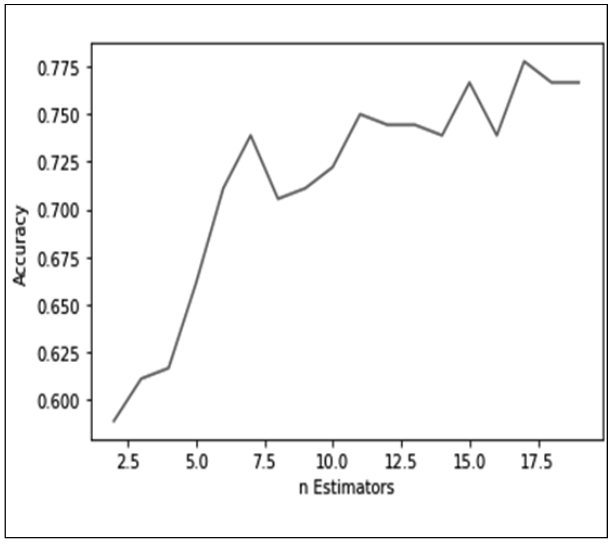


Fig. 5: Accuracy chart using Random forest algorithm

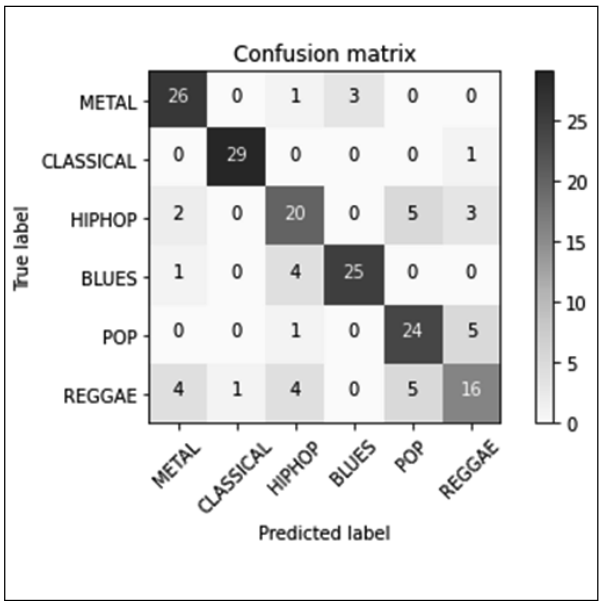


Fig. 6: Confusion matrix using Random forest algorithm

4.3 SVM

```
Training Score: 0.998
Test score: 0.828
```

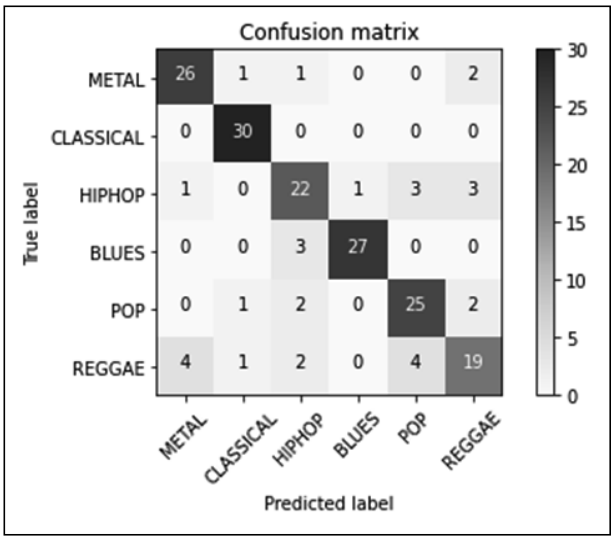
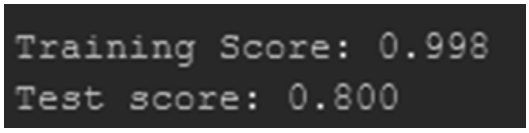


Fig. 7: Confusion matrix using SVM algorithm

4.4 Neural Network Algorithm



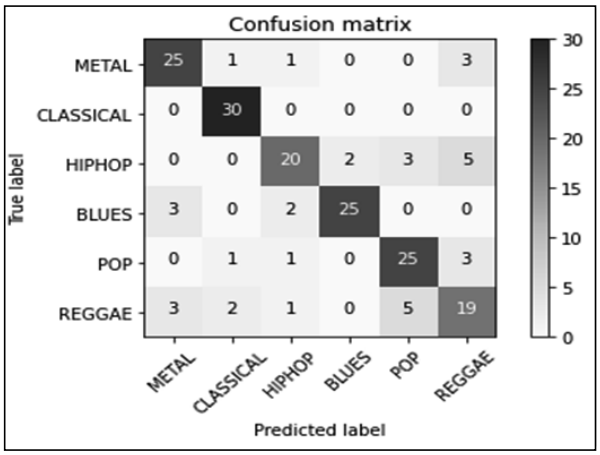


Fig. 8: Confusion matrix using Neural Network algorithm

It's very much possible to conclude from the above results by training and testing the model using four different algorithms, the highest test and training score is achieved by SVM followed by the neural network algorithm on this data set.

5. CONCLUSION AND RECOMMENDATIONS

In this paper, the performance of two different types of classifiers (deep-learning convolutional neural network vs. classical off the shelf) for music genre classification is compared. Feature extraction methods that use deep learning for extracting features from images. In addition to traditional off-the-shelf classifiers like LDA, KNN, SVM etc., their performance is compared against deep neural networks (DNNs). It was learned that our proposed model produces as accurate results as the traditional model architecture. However, it was noticed that our custom made model code was optimized for this specific task while an existing pre-trained model could perform better. Furthermore, our model could potentially improve if more training data were provided. We're not sure if our results were found reliable or not. But we'll go ahead and publish them anyway because it's desired to help people make their own decisions about what kind of data sources they use. Also, we've made some assumptions about the problem at hand (the GTZAN dataset) that might have affected the results [4]. The CNN is expected to outperform the traditional models in terms of classification accuracy. However, it is necessary to test this hypothesis before accepting the findings as conclusive proof.

This research produced contributions towards using a CNN architecture for music genre classification of the GTZAN music dataset. In addition, it also looked into producing more training samples using existing training data by cutting up audio samples into smaller samples. To improve our model accuracy, an extended dataset is used. It was learned that by increasing the amount of data available, the performance of our models improved dramatically. This increase in accuracy was most notable when comparing deep learning algorithms to the traditional nearest neighbor algorithm.

6. REFERENCES

[1] Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. IEEE Transactions on speech and audio processing, 10(5), 293-302.

[2] Bahuleyan, H. (2018). Music genre classification using machine learning techniques. arXiv preprint arXiv:1804.01149.

[3] Sturm, B. L. (2012, November). An analysis of the GTZAN music genre dataset. In Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies (pp. 7-12).

[4] Sturm, B. L. (2013). The GTZAN dataset: Its contents, its faults, their effects on evaluation, and its future use. arXiv preprint arXiv:1306.1461.

[5] Gemmeke, J. F., Ellis, D. P., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., ... & Ritter, M. (2017, March). Audio set: An ontology and human-labeled dataset for audio events. In 2017

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 776-780). IEEE.

[6] Defferrard, M., Benzi, K., Vandergheynst, P., & Bresson, X. (2016). Fma: A dataset for music analysis. arXiv preprint arXiv:1612.01840.

[7] Xu, C., Maddage, N. C., Shao, X., Cao, F., & Tian, Q. (2003, April). Musical genre classification using support vector machines. In 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). (Vol. 5, pp. V-429). IEEE.

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

[9] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.

[10] Choi, K., Fazekas, G., & Sandler, M. (2016). Explaining deep convolutional neural networks on music classification. arXiv preprint arXiv:1607.02444.

[11] Ajoodha, R., Klein, R., & Rosman, B. (2015, November). Single-labelled music genre classification using content-based features. In 2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech) (pp. 66-71). IEEE.