Music Genre Classification Based on Machine Learning Methods

Xundong Ma *

Hwa Chong International School, Singapore

* Corresponding Author Email: ma.xd@stu.hcis.edu.sg

Abstract. A large portion of the population globally are actively downloading and streaming music from online platforms in recent years. In many parts of the world, music has become a lifestyle instead of a luxury. As a result of music's growing popularity, the number of sounds and music released each day and also increased tremendously as the demand growth of music. Music genre is defined as a label that is descriptive of the music category which is used to categorize music based on several characteristics including: harmonic contents, pitch, instrumentation, and rhythmic structure. A classification model used to complete this task of classifying music genres is a machine learning system designed to classify using audio signals from song tracks into different musical genres. This paper aims to investigate the different approaches, including k-nearest neighbours algorithm, random forest, artificial neural network, and convolutional neural network, to classify music genres. Model performance analysis and confusion matrix analysis are applied to compare the advantages of the different algorithms applied on music genre classification problem.

Keywords: Machine learning, Deep learning, KNN, RFC, ANN, CNN, Music genre classification.

1. Introduction

Classifying music is important for consumers who stream music according to the genre they prefer. The sheer volume of the task makes it impractical to be done manually, moreover, the task cannot be reliably done by any average joe as from studies the average person can only reliably and accurately classify and correctly label 70% of the given songs. This is a decent prediction as in comparison, the random chance to get a correct label is 10%, however, both the accuracy and efficiency can be improved by using machine learning models. Furthermore, automatic music genre classification is required behind the concept of "smart playlist" function which generates song suggestions based on various factors one of which being a consumer's favourite genre. This function can be found on many modern popular music streaming platforms and enables consumers to mine for their desired tune quicker. This classification is also used by online radio stations that streams similar songs based on genre preference.

Classifying music using machine learning may sound like a daunting task due to the nature of the music. The task can be ambiguous and subjective to human bias. The goal for this project is to explore different methods of machine learning implementation both supervised and deep learning models to solve this problem.

In this project, labelled data is used for the creation of such machine learning models. This is done by firstly automatically extract features such as instrumentation, pitch, rhythmic structure, and harmonic contents from the input. The input is in the form of raw audio music track that can be found available online in .au format and the system will use these extracted features to determine and classify the audio track into 10 major genres. This means that the problem is a multi-class classification [1] problem.

2. Related work

Wyseand Smoliar's work [2] focused on categorizing general audio signals into three categories namely "speech", "music" and "others". Based on their findings, to identify audio signal that belong to the music class, mean average time period between peaks in narrow frequency range was used.

Afterwards, pitch tracking was used to identify and label signals belonging to the speech class. This was the first step before the further classification into smaller classes such as the music genres.

Historically audio features are designed manually, either inspired by speech procession algorithms or by the acoustic of musical instruments. Identifying the right features are important for supervised learning. Peeters [3] showed in his research features that could be used to characterize the timbre of music instruments. This is further summarized by Scaringella, N., Zoia, G [4] who proposed a list of low-level timbre features that help to characterize music genres, including: temporal features, energy features, spectral shape features and perceptual features.

G. Tzanetakis [5] used models such as the Gaussian model and k-nearest neighbors together with timbral texture, rhythmic content and pitch content features extracted manually. Their work achieved a 61% accuracy while the average human ability to classify music genre is 70% [4].

3. Methodology

3.1. Dataset and feature extraction

The GTZAN dataset is a popular public source of data for training machine learning models. It is widely used in the music genre classification tasks as it contained audio files which consists of songs dating back to the 2000s.

This GTZAN Music Genre Dataset contains 1,000 music samples. Every audio track is kept at 30 seconds long, each audio recording is classified into 1 of the 10 conventional music genres all in .au files. The 10 genres include: blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock. The sampling rate (Hz) used for this project is 22050Hz and the frame size used is 2048 bytes. 96 features are extracted from each sample from 810 samples in total for training purposes. The main aspects of the music include melody, harmony, rhythm, timbre and spatial location. They are taken into consideration when deciding significance of the features.

Timbre is the quality of the sound or tone played on a specific instrument, when two instruments play at the same pitch and volume while sounding perceptually different, timbre is produced. These features of timbre can be used to analyse spectral distribution of audio signal.

Timbre features are important in this project and the focus will be on the following features, temporal features such as zero crossing rate are extracted and computed from the waveform while spectral shape features are calculated from the Short Time Fourier Transformation of the signal.

Another approach is required in order to implement the convolutional neural network (CNN) models [6]. This is because conventionally CNN models only take in 4D arrays as their input type. The four dimensions of the array are typically the batch size of the image, height width and depth which represent the number of colour channels.

Since the raw input type is audio signals, further pre-processing is required. Audio signals are one dimensional. A graph of amplitude against time shown below in Figure 1.

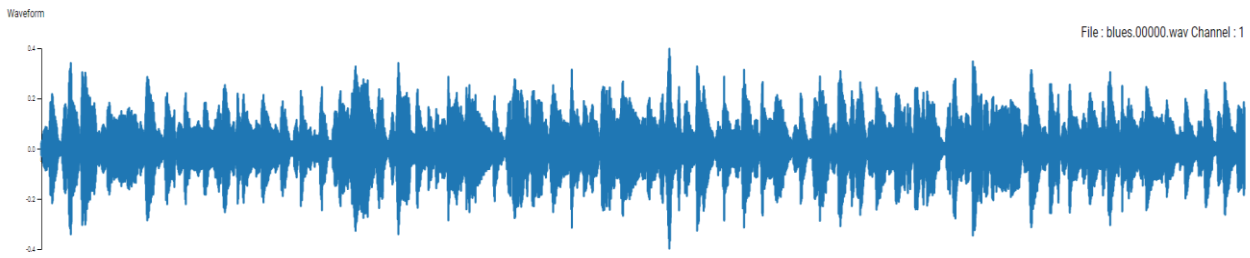

Figure 1. Waveform graph

In order to transform the presentation of the data into a 4 dimensional format, a more detailed and descriptive way of representation can be used. In this study, the raw audio signals are processed and transformed into Mel spectrogram graphic representation. A mel spectrogram logarithmically renders

frequencies above a certain threshold. In the mel spectrogram, the space between those ranges is approximately the same. This process is crucial as it simulates the human perception of music as humans have a higher ability to differentiate between similar low frequency sounds compared to high frequency sounds. This transformation can be done by using the Short time Fourier Transformation (STFT) on the audio signal across short intervals. By using the conventional 20ms as the time interval, and the formula for the STFT is given by formula (1):

$$\sum_{m=-\infty}^{\infty} X_m(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-jn(\omega)} \quad (1)$$

Where the $x(n)$ is the input signal at time n and periodic signal $\omega(n)$ frequency ω and shift m.

This is then projected onto the Mel scale by logarithmically rendering frequencies with the function (2):

$$m = klg\left(1 + f \times \frac{1}{700}\right), k = 2595 \quad (2)$$

The entire process is performed in the background using the Librosa library in python.

A corresponding Mel spectrogram is shown in Figure 2.

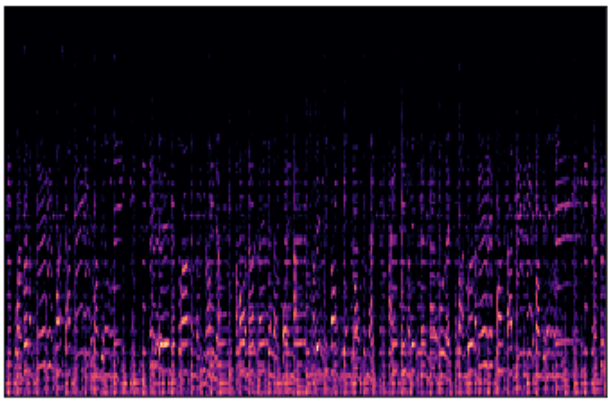

Figure 2. Mel Spectrogram

In this study, two simpler supervised learning models including k-nearest neighbours and, Random Forest, will be used in comparison to two other more complexed models like the Artificial Neural Network and the Convolutional Neural network in order to improve the accuracy of the predictions.

3.2. K-nearest neighbours

K-nearest neighbours algorithm, in short KNN, is a machine learning model classified under supervised learning models. In this case it is used to make classification predictions on the genre of each audio track. Although KNN can be used for both regression and classification problems, it is more commonly used for classification tasks. One assumption on KNN is that data points that are found close to each other belong to the similar classes. The k value in the KNN algorithm represents the k number of neighbours which is crucial in determining the performance for the classifier. One major issue with using KNN is to determine the optimal k value. After experimenting with different k values it is found that k = 10 yield the best result. The model works by using the Euclidean Distance which is calculated between the datapoint in query and all the datapoints in training samples. The distances are sorted and the weights for each of the 10 closest are chosen. It is proportional to the Euclidean distance. The weights would then be used to predict the class that each point falls into.

This process is implemented using the sklearn library in Python.

Some advantages of using the KNN are presented as below:

(1) The algorithm is easy to understand and implement

(2) The different distance metrics can be applied. The KNN is flexible as the model allows for a variety distance metrics. The user can choose the best distance metric that best suit the scenario. The collection of distance metrics includes: Euclidean, Minkowski, and Manhattan distance etc. In this case, the Euclidean distance is used.

The classification report for the KNN model is shown in Table 1:

Table 1. Classification report for the KNN model

Genres	precision	recall	f1-score	support
Blues	0.36	0.21	0.27	19
Classical	0.76	0.84	0.80	19
Country	0.43	0.32	0.36	19
Disco	0.27	0.63	0.38	19
HipHop	0.41	0.47	0.44	19
Jazz	0.44	0.21	0.29	19
Metal	0.55	0.58	0.56	19
Pop	0.82	0.74	0.78	19
Reggae	0.22	0.26	0.24	19
Rock	0.12	0.05	0.07	19
accuracy			0.43	190
macro avg	0.44	0.43	0.42	190
weighted avg	0.44	0.43	0.42	190

The confusion matrix for the KNN model is shown in the Figure 3:

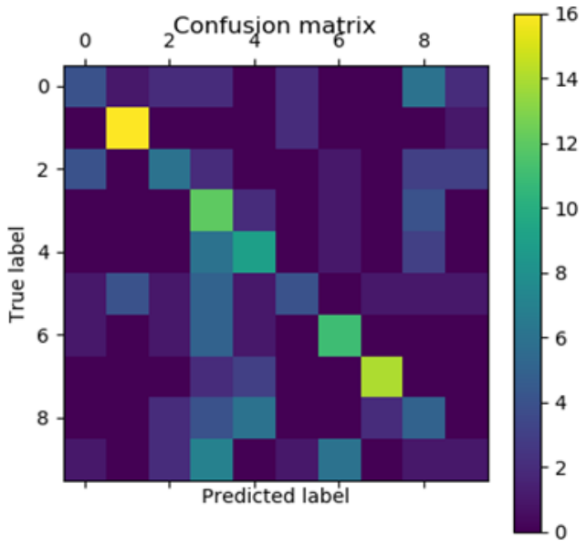

Figure 3. Confusion matrix for the KNN model

3.3. Random Forest Classifier

Random Forest Classifier (RFC) [7] is another supervised training model can be used for classification; it works by using individual decision trees to obtain a collective decision. Each tree produces prediction for the genre of the input audio and the genre with the highest number of votes become the final result obtained from the classifier. Since Decision Trees can be any depth, decision tree tends to overfit every data point. This would result in a 100% accuracy however the model does not generalize to data.

Some advantages of using random forests are presented as below:

(1) Random Forests do not have as many model assumptions as regression-based algorithms or support vector machines. This allows users to quickly build random forests to establish a base score to build on.

(2) Random Forest class provided by the sklearn library allow users to get the feature importance of each of the columns in the dataset. Through this, it can be easily found out which of the features contributes the most to the model prediction.

3.4. Artificial Neural Networks

Artificial Neural Network (ANN) [8] is a form of deep learning model that was inspired by the Biological Neural Networks (BNN) of human brains. This model attempts to replicate the rationale behind a biological neural network, however, although both ANN and BNN are very similar, they are not exactly the same.

ANN model has input requirement limiting to only numeric and structured data. In order to input data in other forms, like speech and music for instance, Convolutional Neural Networks (CNN) can be used instead of ANN which would be explored in the next section. The dataset used previously fits the input requirement for ANN as the input features extracted are in numerical form and organised in a data frame.

Supervised learning models that are less complexed, do not experience an increase in performance when there is an increase in the amout of data available. In comparison, the ANN outperforms these traditional models in this aspect as the size of the dataset and the performance is directly proportional for ANN.

Since Music genre classification is a complex problem, after extraction of audio features using Librosa from the sample files and splitting the dataset into training set and test set, the ANN model is implemented using TensorFlow and Keras. The output shape should be the same as the number of classes. Since the number of music genre is 10, 10 will be the output shape. The architecture of the model will consist of 3 dense layers.

Layer 1 contains 100 neurons, with the input shape being the number of features which is 96. To introduce some non-linearity which can assist in identifying more complexed patterns, Relu activation function is used.

To deal with any overfitting issues, a Dropout layer is introduced. This works by randomly dropping out nodes which is done at the expense of additional computation. However, the additional computation is insignificant while effectively serving as a regularization function to prevent overfitting. For this project, the rate is set at 0.5.

The Relu activation function is given by (3):

$$ReLU(k) = \max(0, k) \tag{3}$$

Layer 2 contains 200 neurons, with the same activation and regularization as Layer 1.

Layer 3 has an identical structure as Layer 1.

The reason for using the ReLU function is that ReLU does not activate all the neurons at the same time, which is more computationally efficient.

Adam optimizer is the best in terms of accuracy which is achieved in a relatively short time, as unlike other optimizers which maintain a single learning rate throughout training, Adam optimizer updates the learning rate for each network weight individually.

The obtained accuracy for the ANN model is around 69%

3.5. CNN

CNN is used as it is effective in recognising patterns in audio signals and images It is different from the ANN due to its unique layers which consists of convolution layers and pooling layers. This enables CNN to detect spatial features. In this study, three convolutional layers with max pooling and regularization are used. The convolution layer is the most important portion in the CNN architecture. It carries the main portion of the network's computational load. The convolutional layer uses multiple filters to traverse through the input to learn the different features and takes the scalar product between the learnable parameters and the kernel before adding the bias. The kernel is spatially smaller than an image but is more in-depth [9]. This means that the CNN has sparse interactions. Few parameters is

identified and stored which boosts the performance of the CNN compared to ANN. Every output unit interacts with every input unit.

The max pooling that is used to reduce the dimensions of the following layers helps in decreasing the amount of computation and weights, and boost efficiency. The most popular max pooling is used, which takes the maximum value.

For the hidden layers, similar Relu function is chosen to avoid overfitting. For the output layer, the model uses softmax activation function. The softmax activation function is given by (4):

$$SM(u)_i = \frac{e^{u_i}}{\sum_k^K e^{u_k}} \tag{4}$$

This softmax function reduces the range to (0,1), which can be viewed as the probability for each element when they all add up to 1.

In order to optimize the CNN, cross entropy loss is used which is given by (5):

$$\min\left(CrossEntropy(\theta)\right) = \min\left(-\sum_{x\in X} y(x)\log\left(\hat{y}(x)\right)\right) \tag{5}$$

The architecture is shown in Figure 4:

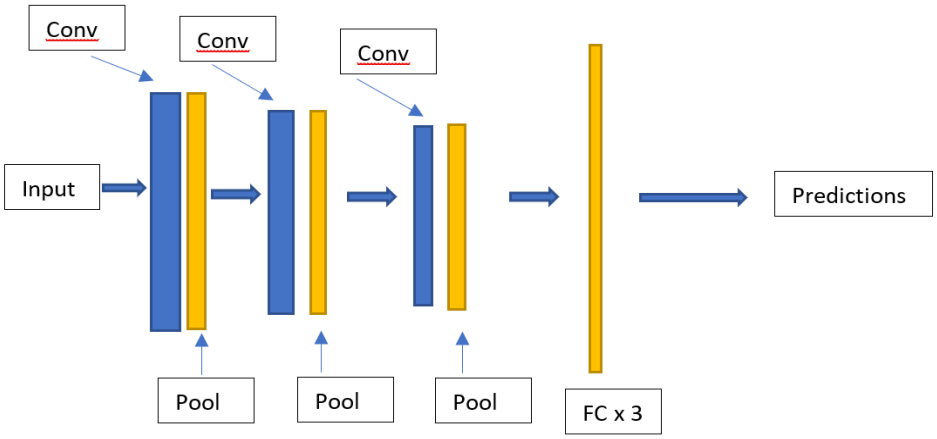

Figure 4. CNN architecture

The validation accuracy achieved by the CNN is around 80%. This model is by far the best model in terms of its accuracy, this model can also be used together with RNN [10] and may improve the accuracy further.

The classification report for CNN is shown below in Table 2:

Table 2. Classification report for CNN

Genres	precision	recall	f1-score	support
Blues	0.83	0.50	0.62	10
Classical	0.71	1.00	0.83	20
Country	0.67	0.20	0.31	10
Disco	1.00	0.80	0.89	10
Hip hop	0.71	0.50	0.59	10
Jazz	1.00	0.20	0.33	10
Halal	1.00	0.60	0.75	10
Pop	1.00	0.80	0.89	10
Reggae	0.67	0.40	0.50	10
Rock	0.00	0.00	0.00	10
micro avg	0.83	0.50	0.62	100
macro avg	0.70	0.50	0.57	100
weighted avg	0.76	0.50	0.57	100
sample avg	0.50	0.50	0.50	100

4. Conclusions

In this study, the main quantitative metric used to measure the performance of the various models is the accuracy of the various models in their prediction of the classes. The CNN model used hyperparameters that are conventionally used based on existing results from the previous academic works which means that through more trials and experimentation, more optimal options for hyperparameters may have yet to be discovered which can further improve the performance of the CNN. Furthermore, in general the models can be improved by using a larger dataset. It is also clear that CNN is the model that performed the best with an accuracy of 80% followed by the ANN with 69% accuracy, then the RFC at 61%, lastly KNN at 43%. Based on the classification reports, in general, the models do not do well with when it comes to identifying Rock music with RFC being the best at doing so with the highest accuracy of only 35%.

Throughout this study, the models all struggled with the problem of over-fitting. To overcome this issue, the number of elements in the hidden layers was reduced and the amount of layers has also been minimized. A higher number of trainable parameters would make memorization more probable, however, reducing this too much would also lead to underfitting and the model would miss out on some important patterns. Weight regularization is also used to add cost to loss function for the higher weights. Different combinations of dropouts were also tested which also improved the training speed.

In this study, two methods of complexed neural network models were explored, which are ANN and CNN. From the analysis, it is verified that these deep learning models tend to work better than the simpler sklearn models like k-nearest neighbour and Random Forest Classifier. The option of CNN working together with Recursive Neural Networks (RNN) could be explored. This is because RNNs performs well in comprehending sequential data by taking into account the progression of time. Spectrograms used in CNN contains the component of time, making the input type for RNN similar to that of the CNN. As RNNs excel identifying the short term and longer term temporal features in the song, the option of CNN-RNN models would combine the advantages of both models. The CNN-RNN model passes the input spectrogram through both CNN and RNN layers at the same time, this is because the RNN would be able to identify patterns from the original temporal information. Next, the outputs would be concatenated sent through a the softmax activation in dense layer to obtain the genre prediction. This method could ensure that both spatial and sequential patterns are taken into consideration when making a prediction.

References

[1] Brownlee, J. 2020. 4 Types of Classification Tasks in Machine Learning. https://machinelearningmastery.com/types-of-classification-in-machine-learning/#:~:text=In%20machine%20learning%2C%20classification%20refers,one%20of%20the%20known%20characters.

[2] A NEW APPROACH FOR CLASSIFICATION OF GENERIC AUDIO DATA | International Journal of Pattern Recognition and Artificial Intelligence. 2021. https://www.worldscientific.com/doi/10.1142/S0218001405003958

[3] Peeters, G., Cornu, F., Doukhan, D., & Regnier, L. 2015. When audio features reach machine learning. https://www.researchgate.net/publication/329878354_When_audio_features_reach_machine_learning

[4] Scaringella, Nicolas & Zoia, Giorgio & Mlynek, Daniel. (2006). Automatic genre classification of music content: a survey. Signal Processing Magazine, IEEE. 23. 133 - 141. 10.1109/MSP.2006.1598089.

[5] Tzanetakis, G., & Cook, P. 2002. Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing. https://doi.org/10.1109/tsa.2002.800560

[6] Verma, S. 2019, August 31. Understanding Input Output shapes in Convolution Neural Network | Keras. https://towardsdatascience.com/understanding-input-and-output-shapes-in-convolution-network-keras-f143923d56ca

[7] Ram, S. 2020. Mastering Random Forests: A comprehensive guide - Towards Data Science. https://towardsdatascience.com/mastering-random-forests-a-comprehensive-guide-51307c129cb1

[8] Rajan, S. 2020. An Introduction to Artificial Neural Networks - Towards Data Science. https://towardsdatascience.com/an-introduction-to-artificial-neural-networks-5d2e108ff2c3

[9] Mishra, M. 2020. Convolutional Neural Networks, Explained - Towards Data Science. https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

[10] Dwivedi, P. 2018. Using CNNs and RNNs for Music Genre Recognition - Towards Data Science. https://towardsdatascience.com/using-cnns-and-rnns-for-music-genre-recognition-2435fb2ed6af