Music Genre Classification: A Comparative Study Between Deep Learning and Traditional Machine Learning Approaches

Chapter · January 2022

DOI: 10.1007/978-981-16-2102-4_22

2 authors:

Dhevan Lau
University of the Witwatersrand
1 PUBLICATION **20** CITATIONS

SEE PROFILE

Ritesh Ajoodha
University of the Witwatersrand
66 PUBLICATIONS **422** CITATIONS

SEE PROFILE

Music Genre Classification: A Comparative Study Between Deep-Learning And Traditional Machine Learning Approaches

Dhevan Lau (1433596)
School of Computer Science and Applied Mathematics
The University of the Witwatersrand, Johannesburg
South Africa
1 December 2020
Supervisor: Dr. Ritesh Ajoodha

***Abstract*—Classifying music by their genres has been an ongoing problem in the field of automatic music classification. This is due to their abstract and subjective nature as overlapping genres for a given song are becoming more prevalent with the evolution of music. The use of deep learning models has risen in popularity and as such, this paper provided a comparative study on music genre classification using a deep-learning convolutional neural network approach against 5 traditional off-the-shelf classifiers. Feature selection included both spectrograms and content-based features. The classifiers were performed on the popular *GTZAN* dataset and our experiments showed similar prediction results on test data at around 66%.**

***Index Terms*—music genre classification, music information retrieval, deep-learning, machine learning, content-based features, spectrograms, comparative study**

I. Introduction

Music has played an important role in society throughout the ages as it is a means of entertainment, bringing people together, building communities and fan-base followings. These communities can take the form of like-minded people having preferences for specific genres of music. In today's age, we have digital music services (Spotify, Soundcloud and Apple Music etc.) responsible for maintaining extremely large databases of music. For music services to make recommendations to its users, machine learning algorithms are put in place for retrieval and classification of songs.

A common method of distinguishing music is by their genre. Musical genres are often defined by songs having common characteristics such as instrumentation, harmonic content and rhythmic structure [1]. Music genre classification is an Automatic Music Classification (AMC) problem in the area of Automatic Music Retrieval (AMR) [2]. Even to this day, it has been a challenging task to classify music genres due to their subjective nature. With the increasing variety of music coming out, the borders between music genres start to blur and overlap. Therefore developing good and accurate machine learning models to automatically classify and provide suitable recommendations to users based on their music library, would prove extremely invaluable to music streaming companies [3].

A. Motivation

Companies seek efficient methods of extracting, structuring and analysing the vast amounts of incoming data that would take humans years to comprehend. Deep-learning has seen its rise with the introductions of big data problems and the evolution of the digital era. Digital music services would benefit tremendously using deep-learning for information retrieval, with their extensive music archives consisting of millions of songs.

The music streaming industry is increasingly growing and looking for faster, more efficient machine learning models to be used for music information retrieval and classification. This research paper compares machine learning approaches within the context of music genre classification.

B. Research Problem

Majority of previous literature make use of content-based features and traditional machine learning approaches for classification. This research will contribute towards further exploring the use of audio signal waves translated into spectrogram images as feature sets for a deep-learning approach. More specifically using a convolutional neural network to classify the GTZAN dataset.

The hypothesis presented is such that the overall music genre classification accuracy of a deep-learning convolutional neural network, using spectrogram images as input, will be greater than the classification accuracy of traditional machine learning approaches, using content-based features for input, for the same given audio dataset.

C. Research Overview

The research presented performs automatic music genre classification using both spectrogram images, and content-based features that have been extracted from a dataset of audio signals. This is a supervised learning problem, making use of deep-learning and traditional off-the-shelf machine learning approaches. The deep-learning approach consists of a convolutional neural network and the traditional approach models consist of logistic regression, k-nearest neighbours,

support vector machine, random forests, and a simple multilayer perceptron.

A preprocessed dataset of features of the GTZAN music dataset was used. This preprocessed dataset consisted of 1000 x 30-second song excerpts uniformly classified into 10 genres. This dataset was duplicated and further divided into 10 000 x 3-second song excerpts to increase the amount of training data provided. Spectrograms images and 57 content-based features were extracted from each song excerpt to produce two prepossessed datasets (One for the 30-second feature set and one for the 3-second feature set). Both these datasets were trained on and results show better classification accuracy for models trained using the 3-second feature set. Additionally, our results show that the spectrogram-input deep-learning model performs at the same level of the content-based feature models approaches in terms of classification accuracy.

II. Background

A. Datasets

The *GTZAN* dataset contains 1000 music excerpts, where each song is 30 seconds long and categorised into 1 of 10 genres: Classical, Hip Hop, Country, Rock, Metal, Blues, Pop, Jazz, and Disco [4]. The dataset has been utilised in over 100 published articles and is one the most popular public datasets available for music genre recognition (MGR) in the field of machine learning [5]. The dataset originated in the early 2000s when it was compiled by [1], however, they never intended for their dataset to become one of the most popular benchmarks for MGR. Despite its popularity, [4] provided an analysis of the dataset composition and found numerous integrity problems. Just to name a few, out of the entirety of the GTZAN dataset, 50 excerpts were exact replicas, 22 excerpts were from the same audio file, 13 excerpts were of the same song but from different recordings. The full analysis can be read at [4].

Audio Set is a substantially large dataset consisting of human-annotated sound clips. The sound clips are each around 10 seconds in length and were extracted from a total of 2.1 million YouTube videos. Each sound clip is carefully annotated into an ontology of 632 audio event categories which include the music category and genre subcategory. The dataset was created by [6] to bridge the gap of image and audio research data availability, and accelerating research in the acoustic audio event detection field. While this dataset consists of an extreme variety of sound effects, it has found a use for MGR as seen in [3].

Free Music Archive is amongst one of the more recent public datasets used for MIR and MGR research. This dataset was introduced by [7] to overcome the limited availability of large-scale music collections. The dataset consists of 106 574 tracks from 16 341 artists and 14854 albums categorised into a hierarchical collection of 161 genres. Audio tracks are of high full-length quality and include pre-computed features, metadata and tags. [7] has also provided a small subset version of the Free Music Archive. This subset consists of 8000 x 30 second sound clips from the top 8 most popular genres of the full dataset. This subset is balanced with 1000 sound clips per genre and is very similar to that of the GTZAN dataset with the addition of pre-computed features, metadata and copy-right free audio files.

B. Spectrogram Features

Spectrograms are 2-dimensional graphical representations of an audio signal. The x-axis represents time and the y-axis represents frequency. Audio signals can be converted into MEL spectrograms where the y-axis represents MEL-frequency bins instead. Figure 1 illustrates spectrograms for a single audio signal obtained from different music genres in the *Audio Set* dataset. These spectrograms were obtained from literature provided by [3].

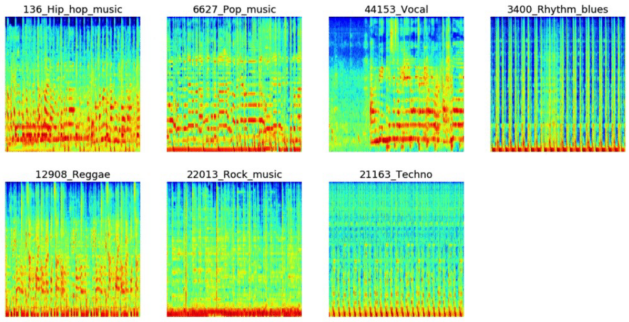

Fig. 1: Examples of spectrograms obtained from a different music genres [3]

C. Content-Based Features

Content-based features are extracted from raw audio signals. These features can be represented using mean, standard deviation, variance, feature histograms, MFCC aggregation and area moments [8]. This subsection will go through a select few of these features and briefly explain them.

Manually extracted or content-based features can be split into the time domain and frequency domain [3].

1) Time Domain Features: These features are extracted directly from the audio signal.

- **Root Mean Square Energy**: The value of the energy in a discrete-time signal after taking the root mean square of it.
- **Zero Crossing Rate (ZCR)**: The frequency at which a discrete-time signal changes sign.
- **Tempo**: Refers to the speed of the musical piece, expressed in beats per minute (BPM). Tempo can vary throughout a given piece of music so it should be aggregated using the mean across several frames in the piece.

2) Frequency Domain Features: A Fourier transform is applied to an audio signal to transform it into the frequency domain. The following features can then be extracted.

- **Mel-Frequency Cepstral Coefficients (MFCC)**: These are the coefficients that make up the Mel-frequency cep-

strum. These are obtained from the cepstral representation of the audio signal.

- **Chroma**: A vector corresponding to the total energy of the signal for each of the 12 semitone pitches, regardless of octave.
- **Spectral Centroid**: The frequency at which the most energy is centred around for a given frame. It is a magnitude weighted frequency.
- **Spectral Bandwidth**: Defined as the band width of an audio wave at one-half the peak maximum.
- **Spectral Roll-off**: The spectral roll-off point is defined as the frequency where 85% of the total signal energy is contained under.

D. Classification Models

The following classification models were chosen in this research paper based on their success in the past.

- **Logistic Regression (LR)**: LR classifiers are generally used for binary classification problems. However, it can be implemented as a one-vs-rest method. I.e. training separate stand-alone classifiers for each music genre and taking the classifier that yields the highest probability for a genre as the final chosen genre label. [8] reported their linear LR achieving an accuracy of 83% using the *GTZAN* dataset.
- **K-Nearest Neighbours (KNN)**: The KNN algorithm can classify songs based on other songs of very similar features. *K* determines how many neighbouring songs are used to determine the label of an unseen song. *K* is always chosen as an odd number to break ties. [8] reported an accuracy of 72.8% for their KNN classifier on the *GTZAN* dataset.
- **Support Vector Machines (SVM)**: This model transforms the given input into a high dimensional space using a kernel [9]. Multi-layer SVM classifiers in the past have been able to produce optimal class boundaries between genres labels. [10] reported the accuracy of 93.14% using a support vector machine, albeit with a very small sample size of 100 samples. [8] reported accuracies of 75.4% on the *GTZAN* dataset. However, [3] reported a low accuracy of 57% using the *Audio Set* dataset.
- **Random Forest (RF)**: RFs are ensemble learning classifiers that combine the predicted labels from a set number of decision trees. Each decision tree is trained using a subset of the training data and required to make a prediction using only a random subset of features. The final predicted class of the Random Forest is determined by the predominant outcome of the individual classifiers. [8] reported an accuracy of 75.7% for their RF classifier on the *GTZAN* dataset.
- **Simple Multilayer Perceptron (MLP)**: MLPs are feed-forward artificial neural networks. Neural networks are a set of algorithms modelled after the human brain. They are used for pattern recognition on numerical input data and output a label. Various forms of inputs such as images, audio and text must first be translated into numerical data for neural networks to interpret them. [8] reported an accuracy of 75.2% for their MLP classifier on the *GTZAN* dataset.
- **Convolutional Neural Network (CNN)**: There are various forms of deep learning neural networks however, CNNs will be solely focused in this paper. CNNs have proven extremely effective at recognising image patterns [11]. As mentioned previously, spectrograms can be referred to as images and therefore can be fed into a CNN for learning. The CNN is broken down into the following operations:
  - **Convolution**: A matrix filter of a given size is applied over the given spectrogram. For each spectrogram element, an element-wise multiplication is calculated between the filter and the overlapping image elements followed by a summation of those values to give final convoluted value.
  - **Pooling/Down Sampling**: This step is required to reduce the processing time and storage of the feature map. It retains the maximum value among elements within a specified window size applied over the convoluted image (i.e among 4 elements using a 2x2 window). This window is moved over the convoluted image over a predefined stride.
  - **Non-linear Activation:** The stronger neural network can be obtained by introducing non-linearity. A Rectifying Linear Unit (ReLU) activation function can be applied to each element of the feature map.

  An example of CNN architecture is illustrated in Figure 2. [3] reports their highest accuracy obtained using a CNN at 64% on the *Audio Set* dataset and [11] reports an accuracy of 75% on a dataset obtained from *Naver Music*.

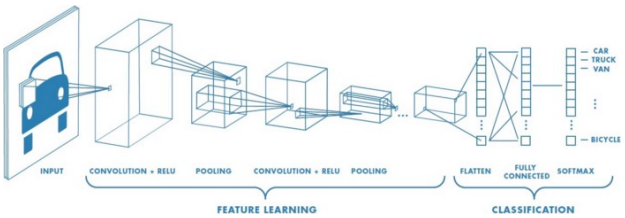

Fig. 2: Example convolutional neural network architecture. [12]

E. Previous Work

Some of the earliest works in music genre classification was by [1], who also created the GTZAN music dataset which would later become one of the most popular benchmark datasets for genre classification. [1] explores the classification of audio signals into musical genre hierarchies, as they believe music genres are categories created by people to label pieces of music based on similar characteristics. They propose three feature sets for representing rhythmic, timbral texture, and pitch characteristics. The following supervised machine learning classifiers were adopted for their experiments: k-

Nearest Neighbour and Gaussian Mixture model. They achieve a classification accuracy of 61% on the GTZAN dataset.

Some of the more recent works in music genre classification can be noted in [8] and [3]. [8] makes use of the GTZAN benchmark dataset, likewise in [1], and categorise features into four feature sets: magnitude spectrum, pitch, tempo and chordal features. The best and worst features that contribute to the learning process are determined using information gain rankings. This paper makes use of six off-the-shelf classifiers: Multilayer Perceptron, Logistic Regression, Support Vector Machines, Random Forests, k-Nearest Neighbours and Naïve Bayes. The results are compared between each classifier. Logistic Regression came out with the best performance. In addition to standard machine learning classifiers, [3] makes use of various Convolutional Neural Networks which are fed MEL spectrograms as inputs. Using the *Audio Set* dataset, the CNN reached performances accuracy up to 64%. Both [8] and [3] attribute Mel-frequency cepstral coefficients (MFCC) as one of the highest contributing features towards music genre classification.

Table I summarises some notable classification models related to this paper on various music datasets.

TABLE I: Notable genre classification models on various music datasets

Author	Dataset	Model	Accuracy
Ajoodha et al. [8]	GTZAN	Logistic Regression	81%
Bahuleyan [3]	Audio Set	VGG-16 CNN + Extreme Gradient Boosting	65%
Chillara et al. [13]	Free Music Archive	CNN	88%
Choi et al. [11]	Naver Music	CNN	75%
Tzanetakis and Cook [1]	GTZAN	Gaussian Mixture Model	61%

III. Methodology

This section outlines the methods and experiments performed for this research paper. This includes further preprocessing of the dataset, the features selected, and implementation details of the machine learning classifiers trained.

A. GTZAN Dataset

For our research, a preprocessed GTZAN dataset, made available at [14], was used. This dataset consisted of the raw audio files, extracted MFCC spectrograms for every given song, as well as several extracted content-based features in a CSV file. This dataset was further split into 3-second audio files with their respective content-based features in an additional CSV file. The 3-second dataset provides 10 times more data to train our models and opens up additional hypotheses of whether 3 seconds of a song is adequate for music classification purposes. The 3-second dataset, however, is not consistent with the number of samples per genre. It was found that some genres had slightly less or slightly more than 1000 samples where 1000 samples is the expected amount for each genre.

B. Features

The spectrograms provided in the dataset provided by [14] were of size 288x432 (height x width), however, they contained large white borders around the image. The spectrograms were each cut down to a size of 217x315 to remove the white borders before training of our deep learning model. An example of the spectrograms is illustrated in Figure 3.

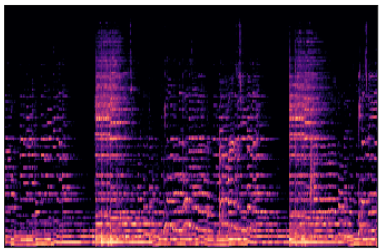

Fig. 3: Spectrogram of a rock song in the provided GTZAN dataset. [14]

Not all features were used that were provided in the 30-second and 3-second CSV files. The following 57 features that were selected for training, can be summarised as follows:

- chroma short-time Fourier transform (mean and var)
- root mean square error (mean and var)
- spectral centroid (mean and var)
- spectral bandwidth (mean and var)
- spectral rolloff (mean and var)
- zero crossing rate (mean and var)
- harmony (mean and var)
- tempo
- 20 MFCC coefficients (mean and var)

The dataset was split into 80% training data and 20% test data. The training data was further split into 10-folds for cross-validation purposes.

C. Deep Learning Approach

Our convolutional neural network architecture was built using Keras and consists of the input layer followed by 5 convolutional blocks. Each convolutional blocks consists of the following:

- convolutional layer using a 3x3 filter, 1x1 stride and mirrored padding.
- relu activation function
- max pooling with a 2x2 windows size, 2x2 stride
- dropout regularisation with a probability of 0.2

The convolutional blocks have filter sizes of (16, 32, 64, 128, 256) respectively. After the 5 convolutional blocks, the 2D matrix is then flattened into a 1D array, regularisation dropout is performed with a probability of 0.5. Lastly, the final layer consists of a dense fully-connected layer that uses a softmax activation function to output the probabilities for each of the 10 label classes. The class with the highest probability

is selected as the classified label for a given input. Categorical cross-entropy is calculated as follows:

$$CE = -\sum_{i}^{C} t_i * log(s_i) \quad (1)$$

Where t_i is the binary indicator (of a given sample) with the value 0 or 1 depending on if it belongs to class C. s_i is the CNN score for each class i in C. The softmax activation function is applied to the scores before the cross-entropy loss computation, $f(si)$ refers to the activations.

Three CNNs were trained using either the spectrograms, 20 MFCCs of the 30-second feature set, or 20 MFCCs of the 3-second feature set. The number of epochs set varied depending on the model. A final classification test was performed on the test sets after training.

D. Traditional Machine Learning Approaches

The following off-the-shelf traditional classifiers were implemented using the Scikit Learn library [15]. The hyperparameters for each classification model are detailed in Table II. 3-Repeated 10-fold cross-validation was performed on these models to reduce bias and present more reliable results. After validation, the models classified the unseen test dataset.

TABLE II: Implementation details of the traditional machine learning algorithms.

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

E. Evaluation Metrics

The following metrics were used to evaluate and measure the performance of the machine learning models:

- **Confusion Matrix**: A useful table\matrix that helps visualise the performance of a classification machine learning model. There are four different combinations of predicted and true values as seen in Table III below. For a given genre label g and a sound-clip s, TP (true positive) is when s is correctly classified as g; FP (false positive) is where the model classified s with g but its true genre is different. FN (false negative) where the model did not classify the sound-clip even though its true genre label is g. Lastly, TN (true negative) where the model correctly identified that s does not have genre label g.

TABLE III: Example of a basic confusion matrix

		True Positive (1)	True Negative (0)
Predicted	Positive (1)	TP	FP
	Negative (0)	FN	TN

- **Accuracy**: The percentage of sound clips that were correctly classified for a specific genre. It is calculated as follows:

$$Accuracy = \frac{TP}{TP + FP + FN} * 100\%$$

.

- **3-Repeated 10-Fold Validation Accuracy**: This is the mean validation accuracy of the model after performing 10-fold cross-validation on a model repeated 3 times. This is to provide a more reliable classification accuracy of a model and remove any bias in the way the data was split.
- **Training Time**: The time required to fit the training data to a given model. Measured in either seconds (s) or milliseconds (ms).

IV. Results and Discussion

This section presents the results of the experiments performed for this research. An analysis and evaluation of these results are detailed at the end of this section.

A. Classification Results

Numerical results for the traditional machine learning approaches can be viewed in Tables IV and V. Table IV displays the results of models that were trained using the full 30-second feature set and Table V shows the results trained on the 3-second feature set. Both tables indicate the average training time is taken to train the model after a single run-through in either milliseconds (ms) or seconds (s). 3-Repeated 10-Fold cross-validation was performed on each model and the mean validation accuracy was recorded to show a more reliable result. Final classification was performed on the unseen test dataset to obtain the test accuracy of the model. The best model for the 30-second and 3-second feature sets was determined based on the averaged validation and test accuracy. These are highlighted in their respective tables.

TABLE IV: Traditional classifier results using the 30-second input feature set.

Classifier (30-second features)	Training Time	3-Repeated 10-Fold Validation Accuracy	Test Accuracy
Logistic Regression	487 ms	66.04%	66.50%
K-nearest Neighbours	5 ms	66.17%	68.50%
Support Vector Machine	73 ms	66.50%	73.50%
Random Forests	**5.72 s**	**69.33%**	**74.50%**
Multilayer Perceptron	60.62 s	62.87%	67.50%

TABLE V: Traditional classifier results using the 3-second input feature set.

Classifier (3-second features)	Training Time	3-Repeated 10-Fold Validation Accuracy	Test Accuracy
Logistic Regression	3672 ms	68.84%	67.52%
K-nearest Neighbours	**78 ms**	**92.66%**	**92.69%**
Support Vector Machine	3872 ms	74.77%	74.72%
Random Forests	52.89 s	80.86%	80.28%
Multilayer Perceptron	134.25 s	80.98%	81.73%

The confusion matrices were plotted for each of the best models for the 30-second and 3-second feature sets. These are illustrated in Figures 4 and 5.

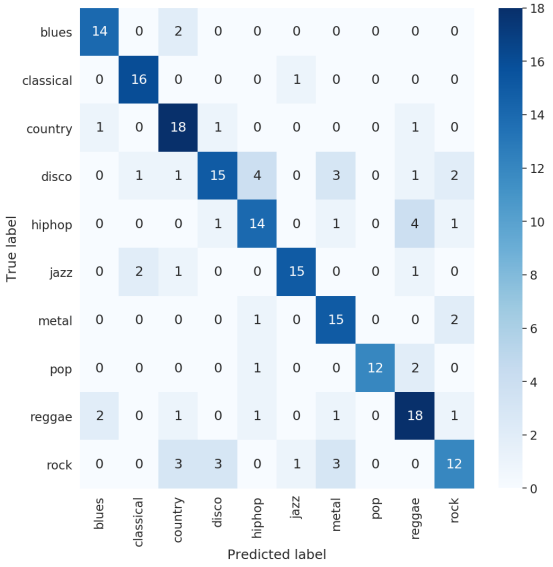

Fig. 4: Confusion Matrix for 10 GTZAN genres using the random forest model on the 30-second feature set.

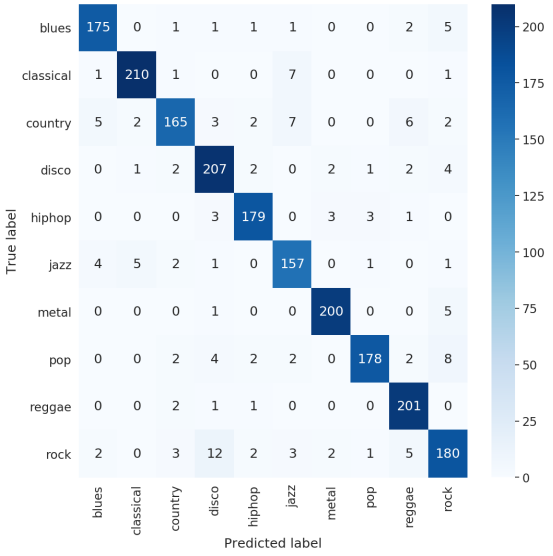

Fig. 5: Confusion Matrix for 10 GTZAN genres using the k-nearest neighbours model on the 3-second feature set.

Three separate convolutional neural networks were trained in our experiments. The first two were trained on 20 MFCCs obtained from the 30-second and 3-second feature sets and the final model took in the spectrogram images as input. Each model was trained on a different number of epochs depending on the visual convergence of the validation accuracy and loss based on previous experiments. Figures 6, 7 and 8 illustrate the accuracy and loss with increasing amounts of epochs, of our respective CNN models during the final experimental phase. These models were then tasked to classify the unseen test set and the numerical results can be viewed in Table VI. The confusion matrix of the spectrogram-fed CNN on the test set is illustrated in Figure 9.

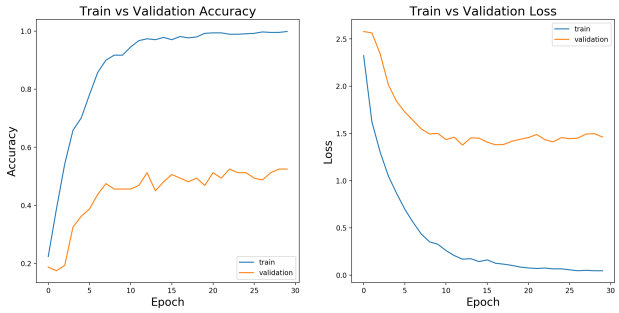

Fig. 6: Line graphs showing the accuracy and loss of training and validation for the CNN model trained on 30-second feature set.

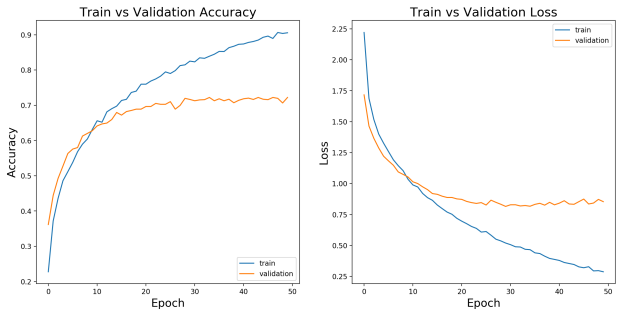

Fig. 7: Line graphs showing the accuracy and loss of training and validation for the CNN model trained on 3-second feature set.

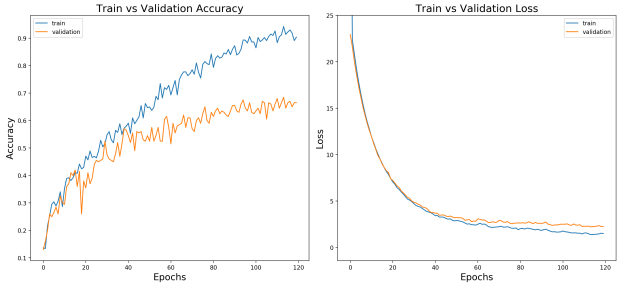

Fig. 8: Line graphs showing the accuracy and loss of training and validation for the CNN model trained on spectrograms.

B. *Evaluation Of Results*

Based on our results in Table IV we see that the random forests classifier comes out as the best performing model with a cross-validation accuracy of 69.33% and a test accuracy of 74.50%. The simple multilayer perceptron (MLP) produced poor results compared to the other models. This may be due to the extremely small training dataset size of 80 samples per genre and that neural networks require vast amounts of data to produce more accurate results. For small dataset sizes, the MLP is not a viable approach as it also takes the lead in the longest training time of 60.62 seconds.

TABLE VI: Convolutional neural network results using the 30-second, 3-second and spectrogram input feature sets.

Classifier	Epochs	Test Loss	Test Accuracy
CNN (30-Sec Features)	30	1.609	53.5%
CNN (3-Sec Features)	**50**	**0.873**	**72.4%**
CNN (Spectrograms)	120	2.254	66.5%

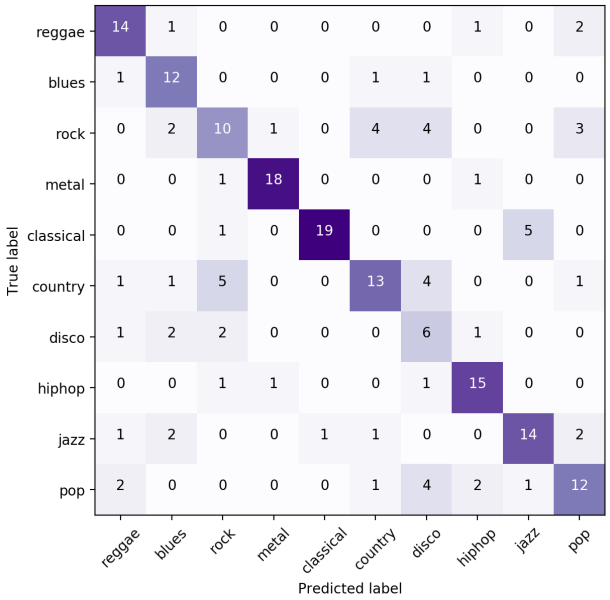

Fig. 9: Confusion Matrix for 10 GTZAN genres using a convolutional neural network model using spectrograms as input.

Taking a look at Table V we can immediately see a huge improvement in classification accuracy across the board compared to Table IV. The k-nearest neighbours (KNN) algorithm produced an exceptionally high test accuracy of 92.69% which is further backed by the 10-fold validation accuracy of 92.66%. Furthermore, the KNN model took the least amount of time to train at 78 milliseconds. These results prove KNNs are an extremely viable option for music genre classification with larger training dataset sizes. The MLP has the second highest classification accuracy which further justifies the need for more training data to produce more viable results. The confusion matrices shown in Figures 4 and 5 have shown that the rock genre has similar content-based features to disco as it appears to be the highest misclassified label out of all misclassifications.

Moving to the convolutional neural networks, the produced numerical results in Table VI are not of what we expected. The classification test accuracy is relatively lower than the traditional models' results for their respective input feature sets. The spectrogram CNN test accuracy at 66.5% is significantly lower than other CNN models in previous literature as seen early on in the paper in Table I. Although, this is a different music dataset that was used and the small training sample sizes may justify its poor performance. We see that in Figures 6 and 7, the validation accuracy and loss starts to converge after specific amount of epochs. Increasing the epochs will not improve the accuracy and therefore changes in the actual CNN model architecture need to be adjusted in hopes of obtaining better results. We do see the trend that the 3-second CNN outperforms the 30-second CNN model which is most likely due to having more training data to learn from. The spectrogram model may perform better if the number of epochs is increased, as we do see a steady increase in accuracy and decrease in the loss at 200 epochs as shown in Figure 8. Since the spectrogram images are based on the full 30-second audio wave, we only compare its results with the traditional classifiers trained on the 30-second feature set. The spectrogram CNN test model accuracy at 66.5% is on par with the validation accuracy of most of the traditional classifiers seen in Table 4. With more data and increasing the epochs, the CNN would be expected to outperform these traditional models. However, due to a shortage of time and computational resources, the epochs could not be increased further than 200 in our final experiments. The confusion matrix in Figure 9 shows good classification for all genres except for disco where some misclassification were to towards the rock genre once again.

V. Conclusion and Recommendations

Musical genre classification classifiers will continue to play an important role in digital music streaming services for music recommendation and retrieval to its users. With these service gaining increasingly large amounts of songs daily, developing faster and more efficient machine learning models for retrieval and recommendation is imperative to these streaming companies. Our paper compares deep-learning convolutional neural networks with traditional off-the-shelf classifiers. We find that the classification accuracy for both types of models produce a very similar result, although, the traditional model architecture was coded from an optimised library of premade models [15], whilst the CNN is a custom developed model coded in Keras which can be further optimised. Additionally, the GTZAN dataset that was used only contains a total of 100 samples each for its limited 10 genres. The integrity of the GTZAN dataset has also proven to be flawed as proven by [5]. Based on this evidence in our findings, our hypothesis stated in Section I-B can neither be accepted nor rejected as both model types produced similar results but the reliability of the results comes down the architecture of models themselves. However, based on previous CNN implementations used to classify genres on other music datasets, further research, optimal implementation of the CNN and more training data, it is expected that the CNN will outperform the traditional models on an average basis and the hypothesis could then be accepted.

This research made contributions towards using a convolutional neural network for music genre classification of the famous GTZAN music dataset, as well as looked into producing more training data using existing training data by further cutting audio samples into smaller samples resulting in more samples. Our findings concluded that using the smaller

but extended dataset resulted in higher classification accuracy in both traditional and deep-learning models. Most notably the k-nearest neighbours classifier produced astonishing validation and test accuracy results at around 92%.

Extensions to the research include using more useful content-based features and feature representations. The choice of features was limited only to the features provided by the preprocessed dataset [14]. No feature analysis was performed and as such the selected features are not proven to be the most useful. [8] presents the most optimal features based on an information gain system which could be considered for optimising feature selection. Training on other verified music datasets with ground truth labels can also be explored.

ACKNOWLEDGEMENT

I would like to give a big thank you to my supervisor, Ritesh Ajoodha, for providing me with the necessary inspiration, mentorship and tools to assist me with the completion of my research project. Despite the consequences of the on-going pandemic and lockdown regulations, he made himself available for online consultation whenever possible.

REFERENCES

[1] G. Tzanetakis, and C .Perry. "Musical genre classification of audio signals." IEEE Transactions on speech and audio processing 10, no. 5, pp. 293-302, 2002.

[2] S. Scardapane, D. Comminiello, M.Scarpiniti, and A Uncini. "Music classification using extreme learning machines." In 2013 8th international symposium on image and signal processing and analysis (ISPA), pp. 377-381. IEEE, 2013.

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

[4] B. L. Sturm. "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. 2012.

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

[6] J. Gemmeke, D. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. Moore, M. Plakal, and M. Ritter. "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, 2017.

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

[8] R. Ajoodha, R. Klein, and B. Rosman. "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, 2015.

[9] C. Cortes, and V Vapnik. "Support-vector networks." Machine learning 20, no. 3 (1995): 273-297.

[10] C. Xu, N. C. Maddage, X. Shao, F. Cao, and Q. Tian. "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, 2003.

[11] K. Choi, G Fazekas, and M Sandler. "Explaining deep convolutional neural networks on music classification." arXiv preprint arXiv:1607.02444. 2016.

[12] Medium, Neural network with many convolutional layers. 2018. [Online] Available: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148.

[13] S. Chillara, A. S. Kavitha, S. A. Neginhal, S. Haldia, and K. S. Vidyullatha. "Music Genre Classification using Machine Learning Algorithms: A comparison." (2019).

[14] A. Olteanu, "GTZAN Dataset - Music Genre Classification", Kaggle.com, 2020. [Online]. Available: https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification.

[15] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.