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

Dhevan S. Lau and Ritesh Ajoodha

Abstract Classifying music by their genres has been an ongoing problem in the field of automatic music classification. 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 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%.

Keywords Music genre classification · Music information retrieval · Deep learning · Machine learning · Content-based features · Spectrograms · Comparative study · GTZAN dataset

1 Introduction

Music has played an important role in society throughout the ages, as it is a means of entertainment and building of fan-base followings. 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]. 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.

D. S. Lau (✉) · R. Ajoodha
The University of the Witwatersrand, Johannesburg, South Africa
e-mail: 1433596@students.wits.ac.za
URL: https://www.wits.ac.za/

R. Ajoodha
e-mail: ritesh.ajoodha@wits.ac.za
URL: https://www.wits.ac.za/

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

X.-S. Yang et al. (eds.), *Proceedings of Sixth International Congress on Information and Communication Technology*, Lecture Notes in Networks and Systems 217,
https://doi.org/10.1007/978-981-16-2102-4_22

1.1 Motivation

In today's age, we have digital music services (Spotify, Soundcloud and Apple Music, etc.) responsible for maintaining extremely large databases of music. 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.

1.2 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 using a deep learning convolutional neural network approach. The hypothesis presented is such that the overall music genre classification accuracy of a deep learning convolutional neural network will be greater than traditional machine learning approaches for the same given audio dataset.

1.3 Research Overview

The research presented performs automatic music genre classification using both spectrogram images and content-based features extracted from a dataset of audio signals. This is a supervised learning problem, making use of a deep learning convolutional neural network and traditional off-the-shelf machine learning approaches consisting of logistic regression, k-nearest neighbours, support vector machine, random forests and a simple multilayer perceptron. A preprocessed dataset of spectrogram images and 57 content-based features of the GTZAN music dataset was used. These features were extracted from each excerpt in the dataset to produce a 30 s feature set and a 3 s feature set. Both of these datasets were trained on and results show better classification accuracy for models trained using the 3 s feature set. Additionally, our results show that the spectrogram-input deep learning model performs at the same level as the content-based feature model approaches in terms of classification accuracy.

Table 1 Notable genre classification models on various music datasets

Authors	Dataset	Model	Accuracy (%)
Ajoodha et al. [3]	GTZAN	Logistic regression	81
Bahuleyan [4]	Audio set	VGG-16 CNN + Extreme gradient boosting	65
Chillara et al. [5]	Free music archive	CNN	88
Choi et al. [6]	Naver music	CNN	75
Tzanetakis and Cook [1]	GTZAN	Gaussian mixture model	61

1.4 Previous Work Results

See Table 1.

2 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.

2.1 GTZAN Dataset

For our research, a preprocessed GTZAN dataset consisting of 1000 × 30 s song excerpts uniformly classified into 10 genres was used [7]. This dataset consisted of the raw audio files, extracted MFCC spectrograms and content-based features. This dataset was duplicated and further divided into 10,000 × 3 s song excerpts to increase the amount of training data provided. The 3 s dataset opens up additional hypotheses of whether 3 s of a song is adequate for music classification purposes. The 3 s 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 are the expected amount for each genre.

2.2 Features

The spectrograms provided in the dataset provided by [7], contained large white borders around the image. The spectrograms were each cut down to a size of 217 × 315 (pixels) to remove the white borders before training of our deep learning model. Not all features were used that were provided in the 30 and 3 s 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.

2.3 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 3 × 3 filter, 1 × 1 stride and mirrored padding.
- relu activation function
- max pooling with a 2 × 2 windows size, 2 × 2 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.

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

Table 2 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

2.4 Traditional Machine Learning Approaches

The following off-the-shelf traditional classifiers were implemented using the Scikit Learn library [8]. The hyperparameters for each classification model are detailed in Table 2. 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.

2.5 Evaluation Metrics

The following metrics were used to evaluate and measure the performance of the machine learning models: **Confusion Matrix**, **Classification Accuracy**, **3-Repeated 10-Fold Validation Accuracy** and **Training Time**.

3 Results and Discussion

This section presents the results and evaluations of the experiments performed for this research.

3.1 Classification Results

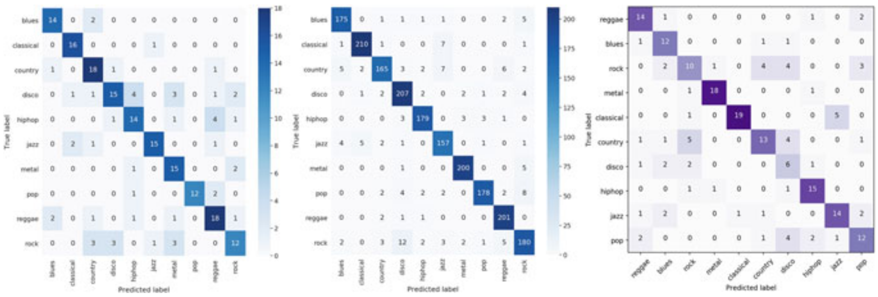
Numerical results for the traditional machine learning approaches can be viewed in Tables 3 and 4. Table 3 displays the results of models that were trained using the full 30 s feature set and Table 4 shows the results trained on the 3 s feature set. The best model for the 30 s and 3 s feature sets was determined based on the averaged validation and test accuracy. These are highlighted in their respective tables. The

Table 3 Traditional classifier results using the 30 s input feature set

Classifier (30 s 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 4 Traditional classifier results using the 3 s input feature set

Classifier (3 s 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



(a) (b) (c)

Fig. 1 Confusion Matrix for 10 GTZAN genres using the (**a**) random forest model on the 30 s feature set. (**b**) k-nearest neighbours model on the 3 s feature set (**c**) convolutional neural network model using spectrograms as input

confusion matrices were plotted for each of the best models for the 30 s and 3 s feature sets. These are illustrated in Fig. 1a and b, respectively.

Three separate convolutional neural networks were trained in our experiments. The first two were trained on 20 MFCCs obtained from the 30 and 3 s feature sets and the final model took in the spectrogram images as input. Each model was trained

Table 5 Convolutional neural network results using the 30 s 3 s and spectrogram-input feature sets

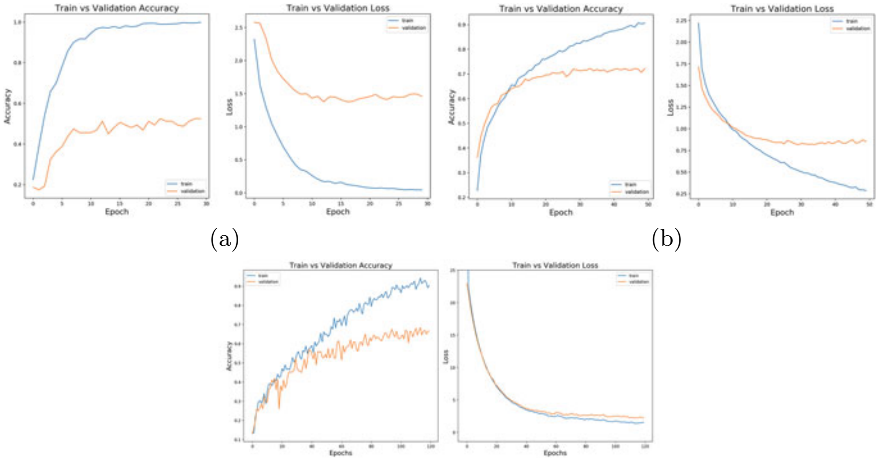
Classifier	Epochs	Test loss	Test accuracy (%)
CNN (30 s features)	30	1.609	53.5%
CNN (3 s features)	**50**	**0.873**	**72.4%**
CNN (Spectrograms)	120	2.254	66.5%

on a different number of epochs depending on the visual convergence of the validation accuracy and loss based on previous experiments. Figure 2 illustrates the final experimental results. These models were then tasked to classify the unseen test set. The numerical results can be viewed in Table 5. The confusion matrix of the spectrogram-fed CNN on the test set is illustrated in Fig. 1c.

3.2 Evaluation of Results

Based on our results in Table 3, 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 which 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. Taking a look at Table 4, we can see an improvement in classification accuracy across the board compared to Table 3. The k-nearest neighbours (KNN) algorithm produced a 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 that KNNs are a viable option for music genre classification with larger dataset sizes. The confusion matrices shown in Fig. 1 have shown that the rock genre has similar content-based features to disco as it appears to be the most misclassified label.

Moving to the CNNs, the produced numerical results in Table 5 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. 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 Fig. 2a and b, the validation accuracy and loss starts to converge after a 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 s CNN outperforms the 30 s CNN model, which is most likely due to having more training data to learn from. The spectrogram model may perform



(a)

(b)

(c)

Fig. 2 Line graphs showing the accuracy and loss of training and validation for the CNN model trained on **a** 30 s feature set **b** 3 feature set **c** spectrograms

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 Fig. 2c. Since the spectrogram images are based on the full 30 s audio wave, we only compare its results with the traditional classifiers trained on the 30 s 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 3. 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 Fig. 1c shows good classification for all genres except for disco, where some misclassification were towards the rock genre once again.

4 Conclusion and Recommendations

Our paper compares deep learning CNNs 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 [8], whilst the CNN is a self-developed model coded in Keras, which can be further optimised. The integrity of the GTZAN dataset has also proven to be flawed as proven by [9]. Based on this evidence in our findings, our hypothesis stated in Sect. 1.2 can neither be accepted nor rejected as both model types produced similar results, but the reliability of the results comes down to the architecture of models and dataset integrity. However, based on previous CNN implementations 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 would then be accepted. This research made contributions towards using a CNN for music genre classification on the GTZAN music dataset, as well as looked into producing more training data using existing training data by further cutting audio samples into smaller 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 high 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 [7]. No feature analysis was performed and as such the selected features are not proven to be the most useful. Ajoodha et al. [3] presents the most optimal features based on an information gain system which could be considered for optimising feature selection.

Acknowledgements I would like to give a big thank you to my supervisor, Ritesh Ajoodha, for providing me with the necessary mentorship and tools to assist me with the completion of my research. This work is based on the research supported in part by the National Research Foundation of South Africa (Grant number: 121835).

References

1. Tzanetakis G, Perry C (2002) Musical genre classification of audio signals. IEEE Trans Speech Audio Process 10(5):293–302
2. Scardapane S, Comminiello D, Scarpiniti M, Uncini A (2013) Music classification using extreme learning machines. In: 2013 8th international symposium on image and signal processing and analysis (ISPA). IEEE, pp 377–381
3. Ajoodha R, Klein R, Rosman B (2015) 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). IEEE, pp 66–71
4. Bahuleyan H (2018) Music genre classification using machine learning techniques. arXiv preprint arXiv:1804.01149
5. Chillara S, Kavitha AS, Neginhal SA, Haldia S, Vidyullatha KS (2019) Music genre classification using machine learning algorithms: a comparison
6. Choi K, Fazekas G, Sandler M (2016) Explaining deep convolutional neural networks on music classification. arXiv preprint arXiv:1607.02444. 2016
7. Olteanu A (2020) GTZAN dataset—music genre classification. Kaggle.com. https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification
8. Pedregosa et al (2011) Scikit-learn: machine learning in Python. JMLR 12:2825-2830
9. Sturm BL (2013) The GTZAN dataset: its contents, its faults, their effects on evaluation, and its future use. arXiv preprint arXiv:1306.1461