Comparison of machine learning techniques for music sub-genre classification

ELEC5305 SOMETHING AWESOME PROJECT
https://github.com/neuromorphic-chair/ELEC5305-SomethingAwesome/wiki

James Walsh
School of Electrical and Information Engineering
University of Sydney
Sydney, Australia
jwal7309@uni.sydney.edu.au

***Abstract*—Music genre classification is a task well suited to machine learning (ML). Classical ML, especially using compact feature representations of audio such as Mel-frequency cepstrum coefficients (MFCCs), produce highly accurate and efficient models for music genre classification. One might ask, how does the performance of ML fair when classifying sub-genres within a genre? It is expected that given the decreased variance between sub-genres, and the increased amount of expert knowledge required to distinguish the genres, that this task would be harder. Here, we demonstrate that classical ML performs well at distinguishing music sub-genres, and that the speed of training and accuracy of the Support Vector Machine approach makes it more attractive than convolutional neural networks (CNNs).**

***Index Terms*—music information extraction, neural audio learning, machine learning**

I. INTRODUCTION

Music genre classification is an interesting area of audio information extraction that shares a methodologies with speaker recognition, instrument identification, sound classifiers, etc. There are two broad frameworks to applying ML to such processes:

- **Classical ML with handcrafted features—**involves extracting a set of audio features (e.g. MFCCs) from a snippet of audio and training a classical ML model (e.g. support vector machine) to identify the audio clips from a set of given classes [1].
- **Deep learning with time-frequency representations—** involves computing the time-frequency representation of a snippet of audio and training a neural network to identify the audio clips from a set of given classes [2].

Deep learning performs better at tasks that require a high degree of generalisation, such as genre classification, and they are also useful if the most descriptive handcrafted feature cannot be determined, since a neural network can be trained to any arbitrary function using any set of features it learns.

Convolutional neural networks (CNNs) are naturally suited to learning from 2-dimensional data; however, the time-frequency representation of audio is not equivalent to an image. The most important consideration is that the x and y dimensions are not interchangeable: in a typical time-frequency representation, the x axis corresponds to time and the y axis corresponds to frequency. Audio will form features which are periodic in time (rhythm, beats) and periodic in frequency (harmonics, timbre), but these are independent physical processes and should be learnt separately (by separate convolutional filters for example). The beauty of a CNN approach is that the computer scientist does not need to know what features (shape, size, intensity) in a time-frequency representation correspond to a particular class, but can train a neural network to identify these features for them, in an efficient manner.

Here, both ML frameworks will be applied to the task of sub-genre (genres within a genre class) music classification. The code provided by [3] was used as a starting point and adapted for the case of two sub-genres (death metal and black metal) within metal. The original code was built for the GTZAN dataset which was collected without copyright permission for a 2002 music genre classification paper [1]. In addition to building the corresponding ML models, an analysis for classical ML is conducted to determine which of the handcrafted features are most important for distinguishing the two sub-genres.

II. DATA PREPARATION

Full song tracks were acquired with varying quality (album, EP, demo, split, live, etc.). The dataset also contained a variety of data file types, including .flac, .mp3, and .ape, which would also produce some variability in audio quality (however the quality difference is unlikely to be large enough to be detected by the ML models). The dataset originally contained 62 black metal songs and 229 death metal songs, however, this caused an issue which be presented in Section III. As a result, a second dataset was created with 206 black metal songs and 229 death metal songs. In addition, a smaller set was used for training the deep learning model due to performance issues, it contained 20 songs for each class.

(a) Death (death metal)

(b) Gorgoroth (black metal)

Fig. 1: Archetypal bands from each sub-genre class.

The two classes of sub-genre were selected based on the following criteria:

- **Death metal—**songs produced by death metal bands in the Florida death metal scene in the 1980s. Bands used in this class include: Death, Mantas, Morbid Angel, Obituary, Pestilence, Possessed and Slaughter. Figure 1a.
- **Black metal—**songs produced by black metal bands in the Norwegian black metal scene in the 1990s. Bands used in this class include: Darkthrone, Emperor, Enslaved, Gorgoroth, Immortal, Mayhem and Satyricon. Figure 1b.

Despite the strict boundary for the selection of these genres, the model would still be useful for a wide range of modern heavy metal bands that follow the sound of either one of these schools of metal religiously (which easily includes thousands of bands across the globe[1]).

It is also worth noting here that the audio content of music is not the only way to classify its genre automatically. A multi-modal approach using audio, album cover (visual) and album reviews (text) to classifying the GTZAN dataset has previously been demonstrated [4]. In the case of metal sub-genres, song lyrics could also be a useful distinguishing feature, since black metal bands tend to sing on topics of battle, nature and folklore, whereas death metal bands tend to sing on topics of violence and death. The song names, album name and band name could also be indicative of the genre.[2]

III. Classical Machine Learning

The classical machine learning framework starts with the extraction of audio features, a principle-component analysis to observe the presence of learnable dimensions and lastly the training and evaluation of several machine learning models. Whole songs were used for the classical ML. The features extracted were left unchanged from the original work by [3]. These features included:

- Spectral centroid
- Spectral roll-off
- Spectral flux
- Spectral contrast
- Spectral bandwidth
- Spectral flatness
- Zero-crossing rate
- Root-mean square
- Sample silence
- MFCCs 1-13
- Tempo

For each feature (except tempo) a set of 6 summary statistics was taken over time since these features were calculated for each time-frame of the audio sample. These summary statistics includes maximum, minimum, mean, standard deviation, kurtosis and skew. This produced 134 audio features overall.

These features were then fed into 6 different classical ML models. These included:

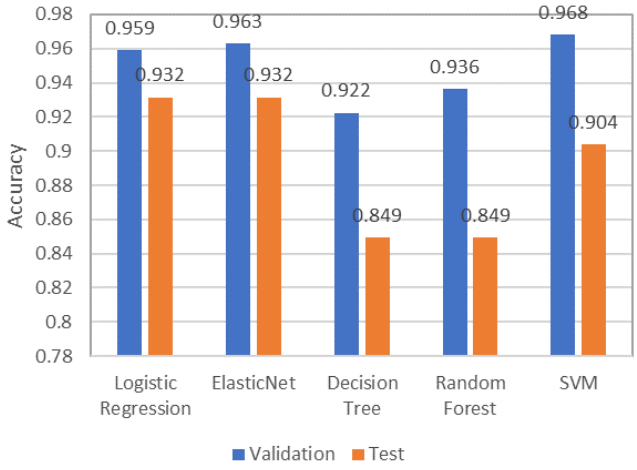
1) Logistic regression
2) ElasticNet
3) Decision tree
4) Random forest
5) Support vector machine (SVM)

The parameters for these models were grid-searched for an ideal configuration according to validation accuracy. The ideal configuration was then used to calculate the final accuracy against a test set. A summary of the grid searched parameters for each model from the above list is as follows:

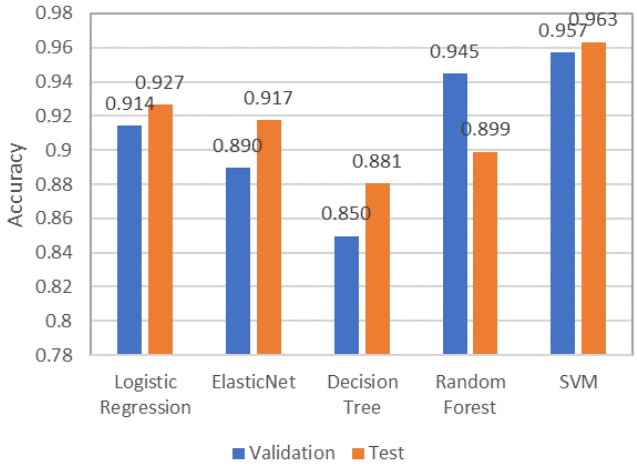
1) Penalty norms = {l1, l2}, regularisation parameter = {0.5, 1, 2, 5}, max iterations = {500}
2) Loss term = {log}, penalty = {elasticnet}, penalty norm l1 ratio = {0.15, 0.25, 0.5, 0.75}
3) Quality criterion = {gini, entropy}, split strategy at each node = {best, random}
4) Number of trees in forest = {100, 250, 500, 1000}, quality criterion = {gini, entropy}, max forest depth = {5, 7, None}

[1]This culture is known as metal elitism and those that do not do not follow the original sounds closely enough are often classified as "poseurs" by the elitists. If the technology in this paper gets into the elitists' hands, all hell could break loose.

[2]Although not many music genres follow such strict naming conventions as metal.



(a) First dataset (62 black, 229 death)



(b) Second dataset (206 black, 229 death)

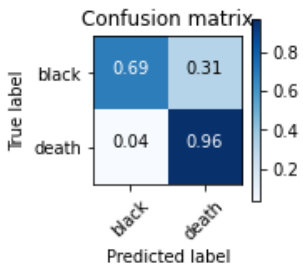
Fig. 2: Comparison of classical ML methods for 2-genre classification.

5) Regularisation parameter = {0.5, 1, 2, 5}, kernel type = {rbf, linear, sigmoid}.

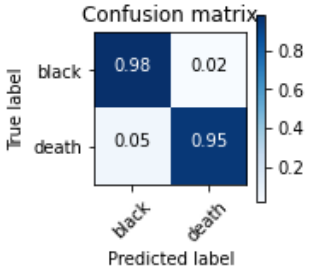
Before the sub-genre datasets were used, the original work was confirmed against the GTZAN dataset. These results showed that SVM produced a superior accuracy over the other techniques for 10-genre classification.

The results for the 2-subgenre classification are shown in Figure 2a and the confusion matrix for the corresponding SVM model is shown in Figure 3a. The original dataset was clearly unbalanced and this was reflected in the poor classification accuracy for the black metal class. To refine this, the second dataset was made balanced by the addition of new data for the black metal class. The result of this improvement is captured in Figure 2b and the confusion matrix for the corresponding SVM model is shown in Figure 3b.

According to the test accuracy, the logistic regression and ElasticNet models were superior at classifying the subgenres



(a) First dataset (62 black, 229 death)



(b) Second dataset (206 black, 229 death)

Fig. 3: Confusion matrices for SVM classifier.

for the first dataset. However, after balancing the dataset, the SVM model produced better performance. What is interesting about the training for the second dataset is the increase in accuracy going from the validation to the test dataset. Academic forums were less than useful for explaining this phenomena in my specific case, but the size of the dataset and train/test split probably had something to do with this phenomenon. As we can see from the last confusion matrix, classical machine learning can yield highly accurate classification models.

IV. Deep Learning

The deep learning training process required preparation of data, which included sampling 30 seconds of each song, and converting these snippets into spectrograms. The model used for this process was a custom CNN by [3]. The CNN architecture for 2-genre classification is shown in Table I. This architecture has over 2 million trainable parameters.

Training this network was incredibly slow. A few minutes into training the model on the GTZAN dataset, it could be projected that the training time would take 8 hours. To reduce this, NVIDIA's cuDNN was installed on the local system to allow the GPU to be used instead of the CPU. This process required some serious version control balancing between CUDA, cuDNN, python, tensorflow and even NumPy (i.e. a lot of trial and error). Once up and running, the network could be trained in a matter of hours rather than a whole day. To test the network, the GTZAN dataset used as the initial training set. The results in [3] were confirmed (see Figure 4a. For the GTZAN dataset, an accuracy gain of only a few percentage points was gained.

Re-configuring the network to a 2-genre CNN (Table I) and training for the second dataset yielded an inferior accuracy (65%) compared to the SVM (96%). The initial learning hyperparameters was chosen for this model, no additional tuning was performed.

V. Discussion

A. Data Balancing

The first dataset used for training the classical ML models was unbalanced. The impact of this was made apparent by analysing the individual classifications in the confusion matrix

TABLE I: CNN architecture used for 2-genre classification. Adapted from [3].

Layer	Shape	# parameters
InputLayer	None, 128, 130, 1	0
Conv2D	None, 128, 130, 16	160
Activation	None, 128, 130, 16	0
MaxPooling	None, 64, 65, 16	0
Dropout	None, 64, 65, 16	0
Conv2D	None, 64, 65, 32	4640
Activation	None, 64, 65, 32	0
MaxPooling	None, 32, 32, 32	0
Dropout	None, 32, 32, 32	0
Conv2D	None, 32, 32, 64	18496
Activation	None, 32, 32, 64	0
MaxPooling	None, 16, 16, 64	0
Dropout	None, 16, 16, 128	0
Conv2D	None, 16, 16, 128	73856
Activation	None, 16, 16, 128	0
MaxPooling	None, 8, 8, 128	0
Dropout	None, 8, 8, 128	0
Conv2D	None, 8, 8, 256	295168
Activation	None, 8, 8, 256	0
MaxPooling	None, 4, 4, 256	0
Dropout	None, 4, 4, 256	0
Flatten	None, 4096	0
Dropout	None, 4096	0
Dense	None, 512	2097664
Dropout	None, 512	0
Dense	None, 2	1026
Total params:		2,491,010

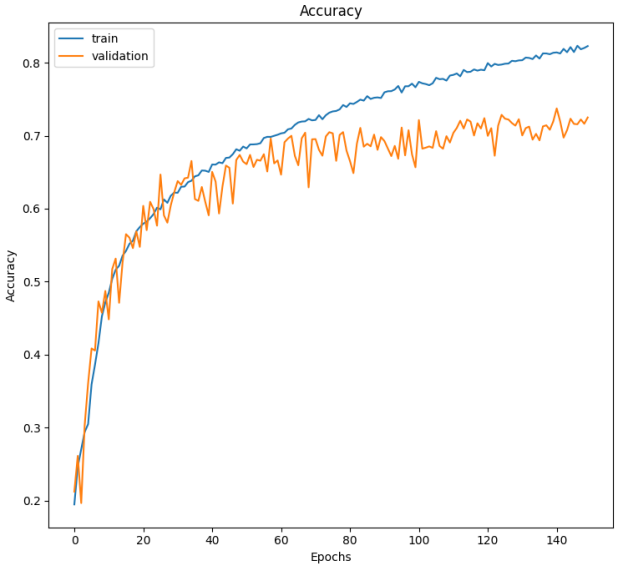
(Figure 3a). Black metal songs were being regularly misclassified as death metal songs. A good online discussion indicated that the likely culprit for this issue was an unbalanced dataset (in this case, containing significantly more death metal songs than black metal songs). The reason misclassifications occurred was obvious, the aim of the classifier is to maximise overall accuracy—if it exists in a reality where most of its situations involve death metal rather than black metal, it will be more likely to mistake black metal for death metal than vice versa.

The solution to this problem in some cases can be to 'balance' the dataset, which typically involves trimming the larger dataset. However, this is not desirable as the overall accuracy of the model will suffer. A better alternative is to increase the number of examples in the deficient class until the classes are of similar size. The effect of this alteration was an improvement in accuracy from 90% to 96%.

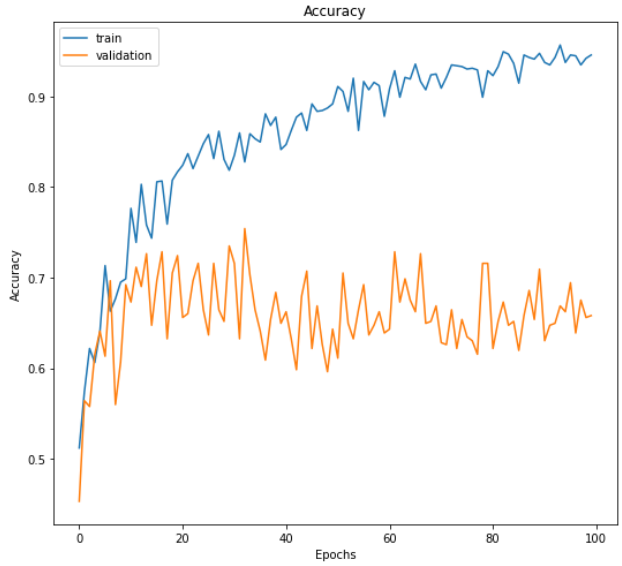
B. Audio Feature Analysis

After completing the classical ML training and testing, a question arises about what audio features were actually the most important for distinguishing the subgenres. This question is mostly academic however, since the models are fast to train and have quick inference times, so there is not really an argument to be made for making them more efficient.

To analyse what features were most important, a principle component analysis was conducted. The 134 audio features used to train the classical ML models were mapped to 32 new dimensions (these new dimensions were not used for any ML training). The resulting transformation from the audio features



(a) GTZAN dataset (10 genres)

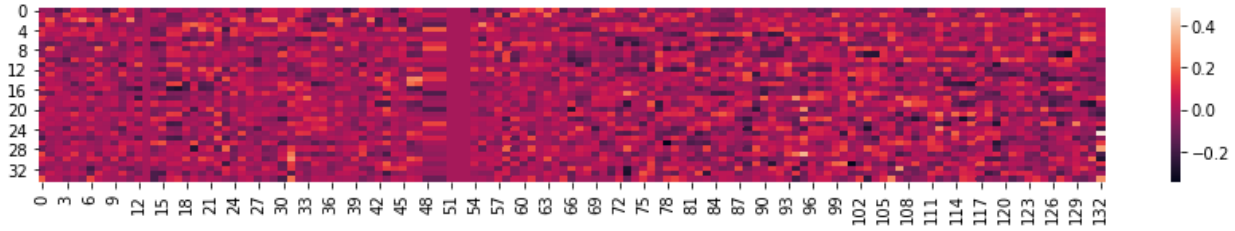


(b) Second dataset (206 black, 229 death)

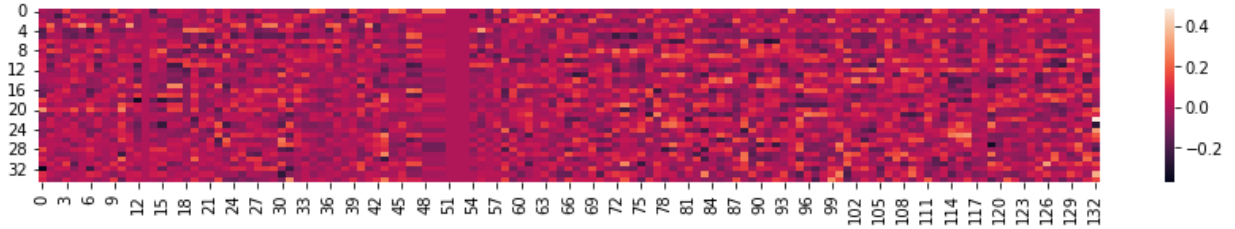
Fig. 4: CNN accuracy curves.

to the new dimensions is shown in Figure 5a. Although not immediately clear from these diagrams, the resulting transformation was analysed to find the most correlated dimensions between the two datasets, for each new dimension. For the first dataset, it was found that features 95 and 132 were the most correlated features with the new dimensions. These correspond to the 6th Mel-frequency cepstrum coefficient skew, and tempo. MFCC6 would correspond to a particular timbral feature in the cepstrum for either black metal or death metal songs. The tempo is, not surprisingly, also an important feature.

The balanced dataset has two prominent features appear in the new dimensions according to PCA. These are again, tempo, but instead of an MFCC being picked up, spectral flatness



(a) First dataset (62 black, 229 death)



(b) Second dataset (206 black, 229 death)

Fig. 5: Principle component analysis mapping from audio features to 32 new arbitrary dimensions.

minimum occurs equally as often as the most correlated feature across the new dimensions. Spectral flatness is an indicator of how tone-like or noise-like an audio signal is. Given the intentionally low-fi audio recording quality of black metal songs, they will likely have a lower spectral flatness than death metal. The balance of full album releases vs demos and other music release types in each class might also cause the correlation to be strong, here, as demos and live recordings will be noisier.

C. Machine Learning Performance

The testing of a diverse number of classical ML models, including a grid search for the best hyper-parameters, allowed a very high testing accuracy to be achieved (96%). A good test for these models would be to extend this 2-class problem to many subgenres within metal, or to include a non-metal control category. The classical ML models were also incredibly quick to train (less than a minute).

The deep learning model was too large (more than 2 million parameters) for any meaningful training to be performed on a local machine. And, it was certainly too large for a ablation testing to determine the effect of adding various data augmentation and architecture alterations. Still, a local GPU was used to speed up the deep learning by a factor of 2–3. For the second (balanced) dataset, the trained accuracy was also relatively poor (65%). In this case, the use of a deep learning model was certainly not an improvement. However, tinkering of the deep learning process was required to achieve optimal hyper parameters (just as a grid search was performed for the classical ML models) which could have yielded a superior accuracy in theory. It is also possible that the time-frequency features were downsampled to a degree that they could not be distinguished by the model (an input size of 128 by 130 is quite small).

For CNNs to be used for this task, a higher fidelity representation of the input data, with less parameters would be required. Given the low number of output classes, a reduction in parameters from the 10-genre case is reasonable. Other tricks to speed up training should also be considered.

VI. Conclusion

Most of the code for this project is based on prior work on the GTZAN dataset by [3]. Adapting this code to the 2-genre case and performing a feature analysis has revealed that SVMs have a number of superior qualities over CNNs. Firstly, audio features for SVMs are already well established, especially MFCCs and tempo. These provide a compact description of the audio to a sufficient fidelity to allow subgenres to be distinguished. This yields a high accuracy (96%) for the 2-genre case. The CNN, however, was incredibly large, and did not result in a superior accuracy once trained. Even with hyper-parameter optimisation, the consumed power to train the CNN was orders of magnitude greater than the SVM. Until obvious issues arise with the SVM, it is not reasonable to adopt a CNN model for subgenre classification. Future work should expand the classical ML models to many subgenres, and attempt to find a more efficient neural network for this task.

References

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002. DOI: 10.1109/TSA.2002.800560.

[2] A. Elbir and N. Aydin, "Music genre classification and music recommendation by using deep learning," *Electronics Letters*, vol. 56, no. 12, pp. 627–629, 2020.

[3] H. Guimarães, *Music genre classification on gtzan dataset using cnns*, https://github.com/Hguimaraes/gtzan.keras, 2020.

[4] S. Oramas, F. Barbieri, O. Nieto Caballero, and X. Serra, "Multimodal deep learning for music genre classification," *Transactions of the International Society for Music Information Retrieval. 2018; 1 (1): 4-21.*, 2018.