Music Genre Classification: A Review of Deep-Learning and Traditional Machine-Learning Approaches

Ndiatenda Ndou
School of Computer Science and Applied Mathematics
The University of the Witwatersrand,
Johannesburg, South Africa
ndiatenda.ndou@students.wits.ac.za

Ritesh Ajoodha
School of Computer Science and Applied Mathematics
The University of the Witwatersrand,
Johannesburg, South Africa
ritesh.ajoodha@wits.ac.za

Ashwini Jadhav
Faculty of Science
The University of the Witwatersrand,
Johannesburg, South Africa
ashwini.jadhav@wits.ac.za

***Abstract*—This research provides a comparative study of the genre classification performance of deep-learning and traditional machine-learning models. Furthermore, we investigate the performance of machine-learning models implemented on three-second duration features, to that of those implemented on thirty-seconds duration features.**

We present the categories of features utilized for automatic genre classification and implement Information Gain Ranking algorithm to determine the features most contributing to the correct classification of a music piece. Machine-learning models and Convolutional Neural Network (CNN) were then trained and tested on ten GTZAN dataset genres. The k-Nearest Neighbours (kNN) provided the best classification accuracy at 92.69% on three-seconds duration input features.

***Index Terms*—machine-learning, deep-learning, music genre classification, CNN, MFCC**

I. Introduction

Genre is one of the most common of factors distinguishing music pieces. Human responses to genre can be biased, however, broad genre definitions exist worldwide. Observing the shift of music to digital platforms, it becomes clear that the automating the task of music classification would be beneficial to all parties involved.

This research explores automatic music genre classification with the aim to show that machine-learning and deep-learning approaches can be utilized to classify music from only the audio signal itself, reducing search-time for music pieces within the large music databases that have emerged with digital music platforms. We compare the deep-learning approach to traditional machine-learning models, furthermore, we investigate the performance of machine-learning classifiers with three-seconds duration features, to those implemented with thirty-second duration features.

This study was conducted in three phases, namely, 'phase A', 'phase B, and 'phase C'. Phase 'A' and 'B' utilized six traditional machine learning classifiers to perform automatic music genre classification, however, the two phases experiment with different dimensions of input features. Phase 'C' provides the deep-learning approach and a machine-learning approach with more audio excerpts but shorter duration.

Related literature shows that digital music platform users are more likely to browse music by genre than artist similarity or recommendations, therefore, successful music genre classification will allow end-users to efficiently browse music within genre categories [10].

This paper continues with a brief background and review of related literature, followed by Section III with the procedures implemented. Section IV presents the automatic music genre classification results, and Section V concludes the paper.

A. Music Features

We present four categories of features utilized for music genre classification. The correct set of features needs to be selected in order to perform correct and informed classification.

1) Magnitude-based features: these features can be described as timbral features, describing the loudness, pitch, and compactness of music [1]. Timbral features of music are essential for humans to categorize and group together different sounds coming from a single source [19]. Some examples of features belonging to this category are spectral features which are embedded in the magnitude spectrum, a spectrum obtained from the absolute value of the Fourier transform of a music chord [1], this examples include: spectral rolloff, spectral flux, spectral centroid, spectral spread, spectral decrease, spectral slope, spectral flatness, and Mel Frequency Cepstral Coefficients (MFCCs).

2) Tempo-based features: these are the features that describe the rhythmic aspects of music such as the rhythm and tempo [1]. Examples of features belonging to this category are; Tempo measured in beats per minute (BPM), Energy (audio signal intensity) measured using the root mean square (RMS), and the Beat Histogram to visualize important properties of audio signals through evaluation of the histogram peak, amplitude, and other statistical measures.

3) Pitch-based features: features belonging to this category describe the pitch of a music piece, this is an essential building

Author(s)	Dataset	Model used	Classification Accuracy
Sturm (2013) [20]	GTZAN	Sparse Representation Classification	83.00%
Bergstra *et al*. (2006) [5]	GTZAN	ADABOOST	82.50%
Li *et al*. (2003) [12]	GTZAN	Support Vector Machines	78.50%
Lidy *et al*. (2007) [13]	GTZAN	Sequential Minimal Optimization	76.80%
Benetos and Kotropoulos (2008) [4]	GTZAN	Non-negative Tensor Factorization	75.00%
Choi *et al*. (2016) [6]	Navier Music	CNN	75.00%
Bahuleyan (2018) [3]	Audio Set	CNN	65.00%
Tzanetakis and Cook (2002) [21]	GTZAN	Gaussian Mixture Mode	61.00%

Table I: Various studies that have shown capability to perform genre classification. The columns list the author(s), dataset utilized, model implemented, and the classification accuracy attained.

block of the harmony, key, and melody of an audio piece [11]. This category is important to explore because pitch perception determines the frequency level of the underlying audio signal [11]. An example of a pitch-based feature is the 'zero crossing rate', which is the count of sign changes in consecutive blocks of an audio excerpt [1].

4) Chordal progression features: these group of features explore the pitch 'chroma', which is a twelve-dimensional vector with each dimension representing one pitch class [11]. Chroma can also be viewed as a distribution, where both the number of occurrences of a pitch and its energy can be deduced from the class values [11].

B. Related Work Results

A review of related literature reveals that several studies have displayed the capability to solve the problem of music genre classification. We present notable genre classification algorithms in Table I.

II. Methodology

This section outlines the method and set of experiments performed in the studies reviewed. The procedures carried out include further preprocessing of the dataset, feature selection, and a description of the machine learning classifiers employed.

A. Data Description

The dataset utilized for all three studies conducted was the GTZAN dataset [17]. This GTZAN dataset is an ensemble of 1000 excerpts of thirty second duration each. The 1000 music pieces are categorized into 10 genres with 100 music pieces for each genre.

For one of the studies conducted [9], the original dataset was duplicated and divided into 10 000 excerpts of three seconds duration each. This procedure provided more training data, however, the dataset did not have a consistent number of samples per genre, with some genres having slightly less or more than 1000 music pieces.

B. Feature Extraction and Representation

We have identified four categories of features that are generally hypothesized to contribute in the correct classification of music genre. Prior to the selection of these features for model implementation, vital preprocessing experiments have to be conducted to make the raw data suitable for the classification task. Feature extraction was performed in this study for two purposes:

- **Dimensionality reduction:** the raw data dimensions are usually too large, that is, an entire raw audio file may be too large to process efficiently. Related studies show that a feature set is used to present the data with fewer values, a single feature value may be produced for an entire audio signal [11].
- **Meaningful representation:** the raw audio fie contains all the information we can possibly extract and use, however, it is important that we represent the musical aspects in an interpretable manner by machines or humans [16].

The computation of features from a music excerpt usually gives rise to an n-dimensional vector, where the value of n is dependent on the length of the audio piece under analysis. If the value of n is large, we deal with high dimensional feature vectors which are inefficient to process, therefore, for a feature vector $V = (v_1, v_2, v_3 \ldots, v_n)$, the following feature representations were explored:

- **Mean:** the average value of feature V, computed as:

$$\mu V = \frac{1}{n}\sum_{i=1}^{n} v_i \tag{1}$$

- **Standard deviation:** a measure of the spread of values of feature V, computed as:

$$\sigma V = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(vi - \mu v)^2} \tag{2}$$

- **The Feature Histogram:** obtained by arranging the feature's local window intensities into bin ranges then taking a count of each bin's contents and modelling a

frequency histogram [1]. The normalized histogram bin values can be used for classification.

- **Mel Frequency Cepstral Coefficients (MFCC) Aggregation:** this representation takes the first n coefficients that form part of the short-term sound power-spectrum [8], [14]. Independently, each dimension is assessed, producing n coefficients per dimension. For this work, $n = 4$ was selected.
- **Area Moments:** this is an important concept in computer vision and image processing. This work follows a classic image moments implementation where 10 area moments were produced for image processing, treating each image as a two-dimensional vector $V(v_1 v_2)$, with v_1, v_2 indexing the underlying matrix [14]. We treat the extracted feature values from the audio signal as two-dimensional images and apply the moments algorithm in the work cited above.

C. Feature Selection

In this section, we present the various features utilized to perform automatic music genre classification in the studies this paper extends [1], [9], [16]. Feature selection is essential for the reduction of irrelevant and redundant data, the reduction of which may result in improved model learning accuracy, and reduced training time. Information gain ranking algorithm was utilized for comparison of the various features' contribution to a correct classification.

Features Maintained	**Rep.**	**Dim. 54**
Spectral Contrast	Mean	7
Spectral Rolloff	Mean + SD	2
Spectral Flux	Mean + SD	2
Spectral Crest	Mean + SD	2
Spectral Flatness	Mean + SD	2
Spectral Decrease	Mean + SD	2
Spectral Kurtosis	Mean + SD	2
Spectral Slope	Mean + SD	2
Spectral Skewness	Mean + SD	2
Spectral Centroid	Mean + SD	2
Spectral Spread	Mean + SD	2
Spectral Entropy	Mean + SD	1
Zero Crossing Rate	Mean + SD	2
Mel Frequency Cepstral Coefficients	Mean	17
Root Mean Square	Mean + SD	2
Beat Histogram	Sum + Mean + SD	3
Temporal Statistic Spread	Mean + SD	2
Features Eliminated	**Rep.**	**Dim. 51**
Spectral Crest Factor	Mean + SD	2
Spectral Tonal Power Ratio	Mean + SD	2
Mel Frequency Cepstral Coefficients	SD	35
Chroma	Mean	12

Table II: The set of features selected for training the employed machine-learning classifiers in 'phase B'. The upper shaded portion the table presents the features maintained after using Information Gain Ranking (IGR) algorithm, while the lower portion presents the eliminated features. The column heading acronym Rep. and Dim. are the feature representation and feature dimension respectively. A total of 54 features were selected in 'phase B', [16](sic).

The work presented in this paper was carried out over three studies referred to as phases, therefore, we continue by presenting three different sets of features selected. TableIII presents the features selected for 'phase A', Table IV presents the features for 'phase B', and TableII presents the feature set selected for 'phase C'.

Features Maintained	**Rep.**	**Dim. 459**
Spectral Flux	MFCC	4
Spectral Variability	MFCC	4
Compactness	Mean + SD	2
MFCCs	MFCC	52
Peak Centroid	Mean + SD	2
Peak Smoothness	SD	1
Complex Domain Onset Detection	Mean	1
Loudness + Sharpness and Spread	Mean	26
OBSI + Radio	Mean	17
Spectral Decrease	Mean	1
Spectral Flatness	Mean	20
Spectral Slope	Mean	1
Shape Statistic Spread	Mean	1
Spectral Centroid	MFCC	4
Spectral Rolloff	SD	1
Spectral Crest	Mean	19
Spectral Variation	Mean	1
Autocorrelation coefficients	Mean	49
Amplitude modulation	Mean	8
Zero Crossing + SF	MFCC	8
Envelope Statistic Spread	Mean	1
LPC and LSF	Mean	12
Root Mean Square	Mean + SD	2
Fraction of low energy	Mean	1
Beat Histogram	SD	171
Strength of Strongest Beat	Mean	1
Temporal Statistic Spread	Mean	1
Chroma	MFCC	48
Features Eliminated	**Rep.**	**Dim. 223**
Peak Fux	20-bin FH	20
Peak Smoothness	Mean	1
Shape Statistic Centroid and Skewness	Mean	1
Shape Statistic Kurtosis	Mean	2
Strongest Frequency of Centroid	MFCC	4
Spectral Rolloff	Mean	1
Strongest Frequency FFT	MFCC	4
Envelope Centroid, Skewness and Kurtosis	Mean	4
Beat Histogram	Mean	171
Strongest Beat	Mean + SD	2
Strength of Strongest Beat	SD	1
Fraction Low Energy	SD	1
Beat Sum	MFCC	4
Relative Difference Function	MFCC	4
Temporal Statistic Centroid	Mean	1
Temporal Statistic Skewness	Mean	1
Temporal Statistic Kurtosis	Mean	1

Table III: The set of features selected for training the employed machine-learning classifiers in 'phase A'. The upper shaded portion the table presents the features maintained after using Information Gain Ranking (IGR) algorithm, while the lower portion presents the eliminated features. The column heading acronym Rep. and Dim. are the feature representation and feature dimension respectively. A total of 459 features were selected in 'phase A', [1](sic).

D. Traditional Machine-Learning Models

For this research, we implemented the following off-the-shelf machine-learning models were implemented through

Features Maintained	**Rep.**	**Dim. 57**
Chroma	Mean + SD^2	2
Root Mean Square	Mean + SD^2	2
Spectral Centroid	Mean + SD^2	2
Spectral Bandwidth	Mean + SD^2	2
Spectral Rolloff	Mean + SD^2	2
Zero Crossing Rate	Mean + SD^2	2
Mel Frequency Cepstral Coefficients	Mean + SD^2	20
Harmony	Mean + SD^2	2
tempo	Mean	3
Features Eliminated	**Rep.**	**Dim. 51**
Spectral Crest Factor	Mean + SD	2
Spectral Tonal Power Ratio	Mean + SD	2
Chroma	Mean	12

Table IV: The set of features selected for training the employed machine learning classifiers in 'phase C'. The upper shaded portion the table presents the features maintained after using Information Gain Ranking (IGR) algorithm, while the lower portion presents the eliminated features. The column heading acronym Rep. and Dim. are the feature representation and feature dimension respectively. A total of 57 features were selected in 'phase C', [9](sic).

the Scikit Learn library [20]: k-Nearest Neighbours, Linear Logistic Regression, Multilayer Perceptron, Random Forests trees, and Support Vector Machines. The hyperparameters for each model are provided in Section III.

E. Deep-Learning Approach

The Convolutional Neural Network (CNN) architecture in this research was constructed using Keras [7]. The CNN built here has an input layer and five convolutional blocks, with each convolutional block consisting the following: The Convolutional Neural Network (CNN) architecture in this research was constructed using Keras [7]. The CNN built here has an input layer and five convolutional blocks, with each convolutional block consisting the following:

- Convolutional layer with mirrored padding, 1x1 stride, and 3x3 filter
- The rectified linear activation function (ReLu)
- Maximum pooling with 2x2 stride and window size
- Probability of 0.2 for dropout regularization

The last layer of the CNN outputs the probabilities of ten label classes through a fully-connected layer implementing the SoftMax activation function. The class that attains the highest probability becomes the classified label for a given input. The CNNs were trained on the spectrograms, twenty Mel Frequency Cepstral Coefficients (MFCC) of the three-or-thirty-seconds feature set, and the extracted spectrograms.

F. Evaluation Metrics

To reduce bias and produce credible results, we performed 3-reapeted 10-fold cross-validation prior to the models classifying the test dataset. We utilize the classification accuracy and training time to evaluate the performance of all employed models.

III. Results and Discussion

This section outlines the results obtained when we use the features provided in Table III, IV, and II to perform genre cassification on ten GTZAN genres.

A. Traditional Machine-Learning Models

The traditional machine-learning models implemented in this research were tested on the GTZAN dataset, the results are presented in Table V, VI, and VII.

Table V presents the results obtained during 'phase A' of this research. The Linear Logistic Regression provided the best classification accuracy at 81%, however, with the exception of the Multilayer Perceptron, we note that the logistic regression had the longest training time. We also note that the níve Bayes classifier was outperformed by all the trained classifiers.

Table VI presents the results obtained during 'phase B' of this research. The Support Vector Machines (SVM) provided the best classification accuracy at 80.80%. The SVM also attained a relatively low training time taking 0.3 seconds to build. Logistic Regression displayed notable accuracy again with a classification accuracy of 75.80%, however, failing to outperform the LogitBoost implementation followed in 'phase A'.

Table VII presents the various models' hyperparameters and performance during 'phase C' of this research. The k-Nearest Neighbour (kNN) provided the best classification accuracy at 92.69%, furthermore, the kNN attained the shortest training time of 78 milliseconds. We note that 'phase C' utilized a three-seconds duration feature set as opposed to the thirty-seconds duration dataset utilized in 'phase A' and 'B' of this research.

Figure 1 presents the confusion matrix attained when classifying music into ten GTZAN genres using Linear Logistic Regression models during 'phase A' of this research. We note the significant overlap between rock and country music, where ten country music excerpts were classified as rock music. Furthermore, rock music was the most misclassified genre, with rock music excerpts classified as blues, country, disco, metal, and pop.

Actual Genre \ Predicted Genre	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}
G_1	84	0	3	3	0	5	1	0	2	2
G_2	0	96	1	0	0	2	0	0	0	1
G_3	3	0	77	2	0	4	0	1	3	10
G_4	1	1	5	76	2	0	0	4	5	3
G_5	1	0	0	1	85	0	4	3	6	0
G_6	3	4	5	1	0	82	1	2	1	1
G_7	2	0	0	1	1	0	90	0	0	6
G_8	0	0	4	4	1	0	0	84	1	6
G_9	2	0	3	6	6	1	1	4	70	7
G_{10}	5	0	7	9	2	0	5	5	1	66

Figure 1: A confusion matrix obtained in the classification of music into ten GTZAN genres using Linear Logistic Regression during 'phase A' of this research. The row labels represent actual genre labels, while the column labels represent the predicted genre labels, where: G_1 = **Blues**, G_2 = **Classical**, G_3 = **Country**, G_4 = **Disco**, G_5 = **Hiphop**, G_6 = **Jazz**, G_7 = **Metal**, G_8 = **Pop**, G_9 = **Reggae**, and G_{10} = **Rock**.

Classifier	Accuracy	Training Time (s)	Hyperparameters
Linear Logistic Regression	81.00%	25.2500	maximum number of itterations for LogitBoost=500
Random Forests	75.70%	18.0800	number of trees = 1000
Support Vector Machines	75.40%	3.8200	kernel degree=3, tolerance=0.001, epsilon for loss function=0.1, used polynomial kernal: $\gamma u'v + coef_o$, and did not normalize
Multilayer Perceptron	75.20%	27.480	number of hidden layers= number of hidden classes, learning rate=0.3, training time=500 epochs, validation threshold=20
k-Nearest Neighbour	72.80%	0.0100	number of neighbours=1, using absolute error for cross-validation, and appied linear search algorithm
naíve Bayes	53.20%	0.5600	used normal distribution for numeric attributes and supervised discretization

Table V: Classification results and implementation details of each of the models employed during 'phase A' of this research. The columns list the accuracy, training time and hyperparameters related to the implementation of each classifier, [1](sic).

Classifier	Accuracy	Training Time (s)	Hyperparameters
Support Vector Machines	80.80%	0.3000	radial basis function kernel, tolerance=0.001, and regularization=0.17
Multilayer Perceptron	77.30%	0.2300	hidden layers=2, learning rate=0.02, activation=ReLu, max iterations=200, solver=adam, and tolerance=0.0001
Logistic Regression	75.80%	0.0800	solver=newton-cg and max itterations=500
Random Forests	72.40%	61.080	split function=gini, number of trees = 100, and max depth 100
k-Nearest Neighbour	69.70%	0.0110	k=7 with manhattan distance metric, weighting=distance
naíve Bayes	54.50%	0.0019	Gaussian naíve Bayes with smoothing

Table VI: Classification results and implementation details of each of the models employed during 'phase B' of this research. The columns list the accuracy, training time and hyperparameters related to the implementation of each classifier, [16](sic).

Classifier	Accuracy	Training Time (s)	Hyperparameters
k-Nearest Neighbours	92.69%	0.0780	nearest neighbours=1
Multilayer Perceptron	81.73%	60.620	activation=ReLu solver lbfgs
Random Forests	80.28%	52.890	number of trees=1000, max depth=10, $\alpha = e^{-5}$, and hidden layer sizes=(5000,10)
Support Vector Machines	74.72%	3.8720	decision function shape=ovo
Logistic Regression	67.52%	3.6720	penaty=12, multi class=multinomial

Table VII: Classification results and implementation details of each of the models employed during 'phase C' of this research. The columns list the accuracy, training time and hyperparameters related to the implementation of each classifier, [9](sic).

B. Deep-Learning Approach

In this subsection, we present the classification results of the Convolutional Neural Network (CNN) when trained on spectrograms, three-seconds features, and thirty-seconds features. Table VIII compares the accuracy attained and brief details of the implementation followed.

We see that the classification accuracy provided by the CNN is relatively lower than that provided by traditional machine learning models. The highest classification accuracy attained with the CNN is 72.40%, where the three-second feature

Classifier	Epochs	Test Loss	Accuracy
CNN (3-Sec Features)	50	0.873	72.40%
CNN (Spectrograms)	120	2.254	66.50%
CNN (30-Sec Features)	30	1.609	53.50%

Table VIII: Classification results attained from CNN implementation using the three-seconds duration, thirty-second duration, and spectrogram input feature sets, [9](sic).

set was utilized. The three-second feature set provides more training data which could explain the higher accuracy attained through it. Thirty-seconds duration features gave the CNN implementation the lowest accuracy at 53.50%. We note that the implementation of CNN with spectrograms attains higher accuracy as the number of epochs is increased, however, time and computational constraints did not allow increasing epochs further than 120 in this research.

IV. Conclusion

This work aimed at automatic music genre classification using deep-learning and traditional machine-learning models. A review of related literature revealed the capability of these classifiers and a benchmark to compare the work of this research. We note that the reliability of a learning model is dependent on the quality of its ground truth, therefore, it is essential to ensure the ground truth is well-founded and motivated.

This research was conducted in three phases, namely, 'phase A', 'phase B', and, 'phase C'. Each phase had a significance that aligns with the contribution made by this research to the current body of work. We present music genre classification via machine-learning and deep-learning approaches, furthermore, this work provides a comparison of the accuracy of machine-learning models and deep-learning models in completing the classification task.

After training several classifiers, the k-Nearest Neighbours (kNN) provided the best accuracy at 92.69%, furthermore, the kNN had a relatively low training time of 78 milliseconds. The higher accuracy attained by kNN relative to related literature can be explained by the three-seconds duration feature set which provides more training data. Backed by these findings, We conclude that three-second duration input features can provide better accuracy than thirty-second duration input features.

Further noteworthy performances were provided by the Linear Logistic Regression and Support Vector Machines (SVM), attaining 81.00% and 80.80% respectively. The Convolutional Neural Network (CNN) implementations followed in this research provided relatively low accuracy, with the most accurate CNN implementation attaining 72.40%.

This work has shown that automatic music genre classification is possible, furthermore, traditional machine learning models tend to outperform deep-learning approaches.

Acknowledgment

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant number: 121835).

References

[1] 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)*, 2015, pp. 66–71.

[2] R. Ajoodha, R. Klein, and M. Jakovljevic, "Using statistical models and evolutionary algorithms in algorithmic music composition," in *Encyclopedia of Information Science and Technology, Third Edition*. IGI Global, 2015, pp. 6050–6062.

[3] H. Bahuleyan, "Music genre classification using machine learning techniques," 2018.

[4] E. Benetos and C. Kotropoulos, "A tensor-based approach for automatic music genre classification," in *2008 16th European Signal Processing Conference*, 2008, pp. 1–4.

[5] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, "Aggregate features and adaboost for music classification," *Machine Learning*, vol. 65, pp. 473–484, 12 2006.

[6] K. Choi, G. Fazekas, and M. Sandler, "Explaining deep convolutional neural networks on music classification," 2016.

[7] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[8] I. Fujinaga, "Adaptive optical music recognition," Ph.D. dissertation, McGill University, CAN, 1997, aAINQ29937.

[9] D. S. Lau and R. Ajoodha, "Music genre classification: A comparative study between deep-learning and traditional machine learning approaches," in *Sixth International Congress on Information and Communication Technology (6th ICICT)*. Springer, 2021, pp. 1–8.

[10] J. H. Lee and J. S. Downie, "Survey of music information needs, uses, and seeking behaviours: preliminary findings." in *ISMIR*, vol. 2004. Citeseer, 2004, p. 5th.

[11] A. Lerch, *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*. Wiley Online Library, 10 2012.

[12] T. Li, M. Ogihara, and Q. Li, "A comparative study on content-based music genre classification," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, ser. SIGIR '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 282–289. [Online]. Available: https://doi.org/10.1145/860435.860487

[13] T. Lidy, A. Rauber, A. Pertusa, and J. Iñesta, "Combining audio and symbolic descriptors for music classification from audio," 2007.

[14] C. McKay, R. Fiebrink, D. McEnnis, B. Li, and I. Fujinaga, "Ace: A framework for optimizing music classification," in *ISMIR*, 2005.

[15] C. McKay and I. Fujinaga, "Musical genre classification: Is it worth pursuing and how can it be improved?" in *ISMIR*, 2006.

[16] T. Nkambule and R. Ajoodha, "Classification of music by genre using probabilistic graphical models and deep learning models," in *Sixth International Congress on Information and Communication Technology (6th ICICT)*. Springer, 2021, pp. 1–6.

[17] A. Olteanu. Gtzan dataset - music genre classification. [Online]. Available: https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Édouard Duchesnay, "Scikit-learn: Machine learning in python," 2018.

[19] B. L. Sturm, "Alexander lerch: An introduction to audio content analysis: Applications in signal processing and music informatics," *Computer Music Journal*, vol. 37, no. 4, pp. 90–91, 2013.

[20] B. L. Sturm, "On music genre classification via compressive sampling," in *2013 IEEE International Conference on Multimedia and Expo (ICME)*, 2013, pp. 1–6.

[21] 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.