Music Genre Classification Using Deep Learning

Navneet Parab[1], Shikta Das[2], Gunj Goda[3], Ameya Naik[4]

[1,2,3]Department of Electronics and Telecommunication Engineering, K. J. Somaiya College of Engineering, Mumbai - 400077, Maharashtra, India
[4]Professor, Department of Electronics and Telecommunication Engineering, K. J. Somaiya College of Engineering, Mumbai - 400077, Maharashtra, India

---***---

Abstract - *In this study, acoustic features of music have been extracted by using digital signal processing techniques and then using neural net, music genre classification have been done. We use the GTZAN database for data analysis and modelling. The dataset uses images of spectrograms generated from songs as the input into a neural net model to classify the songs into their respective musical genres. The objective of this research work is to implement supervised learning techniques like Artificial Neural Networks for classifying musical categories. Thus, comparing music classifiers accuracy for datasets of different nature. Also, the classification accuracy of genre classifiers on the varying number of modifiers and layers is analyzed.*

***Key Words*: Neural Networks, Signal Processing, Audio Processing, Genre Classification, Data Analysis and Modelling**

1.INTRODUCTION

Music plays a very important and impacting role in people's lives. Music brings like-minded people together and is the glue that binds groups and communities together. The widespread usage of the Internet has brought about significant changes in the music industry as well as leading to all kinds of change. Examples of these developments being the widespread usage of online music listening and sales platforms, control of music copyright, classification of music genre, and music recommendations. Today, with the advancement of music broadcast platforms, people can listen to music at any time and at any time and can reach millions of songs through various music listening platforms such as Spotify, SoundCloud , iTunes, Saavn etc.

The music industry has undergone major changes from its conventional form of existence and also in the way music has been created in the last few years. The ever-growing customer base has also increased the market for different styles of music and its consumption. Music not only brings the individuals together, but also provides insight for various cultures. Therefore, it is essential to identify and classify the music according to the corresponding genres to fulfil the needs of the people categorically. The manual ranking and categorisation of music is a repetitive and lengthy task wherein the duty lies with the listener.

Hence, Genre classification of music becomes an important task with many real world applications. As tens of thousands of songs are released every month on the internet, the need for accurate meta- data required for database management and search/storage purposes climbs in proportion.

Being able to instantly and accurately classify songs in any given playlist or library by genre is an important functionality for any music streaming/purchasing service, and the capability for statistical analysis that complete labelling of music provides is essentially limitless.

Genre classification is the way that can classify similar types of data into a single identity (based on its rhythm instrument played, or harmonic content) and give that identity as its name. Experts have been trying for a long time to understand sound & what differentiates one from another. How to visualize sound. What makes one tone different from another. Technology can be used to solve this and make classification easier or more efficient by using its signal properties.

A song can be represented in the form of an audio signal. This audio signal has different features such as frequency, spectral roll-off, root-mean-square (RMS) level, bandwidth, zero-crossing rate, spectral centroid, etc. We are going to analyse these features extracted from the GTZAN dataset and build different types of ensemble models to see how better we can differentiate one genre from another.

We are building a neural network which classifies music into its respective genre. This will automatically classify music based on different features and parameters, instead of manually classifying the genre, into its respective genres. We will compare the accuracies of this model and the pre-existing testing dataset, and draw the necessary conclusions. We will improve the model to reach a high accuracy so that the model identifies and classifies new music into its genre correctly.

2. LITERATURE REVIEW

Music genre labels are useful to organize songs, albums, and artists into broader groups that share similar musical characteristics. With the growth of online music databases and easy access to music content, people find it increasingly hard to manage the songs that they listen to. One of the ways to categorize and organize songs is based on the genre, which is identified by some defining characteristics of the music. Music genre classification has been a widely studied area of research since the early days of the Internet. Musical genres have no strict definitions and boundaries as they arise through a complex interaction between the public, marketing, historical, and cultural factors. This observation has led some researchers to suggest the definition of a new genre classification scheme purely for the purposes of music information retrieval [1][2]

According to Aucouturier and Pachet, 2003 [3] genre of music is possibly the best general information for the music content clarification. Being able to automatically classify and provide tags to the music present in a user's library, based on genre, would be beneficial for audio streaming services such as Spotify and iTunes. Tzanetakis and Cook (2002)[4] addressed this problem with supervised machine learning approaches such K-Nearest neighbour classifiers. More recent deep learning approaches take advantage of visual representations of the audio signal in form of spectrograms. These visual representations are used as input to Convolutional Neural Networks (CNNs)[5]

In Lidy and Rauber (2005),[6] the authors discuss the contribution of psycho-acoustic features for recognizing music genres, especially the importance of STFT taken on the Bark Scale (Zwicker and Fastl, 1999). Mel-frequency cepstral coefficients (MFCCs), spectral contrast and spectral roll-off were some of the features used by (Tzanetakis and Cook, 2002)[1]. A combination of visual and acoustic features are used to train SVM and AdaBoost classifiers in Nanni et al. (2016).

Li, Chan and Chun [7] recommend an alternate technique to concentrate musical examples included in sound music by methods for convolutional neural framework. Their tests demonstrated that convolution neural networks (CNN) has vigorous ability to catch supportive components from the deviations of musical examples with unimportant earlier information conveyed by them. They introduced a system to consequently extricate musical examples high-lights from sound music. Utilizing the CNN relocated from the picture data retrieval field their element extractors require insignificant earlier learning to develop. Their analyses demonstrated that CNN is a practical option for programmed highlight mining. Such revelation supported their hypothesis that the inherent attributes in the assortment of melodic data resemble those of picture data. Their CNN model is exceedingly versatile. They also presented their revelation of the perfect parameter set and best work on using CNN on sound music type arrangement.

With the recent success of deep neural networks, a number of studies apply these techniques to speech and other forms of audio data (AbdelHamid et al., 2014; Gemmeke et al., 2017)[8]. Representing audio in the time domain for input to neural networks is not very straight-forward because of the high sampling rate of audio signals. However, it has been addressed in Van Den Oord et al. (2016)[9] for audio generation tasks. A common alternative representation is the spectrogram of a signal which captures both time and frequency information. Spectrograms can be considered as images and used to train convolutional neural networks (CNNs) (Wyse, 2017)[10]. A CNN was developed to predict the music genre using the raw MFCC matrix as input . In Lidy and Schindler (2016)[11], a constant Q-transform (CQT) spectrogram was provided as input to the CNN to achieve the same task.

Current models have only been focused on CNN models which involve using images (Spectrogram graphs) as the input data for learning attributes of genres. CNN models require higher computational power to process images instead of numerical data. Research over models utilising only key peak features has not been done before extensively. We will be using only the key characteristic features of the audio file instead of the entire spectrogram for data processing and learning.

3. PROJECT DESIGN

This project has been designed into three distinct components viz.

1. Audio Processing (Time and Frequency Domain)
2. Feature Extraction
3. Neural Network Modelling

Audio Processing unit consists of analysing the time domain features such as Tempo, Amplitude, etc of the data along with frequency domain features of the data. This analysis helps us to identify the defining features of an audio wave and understand the required methodology for features which influence the distinction of any musical genre.

Feature Extraction unit consists of gathering all the required and characteristic features of audio and consolidating them all in an ordered format for further ad hoc data analysis and identifying any anomalies or discrepancies in the data. This ordered format (.csv format) has been used for predictive modelling aspect of the Neural Network.

The **Neural Network** unit consists of modelling and training the entire dataset of features extracted from the audio data for genre classification. A number of networks have been modelled and trained by changing and adjusting the network parameters with varying results for accuracy and loss.

3.1 PROBLEM STATEMENT

Classification of music based on genre follows certain rules and distinct characteristics for any musical piece to belong to a certain genre. By using Neural Networks for Supervised Learning, we will be training our neural network to classify music genres based on the parameters obtained through feature extraction of the audio files. The feature extraction will be done by Frequency and Time Domain Audio processing of the audio data.

3.2 BLOCK DIAGRAM

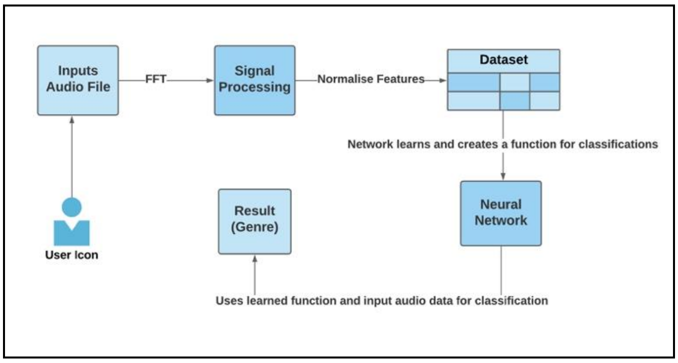

Fig. 1 – Overview of Project Model

3.3 OBJECTIVES

Companies recently have started the use of music classification, either to be able to place recommendations to their customers or simply as a product. Determination of these music genres is the first step for the required objective. Machine Learning techniques have proven to be very successful in extracting trends and patterns from a large set of data. The same principles have also been applied in Music Analysis.

We will be using Neural Networks to model different networks with varying adjustable parameters and observe the accuracy and loss of each model. By comparing these results, we will be able to select the best modelled network with optimum parameters, which will deliver best accuracy.

3.4 AUDIO ANALYSIS (TIME & FREQUENCY DOMAIN)

Sound is typically represented in the form of an audio signal having parameters such as frequency, bandwidth, decibel, etc. A typical audio signal can be expressed in the time domain as a function of Amplitude and Time.

Time Domain - A time domain analysis is the analysis of physical signals, mathematical functions, or time series of any data, with reference to time. Also, in the time domain, the signal or function's value is found for all real numbers at numerous separate instances in the case of discrete time or in the case of continuous time.

A time domain graph can show how a signal changes with respect to time, whereas a frequency domain graph will show what proportion of the signal lies within each given waveband over a range of frequencies.

Frequency Domain - Frequency domain is the analysis of signals or mathematical functions, with reference to frequency. As mentioned earlier, a time domain graph shows the changes in a signal over a period of time, and frequency domain shows what proportion of the signal exists within a given waveband over a range of frequencies. Also, a frequency domain representation can include information on the phase shift that must be applied to each sinusoid to be able to recombine the frequency components to recover the original time signal.

3.5 SUPERVISED LEARNING

Supervised learning is an approach for creating artificial intelligence, wherein a computer algorithm is trained on input data that has been labelled for a particular output. The machine learning model is trained until it can detect the underlying trends and relationships between the output labels and input data, enabling it to obtain accurate labelling results when presented with unseen new data.

Supervised learning is good at classification and regression problems, such as determining what category a news article belongs to or predicting the volume of sales for a given future date. In supervised learning, the aim is to make sense of the data within the context of a specific condition.

In neural network algorithms, the supervised learning process is improved by constantly measuring the resulting outputs of the model and fine-tuning the system to get closer to its target accuracy. The level of accuracy obtained depends on two things: the available labelled data and the algorithm that is used.

In addition:

1. Training data must be balanced and cleaned. Garbage or duplicate data will skew the AI's understanding - hence, we must be careful with the data the model is trained on.
2. The diversity of the data determines how well the AI will perform when presented with new cases; if there are not enough samples in the training data set, the model will falter and fail to yield reliable answers.
3. High accuracy, is not necessarily a good indication: it could mean the model is being overfitted - i.e., it is being over tuned to its particular training data set. Such a dataset might perform well in test scenarios but fail terribly when presented with real-world

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 10 | Oct 2021 www.irjet.net p-ISSN: 2395-0072

data. To avoid overfitting, it is crucial that the test data is different from the training data. This ensures that the model is not drawing answers from its previous experience, but instead that the model's inference is being generalized.

4. IMPLEMENTATION
4.1 DATASET

We analyzed the features extracted from the GTZAN dataset and built different types of ensemble models to see how better we can differentiate one genre from another.

Our Datasets contains 10 genres:

1. Blues
2. Classical
3. Country
4. Disco
5. Hiphop
6. Jazz
7. Metal
8. Pop
9. Reggae
10. Rock

We have 60 columns in our original dataset and we work with 10 of these for making our model.

1. Name
2. length of the file
3. chroma shift mean
4. chroma shift variance
5. rms mean
6. rms variance
7. spectral centroid mean
8. spectral centroid variance
9. spectral bandwidth mean
10. spectral bandwidth variance

4.2 STUDY OF THE SPECTROGRAM

Wave plot of audio file with respect to Amplitude - Since Time Domain analysis is not feasible due to lack of defining features, we implement Frequency Domain analysis for prominent feature detection.

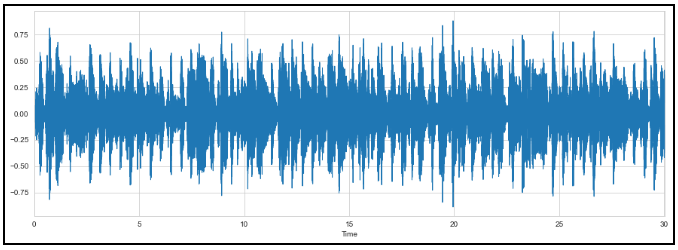

Fig. 2 – Amplitude v/s Time plot of Audio file

Spectrogram with Amplitude - We observe few defining peaks but their visibility is almost negligible. To enhance these features, we use a log scale (decibel) for better visualisation and analysis.

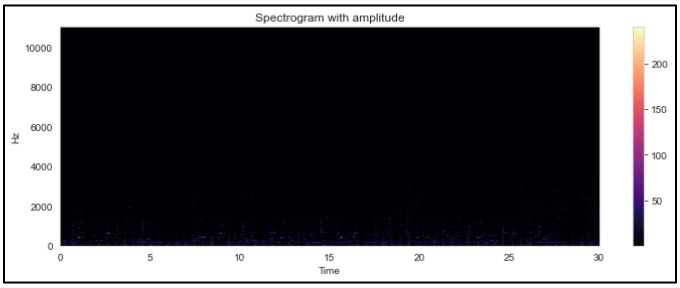

Fig. 3 – Frequency v/s Time plot of Audio file

Hence we use Spectrogram for further analysis.

Spectrogram with Decibel-Log - The decibel-log scale is highly efficient in displaying prominent peaks of frequency range. But frequency observation for the human ear is perceived in variations of the range itself.

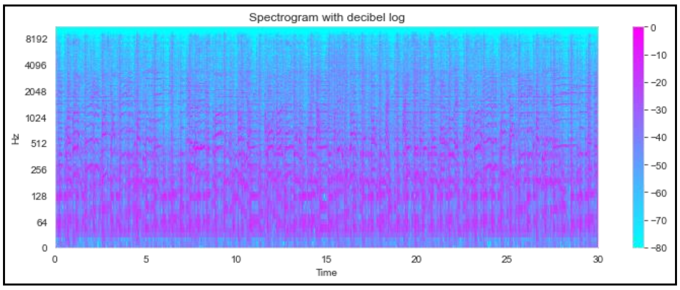

Fig. 4 – Frequency v/s Time plot of Audio (decibel – log)

Hence we use Mel Spectrogram for further analysis.

Mel Spectrogram - The Mel Spectrogram depicts the range of frequencies experienced by the human ear. These ranges are beneficial for us to normalize the features which will be fed into the Neural Network since Genre Classification works on Human interpretation of sound.

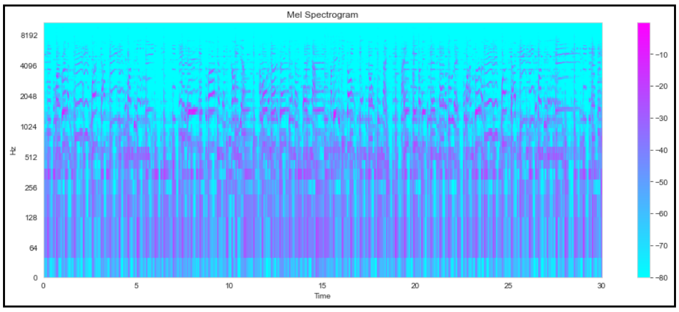

Fig. 5 – Mel Spectrogram of Audio

Chroma Features - In Chroma Features, we scale the pitch class for each instance of the audio, to further create features for better learning for the Neural Network. Pitch class is an essential parameter for genre classification of any song.

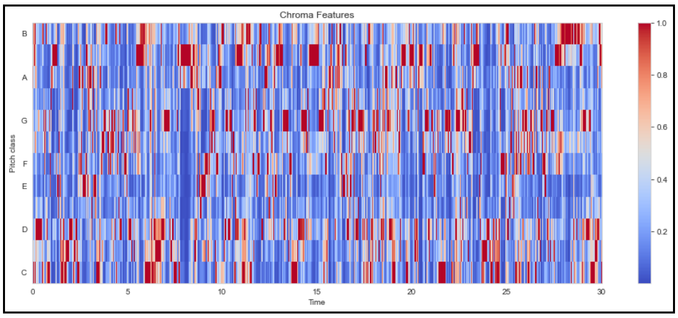

Fig. 6 – Chrome Features of Audio

4.3 DATA PREPARATION

A. Missing Value Treatment:

Before using a dataset for training and testing a Neural Network, we have to make sure that the dataset is void of any null values. The problem of missing data poses a difficulty to the analysis and decision making processes which depend on this data, requiring methods of estimation which are accurate and efficient.

B. Outlier Treatment:

Outliers in any dataset have to be dealt with since these values can skew the final results of the neural network. Outliers are usually eliminated from the dataset to maintain consistency in the data. In our case, the dataset was absent of any outliers, hence further processing was not required.

C. Define Dummy Variables for Categorical Variables:

Any Neural Network can only accept numerical data for processing and learning (Activation Functions and Loss Functions are all formulated for numerical data). In case of categorical data with no numerical value, the network cannot be trained. To overcome this issue, categorical variables are assigned numerical tags called Dummy Variables. These dummy variables can be used as input for the Neural Network and still maintain the categorical nature of the variables. In our case, categorical variables weren't present, thus dummy variables were not required.

D. Encode Genre Label:

As discussed above, Categorical data cannot be used as input for a Neural Network due to its non-numerical nature. Similarly, output of any Neural Network is also of numeric nature. To identify genres (Categorical), we have tagged all the genres to an integer, which the neural network can understand to validate the result.

4.4 TRAINING, TEST AND VALIDATION SPLIT

For training any machine learning or deep learning algorithm, we need to provide some data for it to learn from its parameter and formulate a model. This is performed by splitting our dataset into three parts viz. Training Set, Dev (Validation) Set and Testing Set.

A. Training Set:

A training dataset is a dataset of examples used during the learning process and is used to fit the parameters. A supervised learning algorithm looks at the training dataset to determine, or learn, the optimal combinations of variables that will generate a good predictive model. The goal is to produce a trained (fitted) model that generalises well to new, unknown data. The fitted model is evaluated using "new" examples from the held-out datasets (validation and test datasets) to estimate the model's accuracy in classifying new data. To reduce the risk of issues such as overfitting, the examples in the validation and test datasets should not be used to train the model.

B. Validation Set:

A validation dataset is a dataset of examples used to tune the hyper parameters (i.e. the architecture) of a classifier. In order to avoid overfitting, when any classification parameter needs to be adjusted, it is necessary to have a validation dataset in addition to the training and test datasets. The validation dataset functions as a hybrid: it is training data used for testing, but neither as part of the low-level training nor as part of the final testing.

C. Testing Set:

A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training dataset. If a model fit to the training dataset also fits the test dataset well, minimal overfitting has taken place (see figure below). A better fitting of the training dataset as opposed to the test dataset usually points to overfitting. A test set is therefore a set of examples used only to assess the performance (i.e. generalisation) of a fully specified classifier. To do this, the final model is used to predict classifications of examples in the test set. Those predictions are compared to the examples' true classifications to assess the model's accuracy.

D. Scale Feature:

Feature scaling is a method used to normalise the range of independent variables or features of data. We use StandardScaler for our dataset so that after transformation, the mean of the dataset is zero and the standard deviation is one.

4.5 MODEL BUILDING

Model building comprises designing a Neural Network with specific layers according to the needs of the data. For our model, we tried testing for Four models of a Neural Network with varying parameters. The models with their parameters are as follows:

Table -1: Model 1

Layer Order	Layer Type	Activation Function	Shape
1	Dense	ReLU	256
2	Dense	ReLU	128
3	Dense	ReLU	64
4	Dense	Softmax	10

Epochs = 70
Optimizer = adam

Table -2: Model 2

Layer Order	Layer Type	Activation Function	Shape
1	Dense	ReLU	512
2	Dropout	-	512
3	Dense	ReLU	256
4	Dropout	-	256
5	Dense	ReLU	128
6	Dropout	-	128
7	Dense	ReLU	64
8	Dropout	-	64
9	Dense	Softmax	10

Epochs = 100
Optimizer = adam

Table -3: Model 3

Layer Order	Layer Type	Activation Function	Shape
1	Dense	ReLU	512
2	Dropout	-	512
3	Dense	ReLU	256
4	Dropout	-	256
5	Dense	ReLU	128
6	Dropout	-	128
7	Dense	ReLU	64
8	Dropout	-	64
9	Dense	Softmax	10

Epochs = 500
Optimizer = sgd

Table -4: Model 4

Layer Order	Layer Type	Activation Function	Shape
1	Dense	ReLU	1024
2	Dropout	-	1024
3	Dense	ReLU	512
4	Dropout	-	512
5	Dense	ReLU	256
6	Dropout	-	256
7	Dense	ReLU	128
8	Dropout	-	128
9	Dense	ReLU	64
10	Dropout	-	64
11	Dense	Softmax	10

Epochs = 500
Optimizer = rmsprop

Layer Types:

1. Dense layer:
Dense Layer is the regular deeply connected neural network layer. It is the most common and frequently used layer. Each neuron receives input from all the neurons in the previous layer, thus densely connected. The layer has a weight matrix **W**, a bias vector **b**, and the activations of previous layer **a**.

2. Dropout Layer:
Dropout is a technique used to tackle Overfitting. The Dropout method in keras.layers module takes in a float between 0 and 1, which is the fraction of the neurons to drop. Dropout consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent overfitting.

Activation Function:

1. ReLU:
The rectified linear activation function or ReLU for short is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero. It has become the default activation function for many types of neural networks because a model that uses it is easier to train and often achieves better performance. Equation (1) is the ReLU equation.

$$f(x) = \max(x, 0) \tag{1}$$

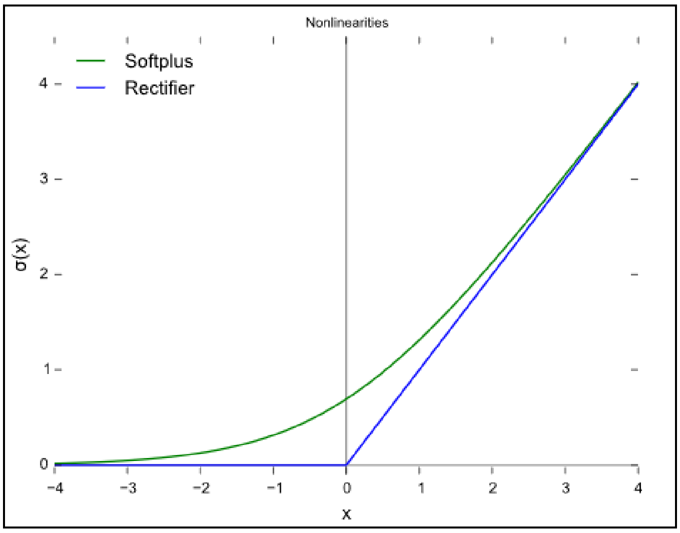

Fig. 7 – ReLU Activation Function

2. Softmax:

Softmax is a mathematical function that converts a vector of numbers into a vector of probabilities, where the probabilities of each value are proportional to the relative scale of each value in the vector. Equation (2) is the Softmax equation for K number of classes.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad for\ j = 1, \ldots, K \tag{2}$$

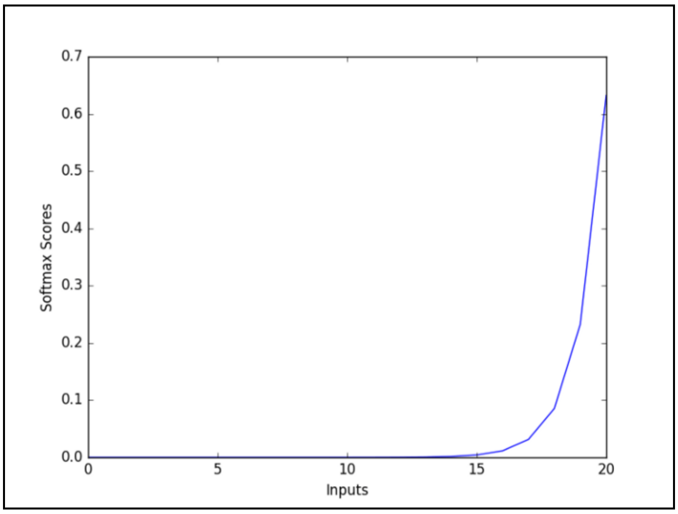

Fig. 8 – Softmax Activation Function

4.5 MODEL EVALUATION

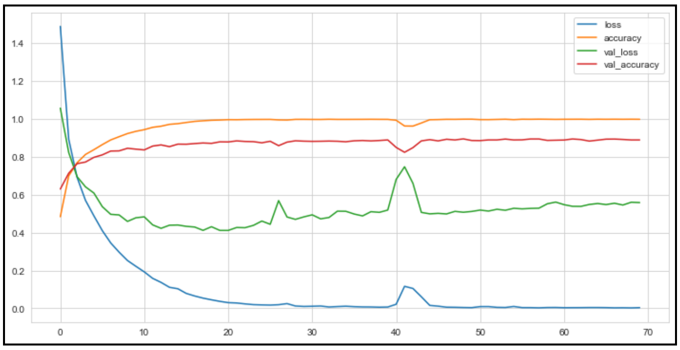

Fig. 9 – Model 1 Performance Metrics

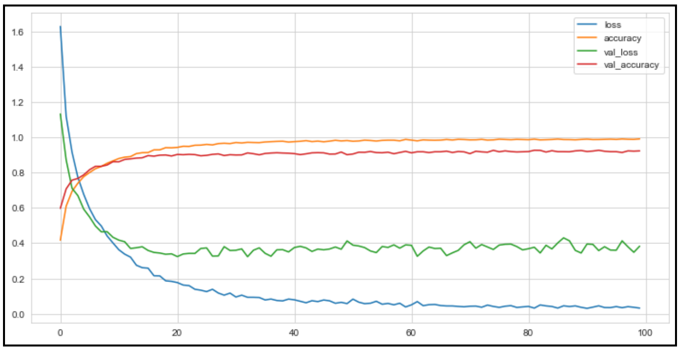

Fig. 10 – Model 2 Performance Metrics

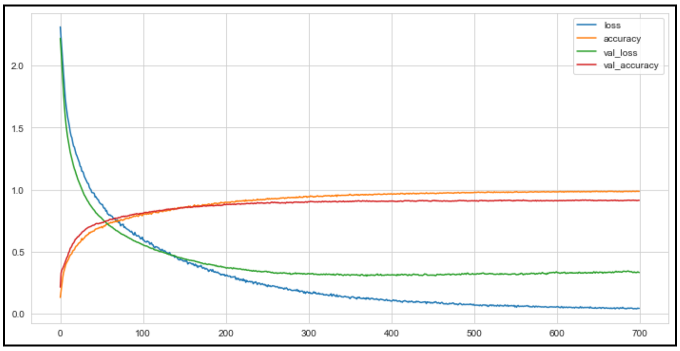

Fig. 11 – Model 3 Performance Metrics

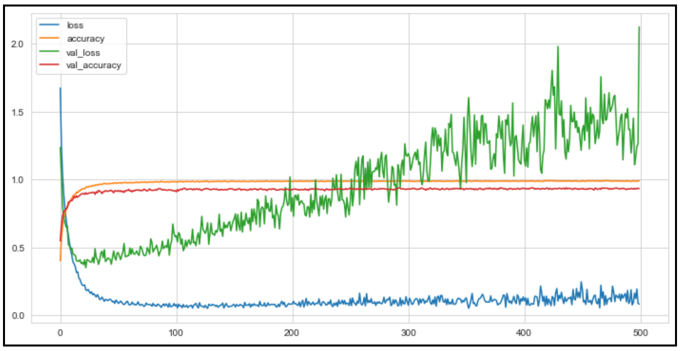

Fig. 12 – Model 4 Performance Metrics

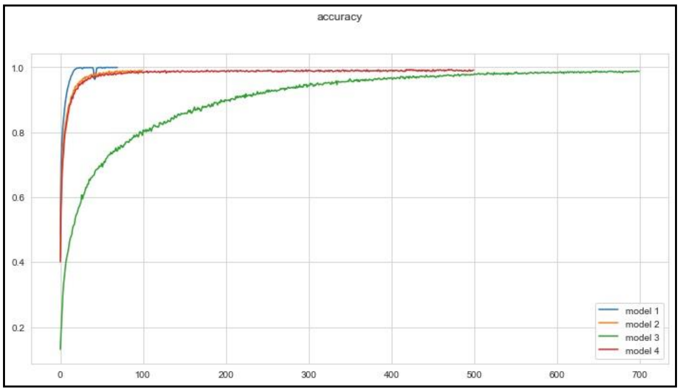

Fig. 13 – Accuracy (All models)

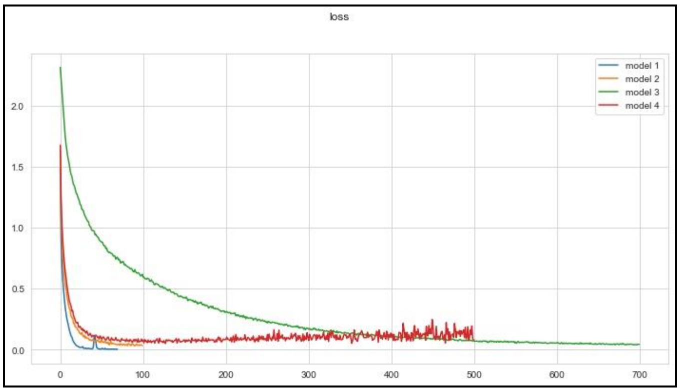

Fig. 14 – Loss (All models)

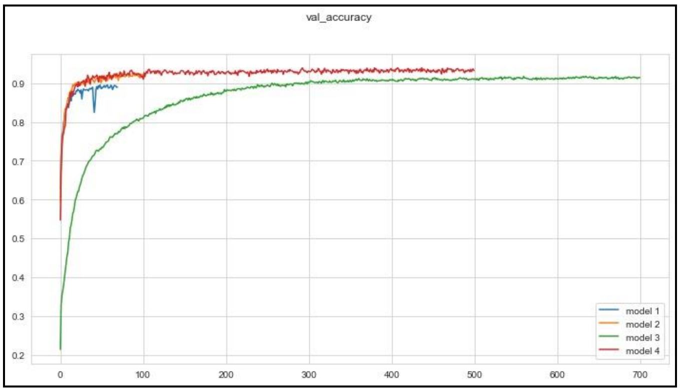

Fig. 15 – Validation Accuracy (All models)

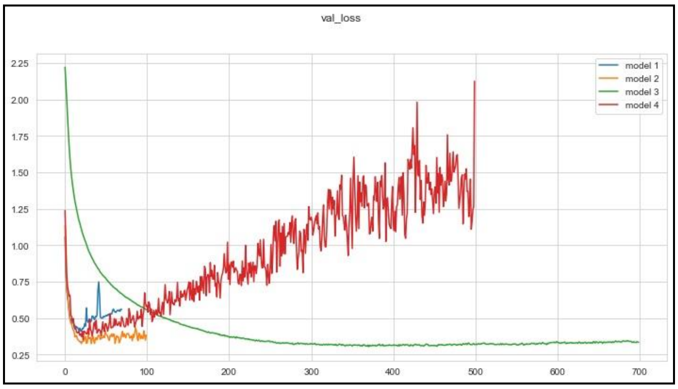

Fig. 16 – Validation Loss (All models)

Table -5: Model Evaluation using Loss and Accuracy

Model Number	Test Loss	Test Accuracy
1	0.5499	0.8950
2	0.3923	0.9166
3	0.3480	0.9176
4	1.6516	0.9264

Table -6: Model Comparison with existing models

Model	Test Accuracy
VGG-16 CNN Transfer Learning	0.63
VGG-16 CNN Fine Tuning	0.64
Feed-forward NN baseline	0.43
ANN – (Dropout Model)	**0.91**

Out of the four models that we modelled and tested, the 3rd model (Accuracy = 91.76%) gave the most reliable and consistent results with a robust validation accuracy and validation loss vs epochs response. Although the 4th model gives better accuracy (92.64%), its validation accuracy and validation loss vs epochs response indicates overfitting over the training data.

5. CONCLUSION

This study compared deep learning techniques in their suitedness to the task of music genre classification. In this project, In this music genre classification project, we have developed a classifier on audio files to predict its genre. We worked through this project on GTZAN music genre classification dataset and built different types of ensemble models and saw how better we can differentiate one genre from another. We used 3 distinct components i.e. time and frequency domain ,feature extraction and neural network modelling. We compared the models based on the kind of input we were receiving. Deep Learning methodology proves to be more accurate with around 90% accuracy in classifying the genres whereas previous models based on Spectrogram models were found to be around 65-70% accurate.

6. FUTURE SCOPE

Future studies can identify ways to pre-process this noisy data before feeding it into a machine learning model, in order to achieve better performance. Key signature and melody could also be worthy of greater focus. Finally, more work on curating high-quality datasets could be performed. This could involve working with and reducing the size of pre-existing datasets, combining existing datasets together, or the collection of new tracks that are not currently in the public domain. Overall, music genre classification remains an interesting and worthwhile challenge for both academic institutions and businesses alike, and there is plenty of room for further study and analysis.

REFERENCES

[1] George Tzanetakis and Perry Cook. 2002. Musical genre classification of audio signals. IEEE Transactions on speech and audio processing 10(5):293– 302.

[2] F. Pachet and D. Cazaly, “A classification of musical genre,” in Proc. RIAO Content-Based Multimedia Information Access Conf., Paris, France, Mar. 2000.

[3] Aucouturier, J., & Pachet, F. (2003). Representing Musical Genre: A State of the Art. Journal of New Music Research, 32(February 2015), 83–93.

[4] Serwach, M., & Stasiak, B. (2016). GA-based parameterization and feature selection for automatic music genre recognition. In Proceedings of 2016 17th International Conference Computational Problems of Electrical Engineering, CPEE 2016.

[5] Keunwoo Choi, George Fazekas, and Mark Sandler. Automatic tagging using deep convolutional neural networks. ISMIR, 2016

[6] Thomas Lidy and Andreas Rauber. 2005. Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In IZMIR pages 34–41.

[7] Li, T. L. H., Chan, A. B., & Chun, A. H. W. (2010). Automatic Musical Pattern Feature Extraction Using Convolutional Neural Network. (IJACSA) of the International MultiConference of Engineers and Computer Scientists (IMECS 2010), I, 546–550.

[8] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. 2014. Convolutional neural networks for speech recognition. IEEE/ACM Transactions on audio, speech, and language processing 22(10):1533–1545.

[9] Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio.

[10] Lonce Wyse. 2017. Audio spectrogram representations for processing with convolutional neural networks.

[11] Thomas Lidy and Alexander Schindler. 2016. Parallel convolutional neural networks for music genre and mood classification. MIREX2016 .

[12] Sturm, B. L. (2012). A survey of evaluation in music genre recognition. In International Workshop on Adaptive Multimedia Retrieval, 29–66

[13] Keunwoo Choi, George Fazekas, and Mark Sandler. Automatic tagging using deep convolutional neural networks. ISMIR, 2016

[14] Yali Amit and Donald Geman. 1997. Shape quantization and recognition with randomized trees. Neural computation 9(7):1545–1588