Proceedings of the Sixth International Conference on Computing Methodologies and Communication (ICCMC 2022)
IEEE Xplore Part Number: CFP22K25-ART; ISBN: 978-1-6654-1028-1

2022 6th International Conference on Computing Methodologies and Communication (ICCMC) | 978-1-6654-1028-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCMC53470.2022.9753953

Music Genre Classification using Deep Learning

Mitt Shah
Computer Science and Engineering
Nirma University
Ahmedabad, India
18bce121@nirmauni.ac.in

Nandit Pujara
Computer Science and Engineering
Nirma University
Ahmedabad, India
18bce130@nirmauni.ac.in

Kaushil Mangaroliya
Computer Science and Engineering
Nirma University
Ahmedabad, India
18bce091@nirmauni.ac.in

Lata Gohil
Computer Science and Engineering
Nirma University)
Ahmedabad, India
lata.gohil@nirmauni.ac.in

Tarjni Vyas
Computer Science and Engineering
Nirma University
Ahmedabad, India
tarjni.vyas@nirmauni.ac.in

Sheshang Degadwala
Computer Engineering
Sigma Institute of Engineering
Vadodara, India
Sheshang13@gmail.com

***Abstract*—One of the fascinating subjects in the area of Music Information Retrieval (MIR) is the classification of music as it is played into different genres. Machine learning is used in this analysis. To predict the genre of the audio signal, models such as Support Vector Machines (SVM), Random Forests, XGB (eXtreme Gradient Boosting), and Convolutional Neural Networks (CNN) are used. The GTZAN dataset was used for model training and testing. Machine learning and deep learning models each had their own set of features. A comparison analysis is proposed between these models, demonstrating that CNN outperforms machine learning models.**

Keywords—Convolutional Neural Networks, Music Information Retrieval, GTZAN, SVM, XGB

I. INTRODUCTION

To have better recommendations, a personalized music experience necessitates categorizing music according to the user's preferences. As defined in [1] music can be classified into various genres based on certain characteristics such as tempo, harmony, and instruments used. The GTZAN dataset was used to train and test machine learning algorithms for categorizing music into different genres. Features are extracted from audio signals and use them to create new features in the first part of the analysis. Support Vector Machines, Decision Trees, and XGBoost are examples of traditional Machine Learning models that have been trained. In the second section, the audio signal's Mel-spectrogram is fed to a convolutional neural network, or CNN. The authors of [2] have used custom CNNs, but it is found that enhancing the custom CNN architecture improves test accuracy. The following is the paper's section structure. Section II is devoted to a review of emerging methods for music genre classification. Dataset and pre-processing are discussed in Section III. Section IV contains the templates and instructions about how to apply them. The findings are discussed in Section V and conclusion in Section VI.

II. RELATED WORK

First, Music genre classification is one of the most intriguing subjects in the field of music information retrieval, and it has been researched extensively. The authors previously used supervised machine learning techniques such as the k-nearest neighbour and Gaussian Mixture model in [1]. Hidden Markov models are widely used in speech recognition tasks, and they have been investigated for music genre classification in [3]. The authors of [4]iinvestigated Support Vector Machines for music genre classification using various metrics. Reference [1] made use of mel-frequency cepstral coefficients (MFCCs), spectral contrast, and spectral roll-off. In [5] AdaBoost and SVM classifiers are also trained using a combination of acoustic and visual features. Deep neural networks are used in many research papers, such as [6] and [7], to analyze speech and other forms of audio data.

Convolutional Neural Networks have gained a lot of traction in the last decade due to their ability to extract highly specific features from image images. Spectrograms may be used to describe features in both the time and frequency domains. Convolutional Neural Networks (CNNs) can be trained using spectrogram images [7]. Reference [8] used the MFCC matrix as an input to CNNs to predict the music genre. Reference [1] attempts to classify music genres by using CNN to extract MFCC Spectrogram features and then identifying them using different classifiers such as J48, Attribute Selected Classifier, and others. This study only used CNN to extract specific features, not to categorize music genres directly.

Researchers in [9] compared the parameters and variables that have an effect on the classification process' efficiency, ranging from the architecture used to the feature representation used to the architecture used. Mel Spectrogram images were found to reflect audio characteristics better than other extracted features in this report. Researchers in [1] use Audio Signal input and RNN architecture directly, while research [10] aims to predict music similarity, but it is not based on music genre prediction, but it is relevant to research [10] because both use RNN. Reference [7]examines the representation of features needed for generative networks, as well as the issues that "lossy" features, such as handcrafted features, computer discovered features, and MFCCs, cannot be used. The emphasis is on the use of spectrograms in the creation of music.

The importance of different layers of Convolutional Neural Networks is explained in [11]. While this study has little to do with music genre classification, it does aid in a better understanding and design of CNN. Researchers in [12] used the Residual Attention Network (RAN) to identify music genres with 71.7% accuracy. They used a dataset from Vietnam that included ten different music genres. Deep

Belief Networks were used to derive features from Discrete Fourier Transforms of audio in [13]. They used a non-linear Support Vector Machine classifier with the qualified neural network activations as inputs. On the GTZAN dataset, they classified music genres with an accuracy of 84.3%.

Since audio is a time-dependent function, researchers used a hybrid model of Long Short-Term Memory (LSTM) and Support Vector Machines (SVM) on the GTZAN dataset to achieve an accuracy of 89% [14]. In music segmentation [15] and speech recognition [16], CNNs were also able to achieve state-of-the-art results.

Researchers used a convolutional neural network-based method to derive valuable information from spectrograms in [17]. On the GTZAN dataset, they used the Squeeze Block (SEBlock) in the convolutional neural network and achieved 92 % accuracy. Researchers in [18] achieved the most recent state-of-the-art performance. A custom CNN is used, which is a bottom-up broadcast neural network. They were able to extract very low level features from spectrograms with 93.9% accuracy on the GTZAN dataset. A comparison of classical machine learning models and CNNs is proposed. For machine learning models, spectrograms are used as input to CNN and handcrafted features.

III. DATASET

For this research, the GTZAN dataset [1] is used. GTZAN is the most widely used public dataset for music genre recognition analysis. The dataset contains 1000 audio tracks, each lasting 30 seconds. It is divided into ten genres, each with 100 songs. All of the tracks are in wave file format and are 22050 Hz Mono 16-bit audio files. For training 90 tracks are used and for testing models ten tracks are used. Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, and Rock are among the ten music genres represented in the dataset. The dataset is approximately 3.71 GB in size.

IV. METHODOLOGY

For In this section, first feature extraction and then model training is described for machine learning models and convolutional neural network models.

A. Machine Learning Model

1) Feature Extraction: Librosa, a python package for audio and music analysis was used for extracting various time domain and frequency domain features from each audio clip *[19]*.

The extracted features are spectral centroid, onset strength, zero-crossing rate, tempo, spectral contrast, spectral band- width, spectral flatness and spectral roll-off with n fft = 1024 (frame length) and hop length = 512. New features are generated for every audio clip by taking maximum, minimum, standard deviation, skew, and kurtosis of all the features extracted before for every frame and is considered to be the representative final feature for feeding the models.

2) Classifier: According to *[20]*, Support Vector Machines (SVMs) are very memory efficient and effective in high dimensional spaces. A pipeline is created and a grid search cross-validation is used to find the best hyperparameters. The first process in the pipeline uses Gaussian distribution to scale the input data and then the Support Vector Classifier from Scikit Learn Library is used as a core model for training *[21]*. A grid search is performed on various hyperparameters with 9-fold cross-validation to get the best model parameters.

Random Forest works based on the concept of combining results of models. It ensembles the prediction of multiple decision trees. Individual decision tree gets trained on subset of the training set which is known as bagging or bootstrap aggregation *[22]*. Random subset of the features used by each individual trees for the prediction *[23]*.

Boosting algorithm combines number of weak classifiers from which it generates strong classifier. Boosting algorithm trains classifiers sequentially using forward stagewise additive modelling *[24]*. XGB refers to eXtreme Gradient Boosting which implements boosting with fast and parallelized training *[25]*. In neural networks, the classic multilayer perception involves the use of arbitrary matrix multiplication in the different layers of the network *[26]*. Deep learning also works well with medical images *[27]*. Music Information is sought by users for assisting the building of a collection of music which is the 4th finding and 5th Finding is Music Information is sought by users to verify or identify the artists, lyrics or work *[28]*.

B. Convolutional Neural Networks

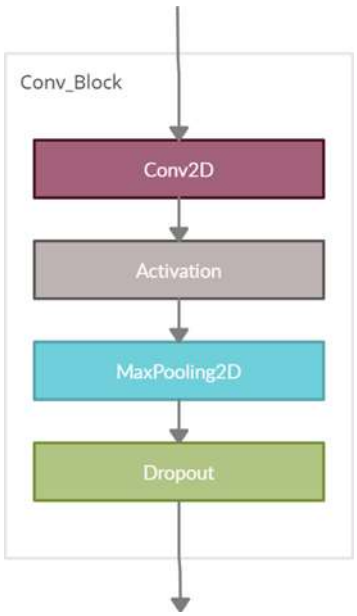


Fig. 1. Convolution Block Architecture

In this section, data preparation for CNN, the CNN model architecture, and implementation details are discussed.

1) Spectrogram Generation: A spectrogram is a matrix where the columns index frames (time) and the rows index frequency bins. A color map is used to express the magnitude of a given frequency within a given time window. The input songs are converted to Mel spectrograms

with n fft = 1024 (frame length) and hop length = 256 and, n mels = 128 (number of Mel bands), using log-scale with a window size of 3 seconds and overlap of 50%.

2) Architecture: Own custom CNN architecture has been used. The 'conv block,' as shown in Fig.2, is a collection of layers that includes a convolutional 2-D layer with 'n' filters, an activation layer with the activation feature 'relu,' and a max- pooling 2-D layer with pool size: (2,2). The Dropout layer is the block's final layer, with a probability of 0.25.

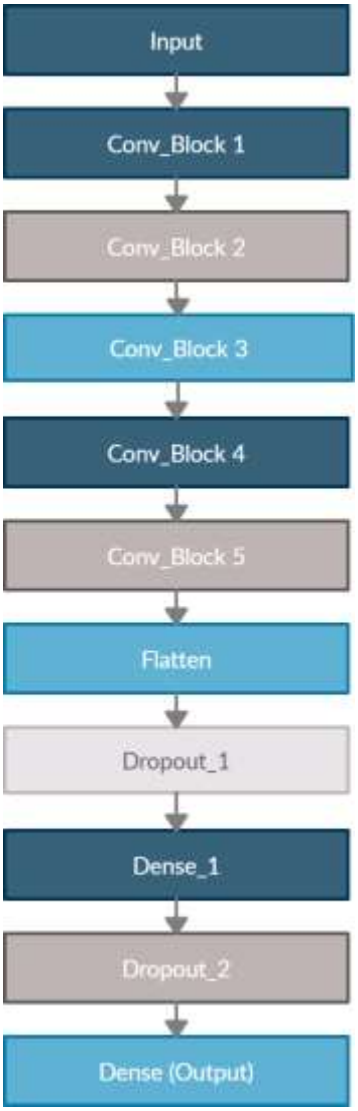


Fig. 2 depicts our custom CNN model architecture. The Input layer accepts spectrogram images with dimensions of 128 pixels in height and 129 pixels in width. 5 'conv block' are used sequentially, and the activation feature for all five blocks is 'relu', the max-pooling 2-D layer of all five blocks has pool size (2, 2), and the likelihood of dropout for all dropout layers in all five 'conv block' is 0.25, but the number of filters in the convolutional layer increases by a factor of 2 beginning with 16 in the first 'conv block'. Let 'N' denotes the number of filters in 'conv block' then:

$$N_t = 2^t + 3 \quad (1)$$

So, from the above equation, 'conv block 1' has 16 filters, 'conv block 2' has 32 and so on till 'conv block 5' having 256 filters in convolutional layer. 'conv block 5' is followed by a Flatten layer and Dropout 1 layer with probability 0.5. Dropout 1 layer is followed by Dense 1 layer which contains 128 neurons, further followed by Dropout 2 layer with probability 0.3 and finally a Dense (Output) layer with softmax activation function for the output.

3) Intuition: Peeking into the 'conv block,' which is repeated in architecture 5 times, reveals 4 layers. The first layer in the block is Conv2D, a convolutional layer that is used because it excels at performing image classification tasks due to its efficiency and accuracy in extracting features from photos. It is accompanied by an Induction layer with a 'relu' activation mechanism because 'relu' is computationally efficient in calculations because neurons are not activated for negative input values and only those neurons with positive input are activated. Following the Activation layer is a Max- Pooling2D layer that will subsample the contribution of the convolutional layer. Pooling aids in reducing the difficulty of feature maps while retaining as much detail about the picture as possible. The final layer in the 'conv block' is the Dropout layer, which prevents the model from overfitting the data.

Five 'conv blocks' are used for the entire custom CNN model, with an increasing number of filters after each 'conv block'. The importance of increasing the number of filters as the CNN progresses is that the convolutional layer initially captures edges, corners, and several other basic features. As the network grows in depth, convolutional layers combine these edges and corners to capture wider patterns and feature combinations, necessitating the use of more filters to capture as many patterns as possible

4) Training: Categorical cross-entropy is used as the loss function and Adam optimizer with learning rate 0.001. The model is trained for 16 epochs with batch size 128. The accuracy and loss over epochs are shown in Fig. 3 and Fig. 5 respectively.

V. RESULT

The metrics used are accuracy, precision, and recall. According to Table 1, SVM has outperformed both Random Forest and Gradient Boosted Trees (XGB) in the classical machine learning approach. However, the highest precision and recall among all the models is achieved by CNN. This shows that Mel spectrograms are highly co-related features for classification..

TABLE I. COMPARISION OF VARIOUS CLASSIFIERS

Model	Accuracy	Precision	Recall
SVM	69.1	70	67

Model	Accuracy	Precision	Recall
Random Forest	66.2	68.2	69.1
XGB	71.2	72.3	70.5
CNN	74.1	77.2	73.5

The result shows that the feature extraction task is very well done by CNNs and it achieves higher accuracy than traditional machine learning models. Transfer learning has been proved to be very effective in image recognition task [28]. But research in [2] clarifies that complex architecture and large models fail to classify the audio clips in GTZAN dataset. Larger architectures can easily over-fit and they tend to perform well when the dataset is very complex and large. They also have a large amount of parameters and take more time and computing resource in training and testing on dataset. Whereas, small and simple CNNs can under-fit but they perform very well when dataset is small and not much complex. They can also be trained faster and use comparatively less computing resource while training and testing.

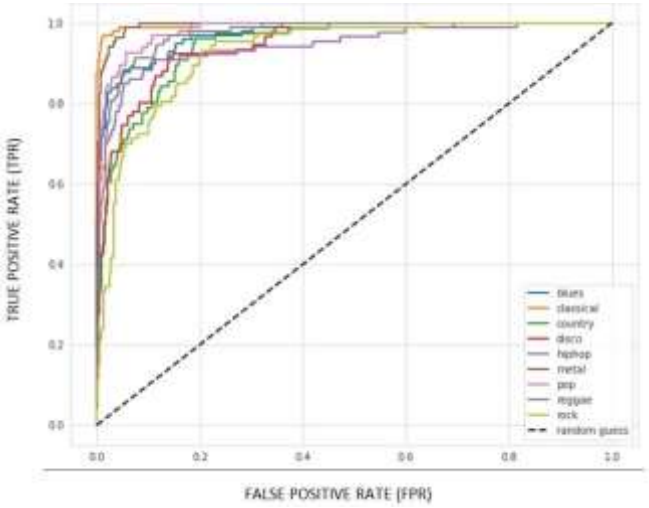
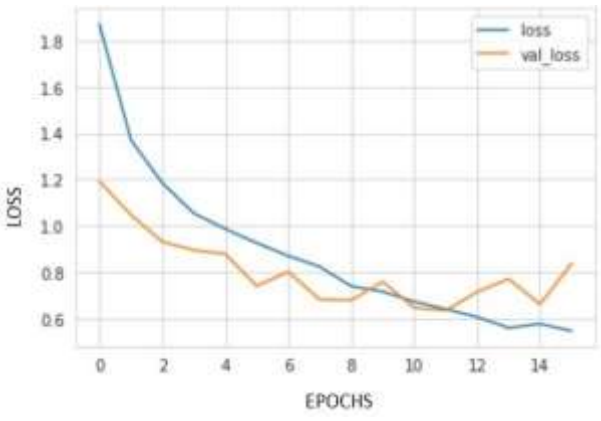


Fig. 2 Accuracy over Epochs of CNN model



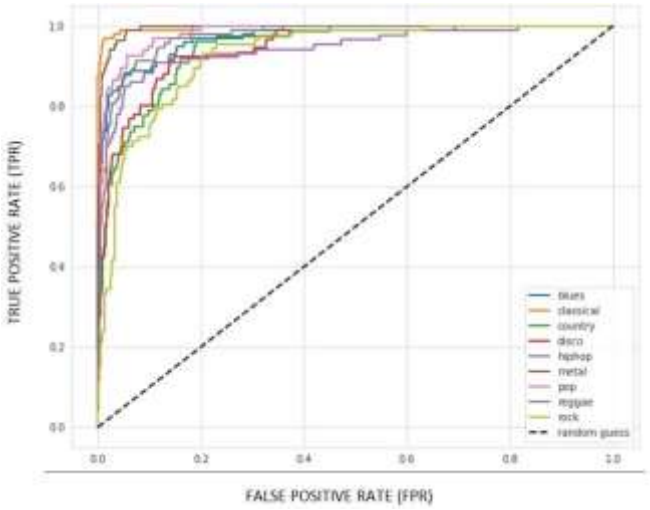


Fig. 3 ROC curve of CNN model

VI. CONCLUSION

GTZAN dataset is used for classifying music genres and achieved the highest accuracy using the CNN model. Two different approaches for this task were used. In the first approach, the time domain and frequency domain features are manually extracted and fed into classical machine learning models. In the second approach, spectrogram images are extracted from audio files and fed them to CNNs. Finally, the performance of all the models are evaluated.

Future work aims at preprocessing data and using data augmentation techniques to generate more data to reduce variance and increase model performance. Some new technique can be used to extract more information from the spectrogram images and use more complex but not large architectures designed manually which can increase the accuracy.

VII. REFERENCES

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing,* vol. 10, p. 293–302, 2002.

[2] S. Sugianto and S. Suyanto, "Voting-based music genre classification using melspectogram and convolutional neural network," in *2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2019.

[3] N. Scaringella and G. Zoia, "On the Modeling of Time Information for Automatic Genre Recognition Systems in Audio Signals.," in *ISMIR*, 2005.

[4] M. I. Mandel and D. P. W. Ellis, "Song-level features and support vector machines for music classification," 2005.

[5] L. Nanni, Y. M. G. Costa, A. Lumini, M. Y. Kim and S. R. Baek, "Combining visual and acoustic features for music genre classification," *Expert Systems with Applications,* vol. 45, p. 108–117, 2016.

[6] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G.

Penn and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on audio, speech, and language processing,* vol. 22, p. 1533–1545, 2014.

[7] L. Wyse, "Audio spectrogram representations for processing with convolutional neural networks," *arXiv preprint arXiv:1706.09559,* 2017.

[8] T. L. Li, A. B. Chan and A. H. Chun, "Automatic musical pattern feature extraction using convolutional neural network," *Genre,* vol. 10, p. 1x1, 2010.

[9] J. Yang, "Music Genre Classification With Neural Networks: An Examination Of Several Impactful Variables," 2018.

[10] A. Balakrishnan and K. Dixit, "Deepplaylist: using recurrent neural networks to predict song similarity," *Stanfort University,* p. 1–7, 2014.

[11] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 international conference on engineering and technology (ICET)*, 2017.

[12] Q. H. Nguyen, T. T. T. Do, T. B. Chu, L. V. Trinh, D. H. Nguyen, C. V. Phan, T. A. Phan, D. V. Doan, H. N. Pham, B. P. Nguyen and others, "Music genre classification using residual attention network," in *2019 International Conference on System Science and Engineering (ICSSE)*, 2019.

[13] P. Hamel and D. Eck, "Learning features from music audio with deep belief networks.," in *ISMIR*, 2010.

[14] P. Fulzele, R. Singh, N. Kaushik and K. Pandey, "A hybrid model for music genre classification using LSTM and SVM," in *2018 Eleventh International Conference on Contemporary Computing (IC3)*, 2018.

[15] K. Ullrich, J. Schlüter and T. Grill, "Boundary Detection in Music Structure Analysis using Convolutional Neural Networks.," in *ISMIR*, 2014.

[16] T. N. Sainath, A.-r. Mohamed, B. Kingsbury and B. Ramabhadran, "Deep convolutional neural networks for LVCSR,[in:] 2013 IEEE International Conference on Acoustics," *Speech and Signal Processing,* p. 8614–8618, 2013.

[17] Y. Xu and W. Zhou, "A deep music genres classification model based on CNN with Squeeze & Excitation Block," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2020.

[18] C. Liu, L. Feng, G. Liu, H. Wang and S. Liu, "Bottom-up broadcast neural network for music genre classification," *Multimedia Tools and Applications,* vol. 80, p. 7313–7331, 2021.

[19] B. McFee, V. Lostanlen, A. Metsai, M. McVicar, S. Balke, C. Thom_, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, J. Mason, D. Ellis, E. Battenberg, S. Seyfarth, R. Yamamoto, K. Choi, viktorandreevichmorozov, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, D. Here__, F.-R. St_ter, P. Friesch, A. Weiss, M. Vollrath and T. Kim, *librosa/librosa: 0.8.0,* Zenodo, 2020.

[20] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning,* vol. 20, p. 273–297, 1995.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg and others, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research,* vol. 12, p. 2825–2830, 2011.

[22] L. Breiman, "Random forests," *Machine learning,* vol. 45, p. 5–32, 2001.

[23] Y. Amit and D. Geman, "Shape quantization and recognition with randomized trees," *Neural computation,* vol. 9, p. 1545–1588, 1997.

[24] T. Hastie, R. Tibshirani and J. Friedman, "The elements of statistical learning. Springer series in statistics," *New York, NY, USA,* 2001.

[25] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016.

[26] M. Hussain, J. J. Bird and D. R. Faria, "A study on cnn transfer learning for image classification," in *UK Workshop on computational Intelligence*, 2018.

[27] T. Vyas, R. Yadav, C. Solanki, R. Darji, S. Desai and S. Tanwar, "Deep learning-based scheme to diagnose Parkinson's disease," *Expert Systems,* p. e12739, 2021.

[28] D. Jayati, P. Dhara, F. Pranav and T. Vyas, "Music Information Retrieval: A Window into the Needs and Challenges," in *International Conference on Emerging Research in Computing, Information, Communication and Applications*, 2016.

[29] T. Vijayakumar, "Posed inverse problem rectification using novel deep convolutional neural network," *Journal of Innovative Image Processing (JIIP),* vol. 2, p. 121–127, 2020.

[30] M. H. Pimenta-Zanon, G. M. Bressan and F. M. Lopes, "Complex Network-Based Approach for Feature Extraction and Classification of Musical Genres," *arXiv preprint arXiv:2110.04654,* 2021.