IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 1, 2021, accepted June 17, 2021, date of publication July 1, 2021, date of current version July 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3093911

Classification of Indian Classical Music With Time-Series Matching Deep Learning Approach

AKHILESH KUMAR SHARMA[1], (Member, IEEE), GAURAV AGGARWAL[1], SACHIT BHARDWAJ[1], PRASUN CHAKRABARTI[2], (Senior Member, IEEE), TULIKA CHAKRABARTI[3], JEMAL H. ABAWAJY[4], (Senior Member, IEEE), SIDDHARTHA BHATTACHARYYA[5], (Senior Member, IEEE), RICHA MISHRA[6], ANIRBAN DAS[7], AND HAIRULNIZAM MAHDIN[8], (Member, IEEE)

[1]Department of Information Technology, Manipal University Jaipur, Jaipur, Rajasthan 303007, India
[2]Department of Computer Science and Engineering, Techno India NJR Institute of Technology, Udaipur, Rajasthan 313003, India
[3]Department of Chemistry, Sir Padampat Singhania University, Udaipur, Rajasthan 313601, India
[4]School of Information Technology, Deakin University at Waurn Ponds, Geelong, VIC 3216, Australia
[5]Rajnagar Mahavidyalaya, Rajnagar, West Bengal 731130, India
[6]Vishwaniketan's Institute of Management Entrepreneurship and Engineering Technology, Khalapur, Maharashtra 410202, India
[7]Department of Computer Science, University of Engineering and Management, Kolkata, West Bengal 700156, India
[8]Center of Intelligent and Autonomous System, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Batu Pahat 86400, Malaysia

Corresponding authors: Prasun Chakrabarti (drprasun.cse@gmail.com), Jemal H. Abawajy (jemal.abawajy@deakin.edu.au), and Hairulnizam Mahdin (hairuln@uthm.edu.my)

This work was supported in part by Universiti Tun Hussein Onn Malaysia (UTHM) and in part by the UTHM Publisher's Office through the Publication Fund under Grant E15216.

ABSTRACT Music is a heavenly way of expressing feelings about the world. The language of music has vast diversity. For centuries, people have indulged in debates to stratisfy between Western and Indian Classical Music. But through this paper, an understanding can be fabricated while differentiating the types of Indian Classical Music. Classical music is one of the essential characteristics of Indian Cultural Heritage. Indian Classical Music is divided into two major parts, i.e. Hindustani and Carnatic. Models have been sculptured and trained to classify between Hindustani and Carnatic Music. In this paper, two approaches are used to implement classification models. MFCCs are used as features and implemented models like DNN (1 Layer, 2 Layers, 3 Layers), CNN (1 Layer, 2 Layers, 3 Layers), RNN-LSTM, SVM (Sigmoid, Polynomial & Gaussian Kernel) as one approach. A 3 channels input is created by merging features like MFCC, Spectrogram and Scalogram and implemented models like VGG-16, CNN (1 Layer, 2 Layers, 3 Layers), ResNet-50 as another approach. 3 Layered CNN and RNN-LSTM model performed best among all the approaches.

INDEX TERMS DNN, CNN, SVM, Sigmoid Kernel, Polynomial Kernel, Gaussian Kernel, RNN-LSTM, VGG-16, ResNet-50, MFCCs, Spectrogram, Scalogram.

I. INTRODUCTION

India has the most extensive Intangible Cultural Heritage globally, and Music is one of the most crucial aspects of this Cultural Heritage. Indian Classical Music is divided into two major parts, i.e. Hindustani and Carnatic [1]. The Classical Music tradition followed in India's Northern region is known as Hindustani, while tradition followed in the Southern region is Carnatic [1]. This distinction of music was observed around 16th century. Both aspects were evolved from a common ancestor. Bhakti Movement gave birth to Carnatic strain while Hindustani strain during the Vedic phase [2]. Dhrupad, Khayal, Tarana, Thumri, Dadra and Gazals are the main vocal forms of Hindustani Music while Alpana, Niraval, Kalpnaswaram and Ragam Thana Pallavi for Carnatic Music. Carnatic Music comprises 72 ragas with the employment of Veena, Mandolin and Mridangam. Hindustani Music whereas comprises 6 major ragas with the employment of Sarangi, Tabla, Santoor and Sitar [3]. There is only one definite technique directed style of chanting in Carnatic while various sub-styles in Hindustani. The vocal part is prioritized in Hindustani Music, where both are given indistinguishable importance in Carnatic [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson.

Indian Classical Music based Raga Music classification is the upcoming area under music information retrieval. These studies are conceivable due to the availability of a considerable amount of musical data on the Internet. Significant work has been done in multimedia such as text and video. But the audio processing is still in the developing phase. It involves the processing of music and speech. This paper discusses speech processing, which could be used as the basis of Classical music classification using feaures such as MFCCs, Spectrogram, Scalogram. The sound and music features are studied and the features are extracted to perform the classification on these categories of the music. The initial phase are discussed, which used the music signals. The pitch class profiles based features and their acoustic characteristics based statistical measures are also considered along with algorithms applied. The promising results are depicted in this study along with performance comparison.

Studying music is an upcoming area that involves various computational techniques for investigating different forms of music. The computations of music used to understand society's heritage and culture from where the music evolved. It also pulls out the science behind the music and helps in developing the scientific model. Usually, researches focus on Western Music, while some studies have explored Indian Classical Music Sound [4]. Indian Classical Music is mainly classified into Carnatic music and Hindustani music. These two music form frameworks are similar with some stylistic differentiation. Hindustani Music is hinged on Raga based composition while Carnatic Music over Kriti. However, they have grown differently under diverse cultural inspirations [3].

Indian Classical Music is broadly categorized into Carnatic Music and Hindustani Music. Both are heaving a wide following in their way, but Carnatic Music's complexity is much higher in the means the notes are rendered and arranged [4]. Indian Classical Music is generally based on Raga and Talam. Talam can be considered equivalent to the melody in Western Music. The complexity of ragas is more as compared to Western Music in the context of melody and scale. Ragas notes are sequentially arranged so they can invoke the emotion of the song. A note is defined as Swara in Carnatic Music. Every note has a set frequency associated with it [5]. Every Carnatic Music has a Talam associated. The time duration of a song in Carnatic Music is an integral multiple of Talam. Talam is just like a beat in Western Music. It signifies the placement of the syllables and the tempo of the music in the composition. In Carnatic Music, Talam is indicated by singer hand gestures. Hence in this paper, Ragas patterns are considered to categorized Hindustani Classical Music from Carnatic Classical Music.

Music Note is considered to be an atomic unit of Indian Classical Music. In a real sense, the musical note considered as an identifiable fundamental frequency component (also known as pitch) of a singer with an appropriate duration [5]. The ratio of the fundamental frequencies of two notes is referred to as an interval [5].

Sa, Ri, Ga, Ma, Pa, Da, Ni are the seven musical notes carrying frequencies which further subdivided into semitones or microtones [31]. Full forms of Sa, Ri, Ga, Ma, Pa, Da, Ni are Shadja, Rishaba, Gandhara, Madhyama, Panchama, Dhaivatha and Nishadha [4]. 16-note scale, 12-note scale, & 22-note micro-tone are 3 kinds of scales utilised in Hindustani and Carnatic Music [1], [20], [4], [8]. The 16-note scale has been used in Carnatic Music while the 12-note scale in Hindustani Music. 12 various frequency components have been observed in Carnatic Music.

Melodic audio generated when combining and playing together notes is known as Raga, similar to Western Music [9]. Arohana-avarohana patterns play a crucial role in different melody while having the same set of notes. The progression of notes, i.e. descending and ascending, is known as Arohana-avarohana patterns. It gives the knowledge about the transformation of notes which may go through in a raga [7].

II. LITERATURE REVIEW

The literature availability based on Carnatic and Hindustani Music is minimal as compared to Western Music. Very few studies have been conducted on Singer identification and Swara pattern recognition on Carnatic Music [5], [6]. Simultaneously, some works are being done to identify the Ragas in Hindustani Music [7]. In [7] the authors constructed an HMM-based model which recognized two Ragas of Hindustani Classical Music. In [8] authors has suggested a primary difference between the Raga patterns of Hindustani and Carnatic Music. It says we have R1 and R2 raga patterns in Hindustani music compared to R1, R2 and R3 in Carnatic. Likewise, G, D and N all have three different frequencies in Carnatic Classical Music against two Hindustani Classical frequencies, enhancing frequency identifications. The input signal used is a monophonic, voice-only music signal. The signal's fundamental frequency was also examined, and based on these features, the raga identification process was conducted for two Hindustani ragas. In the aspect of Western Music, researchers investigated the function of melody retrieval. In this paper, we have taken a song dataset consisted of Hindustani and Carnatic Classical Music. Apply speech signal processing algorithms to extract Mel frequency cepstral coefficients (MFCC) features for each song. Different classification algorithms have applied to classify Hindustani and Carnatic Classical Music.

III. METHODOLOGY

The proposed methodology contains dataset which comprises of audio files of Carnatic and Hindustani Music. Dataset consists of 28 Carnatic files while 36 Hindustani with 160 seconds as track duration. '0' has been classified as Carnatic while '1' as Hindustani in this study. This study consists of several layers of comparisons, as shown in Fig-1. The first layer consists of extracting 3 features, i.e. MFCC, Spectrogram, Scalogram while the second layer consists of integrating features. The third layer consists of comparing model's

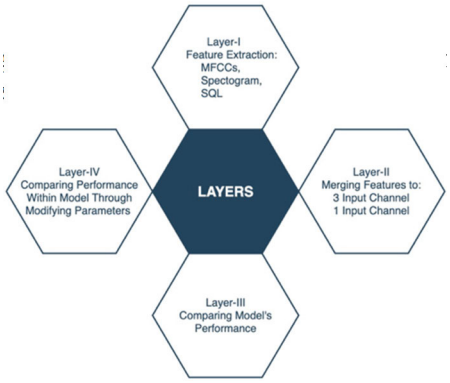

FIGURE 1. Layers of this study.

performance, while the fourth layer consists of comparing performance within models through modifying parameters.

Here, 2 type of approaches. MFCCs, Spectrogram and Scalogram features are merged into 3 channel configuration followed by training through VGG-16, ResNet-50 and CNN models as one approach. As another approach towards this, MFCCs are integrated into 1 channel configuration followed by training through DNN, SVM, RNN-LSTM and CNN models.

This study has been carried out on Google Colaboratory environment, with 13 GB of RAM and Intel(R) Xeon(R) CPU @ 2.20GHz. Training 3 channel input data has carried over TPU provided by Google Colaboratory. In this study, the Sample Rate assumed to be 22050. The dataset is divided into training and test data in the ratio of 7:3. Cross-validation is used to check the reliability of the dataset.

IV. FEATURE EXTRACTION

A. MEL FREQUENCY CEPSTRAL COEFFICIENTS (MFCCs)

The primary feature extraction technique used in this study is MFCC. MFCC extraction from audio files has been carried out in significant 5 steps. These 5 steps are Pre Emphasis, Frame Blocking and Windowing, Discrete Fourier Transform, Mel Spectrum, Discrete Cosine Transform [21]. The balancing of sound is performed in Pre Emphasis, which filter higher frequencies. Eq-1 shows the executed pre-emphasis filter in this study.

$$\mathbf{H}(\mathbf{z}) = \mathbf{1} - \mathbf{b}\mathbf{z}^{-\mathbf{1}} \tag{1}$$

where **b** is the slope of the filter.

Segmentation of audio is executed in the second step, i.e. Frame Blocking and Windowing, to achieve a windowed section. This technique is effective against the edge effect, which is usually observed in Fourier Transform [21]. In this study, the audio is segmented into 10 subparts with a windowed length of 512. Any sampled signal can be constituted as a finite series of sinusoids. This alteration is known as Fourier Transform **X** (**k**) that is shown in Eq-2. The spectrum is obtained through Discrete Fourier Transform from each

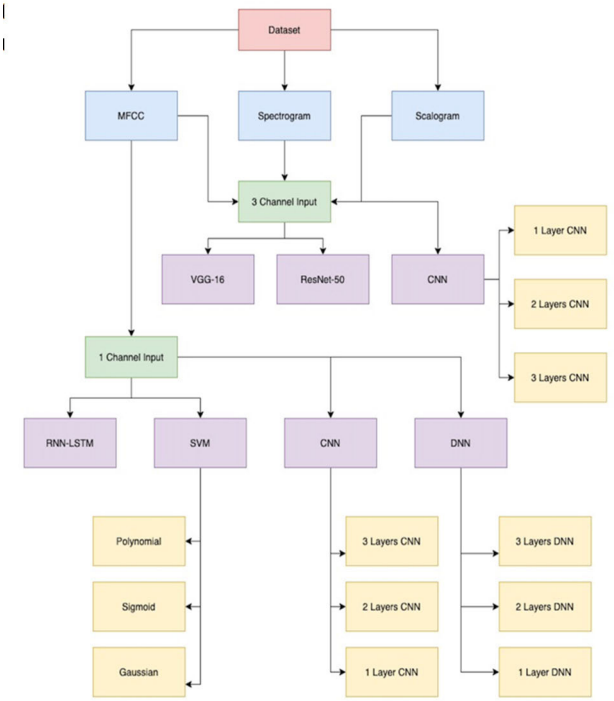

FIGURE 2. Methodology.

windowed frame in the third step [21].

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{\frac{-j2\pi nk}{N}}; \quad 0 \le k \le N-1 \tag{2}$$

where **N** is the number of points, the value of FFT is assumed to be 2048 in this study.

Usually, after processing DFT, the spectrum is observed to be very extensive. Hence to make frequencies range linear, they are passed through Mel-filter banks. Mel-filter bank is a set of bandpass filters. The Mel scale is shown in Eq-3.

$$f_{Mel} = 2595\log_{10}\left(1 + \frac{f}{700}\right) \tag{3}$$

where $\mathbf{f_{Mel}}$ denotes the perceived frequency, and f represents the physical frequency in Hz [25].

The triangular Mel Weighting filter is multiplied by spectrum to evaluate Mel Spectrum [**s** (**m**)] as shown in Eq-4.

$$s(m) = \sum_{k=0}^{N-1} \left[|X(k)|^2 H_m(k) \right]; \quad 0 \le m \le M-1 \tag{4}$$

where $\mathbf{H_m}$ is weight to $\mathbf{k^{th}}$ energy spectrum granting to $\mathbf{m^{th}}$ output band, **M** is total triangular Mel weighting filters.

The Illustration of Mel spectrum over logarithmic scale has been performed in the last step, followed by execution of Discrete Cosine Transform (DCT) execution through which production of cepstral coefficients occurs [21].

$$c(n) = \sum_{m=0}^{M-1} \log_{10}(s(m)) \cos\left(\frac{\pi n(m-0.5)}{M}\right). \tag{5}$$

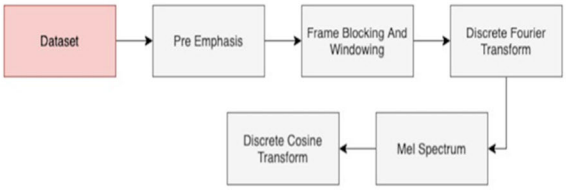

FIGURE 3. Flow chart of MFCC.

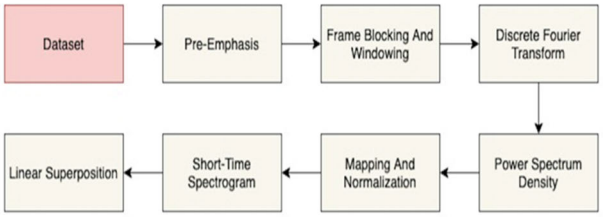

FIGURE 4. Flow chart of spectrogram.

where C is the number of MFCCs, $\mathbf{n} = \mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots, \mathbf{C-1}$, **and** $\mathbf{c}(\mathbf{n})$ are the cepstral coefficients. The value of C is assumed to be 13 in this study.

B. SCALOGRAM

The portrayal of a signal's time-frequency domain through wavelet transformation is known as Scalogram. It is used to identify coefficient estimates at respective time-frequency positions [22]. Scalogram provides an in-depth visualization of the signal. Scalogram is the modulus of the multiscale wavelet transform, which elucidate time-frequency visualization. The spectro-temporal nature of scalograms makes them desirable for neural networks due to the signal's mapping properties with minimalistic information loss [23]. Scalogram gives the insight into the frequency, energy in time which shown as a function $\mathbf{Ix}(\mathbf{t}, \boldsymbol{\lambda})$ in Eq-6.

$$\mathbf{Ix}(\mathbf{t}, \boldsymbol{\lambda}) = |\mathbf{Wx}(\mathbf{t}, \boldsymbol{\lambda})| = \left|\mathbf{x} \star \boldsymbol{\psi}_{\boldsymbol{\lambda}}(\mathbf{t})\right|. \tag{6}$$

where $\mathbf{Wx}(\mathbf{t}, \boldsymbol{\lambda})$ is the wavelet transform, $\mathbf{x}$ is energy, $\boldsymbol{\psi}_{\boldsymbol{\lambda}}(\mathbf{t})$ is dilated wavelet and $\mathbf{2}^{\boldsymbol{\lambda}}$ is frequency [23].

C. SPECTROGRAM

The visualization of signal's robustness at several frequencies over time in a waveform is known as Spectrogram. In other words, it is also known as the representation of signal's loudness. It's the intensity plot of the Short-Time Fourier Transform (STFT) magnitude. The succession of data segment's Fast Fourier Transform (FFT) is known as Short-Time Fourier Transform (STFT). The Spectrogram extraction is carried out in 8 broad steps, i.e. Pre-emphasis, Frame Blocking, Windowing, Discrete Fourier Transform (DCT), Power Spectrum Density (PSD), Mapping and Normalization, Short-time Spectrogram, and Linear Superposition [26].

Power Spectral Density $\mathbf{S_X}(\mathbf{f})$ of signal X(t) is computed as the Fourier Transform of an autocorrelation function $\mathbf{R_X}(\boldsymbol{\tau})$ as shown in Eq-7.

$$\mathbf{S_X(f)} = \mathcal{F}\{\mathbf{R_x}(\boldsymbol{\tau})\} = \int_{-\infty}^{\infty} \mathbf{R_x}(\boldsymbol{\tau}) \mathbf{e}^{-\mathbf{2j\pi f\tau}} \mathbf{d}\boldsymbol{\tau} \tag{7}$$

where $\mathbf{j} = \sqrt{-\mathbf{1}}$.

V. MODELS

A. DEEP NEURAL NETWORK (DNN)

In this study, 3 configurations of Deep Neural Networks are used, i.e. 3 Layers, 2 Layers, 1 Layer. DNN model is sculptured as a sequential model and trained for one channel input i.e. MFCCs features. A Flatten layer is used before passing inputs through layers to make inputs a whole vector. These models are trained through 200 epochs with numerous configurations of parameters like Batch Size, Adam Learning Rate, etc., as shown in result's section in this paper. The Flatten layer changed the input shape to 5980. The dropout layer of 0.3 rate is added to the model followed by the L2 regularization technique with value 0.001 to prevent overfitting of the model at every hidden layer. ReLU activating function is used in all hidden layers while Softmax activating function at the output layer. Adam Optimizer is used as an optimizer

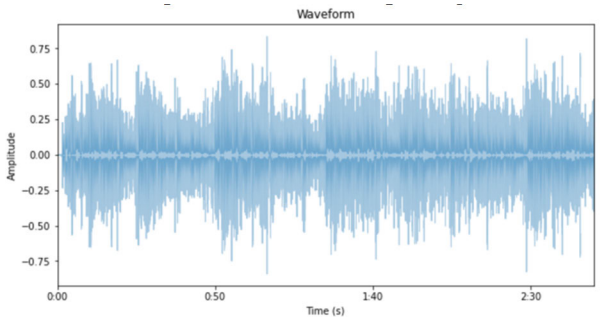

FIGURE 5. Waveform of sample audio file.

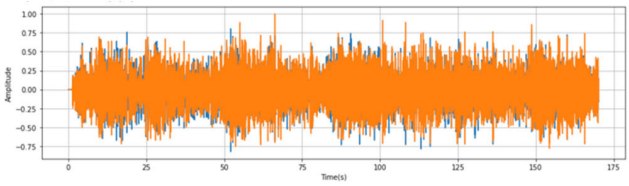

FIGURE 6. Normalized audio of sample audio file.

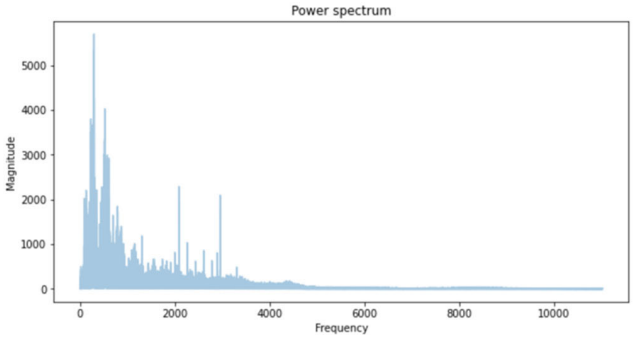

FIGURE 7. Power spectrum of sample audio file.

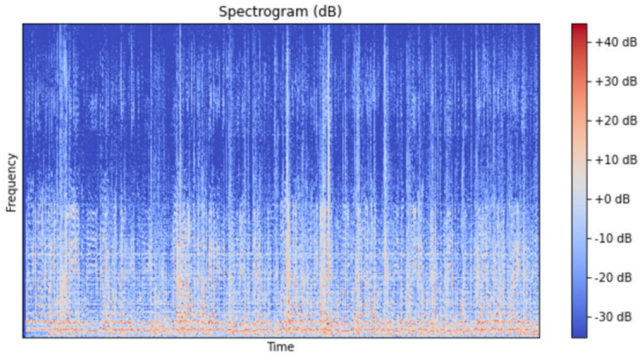

FIGURE 8. Spectrogram of sample audio file.

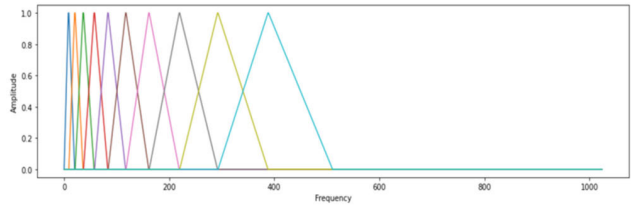

FIGURE 9. Filters.

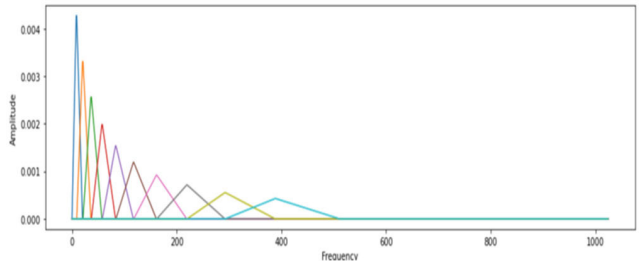

FIGURE 10. Filter Bank on Mel scale of sample audio file.

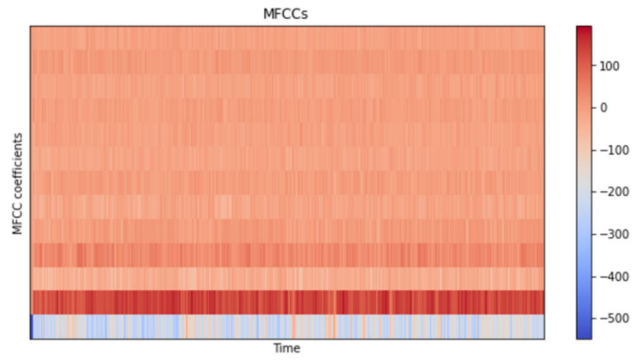

FIGURE 11. MFCCs of sample audio file.

to this model while Sparse Categorical Cross Entropy as loss function.

In 3 layers DNN model, total parameters have observed to be 3,210,698, out of which all are trainable parameters. In 2 layers DNN model, total parameters have observed to be 3,196,170, out of which all are trainable parameters. In a single layer DNN model, total parameters have observed to be 3,067,402, out of which all are trainable parameters.

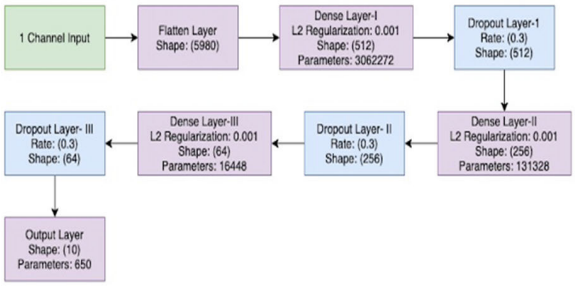

FIGURE 12. 3 Layered DNN.

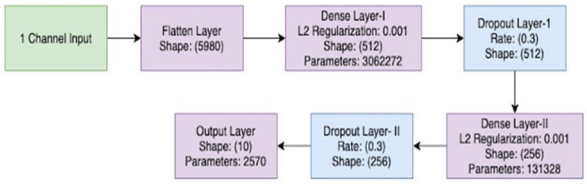

FIGURE 13. 2 Layered DNN.

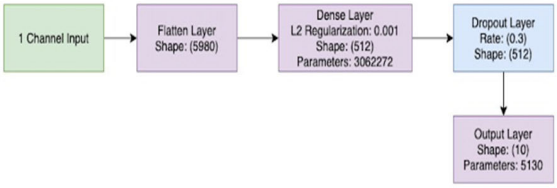

FIGURE 14. Single Layered DNN.

B. LONG TERM SHORT MEMORY-RECURRENT NEURAL NETWORK (RNN-LSTM)

In this study, a 2 layer LSTM is used with one hidden layer. RNN-LSTM model is sculptured and trained for one channel input, i.e. MFCCs features. In this model, total parameters have observed as 57,802, in which all are trainable. This model is trained through 100 epochs with numerous configurations of parameters like Batch Size, Adam Learning Rate, etc., as shown in result's section in this paper. Dropout layers of 0.3 rate is adopted to prevent overfitting of the model. ReLU activating function is used in a single hidden layer while Softmax activating function at the output layer followed by 2 LSTM layers before. Adam Optimizer is used as an optimizer to this model while Sparse Categorical Cross Entropy as loss function.

C. CONVOLUTION NEURAL NETWORK (CNN)

In this study, 3 configurations of Convolution Neural Networks are used, i.e. 3 Layers, 2 Layers, 1 Layer with 2 types of inputs. This model is trained through 30 epochs with numerous configurations of parameters like Batch Size, Adam Learning Rate, etc, as shown in result's section in this paper.

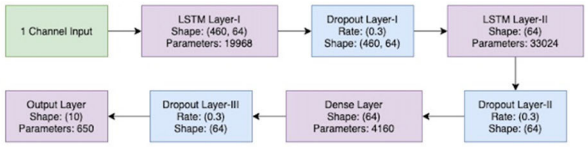

FIGURE 15. RNN-LSTM.

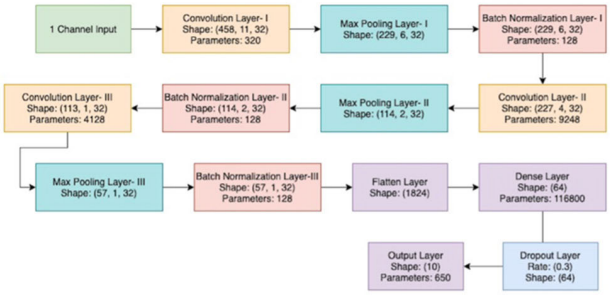

FIGURE 16. 3 Layered CNN Model-A.

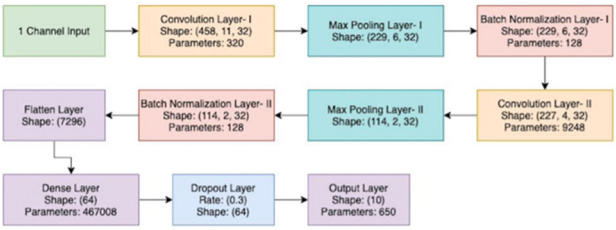

FIGURE 17. 2 Layered CNN Model-A.

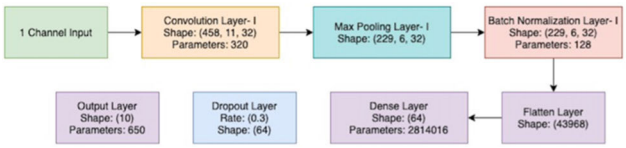

FIGURE 18. Single Layered CNN Model-A.

A Layer of CNN Model consists of a 2D Convolution Layer with ReLU activation function followed by a 2D Max Pooling layer. Padding is adopted to be the same and strides of (2, 2) followed by a Batch Normalization Layer. These all aspects joint together become a single convolution layer. After passing through various layers of the convolutional network, a flatten layer has changed the output to a single vector. It can further train in the dense layer. A Dropout layer is used and the dense layer to prevent overfitting, followed by an output layer consisting of a softmax acting function. Adam Optimizer is used as an optimizer to this model while Sparse Categorical Cross entropy as loss function.

CNN Model-A is sculptured as sequential model and trained for 1 channel input, i.e. MFCCs features. In 3 layers CNN model-A, total parameters has observed to be 131,530, out of which 131,338 are trainable parameters whereas 192 non-trainable. In 2 layers CNN model-A,

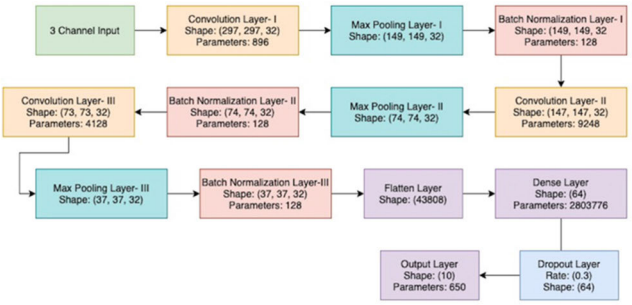

FIGURE 19. 3 Layered CNN Model-B.

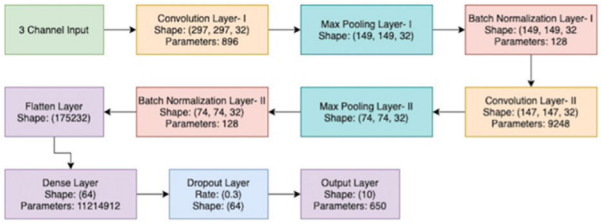

FIGURE 20. 2 Layered CNN Model-B.

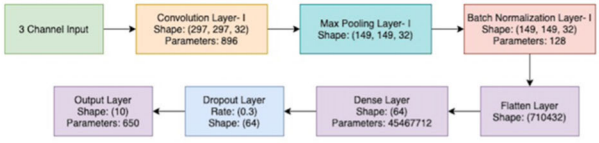

FIGURE 21. Single Layered CNN Model-B.

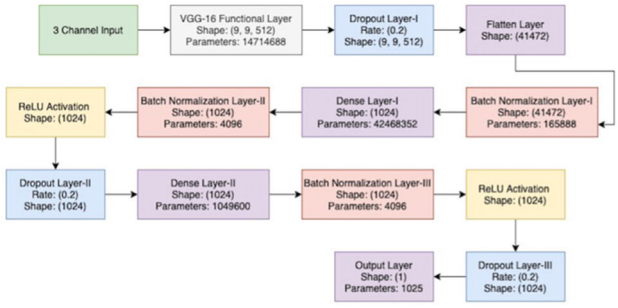

FIGURE 22. VGG-16.

total parameters has observed to be 477,482, out of which 477,354 are trainable parameters whereas 128 non-trainable. In a single layer CNN model-A, total parameters has observed to be 2,815,114, out of which 2,815,050 are trainable parameters whereas 64 non-trainable.

CNN Model-B is sculptured as a sequential model and trained for 3 channels input i.e. MFCCs, Spectrogram & Scalogram. In 3 layers CNN model-B, total parameters has observed to be 2,819,082 out of which 2,818,890 are trainable parameters whereas 192 non-trainable. In 2 layers CNN model-B, total parameters has observed to be 11,225,962 out of which 11,225,834 are trainable parameters whereas 128 non-trainable. In a single layer CNN

TABLE 1. Neural Network Evaluation For Learning Rate- 0.0001.

Input Channel	Model	Epochs	Batch Size	Adam Learning Rate – 0.0001					
				Accuracy	Validation Accuracy	Precision	Recall	F1 Score	ROC
1	3 Layers DNN	200	32	97.82%	89.16%	90.20%	94.16%	92.14%	96.47%
1	3 Layers DNN	200	64	98.12%	90.64%	89.18%	97.77%	93.28%	97.12%
1	3 Layers DNN	200	128	96.89%	88.67%	90.76%	91.47%	91.11%	97.62%
1	3 Layers DNN	200	512	91.95%	88.67%	86.42%	96.80%	91.32%	96.22%
1	2 Layers DNN	200	32	98.37%	89.16%	90.34%	94.24%	92.25%	96.18%
1	2 Layers DNN	200	64	97.72%	92.61%	91.36%	97.69%	94.42%	90.82%
1	2 Layers DNN	200	128	96.42%	89.66%	88.40%	96.06%	92.07%	97.50%
1	2 Layers DNN	200	512	98.52%	85.71%	83.68%	95.16%	89.05%	93.02%
1	1 Layer DNN	200	32	99.50%	93.60%	93.43%	96.96%	95.16%	92.14%
1	1 Layer DNN	200	64	99.50%	88.67%	88.27%	95.52%	91.75%	85.44%
1	1 Layer DNN	200	128	98.84%	89.16%	85.98%	100%	92.46%	93.82%
1	1 Layer DNN	200	512	100%	91.13%	92.80%	94.16%	93.47%	89.50%
1	RNN-LSTM	100	32	98.45%	92.16%	91.58%	95.14%	93.33%	90.75%
1	RNN-LSTM	100	64	92.22%	93.14%	99.01%	84.16%	90.99%	91.06%
1	RNN-LSTM	100	128	89.11%	85.29%	87.90%	100%	93.56%	87.50%
1	RNN-LSTM	100	512	63.37%	64.71%	60.35%	100%	75.27%	50%
1	3 Layers CNN	30	32	99.07%	96.08%	94.25%	94.21%	94.22%	97.08%
1	**3 Layers CNN**	**30**	**64**	**98.53%**	**96.08%**	**95.70%**	**95.70%**	**95.70%**	**95.70%**
1	3 Layers CNN	30	128	94.99%	71.57%	75%	80.13%	70.94%	80.13%
1	3 Layers CNN	30	512	68.81%	32.35%	10.18%	22.91%	14.10%	59.53%
1	2 Layers CNN	30	32	99.93%	91.18%	90.16%	90.65%	90.40%	90.65%
1	2 Layers CNN	30	64	99.21%	93.14%	91.66%	94.77%	92.69%	94.77%
1	2 Layers CNN	30	128	98.93%	45.10%	68.88%	58.82%	42.41%	58.82%
1	2 Layers CNN	30	512	93.56%	36.27%	67.50%	51.49%	28.82%	51.49%
1	1 Layer CNN	30	32	98.39%	89.22%	89.22%	88.91%	88.58%	77.47%
1	1 Layer CNN	30	64	99.41%	84.31%	83.38%	85.36%	85.36%	87.82%
1	1 Layer CNN	30	128	98.15%	58.82%	72.36%	69.11%	58.56%	89.11%
1	1 Layer CNN	30	512	91.58%	34.31%	17.15%	50%	25.54%	50%
3	3 Layers CNN	30	32	57.47%	58.33%	29.16%	50%	36.84%	50%
3	3 Layers CNN	30	64	56.45%	56.25%	28.12%	50%	36%	50%
3	3 Layers CNN	30	128	53.82%	60.42%	30.20%	50%	37.66%	50%
3	**3 Layers CNN**	**30**	**512**	**57.29%**	**60.42%**	**30.20%**	**50%**	**37.66%**	**50%**
3	2 Layers CNN	30	32	59.11%	58.33%	29.16%	50%	36.84%	50%
3	2 Layers CNN	30	64	60.68%	56.25%	28.12%	50%	36%	50%
3	2 Layers CNN	30	128	54.95%	58.33%	29.16%	59%	36.84%	50%
3	2 Layers CNN	30	512	58.85%	58.33%	29.16%	50%	36.82%	50%
3	1 Layer CNN	30	32	49.29%	60.42%	30.20%	50%	37.66%	50%
3	1 Layer CNN	30	64	56.77%	52.08%	26.04%	50%	34.24%	50%
3	1 Layer CNN	30	128	56.86%	47.92%	23.95%	50%	32.39%	50%
3	1 Layer CNN	30	512	56.77%	50%	25%	50%	33.33%	50%

TABLE 2. Neural Network Evaluation For Learning Rate-0.001.

Input Channel	Model	Epochs	Batch Size	Adam Learning Rate – 0.001					
				Accuracy	Validation Accuracy	Precision	Recall	F1 Score	ROC
1	3 Layers DNN	200	32	71.32%	81.77%	87.40%	84.09%	85.71%	80.77%
1	3 Layers DNN	200	64	93.46%	80.79%	85.49%	84.84%	85.17%	79.04%
1	3 Layers DNN	200	128	93.42%	87.68%	93.12%	88.40%	90.70%	87.27%
1	3 Layers DNN	200	512	77.33%	82.27%	81.37%	92.91%	86.76%	78.69%
1	2 Layers DNN	200	32	69.43%	57.14%	57.14%	100%	72.72%	50%
1	2 Layers DNN	200	64	70.21%	66.50%	100%	49.25%	66%	74.62%
1	2 Layers DNN	200	128	97.91%	91.13%	89.78%	96.85%	93.18%	89.21%
1	2 Layers DNN	200	512	93.43%	86.70%	93.96%	84.49%	88.97%	87.51%
1	1 Layer DNN	200	32	98.71%	87.68%	95.49%	84.12%	89.45%	88.81%
1	1 Layer DNN	200	64	99.45%	92.61%	91.66%	97.77%	94.62%	90.06%
1	1 Layer DNN	200	128	99.64%	82.27%	80.48%	97.05%	88%	74.64%
1	1 Layer DNN	200	512	99.58%	83.25%	88.73%	87.50%	88.11%	80.19%
1	RNN-LSTM	100	32	99.93%	94.12%	97.32%	96.46%	96.88%	95.55%
1	**RNN-LSTM**	**100**	**64**	**99.73%**	**96.08%**	**96.03%**	**97%**	**96.51%**	**95.60%**
1	RNN-LSTM	100	128	94.50%	95.10%	98.96%	86.48%	92.30%	92.38%
1	RNN-LSTM	100	512	63.37%	64.71%	60.35%	100%	75.27%	50%
1	3 Layers CNN	30	32	99.63%	94.12%	94.28%	93.73%	93.96%	93.73%
1	3 Layers CNN	30	64	99.75%	81.37%	84.67%	83.89%	81.35%	83.89%
1	3 Layers CNN	30	128	99.52%	64.71%	75.67%	71.87%	64.36%	71.87%
1	3 Layers CNN	30	512	100%	33.33%	16.66%	50%	25%	50%
1	2 Layers CNN	30	32	99.74%	94.12%	93.70%	93.70%	93.70%	93.70%
1	2 Layers CNN	30	64	100%	89.22%	85.89%	92.56%	87.77%	92.56%
1	2 Layers CNN	30	128	99.17%	48.04%	69.18%	61.59%	46.55%	61.59%
1	2 Layers CNN	30	512	99.26%	43.14%	21.56%	50%	30.13%	50%
1	1 Layer CNN	30	32	88.58%	98.55%	85.92%	86.36%	86.13%	86.36%
1	1 Layer CNN	30	64	99.68%	90.20%	89.51%	91.04%	89.94%	91.04%
1	1 Layer CNN	30	128	98.75%	47.06%	67.46%	63.03%	46.54%	63.01%
1	1 Layer CNN	30	512	92.33%	41.18%	20.58%	50%	29.16%	50%
3	3 Layers CNN	30	32	61.61%	47.92%	23.95%	50%	32.39%	50%
3	3 Layers CNN	30	64	56.84%	47.92%	23.95%	50%	32.39%	50%
3	3 Layers CNN	30	128	56.68%	41.67%	20.83%	50%	29.41%	50%
3	3 Layers CNN	30	512	58.33%	47.92%	23.95%	50%	32.39%	50%
3	2 Layers CNN	30	32	55.48%	50%	25%	50%	33.33%	50%
3	**2 Layers CNN**	**30**	**64**	**57.29%**	**60.42%**	**30.20%**	**50%**	**37.66%**	**50%**
3	2 Layers CNN	30	128	41.23%	43.75%	21.87%	50%	30.43%	50%
3	2 Layers CNN	30	512	55.73%	47.92%	23.95%	50%	32.39%	50%
3	1 Layer CNN	30	32	54.46%	68.75%	35.37%	50%	40.74%	50%
3	1 Layer CNN	30	64	57.75%	58.33%	29.16%	50%	36.84%	50%
3	1 Layer CNN	30	128	45.31%	45.83%	22.91%	50%	31.42%	50%
3	1 Layer CNN	30	512	56.77%	56.25%	28.12%	50%	36%	50%

model-B, total parameters has observed to be 45,469,386, out of which 45,469,322 are trainable parameters whereas 64 non-trainable.

D. VGG-16

A special kind of Convolution Neural Network is proposed by Simonyan *et al.* which is used in this paper [27]. VGG-16

TABLE 3. Evaluation of pre-trained networks.

Input Channel	Model	Epochs	Adam Learning Rate	Accuracy	Validation Accuracy	Precision	Recall	F1 Score	ROC
3	VGG-16	30	0.0001	55.07%	41.67%	20.83%	50%	29.41%	50%
3	VGG-16	30	0.001	49.74%	58.33%	20.83%	50%	29.41%	50%
3	ResNet-50	20	0.0001	56.31%	47.92%	26.04%	50%	34.24%	50%
3	ResNet-50	20	0.001	68.08%	54.17%	22.91%	50%	31.42%	50%

TABLE 4. SVM models evaluation.

Input Channel	SVM Kernel	Accuracy	Precision	Recall	F1 Score	ROC
1	Sigmoid	60.74%	67.70%	74.71%	71.03%	55.06%
1	Gaussian	78.51%	77%	92.77%	84.15%	74.27%
1	Polynomial	75.55%	71.55%	97.50%	82.53%	70.56%

has been sculptured and trained for 3 channels input, i.e. MFCCs, Spectrogram & Scalogram. VGG-16 is the combination of A to E ConvNets. ConvNet A consists of 8 convolution layers and 3 fully connected layers resulting in 11 weight layers, whereas E consists of 16 convolution layers and 3 fully connected layers resulting in 19 weight layers. This model is implemented through TensorFlow library. VGG-16 functional layer has been used, which is already pre-configured, followed by the dropout layer to avoid overfitting and batch normalization. Dense layers are used with the ReLU activation function and Softmax activation function in the output layer. This model is trained through 30 epochs with numerous configurations of parameters like Batch Size, Adam Learning Rate, etc, as shown in result's section in this paper. Adam Optimizer is used as an optimizer to this model while Binary Cross-Entropy as loss function. In this model, total parameters has observed to be 58,407,745 out of which 43,606,017 are trainable parameters, whereas 14,801,728 non-trainable.

E. ResNet-50

A special kind of Convolution Neural Network is proposed by K He *et al.* which is used in this paper [28]. ResNet-50 is sculptured and trained for 3 channels input i.e. MFCCs, Spectrogram & Scalogram. This model consists of 50 deep convolutional neural network followed by an output layer. This model is implemented through the TensorFlow library. This model is trained through 20 epochs with numerous configurations of parameters like Batch Size, Adam Learning Rate, etc as shown in result's section in this paper. Adam Optimizer is used as an optimizer to this model while Binary Cross-Entropy as loss function. In this model, total parameters has observed to be 23,589,761 out of which 23,536,641 are trainable parameters, whereas 53,120 non-trainable.

F. SVM

SVM can be used to solve classification problems irrespective of linearity and non-linearity. A non-linear transformation is used to uplift the training data into higher dimensions through non-linear mapping. An SVM with a non-linearity function has been shown in Eq-8.

$$\mathrm{f}(\mathrm{x}) = \mathbf{sign}\left(\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{L}} \boldsymbol{\alpha}_{\mathbf{i}}\mathbf{y}_{\mathbf{i}}\mathbf{K}(\mathbf{x}_{\mathbf{i}}, \mathbf{x}) + \mathbf{b}\right) \tag{8}$$

where K(X,Y) is kernel [29].

Decision Boundaries, also known as Hyperplane, have been used in SVM to support classifying data points. The Equation of Hyperplane has been shown in Eq-9.

$$\mathbf{g}(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\mathbf{x} + \mathbf{b} \tag{9}$$

where $\mathbf{w}^{\mathbf{T}}$ is the weight vector and b is scalar.

In this study, 3 separate kernels are used to classify between Carnatic and Hindustani Music, i.e. Polynomial, Sigmoid & Gaussian.

G. ADAM OPTIMIZER

In this study, Adam Optimizer is used in every model due to it's combining properties of AdaGrad and RMSProp. Other advantages of using Adam optimizer above other traditional optimizers are: memory efficient, easy to implement, handle highly noisy or sparse gradient easily, etc. Adam Learning rate have varied from 0.0001 to 0.001 in this study.

H. LOSS FUNCTION

Sparse categorical cross-entropy and Binary categorical cross-entropy are used in this study. Both of the loss functions are the special cases of cross entropy loss function. Therefore, using the same computational relationship, as shown in Eq-10.

$$\mathbf{J}(\mathbf{w}) = -\frac{\mathbf{1}}{\mathbf{N}}\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{N}}\left[\mathbf{y}_{\mathbf{i}}\mathbf{log}\left(\hat{\mathbf{y}}_{\mathbf{i}}\right) + (\mathbf{1}-\mathbf{y}_{\mathbf{i}})\,\mathbf{log}\left(\mathbf{1}-\hat{\mathbf{y}}_{\mathbf{i}}\right)\right] \tag{10}$$

where $\mathbf{W}$ is the weight of the neural network, $\mathbf{y}_{\mathbf{i}}$ is true label, $\hat{\mathbf{y}}_{\mathbf{i}}$ is the predicted label.

I. ACTIVATION FUNCTION

ReLU and Softmax activating functions are used in this study. Rectified Linear Activation Function (ReLU) is a linear function that ranges from zero to infinity, as shown in Eq-11.

$$\mathbf{R}(\mathbf{z}) = \mathbf{max}(\mathbf{0}, \mathbf{z}) \tag{11}$$

where z is above or equal to 0.

Sigmoid Activation Function is used at the output layer for finding the maximum probable answer to classification problems, as shown in Eq-12.

$$\boldsymbol{\sigma}\left(\vec{\mathbf{z}}\right)_{\mathbf{i}} = \frac{\mathbf{e}^{\mathbf{z}_{\mathbf{i}}}}{\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{K}}\mathbf{e}^{\mathbf{z}_{\mathbf{j}}}} \tag{12}$$

where $\boldsymbol{\sigma}$ is softmax, $\vec{\mathbf{z}}$ is vector, K is the classes number.

J. PRECISION

Precision is used as one of the evaluation metrics in this study. Precision gives the insights of positive identification's proportion which are actually veracious, as shown in Eq-13.

$$\mathbf{P}=\frac{TP}{TP+FP} \tag{13}$$

where TP is True Positive and FP is False Positive.

K. RECALL

Recall is used as one of the evaluation metrics in this study. Recall gives the insights of veraciously identified actual positive's proportion, as shown in Eq-14.

$$\mathbf{R}=\frac{TP}{TP+FN} \tag{14}$$

where TP is True Positive and FP is False Positive.

L. F1 SCORE

F1 Score is used as one of the evaluation metrics in this study. F1 score fetches an equilibrium among precision and recall, as shown in Eq-15.

$$F_1 = 2*\frac{\mathbf{P}*\mathbf{R}}{\mathbf{P}+\mathbf{R}} \tag{15}$$

where P is precision and R is recall.

M. RECEIVER OPERATOR CHARACTERISTIC (ROC)

Receiver Operator Characteristic (ROC) is used as one of the evaluation metrics in this study. ROC is an evaluatic metric which yield's the probability over the curve between True Positve Rate (TPR) and False Positive Rate (FPR), as shown in Eq-16,17.

$$\mathbf{TPR} = \frac{TP}{TP+FN} \tag{16}$$

$$FPR = \frac{FP}{TN+FP} \tag{17}$$

where TP is True Positive, TN is True Negative, FP is False Positive and FN is False Negative.

VI. RESULTS

RNN-LSTM and 3 layer CNN performed the best with evaluation metrices i.e Accuracy, Validation Accuracy, Precision, Recall, F1 score & ROC as 98.53%, 96.08%, 95.70%, 95.70% while 3 layer CNN as 99.73%, 96.08%, 96.03%, 97%, 96.51%, 95.60%, respectively using 1 input channel. Loss & Validation Loss have observed as 0.0046 & 0.3056 for RNN-LSTM while 3 Layer CNN as 0.1048 & 0.1940. 2-Layers & 3 Layers CNN with Adam Learning Rate- 0.0001 & 0.001 and Batch Size 512 & 64 surprisingly best performed the same with evaluation metrices i.e Accuracy, Validation Accuracy, Precision, Recall, F1 score & ROC as 57.29%, 60.42%, 30.20%, 50%, 37.66%, 50%, respectively using 3 input channel. Pre-trained models used in this study, i.e. ResNet-50, performed best with evaluation metrices i.e Accuracy, Validation Accuracy, Precision, Recall, F1 score & ROC as 68.08%, 54.17%, 22.91%, 50%, 31.42%, 50%, while for VGG-16 as 49.74%, 58.33%, 20.83%, 50%, 29.41%, 50% using 3 channels input. Loss & Validation Loss have been observed as 1.1492 & 1.6148 for ResNet-50 while 0.7589 & 0.6858 for VGG-16. Evaluation metrices i.e Accuracy, Validation Accuracy, Precision, Recall, F1 score & ROC for SVM has been observed as 78.51%, 77%, 92.77%, 84.15%, 74.27%

TABLE 5. Loss of neural networks.

Input Channel	Model	Batch Size	Epochs	Adam Learning Rate – 0.0001		Adam Learning Rate – 0.001	
				Loss	Validation Loss	Loss	Validation Loss
1	3 Layers DNN	32	200	1.0625	1.6210	0.6179	0.9561
1	3 Layers DNN	64	200	1.1219	2.0002	0.8988	1.3687
1	3 Layers DNN	128	200	1.1100	1.7621	1.1187	1.0875
1	3 Layers DNN	512	200	1.4308	1.8267	1.4253	2.2453
1	2 Layers DNN	32	200	1.0474	2.9661	0.9221	0.5714
1	2 Layers DNN	64	200	1.0292	1.8105	1.2423	2.1788
1	2 Layers DNN	128	200	1.1454	2.1562	0.9608	2.1854
1	2 Layers DNN	512	200	1.2811	4.7790	1.2811	4.7790
1	1 Layer DNN	32	200	0.8036	2.3571	0.5273	1.2434
1	1 Layer DNN	64	200	0.7730	2.7429	0.5831	1.8691
1	1 Layer DNN	128	200	0.9718	5.7310	0.7441	11.8691
1	1 Layer DNN	512	200	0.8088	3.7081	0.9709	35.9172
1	RNN-LSTM	32	100	0.0814	0.1865	0.1202	0.2586
1	**RNN-LSTM**	**64**	**100**	0.1666	0.1676	**0.0046**	**0.3056**
1	RNN-LSTM	128	100	0.3116	0.3214	0.0658	0.1410
1	RNN-LSTM	512	100	0.4411	0.4523	0.3243	0.3386
1	3 Layers CNN	32	30	0.0682	0.2138	0.0124	0.2065
1	**3 Layers CNN**	**64**	**30**	**0.1048**	**0.1940**	0.0096	0.5874
1	3 Layers CNN	128	30	0.2081	0.9494	0.0141	1.5064
1	3 Layers CNN	512	30	0.9878	2.4906	0.0084	12.0540
1	2 Layers CNN	32	30	0.0127	0.3455	0.0119	0.2750
1	2 Layers CNN	64	30	0.0205	0.2515	0.0117	0.2686
1	2 Layers CNN	128	30	0.0674	1.9545	0.0253	2.3188
1	2 Layers CNN	512	30	0.1485	2.8651	0.0258	8.3208
1	1 Layer CNN	32	30	0.0710	0.2816	0.0445	0.5300
1	1 Layer CNN	64	30	0.0455	0.3737	0.0140	0.3958
1	1 Layer CNN	128	30	0.0619	1.6848	0.0399	5.2801
1	1 Layer CNN	512	30	0.2111	8.4889	0.4864	23.0315
3	3 Layers CNN	32	30	2.2750	2.2743	2.0389	2.0331
3	3 Layers CNN	64	30	2.2886	2.2883	2.1663	2.1628
3	3 Layers CNN	128	30	2.2933	2.2930	2.2849	2.2845
3	3 Layers CNN	512	30	2.2979	2.2978	2.2565	2.2549
3	2 Layers CNN	32	30	2.2750	2.2742	2.0391	2.0327
3	2 Layers CNN	64	30	2.2887	2.2883	2.1664	2.1628
3	2 Layers CNN	128	30	2.2933	2.2930	2.2111	2.2083
3	2 Layers CNN	512	30	2.2569	2.2549	2.2565	2.2549
3	1 Layer CNN	32	30	2.2750	2.2743	2.0396	2.0329
3	1 Layer CNN	64	30	2.2886	2.2883	2.1660	2.1624
3	1 Layer CNN	128	30	2.2933	2.2930	2.2979	2.2978
3	1 Layer CNN	512	30	2.2979	2.2978	2.2565	2.2549

TABLE 6. Loss of pre-trained networks.

Input Channel	Model	Epochs	Adam Learning Rate	Loss	Validation Loss
3	VGG-16	30	0.0001	0.7271	0.7059
3	VGG-16	30	0.001	0.7589	0.6858
3	ResNet-50	20	0.0001	1.5466	1.4658
3	ResNet-50	20	0.0001	1.1492	1.6148

for Gaussian Kernel, 75.55%, 71.55%, 97.50%, 82.53%, 70.56% for Polynomial Kernel and 60.74%, 67.70%, 74.71%, 71.03%, 55.06%.

VII. CONCLUSION

This study primarily focused on classification between Carnatic and Hindustani Music through audio files using two broad feature extraction approaches. 1 channel input, i.e. MFCCs features, have observed to be more effective than 3 channel input, thus outperforming it with 96.08%. RNN-LSTM and 1 layer CNN has performed the best by yielding the same validation accuracy, i.e. 96.08%, but validation loss as 0.1356 & 0.1111 respectively for 1 input channel. This type of classification can open many insights of Indian music industry. Raga motifs are the foundation of melody in Indian classical music. Same raga can be redeveloped using different compositions and improvisation. This classification technique can also be used to check the similarity of the raga in different formations. The representation of the melody and their similarity characteristics are very essential for Indian classical music. Future works can be based on the humdrum-based sequences. The reinforcement learning models can be generated to predict the music segments by associating and considering Raga and using an action reward approach to classify it better using an intensive agent-based learning and reward approach. Further works will also focus on the melodic shape on tempo range which covers a wide performing tempo range in Indian classical concerts.

REFERENCES

[1] R. Sridhar and T. V. Geetha, "Swara indentification for south indian classical music," in *Proc. 9th Int. Conf. Inf. Technol. (ICIT)*, Dec. 2006, pp. 143–144.
[2] R. Sridhar and T. V. Geetha, "Music information retrieval of carnatic songs based on carnatic music singer identification," in *Proc. Int. Conf. Comput. Electr. Eng.*, Dec. 2008, pp. 407–411.
[3] G. Pandey, C. Mishra, and P. Ipe, "TANSEN: A system for automatic raga identification," *IICAI*, Dec. 2003, pp. 1350–1363.
[4] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 1035–1047, Sep. 2005.
[5] A. Klapuri and M. Davy, *Signal Processing Methods for Music Transcription*. New York, NJ, USA: Springer-Verlag, 2006.
[6] P. Chordia, "Automatic raag classification of pitch-tracked performances using pitch-class and pitch-class dyad distributions," in *Proc. ICMC*, 2006, pp. 1–7.
[7] G. E. Poliner, D. P. W. Ellis, A. F. Ehmann, E. Gomez, S. Streich, and B. Ong, "Melody transcription from music audio: Approaches and evaluation," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 4, pp. 1247–1256, May 2007.
[8] S. Samsekai Manjabhat, S. G. Koolagudi, K. S. Rao, and P. B. Ramteke, "Raga and tonic identification in carnatic music," *J. New Music Res.*, vol. 46, no. 3, pp. 229–245, Jul. 2017.
[9] *Theory of Indian Music*, Pankaj, New Delhi, India, 1999.
[10] S. Shetty and S. Hegde, "Automatic classification of carnatic music instruments using MFCC and LPC," in *Data Management, Analytics and Innovation*. Singapore: Springer, 2020, pp. 463-474.
[11] H. Mukherjee, S. M. Obaidullah, S. Phadikar, and K. Roy, "MISNA–A musical instrument segregation system from noisy audio with LPCC-S features and extreme learning," *Multimedia Tools Appl.*, vol. 77, no. 21, pp. 27997–28022, Nov. 2018.
[12] H. G. Ranjani and T. V. Sreenivas, "Multi-instrument detection in polyphonic music using Gaussian mixture based factorial HMM," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 191–195.
[13] X. He and X. Zhou, "Audio classification by hybrid support vector machine/hidden Markov model," *World J. Model. Simul.*, vol. 1, no. 1, pp. 56–59, 2005.
[14] T. P. Vinutha and P. Rao, "Audio segmentation of hindustani music concert recordings," in *Proc. Int. Symp., Frontiers Res. Speech Music (FRSM)*, Mar. 2014, pp. 1–5.
[15] T. Virtanen and T. Heittola, "Interpolating hidden Markov model and its application to automatic instrument recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 49–52.
[16] H. Sundar, R. H. G., and T. V. Sreenivas, "Student's-t mixture model based multi-instrument recognition in polyphonic music," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 216–220.
[17] A. KSharma, A. Panwar, and P. Chakrabarti, "Analytical approach on indian classical raga measures by feature extraction with EM and naive bayes," *Int. J. Comput. Appl.*, vol. 107, no. 6, pp. 41–46, Dec. 2014.
[18] A. K. Sharma, A. Panwar, P. Chakrabarti, and S. Vishwakarma, "Categorization of ICMR Using feature extraction strategy and MIR with ensemble learning," *Procedia Comput. Sci.*, vol. 57, pp. 686–694, Apr. 2015.
[19] E. Patel and S. Chauhan, "Raag detection in music using supervised machine learning approach," *Int. J. Adv. Technol. Eng. Explor.*, vol. 4, no. 29, pp. 58–67, Jun. 2017.
[20] B. Kumaraswamy and P. P G, "Recognizing ragas of carnatic genre using advanced intelligence: A classification system for indian music," *Data Technol. Appl.*, vol. 54, no. 3, pp. 383–405, May 2020.
[21] G. C. Batista and W. L. S. Silva, "Application of support vector machines and two dimensional discrete cosine transform in speech automatic recognition," in *Proc. IEEE SAI Intell. Syst. Conf.*, Nov. 2015, pp. 687–691.
[22] Z. Ren, K. Qian, Z. Zhang, V. Pandit, A. Baird, and B. Schuller, "Deep scalogram representations for acoustic scene classification," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 3, pp. 662–669, May 2018.
[23] A. Copiaco, C. Ritz, S. Fasciani, and N. Abdulaziz, "Scalogram neural network activations with machine learning for domestic multi-channel audio classification," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT)*, Dec. 2019, pp. 1–6.
[24] G. Wolf, S. Mallat, and S. Shamma, "Audio source separation with time-frequency velocities," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2014, pp. 1–6.
[25] J. R. Deller, J. G. Proakis, and J. H. Hansen, *Discrete-Time Processing of Speech Signals*. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers, 2000
[26] A. K. Sharma and P. Ramani, "Rigorous data analysis and performance evaluation of Indian classical raga using RapidMiner," in *Soft Computing: Theories and Applications*. Singapore: Springer, 2018, pp. 97–106.
[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*. [Online]. Available: http://arxiv.org/abs/1512.03385
[29] G. Aggarwal and L. Singh, "Comparisons of speech parameterisation techniques for classification of intellectual disability using machine learning," *Int. J. Cognit. Informat. Natural Intell.*, vol. 14, no. 2, pp. 16–34, Apr. 2020.
[30] G. Aggarwal, R. Monga, and S. P. Gochhayat, "A novel hybrid PSO assisted optimization for classification of intellectual disability using speech signal," *Wireless Pers. Commun.*, vol. 113, no. 4, pp. 1955–1971, Aug. 2020.

[31] Y. Jia, X. Chen, J. Yu, L. Wang, Y. Xu, S. Liu, and Y. Wang, "Speaker recognition based on characteristic spectrograms and an improved self-organizing feature map neural network," *Complex Intell. Syst.*, vol. 4, pp. 1–9, Jun. 2020.
[32] A. Krishnaswamy, "Application of pitch tracking to South Indian classical music," in *Proc. Int. Conf. Multimedia Expo.*, vol. 3, Jul. 2003, p. 389.
[33] A. Klapuri, *Signal Processing Methods for the Automatic Transcription of Music*. Helsinki, Finland: Tampere Univ. Technol., 2004, pp. 6–35.

AKHILESH KUMAR SHARMA (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in CSE. He is currently working in Jaipur, India, as an Associate Professor. He is also the Founder of the Cognitive Intelligence Research (CIDCR) Laboratory, Jaipur. He has delivered keynotes in IITs, NITs, Vietnam, Thailand, Malaysia, Australia, Singapore, and China. He is also affiliated with IEEE, ACM, CSI, IUCEE, and MIR Labs, USA. He is a member and the Joint Secretory of the ACM Professional Chapter Jaipur and has won the Second Best Student Chapter Award from ACM India. He received various prestigious awards.

GAURAV AGGARWAL received the B.Tech. degree in instrumentation from the University Science Instrumentation Centre, Kurukshetra University, India, in 2006, the M.Tech. degree in computer science and engineering from the Department of Computer Science and Application, Kurukshetra University, in 2008, and the Ph.D. degree in speech signal processing and machine learning from The NorthCap University, Gurugram, India, in 2019. He is currently an Assistant Professor with the Department of Information Technology, Manipal University Jaipur, India. His research interests include signal processing, machine learning, and cognitive sciences. He has published more than 20 technical articles and reports in the above research areas at the national and international platform.

SACHIT BHARDWAJ is currently pursuing the bachelor's degree with the Department of Information Technology, Manipal University Jaipur. He has finished specialization courses in machine learning and deep learning from the University of Washington and DeepLearning.AI. He has gained his skills through various courses from Universities like the Higher School of Economics—National Research University Moscow, the University of California San Diego, the University of Michigan, and worked on various projects. He has published several technical articles and chapter in the area of machine learning. His research interests include machine learning, deep learning, computer vision, time series forecasting, signal processing, and natural language processing.

PRASUN CHAKRABARTI (Senior Member, IEEE) received the Ph.D. (Engg.) degree from Jadavpur University, in 2009. He is currently working as a Provost and an Institute Endowed Distinguished Senior Chair Professor with the Techno India NJR Institute of Technology. On various research assignments, he has visited Waseda University, Japan, in 2012, availing prestigious INSA-CICS travel grant; the University of Mauritius, in 2015; Nanyang Technological University, Singapore, in 2015, 2016, and 2019; Lincoln University College, Malaysia, in 2018; the National University of Singapore, in 2019; the Asian Institute of Technology, Bangkok, Thailand, in 2019; and ISI Delhi, in 2019. He has several publications, books, 30 granted patents, and two granted copyrights. He has successfully supervised 11 Ph.D. students. He is a fellow of IET (U.K.) and the Royal Society of Arts London. He is also an Honorary Fellow of the Iranian Neuroscience Society, IETE, ISRD (U.K.), IAER (London), AE (I), CET (I), and Nikhil Bharat Shiksha Parisad.

TULIKA CHAKRABARTI received the Ph.D. (Sc.) degree from the Indian Institute of Chemical Biology, Jadavpur University, in 2013. She is currently working as an Assistant Professor (Senior Grade) with Sir Padampat Singhania University, Udaipur. She has visited the TLI-AP, National University of Singapore; Nanyang Technological University, Singapore; Lincoln University College, Malaysia; and the Asian Institute of Technology, Bangkok, on several academic assignments. She has several publications, books, 27 granted international patents, and two granted copyrights to her credit. She is also a national merit scholarship holder in both 10th and 12th grade. She is a fellow of the International Society for Development and Sustainability, Japan, and an Honorary Fellow of the Iranian Neuroscience Society, Iran; the Royal Society of Arts, London; and Nikhil Bharat Shiksha Parisad.

JEMAL H. ABAWAJY (Senior Member, IEEE) is currently a Full Professor with the Faculty of Science, Engineering, and Built Environment, Deakin University, Australia. His leadership is extensive spanning industrial, academic, and professional areas. He is also the Director of the Distribution System Security (DSS). He has been actively involved in the organization of more than 200 national and international conferences, including the chair, the general co-chair, the vice-chair, the best paper award chair, the publication chair, the session chair, and a program committee member. He is also actively involved in funded research supervising, a large number of Ph.D. students, postdoctoral researchers, research assistants, and visiting scholars, in the area of cloud computing, big data, network and system security, and e-health. He is the author/coauthor of five books and ten conference volumes, and more than 250 referenced articles in conferences, book chapters, and journals. He is a Senior Member of the IEEE Technical Committee on Scalable Computing (TCSC), the IEEE Technical Committee on Dependable Computing and Fault Tolerance, and the IEEE Communication Society. He served on the editorial board for numerous international journals.

SIDDHARTHA BHATTACHARYYA (Senior Member, IEEE) received the bachelor's degree in physics and the bachelor's and master's degrees in optics and optoelectronics from the University of Calcutta, Kolkata, India, in 1995, 1998, and 2000, respectively, and the Ph.D. degree in computer science and engineering from Jadavpur University, Kolkata, in 2008. He is currently serving as the Principal of Rajnagar Mahavidyalaya, Birbhum, India. Prior to this, he was a Professor with the Department of Computer Science and Engineering, Christ University, Bengaluru, India. He served as the Principal for the RCC Institute of Information Technology, Kolkata. He served as a Senior Research Scientist for the Faculty of Electrical Engineering and Computer Science, VSB-Technical University of Ostrava, Ostrava, Czech Republic. He has coauthored six books, co-edited 76 books, and authored or coauthored more than 300 research publications in international journals and conference proceedings. He holds four patents. His research interests include soft computing, pattern recognition, multimedia data processing, hybrid intelligence, social networks, and quantum computing. He is also a full Foreign Member of the Russian Academy of Natural Sciences.

RICHA MISHRA received the Ph.D. degree. She is currently pursuing the Short Term Post-Ph.D. degree with Deakin University, Australia, under the supervision of Prof. Jemal H. Abawajy. She is also doing the Post-Ph.D. Pilot Research Physical Project under a Professor and the Director of the Cyber Systems Laboratory, Deakin University. She has 13 years of comprehensive experience in the field of academics, training, and counseling in universities and NGO's, along with strategic consulting in behavior and usage of technology. Besides being the Ph.D. holder with one international (Australian) and six filed/published Indian patents, she is an avid thinker along with teaching consciousness and strong academic sense. She is a Senior Member of the Iranian Neuroscience Society. She received the Research Fellowship, for the duration term of two years, in February 2021, at the Dana Brain Health Institute (DBHI) and DBHI/NBML, Iran.

ANIRBAN DAS is currently associated with the University of Engineering and Management, Kolkata, as a Full Professor in computer science and engineering. He is also a Visiting Scientist with the University of Malaya, Malaysia. He is also the honorary Vice President of the Scientific and Technical Research Association, Eurasia Research, USA. He is also a Visiting Faculty with the Central University of Jharkhand, Wall of Fame of myGov as "The Confederation of Elite" Ranchi. He has authored 11 books, 47 patents filed, and more than 50 research publications mostly in journals and conferences of international repute. His research interests include machine learning optimization techniques and blockchain. He is a fellow of the Royal Society, U.K.; Nikhil Bharat Siksha Parishad; IETE, India; and RSA, U.K. He is also the invitee as delegate from academics of several national bodies like, CII, NASSCOM, FICCI, and EDCN. He was nominated in Wall Academicians of IICDC powered by DST, AICTE, myGov, and Texas Instruments, in 2019. He is also the Innovation Ambessabor certified by MHRD Innovation Cell, Government of India.

HAIRULNIZAM MAHDIN (Member, IEEE) is currently an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia. He has been actively involved in many conferences internationally serving in various capacity, including the chair, the general co-chair, the vice-chair, the best paper award chair, the publication chair, the session chair, and a program committee member. His current research interests include data management, the IoT, and blockchain. He is a member of the Malaysia Board of Technologist (MBOT). He has also guest edited many special issue journals.