IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 18, 2020, accepted December 1, 2020, date of publication December 7, 2020,
date of current version December 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043142

A Globally Regularized Joint Neural Architecture for Music Classification

MOHSIN ASHRAF[1,2], GUOHUA GENG[1], XIAOFENG WANG[1], FAROOQ AHMAD[3], (Member, IEEE), AND FAZEEL ABID[1,4]

[1]School of Information Science and Technology, Northwest University, Xi'an 710069, China
[2]Department of CS and IT, The University of Lahore, Lahore 54590, Pakistan
[3]COMSATS University Islamabad, Lahore Campus, Lahore 54000, Pakistan
[4]Department of Information Systems, School of Business and Economics, University of Management and Technology (UMT), Lahore 54770, Pakistan

Corresponding author: Xiaofeng Wang (xfwang@nwu.edu.cn)

This work was supported in part by the National Key Research and Development Projects under Grant 2019YFC1521103, Grant 2020YFC1523300, Grant 2017YFB1402103, and Grant 2017YFB1002702; in part by the National Natural Science Foundation of China under Grant 61673319, Grant 61772421, and Grant 61731015; and in part by the Northwest University Education and Teaching Project under Grant YJG17013 and Grant YKC17021.

ABSTRACT Music classification is an essential application of Music Information Retrieval (MIR) in organizing extensive collections of music. The tasks to classify different music with reliable accuracy observed to be challenging. Most of these tasks employ handcrafted feature engineering to build a classifier, yet unable to identify the original characteristics of music. Several combinations of neural networks using convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been in consideration of many researchers. However, it has been noticed that the joint architecture of CNN and RNN suffers some problems due to batch normalization, which causes low accuracy and more training time. To handle these issues, the Global Layer Regularization (GLR) technique is proposed on the hybrid model of CNN and RNN using Mel-spectrograms for the evaluation of training and accuracy. Our experiments, with few hyper-parameters, improve performance on GTZAN and Free Music Achieve (FMA) datasets by achieving modest accuracy of 87.79% and 68.87% respectively. Empirically, our proposed model takes the advantages of spatiotemporal domain features and the global layer regularization technique to accomplish reliable accuracy as compared to the other state of art works.

INDEX TERMS Information retrieval, information systems, convolutional neural networks (CNNs), recurrent neural networks (RNNs), global layer regularization (GLR), music classification, spatiotemporal domain.

I. INTRODUCTION

The purpose of music classification is obvious due to the rapid increase in the volume of music in recent years. Music samples are continually growing, making it difficult to analyze and maintain the order of music databases manually. Automating the task of music classification and analysis can result better in music organization that has a significant role in Music Information Retrieval (MIR), Music Recommendation, and Online Access. However, music classification is a challenging task due to the presence of fuzzy nature among different music samples. Thus, music classification with reliable accuracy is worth investigating.

The associate editor coordinating the review of this manuscript and approving it for publication was Nadeem Iqbal.

The emergence of digital skills and complex models gained significant consideration of the researchers in music classification. Most of the music classification techniques use acoustic features of the audio signal for comparisons like rhythm, pitch, tonality, intensity, timbre, and Mel-frequency cepstrum coefficients (MFCCs). The handcrafted features, for instance, Local Binary Pattern (LBP), Robust Local Binary Pattern(RLBP), Rotational Invariant Co-occurrence (RIC), Local Ternary Pattern (LTP) have also performed well in the field of music [1]. Despite the extensive use of acoustic and handcrafted features, the Visual Domain Features using Mel-spectrograms are found to be similar to the human auditory system and ideal for the methodologies based on Deep learning [2].

Deep learning helps in designing the end-to-end systems for numerous applications. These systems are capable enough

for automatic feature extraction, free from feature biases, and beneficial in comparison with the traditional methods. Neural Networks such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are among the influential methods for music classification [3]–[5]. CNNs are favorable to record the spatial dependencies concerning feature domain [6], [7], whereas RNNs reasonably handle temporal dependencies in sequential data [8]. Many of these works performed music classification by using GTZAN [9] dataset that became the benchmark for music analysis. Later on, another dataset called Free Music Achieve (FMA) [10] has been developed for public use, whose small-subset (small -FMA) is similar to GTZAN.

The normalization techniques in these neural networks are sufficient in reducing the activities of the neurons to improve training time through computing mean and variance for fixed-size input [11]. However, the major shortcomings are the presence of feature biases and training complexity due to a fixed batch size, which is not appropriate to achieve consistent training performance [12]. In this paper, a new layer regularization is introduced, which is an effective and robust technique that is compatible with both CNN and RNN referred to Global Layer Regularization (GLR) to address the issues. This technique adequately removes the limitation of fixed batch sizes and discourages the use of traditional regularization techniques. Therefore, in this work, a global regularization term is used for the joint CNN-RNN architecture as an alternative for batch normalization to impact the influence of training samples as a whole for the improved results. Our method extracts features from the spatiotemporal domain that is independent of feature biases. Further, this technique is capable of computing statistics with layer regularization from the input where each neuron has its own adaptive bias and gain before non-linearity operation. Following are the main contributions of this work:

- A unified architecture that first performs convolution operations through CNN to perform feature extraction from Mel-spectrograms and makes it free from feature biases, and then the temporal aggregation of extracted features with RNN.
- A Global Layer Regularization (GLR) technique to compute the statistics from summed inputs directly to the hidden layer neurons without involving any new dependencies.
- A novel degree of compatibility of GLR and joint CNN-RNN architecture by operating the spatiotemporal features that result in better training and classification.

The rest of this paper is organized as follows: Section II contains related work while the proposed architecture is elaborated in section III. Section IV consists of data description and experimentation. Discussion is in section V. The paper is concluded in section VI.

II. RELATED WORK

Music Classification techniques are mostly built on feature descriptors extracted from the music files. The frequently adopted features with some nonconventional engineering methods are Zero-Cross Rate (ZRC), Spectral Roll Off (SR), Spectral Centroid (SC), Chroma, and MFCCs, as explained in [13]–[15]. Among these, MFCCs are considered an appropriate feature-set to provide strength for the classification of diverse genres. However, the Low-Level Audio Features of MFCCs affect model performance. Further, the handcrafted feature engineering has also been used for music classification in [16]. Although it was better compared to previous methodologies, it still required intense manual labeling and a robust understanding of signal processing methods. Another work demonstrated that the perceptual nature of features with the primary auditory cortex is not enough to provide discriminative strength for classification; therefore, it is desirable to combine generation and perception phenomena in representing music [17].

Currently, deep learning techniques are emerging as an alternative to handcrafted feature engineering due to automatic feature extraction, as introduced in [18]. These techniques perform automatic feature extraction for the music classification tasks [6], [19], [20].

Dieleman *et al.* [21] used the unsupervised feature extraction method by stacked Restricted Boltzmann Machine (RBM). They learned arguments to initialize the CNN perceptron for the classification of music genre but obtained low accuracy. Vishnupriya and Meenakshi [20] proposed a model built on CNNs for the classification of music genres by utilizing MFCCs for feature extraction and CNN for training and classification but unable to achieve a reliable outcome.

Costa *et al.* [2] successfully attained remarkable results and found to be more favorable in music classification by utilizing CNNs as compared to the approaches based on handcrafted features and SVM classifiers. Many methods using RNN variants such as Long-Short-Term-Memory (LSTM), constructed on segment features for classification by focusing on the sequential nature of the music, have been defined in [22], [23] to capture a long-range. Similarly, Soboh *et al.* [24] developed the RNNs model for the Arabic music classification and found satisfactory results in terms of training and classification accuracy.

Furthermore, Fulzele *et al.* [25] presented a joint model of LSTM and SVM for music classification and resulted in improved accuracy of prediction of the individual methods. Similarly, Choi *et al.* [26] used CRNN (Convolutional Recurrent Neural Network) for the music auto-tagging. They compared CRNN findings with three CNN structures when monitoring the number of parameters in terms of output and training time per sample. Adiyansjah *et al.* [27] used a hybrid form of convolutional and recurrent neural network for the music recommendation by considering the frequency as well as time-domain features. Their work indicated the users prefer recommendations for music genres compared with recommendations based on solely similarity. The works presented in [28]–[30] utilized joint neural networks composed of Recurrent and Convolution Neural Networks. Thus, many

works, as mentioned above, provided consistent performance, but were not adequate in training and density, which leads to computational overheads.

In neural networks, training through Stochastic Gradient Descent (SGD) has given significant results in computer vision [31] and processing speech [32], but due to data-intensive nature, it required massive time for training data. The training speed could be enhanced by splitting the dataset on different machines [33]. Having a complex communication setup and software also increases the concept of parallelization. In all the mentioned works, the authors utilized a technique in terms of normalization to reduce training time. This technique stores statistics separately for every hidden layer with static depth. However, RNNs often require summed input of dynamic sequence length and distinct computations for various time-steps. Empirically, this normalization is not suitable to apply on large distributed architectures where batches are small. It is beneficial to stabilize the dynamics of hidden states by using global layer regularization on the architecture based on CNN and RNN.

We propose a CNN-RNN based model for the music classification, which exploits CNN for automatic feature extraction from Mel-spectrogram that eliminates feature extraction biases while RNN to capture temporal aggregation distinctively by employing a global layer regularization that can significantly improve model training and classification accuracy.

III. THE PROPOSED GLOBALLY REGULARIZED CNN-RNN ARCHITECTURE

The proposed globally regularized CNN-RNN architecture aims to classify the music by capturing the spatiotemporal statistics directly from music files, as in FIGURE 1.

The workflow initiates to generate Mel-spectrograms by using the Librosa Library (a python package used for music analysis). These Mel-spectrograms are given to our joint architecture that consists of CNN and RNN. CNN transforms low-level features of Mel-spectrograms into high-level semantics by using some specific kernels. Just like images and textual data, music also consists of meaningful information in a graded structure that can be extracted by CNNs, as defined in [34]. The output of CNN is pooled to a feature map of a smaller dimension as an input to RNN, keeping an aptitude to process sequential data by utilizing its memory unit. By this ability, it examines the long term dependency and important patterns hidden in sequential data explained in [35]. A Global Layer Regularization (GLR) is applied to the combined model that reduces the dimensions by computing statistics across each feature and helps in finding optimal parameters quickly for training. No restriction on the size of the mini-batch is applied by this technique. It also prevents the summed input from being rescaled to a layer and allows the hidden state dynamics more stable by computing statistics along feature dimension rather than batch dimension. Our proposed architecture has succeeding steps as follows:

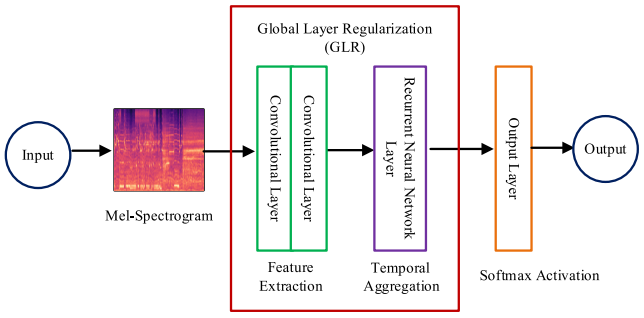


FIGURE 1. Proposed globally regularized CNN-RNN architecture.

1) Generating Mel-Spectrograms
2) CNN to Extract Features
3) RNN to Perform Temporal Aggregation
4) Global Layer Regularization

A. GENERATING MEL-SPECTROGRAMS

A music signal requires suitable representation to comprehend by the neural network architectures. A spectrogram is a 2-dimensional depiction of frequency facts over time that refers to the squared degree of Short-Time Fourier Transformation (STFT) of the audio signal [36]. Mathematically, an isolated STFT is calculated as in Eq. (1) :

$$STFT\{x(n)\}(m,\omega) = \sum_{n=-\infty}^{\infty} x[n]\omega[\text{n-m}]e^{-j\omega n} \quad (1)$$

where $x[n]$ refers to an input signal, and $\omega[n]$ is a window function [37].

Usually, the representation for the music data is a time-domain; however, distinct features are related to the frequency domain entirely for content-based audio classification [38]. Comparatively, Mel-Scale refers to the nonlinear transformation of the frequency scale that reflects how humans hear the sound. The Mel-scale is essentially a perceptual scale of frequencies that people consider to be equal in distance from each other. It is computed as a power spectral density $P(f, t)$ which is useful in testing the various points for time (t_i) and frequency (f_j) instants that are equally spaced. The frequency at Mel-scale is computed as:

$$Mel = 2,595 * log_{10}(1 + {}^{hertz}/_{700}) \quad (2)$$

The inverse can be calculated as:

$$Hertz = 700 * \left(10.0^{mel/2,595.0} - 1\right) \quad (3)$$

To acquire this representation, all the music samples are encoded with mp3 format, and each is having a duration of 30 seconds. We divide the datasets with the ratio 8:1:1 into training, validation, and test datasets respectively, and perform three iterations on both GTZAN and FMA datasets. For every iteration, each dataset contains an equal number of randomly shuffled samples. Shuffling data resolves the issue of reducing variance and assures that models remain general and less overfit [39]. In this work, we tend to use Mel-spectrograms to record temporal form for music analysis, as elaborated in [40].

B. CNNs TO EXTRACT FEATURES

CNNs have been developed specifically to examine the image data [41]–[43]. In CNNs, multiple layers act as feature extractor and can be used to balance the learning cost and extensive use [44]. CNNs use various filters to drive the final output feature map. Then an activation function is used to reduce the dimensions into a specific range. A typical framework of CNN [45] is demonstrated in FIGURE 2.

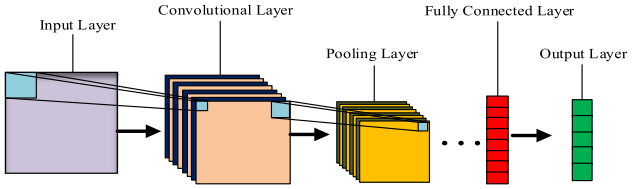


FIGURE 2. Block diagram of typical convolutional neural network.

In this framework, convolution operation represents the computations on element-wise multiplication of the specific kernel function with the input source. This filter traverses the whole source input to detect patterns. Consequently, CNNs automatically extract the real and natural patterns from the Mel-spectrograms [46] and requires fewer computations required in [47]. Each convolution layer uses kernels to learn a feature map that gives results to the next layer. Following Eq.(4) demonstrates to learn a feature map m_i^t with the use of specified filters:

$$m_i^t = f\left(V_{i;i+w-1} \odot W^t + b^t\right) \tag{4}$$

where W^t is the weight matrix, $V_{i;i+w-1}$ shows vectors, b is bias value and $\odot$ refers to the convolution operation. However, various kernels with different lengths and weight matrices are also considered to get an adequate feature map. The obtained result is transformed into a non-linear form by the use of the Rectified Linear Unit (ReLU) activation function.

The pooling layer in CNN architecture is used between the convolutional layers. This layer performs downsampling on the output given by the convolutional layer. It also creates nonoverlapping partitions for the output, and for each such partition, the maximum value is taken as a final result. So, we utilized max-pooling after each convolutional layer to reduce the dimensions of the input feature map.

C. RNN TO PERFORM TEMPORAL AGGREGATION

In our work, RNNs perform an essential task to accommodate sequential data through a single RNN layer. A directed cycle (DC) is created through the connections among artificial neurons. The unfolding structure of RNN can be seen in FIGURE 3. Multiple time steps share the same weights, such as U, V, and W. Sharing weights refer to capture temporal dynamic behavior in RNN.

To design music classification systems, the sequential and contextual nature of the data is of much significance. The RNN performs temporal aggregation on the feature map by keeping long term dependencies. So, we use the temporal

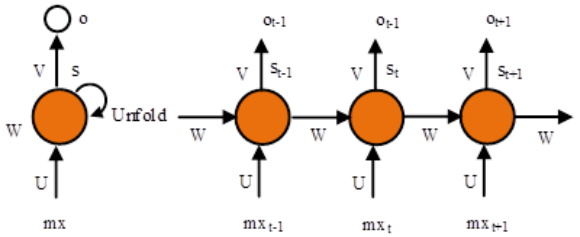


FIGURE 3. Unfolding structure of recurrent neural network [3].

structure in RNN by following a convolution that can achieve better performance in music classification without losing rich information in feature maps.

The statistics about the hidden state and output can be calculated as:

$$hs_t = f\left(U.m_i^t + Whs_{t-1} + b\right) \tag{5}$$

$$O_t = softmax\left(V.hs_t\right) \tag{6}$$

where m_i^t is input at time step (t), W is the weight and O_t refers to the output at the time step (t). Normally hidden state (hs_{t-1}) is initialized with all zeros. Further, regularization embedded with neural networks enhances the ability of neural networks that are committed to yielding an output in terms of training performance and classification accuracy.

D. GLOBAL LAYER REGULARIZATION (GLR)

One uncertainty in the neural network is related to overfitting that can be addressed with the use of regularization in the dense connections [48]. During the training, some nodes are kept with the probability p, and some are dropped with the probability *1-p* to reduce the network size. The batch normalization (BN) technique has been implemented to normalize the input data over a mini-batch [49], but some shortcomings have also been observed, such as fixed batch-size limitation and incompatible with the recurrent connections of RNN [12].

BN normalizes the summed input to every hidden unit over the training cases and rescales summed input according to the variances of the distribution of the data. In contrast by using GLR, every hidden unit of a layer shares the same normalization terms (μ, σ) but these terms are different for different training cases. This technique does not impose any limit on the size of the mini-batch. It also avoids rescaling the summed input to a layer that makes the hidden state dynamics more stable. Thus, we implement Global Layer Regularization (GLR) for normalizing the data by computing statistics in terms of improving the training performance by capturing the local and global parts of the music information. We used the term global to preserve the relative features of the music, in which the adaptive mechanism focuses on the combination of the local tasks and the relative parameters of the local tasks automatically configured. GLR computes statistics along the feature dimension to provide high compatibility to the joint unified architecture. In GLR, each feature is incorporated for computing the statistics independently with no batch size limit and causes to stabilize the hidden state dynamics [50]. Note that variations in the behavior of

one layer, particularly with ReLU units whose outputs can change by a lot, tend to cause highly correlated changes in the summed inputs to the next layer. This implies that by correcting the mean and variance of the summed inputs within each layer, the ''covariate change'' issue can be minimized. So, we calculated the GLR as:

$$\mu^l = \frac{1}{H}\sum_{i=1}^{H} a_i^l \tag{7}$$

$$\sigma^l = \sqrt{\frac{1}{H}\sum_{i=1}^{H}\left(a_i^l - \mu^l\right)^2} \tag{8}$$

where H represents the number of hidden units in a layer and a^l is the vector of summed input to the neurons. Additionally, Backpropagation through time (BPTT) is applied to deal with errors by incorporating different time steps [51]. This technique helps in acquiring gradients as in [52] by keeping resident data of hidden layers for various time steps.

SoftMax layer provides the range of probabilities to samples space. In our case, we used a SoftMax output layer with specific hidden units (neurons) to determine the probabilities for the concerned music files. The following expression determines the probability measures. The following expression determines the probability as:

$$P\left(i_j \mid k, \theta\right) = \frac{exp\left(x_j\left(k, \theta\right)\right)}{\sum_{1 \le i \le |X|} exp\left(x_i\left(k, \theta\right)\right)} \quad 1 \le j \le |X| \tag{9}$$

where $x_j(k, \theta)$ refers to the input vector with θ representing parameters. j refers to the output instance of the class, and X represents class space. The above expression calculates the exponential value of the input and then the sum of all the exponential terms for all input values. Then the ratio between the exponential of the input value and the aggregated sum of all the exponential terms refers to the final result of this activation function.

In our formulation, we build a single model based on CNN-RNN as joint learning and take global information as the regularization. We focused that both the local and global parts play important roles in capturing the music information. By dividing the target music files into local batches, most problems in music classification can be effectively detected and addressed. Meanwhile, the global term can preserve the relative features of the music, which improves the robustness of the model. From another point of view, the combination of the local tasks is considered as the adaptive mechanism, where the parameters of the local tasks are automatically adjusted by the global regularization term. The global regularization is completely data-driven, which well retains the overall structure of the music file. These local tasks adequately address the music classification problem as well and guide the update of the model.

IV. EXPERIMENTS

This section presents the experimentation of our proposed model that includes dataset description, baseline models, experimentation setup, and results and analysis.

A. DATASET DESCRIPTION

We have used two public datasets for the evaluation of our proposed model. Each of the datasets has some common and distinct features with associated labels. The first dataset is GTZAN which consists of 1000 samples with 10 different genres, each having 100 songs as ''blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock'' with a duration of 30 sec each as in [3], [5]. All files are mp3 and encoded with the sample rate of 22050 Hz with a size of 16 bit with the mono channel. The second dataset is Free Music Achieve (FMA), a suitable music dataset for the various evaluating tasks in MIR, such as searching, browsing, and organizing extensive music collections [10]. This dataset is available in different sizes, such as Full, Medium, and Small size depending upon the number of samples in it. With the definite objective, we selected a small-FMA dataset containing 8000 music files distributed over the 8 genres, such as rock, pop, instrumental, international, hip-hop, folk, experimental, and electronic, each having 1000 clips with the 30s each. All files are mp3 encoded with a sample rate of 44.1KHz and a sample size of 320 kb/s with stereo channels.

B. BASELINE MODELS

Only for the reference related to the CNN and RNN assessment, we first illustrate baseline models by using batch normalization performed on GTZAN and FMA datasets, as shown in TABLE 1. Then we tuned these baseline models by employing the Global Layer regularization (GLR) on the proposed architecture, as shown in TABLE 2.

TABLE 1. Baseline models with GTZAN and FMA datasets.

Baseline Models	Description
Convolution Neural Network (CNN) [2] Recurrent Neural Network (RNN) [23] CNN-RNN [4]	Using Batch Normalization

TABLE 2. Proposed model with GTZAN and FMA datasets.

Proposed Model	Dataset	Description
Convolutional Recurrent Neural Network (CNN-RNN)	GTZAN FMA	Using Global Layer Regularization (GLR) on a unified view of CNN-RNN model

C. EXPERIMENTATION SETUP

Our experimentations involved the assessments of the parameters of our proposed architecture. During the process of model building, we first transformed samples of both the datasets into its equivalent Mel-spectrograms by using the Librosa library. The outcome is scaled by the log function of

TABLE 3. Parameter configurations for building proposed model.

Parameters	Candidate Set	Optimized
Window Length	-	2048
Hop Length	-	512
Number of CNN layers	{1, 2, 3, 4}	2
Kernel Dimensions	{3,5,7, 9}	5
Number of Kernels	{32, 64, 128, 256}	128
Number of RNN layers	-	1
Number of hidden units in RNN	{64, 96, 128, 256}	96
Number of Epochs	{20, 30, 40, 50}	50
Learning Rate	{0.1, 0.01, 0.001, 0.0001}	0.001

TABLE 4. Accuracy of the baseline models using GTZAN dataset.

Iterations	CNN	RNN	CNN-RNN
Iteration-1	72.09	69.40	82.39
Iteration-2	88.59	76.53	80.00
Iteration-3	83.01	71.96	88.71

the audio files by setting the window length 2048, hop length 512, and obtained the shape of (640, 128). This technique determines loudness in decibels (dB) according to human perception.

Additionally, it is significant to decide network size and hyperparameter settings for the training of neural network models. Applying the same hyper-parameters for all datasets is not appropriate as various datasets have different impacts on different architectures. So, we performed a series of experiments to find the best parameters, including the number of CNN layers, kernel length, number of kernels, number of neurons (hidden units) in RNN, and learning rate. All optimized parameters from candidate sets can be seen in TABLE 3.

To determine the number of convolutional layers, we made the candidate set of 1, 2, 3, and 4 layers and observed the highest accuracy for layer size 2. Then dimension 5 of the convolutional filters was found to be better than 3,7,9. Similarly, 128 number of filters in each convolutional layer provided the highest score among the values of 32, 64, 96, and 128. For RNN, 96 hidden units performed well among the candidate set of 64, 96, 128, 256. Further, we tuned the proposed model by implementing the Global Layer Regularization (GLR) in favor of improving the training time. We used Adam optimizer, with 10 to 50 epochs, and a learning rate of 0.001 with categorical cross-entropy as a loss function.

D. RESULTS AND ANALYSIS

To assess the performance of our proposed model, we compared the accuracies of baseline models and the proposed model in Tables 4 to 6 and Figures 4 to 6. Our results revealed a remarkable performance in an average accuracy of 87.79% for the GTZAN and 68.87% for the FMA dataset, as in

TABLE 5. Accuracy of the baseline models using FMA dataset.

Iterations	CNN	RNN	CNN-RNN
Iteration-1	59.40	61.34	68.23
Iteration-2	44.23	51.50	57.44
Iteration-3	54.20	64.28	73.23

TABLE 6. Average accuracy of the proposed model with GTZAN and FMA datasets.

Proposed Model	Dataset	Iterations	Accuracy	Average
Convolutional Recurrent Neural Network (CNN-RNN) with GLR	GTZAN	Iteration-1	90.21	87.79
		Iteration-2	83.68	
		Iteration-3	89.48	
	FMA	Iteration-1	64.73	68.87
		Iteration-2	69.81	
		Iteration-3	72.07	

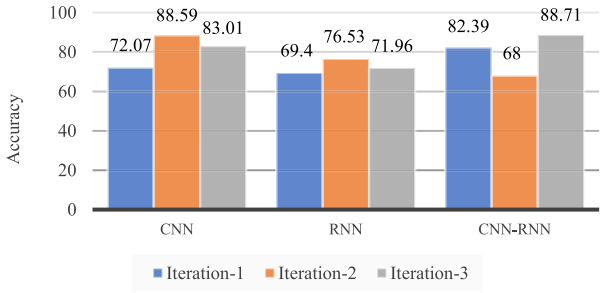


FIGURE 4. Accuracy of baseline models with GTZAN dataset.

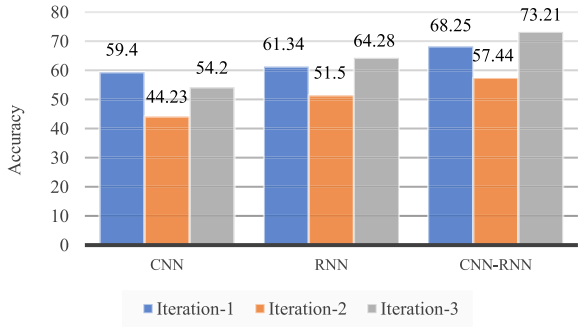


FIGURE 5. Accuracy of baseline model with FMA dataset.

TABLE 6. In the baseline models for the GTZAN dataset, we obtained the accuracies of the CNN model for three different iterations as 72.09%, 88.59%, and 83.01%. Similarly, we obtained 69.40%, 76.53%, and 71.96% for RNN. And in the last for CNN-RNN by using batch normalization technique, we found 82.39%, 80.00, and 88.71 as shown in TABLE 4 and *FIGURE 4*. Conversely, for the FMA dataset by using the CNN model, we obtained 59.40%, 44.23%, and 54.20%. For RNN, we observed 61.34%, 51.50%, and 64.28% accuracies. And further for CNN-RNN, we obtained 68.23%, 57.44%, 73.23% respectively for three iterations, as in TABLE 5 and *FIGURE 5*.

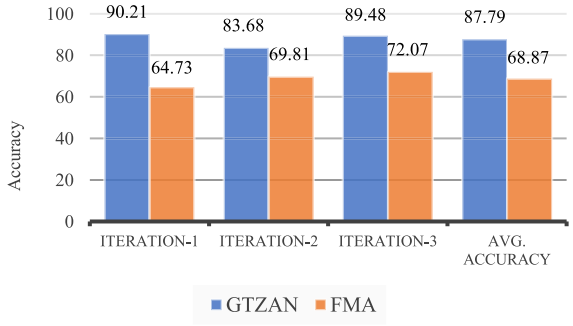


FIGURE 6. Accuracy of proposed model with GTZAN and FMA datasets.

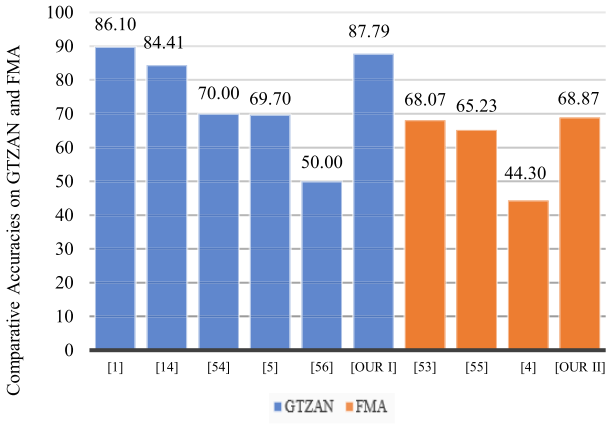


FIGURE 7. Comparison between proposed model and literature models.

To test the proposed model, we performed three iterations on both datasets. In observing individual accuracies, Iteration-1 attained the highest of 90.12% for the GTZAN, whereas Iteration-3 exhibited the top accuracy of 72.07% for FMA, as in TABLE 6 and FIGURE 6. We can also observe the modest accuracy of the proposed model compared with other models, as shown in TABLE 7 and FIGURE 7.

Implementing GLR on the proposed model computes the statistic along with the feature domain instead of the mini-batch. These computations are considered independent for the samples rather than making groups of all the elements together and computing the mean and variance. This mechanism considerably reduces the training complexities with lesser hyperparameters.

To determine how well the model is performing, we have also computed the precision, recall, and F1-score for the individual iterations of both datasets as shown in TABLE 8 & TABLE 9. For the GTZAN dataset, iteration-2 is showing the best value whereas iteration-3 is showing the poor result. Conversely, for the FMA dataset, iteration-3 is exhibiting the best performance where iteration-2 is lacking in performing well. The performance can be described by the fact that some music samples are highly distinct and some are overlapping. For example, the beats and rhythm of some music samples are quite different causing high values of precision, recall, and f1-score whereas some music samples have fairly similar beats and rhythm causing low values of performance measures [57].

TABLE 7. Comparison between the proposed GLR model and literature models without GLR.

Methods	GTZAN	FMA
L Nanni et al. [1]	86.10	-
Zlatintsi et al. [14]	84.41	-
Guo et al. [53]	-	68.07
M. Dong. [54]	70.00	-
Jakubec et al. [5]	69.70	-
Chen et al. [55]	-	65.23
Huang et al. [4]	-	44.30
Tang et al. [56]	50.00	-
Our Proposed	87.79	68.87

TABLE 8. Performance measures for each genre of GTZAN dataset.

Iterations	Precision	Recall	F1-Score
Iteration-1	0.89	0.86	0.87
Iteration-2	0.92	0.87	0.89
Iteration-3	0.86	0.86	0.86

TABLE 9. Performance measures for each genre of GTZAN dataset.

Iterations	Precision	Recall	F1-Score
Iteration-1	0.72	0.66	0.69
Iteration-2	0.67	0.62	0.64
Iteration-3	0.75	0.66	0.70

V. DISCUSSION

This section discusses the analysis of the proposed CNN-RNN model by integrating Global Layer Regularization (GLR) for music classification tasks. As mentioned earlier, in [13], [15], utilization of music features like Zero-Cross Rate, Spectral Centroid, MFCCs, and nonconventional engineering techniques for classification tasks that restrict model performance. Comparatively, this work holds spatiotemporal dependencies and normalizes the input feature map to excellent performance related to the complexity of training and accuracy. Interestingly, spectrograms generate Mel-spectrograms by using Mel-Scale. This Mel-Scale efficiently visualize the samples into the number of points equally spaced with times (t_i) and frequencies (f_j). Our work takes this advantage of the spatiotemporal domain and uses Mel-spectrograms for better analysis of music. To identify the patterns from Mel-spectrograms, CNN is the most appropriate choice for feature extraction through filters of certain lengths [34], [42] along with RNN significantly holding long term dependencies by the temporal aggregation to manage sequential data [6].

Multiple convolutional layers can be involved in a neural network that increases computations such as time, as mentioned in [31]. These computations depend upon the size of

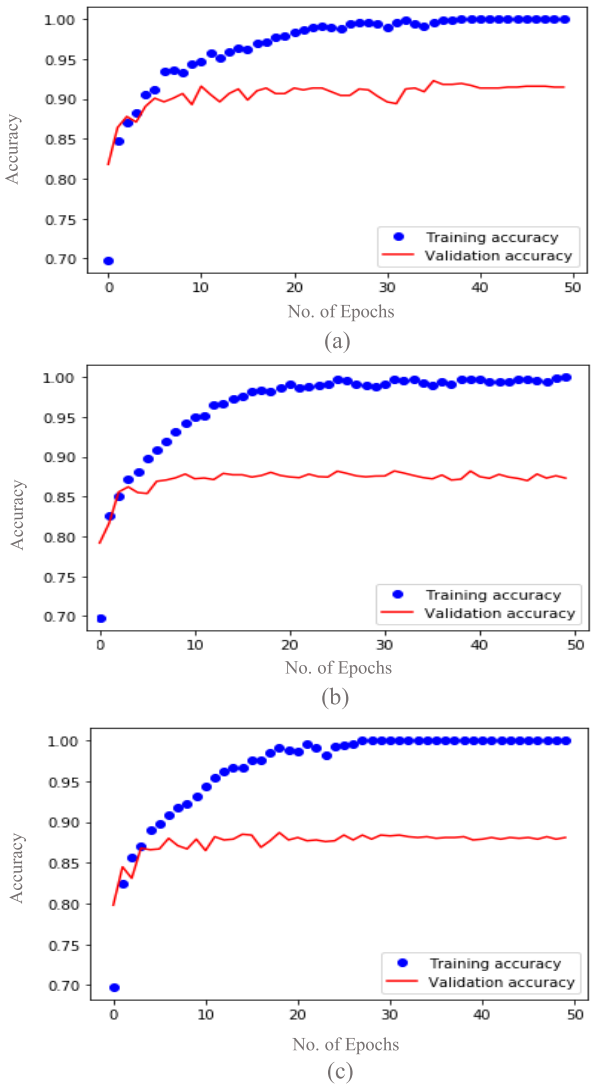


FIGURE 8. Training and validation accuracies of proposed model for three iterations (a)(b)(c) with GTZAN dataset.

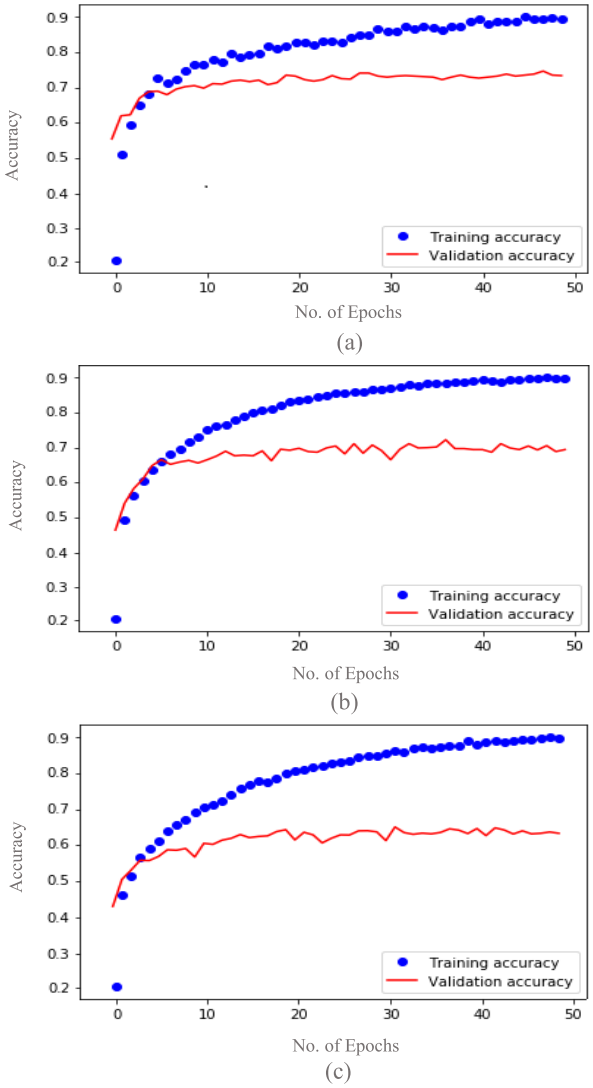


FIGURE 9. Training and validation accuracies of proposed model for three iterations (a)(b)(c) with FMA dataset.

the input image, number of layers, number of kernels, size of the kernels used in a network.

In this work, we used 2 CNN layers to extract features from the spatiotemporal domain that have an insignificant effect on network density. Furthermore, the explanation about the degree of complexity tends to inspect the depth of the network described in [58], and the width of the network described in [59], indicated in extending layers and filters. Conversely, computational complexity also depends upon the power of hardware that offers additional cost and memory concerns to the model. The following FIGURE 8 and FIGURE 9 show the accuracy of our model in terms of training and validation along the vertical axis concerning the increasing number of epochs along the horizontal axis.

However, time complexity depends upon the model instead of the real running time due to the cost of executions related to hardware [60]. The trainable parameters regulate the complexity of a neural network. We then accumulate the upsides of both the neural networks into a single model and implement New technique; Global Layer Regularization to improve performance as compared to the standard normalization techniques implemented in the various works [1], [14].

Some works have accumulated the benefits of joining neural networks such as [4], [19], [26], [27]. They suggested convolutional recurrent neural network (CRNN) models for the music auto-tagging and genre classification. The hybrid nature of neural networks has also been used in [3], [5] for music recognition tasks. They used batch normalization on the input feature map, but fail to perform well in the situation where statistics change for various time steps, as mentioned in [12]. To overcome this issue, we use the GLR (global layer regularization), which computes statistics along the feature dimension and provides high compatibility to the joint unified architecture. In GLR, each feature is incorporated for computing the statistics independently. This implies that each input uses a different normalization function with no batch size limit, and causes to stabilize the hidden state dynamics [50].

Our key findings during the experiments are that the observing patterns identified with Mel-spectrograms, utilizing GLR with CNN-RNN give comparable average accuracy as in TABLE 6 and TABLE 7. Moreover, we analyzed the causes of low accuracy for FMA as compared to GTZAN due to its complex nature. This can be addressed in future works by focusing on the increment of the number of samples or exploiting metadata associated with it.

VI. CONCLUSION

We have proposed a hybrid model of CNN and RNN to evaluate the training and accuracy of GTZAN and FMA datasets for the music classification tasks. The model generates Mel-spectrograms directly from music files, that are capable of retrieving the original characteristics of music and reduces the feature biases. Our work takes the advantages of both the Convolutional neural network (CNN) and the Recurrent neural network (RNN) to extract local features and perform temporal summarization of the extracted features. We introduce a novel Global Layer Regularization (GLR) technique in which every hidden unit of a layer shares the same normalization terms as well as independent of the size of the mini-batch. Further, to improve the training and accuracy, this technique effectively avoids the rescaling of the summed input to a layer that is useful to make the hidden state dynamics more stable. We performed a comparison between the proposed framework with baseline methods having conventional batch normalization techniques together with the setting of few parameters. Finally, our model obtained robust performance which signifies the success of the hybrid nature of the neural network towards feature extraction and temporal aggregation. In the future, diverse techniques of data augmentation using spectrograms can be employed by using deep learning models on substantial datasets for improving the training. We also aim to explore other datasets related to music classification tasks such as mood classification, artist, and instrument recognition.

REFERENCES

[1] L. Nanni, Y. M. G. Costa, D. R. Lucio, C. N. Silla, and S. Brahnam, "Combining visual and acoustic features for audio classification tasks," *Pattern Recognit. Lett.*, vol. 88, pp. 49–56, Mar. 2017, doi: 10.1016/j.patrec.2017.01.013.

[2] Y. M. G. Costa, L. S. Oliveira, and C. N. Silla, "An evaluation of convolutional neural networks for music classification using spectrograms," *Appl. Soft Comput.*, vol. 52, pp. 28–38, Mar. 2017.

[3] D. Bisharad and R. H. Laskar, "Music genre recognition using convolutional recurrent neural network architecture," *Expert Syst.*, vol. 36, no. 4, pp. 1–13, Aug. 2019, doi: 10.1111/exsy.12429.

[4] A. Huang and R. Wu, "Deep learning for music," 2016, *arXiv:1606.04930*. [Online]. Available: http://arxiv.org/abs/1606.04930

[5] M. Jakubec and M. Chmulik, "Automatic music genre recognition for in-car infotainment," *Transp. Res. Procedia*, vol. 40, pp. 1364–1371, Jan. 2019, doi: 10.1016/j.trpro.2019.07.189.

[6] S. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-end environmental sound classification using a 1D convolutional neural network," *Expert Syst. Appl.*, vol. 136, pp. 252–263, Dec. 2019.

[7] A. Murad and J.-Y. Pyun, "Deep recurrent neural networks for human activity recognition," *Sensors*, vol. 17, no. 11, p. 2556, Nov. 2017.

[8] W. Wu, F. Han, G. Song, and Z. Wang, "Music genre classification using independent recurrent neural network," in *Proc. Chin. Autom. Congr. (CAC)*, Nov. 2018, pp. 192–195.

[9] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 5, p. 293, 2002, doi: 10.1109/TSA.2002.800560.

[10] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A dataset for music analysis," 2016, *arXiv:1612.01840*. [Online]. Available: http://arxiv.org/abs/1612.01840

[11] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167

[12] D. Masters and C. Luschi, "Revisiting small batch training for deep neural networks," 2018, *arXiv:1804.07612*. [Online]. Available: http://arxiv.org/abs/1804.07612

[13] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "A survey of audio-based music classification and annotation," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 303–319, Apr. 2011.

[14] A. Zlatintsi and P. Maragos, "Comparison of different representations based on nonlinear features for music genre classification," in *Proc. 22nd Eur. Signal Process. Conf. (EUSIPCO)*, 2014, pp. 1547–1551.

[15] Z. Fu, G. Lu, K.-M. Ting, and D. Zhang, "On feature combination for music classification," in *Proc. Joint IAPR Int. Workshops Stat. Techn. Pattern Recognit. (SPR), Struct. Syntactic Pattern Recognit. (SSPR)*, 2010, pp. 453–462.

[16] Y. M. G. Costa, L. S. Oliveira, A. L. Koerich, F. Gouyon, and J. G. Martins, "Music genre classification using LBP textural features," *Signal Process.*, vol. 92, no. 11, pp. 2723–2737, Nov. 2012.

[17] B. L. Sturm and P. Noorzad, "On automatic music genre recognition by sparse representation classification using auditory temporal modulations," *Proc. Comput. Music Model. Retr.*, 2012, pp. 379–394.

[18] D. P. Kumar, B. J. Sowmya, and K. G. Srinivasa, "A comparative study of classifiers for music genre classification based on feature extractors," in *Proc. IEEE Distrib. Comput., VLSI, Electr. Circuits Robot. (DISCOVER)*, Aug. 2016, pp. 190–194.

[19] Z. Nasrullah and Y. Zhao, "Music artist classification with convolutional recurrent neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8851988.

[20] S. Vishnupriya and K. Meenakshi, "Automatic music genre classification using convolution neural network," in *Proc. Int. Conf. Comput. Commun. Informat. (ICCCI)*, Jan. 2018, pp. 1–4.

[21] S. Dieleman, P. Brakel, and B. Schrauwen, "Audio-based music classification with a pretrained convolutional network," in *Proc. 12th Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, 2011, pp. 669–674.

[22] J. Dai, S. Liang, W. Xue, C. Ni, and W. Liu, "Long short-term memory recurrent neural network based segment features for music genre classification," in *Proc. 10th Int. Symp. Chin. Spoken Lang. Process. (ISCSLP)*, Oct. 2016, pp. 1–5.

[23] G. Song, Z. Wang, F. Han, S. Ding, and M. A. Iqbal, "Music auto-tagging using deep recurrent neural networks," *Neurocomputing*, vol. 292, pp. 104–110, May 2018, doi: 10.1016/j.neucom.2018.02.076.

[24] L. Soboh, I. Elkabani, and Z. Osman, "Arabic cultural style based music classification," in *Proc. Int. Conf. New Trends Comput. Sci. (ICTCS)*, Oct. 2017, pp. 6–11.

[25] P. Fulzele, R. Singh, N. Kaushik, and K. Pandey, "A hybrid model for music genre classification using LSTM and SVM," in *Proc. 11th Int. Conf. Contemp. Comput. (IC)*, Aug. 2018, pp. 1–3, doi: 10.1109/IC3.2018.8530557.

[26] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Convolutional recurrent neural networks for music classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2392–2396, doi: 10.1109/ICASSP.2017.7952585.

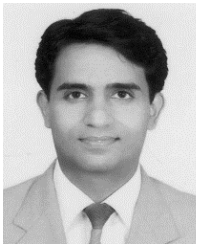
[27] A. A. S. Gunawan and D. Suhartono, "Music recommender system based on genre using convolutional recurrent neural networks," *Procedia Comput. Sci.*, vol. 157, pp. 99–109, Jan. 2019, doi: 10.1016/j.procs.2019.08.146.

[28] F. Abid, M. Alam, and A. Abid, "Representation of words over vectors in recurrent convolutional attention architecture for sentiment analysis," in *Proc. Int. Conf. Innov. Comput.*, Nov. 2019, pp. 1–8, doi: 10.1109/ICIC48496.2019.8966730.

[29] F. Abid, C. Li, and M. Alam, "Multi-source social media data sentiment analysis using bidirectional recurrent convolutional neural networks," *Comput. Commun.*, vol. 157, pp. 102–115, May 2020, doi: 10.1016/j.comcom.2020.04.002.

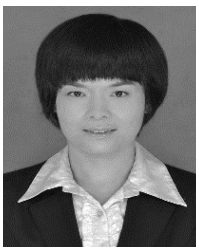
[30] M. Alam, F. Abid, C. Guangpei, and L. V. Yunrong, "Social media sentiment analysis through parallel dilated convolutional neural network for smart city applications," *Comput. Commun.*, vol. 154, pp. 129–137, Mar. 2020, doi: 10.1016/j.comcom.2020.02.044.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.
[32] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
[33] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, and Q. Le, "Large scale distributed deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1223–1231.
[34] L. Wu, Y. Wang, X. Li, and J. Gao, "Deep attention-based spatially recursive networks for fine-grained visual recognition," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1791–1802, May 2019.
[35] J. Jakubik, "Evaluation of gated recurrent neural networks in music classification tasks," in *Proc. Int. Conf. Inf. Syst. Archit. Technol.*, 2017, pp. 27–37.
[36] B. McFee, C. Raffel, D. Liang, D. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "Librosa: Audio and music signal analysis in Python," in *Proc. 14th Python Sci. Conf.*, vol. 8, 2015, pp. 18–25.
[37] W. J. Pielemeier, G. H. Wakefield, and M. H. Simoni, "Time-frequency analysis of musical signals," *Proc. IEEE*, vol. 84, no. 9, pp. 1216–1230, Sep. 1996.
[38] T. Li, M. Ogihara, and Q. Li, "A comparative study on content-based music genre classification," in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. informaion Retr. (SIGIR)*, 2003, pp. 282–289.
[39] D. J. Schad and R. Engbert, "The zoom lens of attention: Simulating shuffled versus normal text reading using the SWIFT model," *Vis. Cognition*, vol. 20, nos. 4–5, pp. 391–421, Apr. 2012.
[40] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," 2016, *arXiv:1606.00298*. [Online]. Available: http://arxiv.org/abs/1606.00298
[41] W. Zhang, W. Lei, X. Xu, and X. Xing, "Improved music genre classification with convolutional neural networks," in *Proc. Interspeech*, Sep. 2016, pp. 3304–3308.
[42] Y. Guo, Y. Liu, E. M. Bakker, Y. Guo, and M. S. Lew, "CNN-RNN: A large-scale hierarchical image classification framework," *Multimedia Tools Appl.*, vol. 77, no. 8, pp. 10251–10271, Apr. 2018.
[43] F. Abid, M. Alam, M. Yasir, and C. Li, "Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter," *Future Gener. Comput. Syst.*, vol. 95, pp. 292–308, Jun. 2019.
[44] K. Choi, G. Fazekas, and M. Sandler, "Explaining deep convolutional neural networks on music classification," 2016, *arXiv:1607.02444*. [Online]. Available: http://arxiv.org/abs/1607.02444
[45] Q. Shen, Z. Wang, and Y. Sun, "Sentiment analysis of movie reviews based on CNN-BLSTM," in *Proc. Int. Conf. Intell. Sci.*, 2017, pp. 164–171.
[46] T. L. H. Li, A. B. Chan, and A. H. W. Chun, "Automatic musical pattern feature extraction using convolutional neural network," in *Proc. Int. MultiConf. Eng. Comput. Sci. (IMECS)*, 2010, pp. 546–550.
[47] N. Karunakaran and A. Arya, "A scalable hybrid classifier for music genre classification using machine learning concepts and spark," in *Proc. Int. Conf. Intell. Auto. Syst. (ICoIAS)*, Mar. 2018, pp. 128–135, doi: 10.1109/ICoIAS.2018.8494161.
[48] D. R. Tobergte and S. Curtis, "Improving neural networks with dropout," *J. Chem. Inf. Model.*, vol. 5, no. 13, pp. 1689–1699, 2013.
[49] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2483–2493.
[50] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*. [Online]. Available: http://arxiv.org/abs/1607.06450
[51] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cogn. Model.*, vol. 5, no. 3, p. 1, 1988.
[52] P. J. Werbos, "Backpropagation through time: What it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, Oct. 1990.
[53] L. Guo, Z. Gu, and T. Liu. *Music Genre Classification via Machine Learning Category: Audio and Music*. Accessed: 2017. [Online]. Available: http://cs229.stanford.edu/proj2017/final-reports/5244969.pdf
[54] M. Dong, "Convolutional neural network achieves human-level accuracy in music genre classification," 2019, *arXiv:1802.09697*. [Online]. Available: https://arxiv.org/abs/1802.09697, doi: 10.32470/ccn.2018.1153-0.
[55] C. Chen. *SongNet: Real-Time Music Classification*. Accessed: Oct. 24, 2019. [Online]. Available: http://cs229.stanford.edu/proj2018/report/53.pdf
[56] C. P. Tang, K. L. Chui, Y. K. Yu, Z. Zeng, and K. H. Wong, "Music genre classification using a hierarchical long short term memory (LSTM) model," *Proc. SPIE*, vol. 10828, Jul. 2018, Art. no. 108281B, doi: 10.1117/12.2501763.
[57] J. de Sousa, E. T. Pereira, and L. R. Veloso, "A robust music genre classification approach for global and regional music datasets evaluation," in *Proc. IEEE Int. Conf. Digit. Signal Process. (DSP)*, vol. 26, Oct. 2016, pp. 109–113. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7868526/
[58] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[59] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[60] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5353–5360.



MOHSIN ASHRAF received the M.S. degree in computer science from the University of Central Punjab. He is currently pursuing the Ph.D. degree with Northwest University, Xi'an, China. He is also with the School of Information Science and Technology, where he is working on music information retrieval systems. His research interests include machine learning and deep learning applications, and pattern recognition.

GUOHUA GENG received the Ph.D. degree in computer software and theory from Northwest University. She is currently working as a Professor with the School of Information Science and Technology, Northwest University. Her current research interests include preservation of cultural relics, model processing, and intelligent processing.



XIAOFENG WANG received the Ph.D. degree in computer software and theory from Northwest University. She is currently working as an Associate Professor with the School of Information Science and Technology, Northwest University. Her research interests include pattern recognition, image processing, multimedia processing, music retrieval, and data mining.

FAROOQ AHMAD (Member, IEEE) received the Ph.D. degree in computer science from the Harbin Institute of Technology, China. He is currently working as an Associate Professor with COMSATS University Islamabad, Lahore, Pakistan. His research interests include formal modelling and simulation of systems, and Petri net theory and its applications.

FAZEEL ABID received the Ph.D. degree in information technology specialized in data mining and big data from Northwest University, Xi'an, China. He is currently working as an Assistant Professor with the University of Management and Technology (UMT). His research interests include computer science education and social data mining in the aspect of deep learning.