applied sciences

Article

A Hybrid CNN and RNN Variant Model for Music Classification

Mohsin Ashraf [1], Fazeel Abid [2], Ikram Ud Din [3], Jawad Rasheed [4], Mirsat Yesiltepe [5], Sook Fern Yeo [6,7,*] and Merve T. Ersoy [4]

[1] Department of Computer Science, University of Central Punjab, Lahore 54700, Pakistan
[2] Department of Information Systems, University of Management and Technology, Lahore 54700, Pakistan
[3] Department of Information Technology, University of Haripur, Haripur 22610, Pakistan
[4] Department of Software Engineering, Nisantasi University, Istanbul 34398, Turkey
[5] Department of Mathematical Engineering, Yildiz Technical University, Istanbul 34220, Turkey
[6] Faculty of Business, Multimedia University, Melaka 75450, Malaysia
[7] Department of Business Administration, Daffodil International University, Dhaka 1207, Bangladesh
* Correspondence: yeo.sook.fern@mmu.edu.my

Abstract: Music genre classification has a significant role in information retrieval for the organization of growing collections of music. It is challenging to classify music with reliable accuracy. Many methods have utilized handcrafted features to identify unique patterns but are still unable to determine the original music characteristics. Comparatively, music classification using deep learning models has been shown to be dynamic and effective. Among the many neural networks, the combination of a convolutional neural network (CNN) and variants of a recurrent neural network (RNN) has not been significantly considered. Additionally, addressing the flaws in the particular neural network classification model, this paper proposes a hybrid architecture of CNN and variants of RNN such as long short-term memory (LSTM), Bi-LSTM, gated recurrent unit (GRU), and Bi-GRU. We also compared the performance based on Mel-spectrogram and Mel-frequency cepstral coefficient (MFCC) features. Empirically, the proposed hybrid architecture of CNN and Bi-GRU using Mel-spectrogram achieved the best accuracy at 89.30%, whereas the hybridization of CNN and LSTM using MFCC achieved the best accuracy at 76.40%.

Keywords: music classification; music information retrieval; convolutional neural network; recurrent neural network; Mel-spectrogram

check for updates

Citation: Ashraf, M.; Abid, F.; Din, I.U.; Rasheed, J.; Yesiltepe, M.; Yeo, S.F.; Ersoy, M.T. A Hybrid CNN and RNN Variant Model for Music Classification. *Appl. Sci.* **2023**, *13*, 1476. https://doi.org/10.3390/app13031476

Academic Editor: David Ruano Ordás

Received: 5 November 2022
Revised: 26 December 2022
Accepted: 28 December 2022
Published: 22 January 2023

1. Introduction

The recognition and classification of music genres from audio data is a significant task known as "music classification". Due to the rapid increase in music archives, the goal of music classification is self-evident. A massive increase in the number music samples has been observed, making it challenging to retain the music order manually. Music classification and analysis can be improved by automating the task, which is essential in MIR, music recommendation, and online access. However, music classification is a difficult task caused by the presence of a fuzzy nature in various music samples. As a result, music classification with consistent accuracy is worth investigating.

The advent of digital abilities and sophisticated techniques has stimulated the interest of music classification researchers. Audio signal acoustics such as rhythm, pitch, tonality, intensity, timbre, and MFCCs are usually compared in music classification techniques. Local binary pattern (LBP) and local ternary pattern (LTP) are handcrafted features that have not performed well and introduced biases as in References [1,2]. The visual domain features based on Mel-spectrogram are similar to the human auditory system, and they are suited for deep-learning approaches despite using acoustical and handcrafted features [3].

Deep learning assists in the design of end-to-end systems for a wide range of applications. These systems can automatically extract features without biases and outperform traditional techniques. CNN and RNN are two of the most effective approaches to classifying music data [3,4], where CNN is better at recording spatial dependencies in feature domains [5,6], and RNN satisfactorily handles sequential data temporal dependencies [7,8]. In literature, many of the works classified GTZAN [9] dataset, which became the musical analysis benchmark. Further, a computational model that can automatically explore the digital content of the large and growing library of music is still lacking. Another reason to apply the CNN and variants of RNN jointly is to learn features incorporated with estimated parameters, in addition to using sequential modeling for neural networks that should be capable enough to analyze the inherently sequential nature of music files. The following are the main contributions of this work:

- Utilizing MFCC and Mel-spectrograms to determine which neural architecture is most effective for classifying music.
- For the classification of music, multiple hybridization such as CNN-LSTM, CNN-Bi-LSTM, CNN-GRU, and CNN-Bi-GRU are taken into consideration.
- Lastly, the performance of the various hybridizations for the various extracted features were also evaluated using the same musical dataset.

The remainder of this paper is structured as follows: Section 2 contains related work, while Section 3 elaborates on the methodology and proposed architecture. Section 4 is made up of data descriptions and experiments. Section 5 has the results of extracted features and discussion, and the work is concluded in Section 6.

2. Related Work

Music genre classification has a wide range of applications. These applications have used various methods of feature extraction and classifiers for the classification of music genre using machine- and deep-learning techniques as described in [10–12].

A machine-learning method based on support vector machine (SVM) and k-nearest neighbor (k-NN) was suggested in Reference [13]. This method was performed on the GTZAN dataset for the classification task. In order to extract the features from music samples, it used MFCC, and it obtained 64.4% and 77.78% accuracy for k-NN and SVM, respectively.

Feature extraction with statistical description was also performed in Reference [14]. This work extracted eight features that were used with different machine-learning algorithms and achieved an accuracy of 72%. Further, they also extended their approach by using deep-learning techniques such as CNN. They implemented their work in three steps—creating raw data, using short-time Fourier transform (STFT) (hop count = 1024, window size = 2048), and MFCC by employing 13 coefficients—and obtained 66% accuracy.

Similarly, multiple techniques for detecting and classifying GTZAN music datasets were compared in Reference [15], in which FFT and MFCC for the feature extraction is considered. They performed analysis using the machine-learning approaches “decision tree”, “k-NN,” and “RNN” as classifiers. By implementing RNN, they obtained the highest accuracy of 86%. Further, in Reference [16], another model was implemented in two parts. First, the extracted features from music files were arranged in a specific format to be used in two architectures. Second, the sum rule was used to merge the two models. To show the final result, the discrete posterior probabilities were jointly used. The GTZAN dataset was used for the experiments, and nine distinct features were extracted in order to analyze the music pattern. Every feature considered input to two models, and results were merged for the final prediction. By combining the SVM and LSTM, they achieved an accuracy of 89%.

To classify music, a model in Reference [17] considered eight characteristics of extracted audio files: beat periodicity, loudness, energy, speech, acoustic, valence, and danceability (discreet wavelet transformation, DWT). A neural network then took the input of these features. They performed the experiments on only two genres: classical and Sufi songs. Classical songs were predicted correctly with an accuracy of 87%, and the Sufi genre was predicted with an accuracy of 82%. Similarly, the study in Reference [18] used the GTZAN dataset with limited parameters and the global pooling approach to investigate a CNN-based network. However, they could only achieve a test accuracy of 70.60%, since they could not locate the temporal elements. The authors in Reference [19] also suggested the CNN design combined with a residual network. Still, due to a shortage of training examples, their experiments on the PMG dataset showed an accuracy of 86%. Additionally, a model suggested in [20] that attempted to classify the Free Music Achieve (FMA) dataset using CNN-based architecture showed limited performance, obtaining an accuracy rate of only 66.4%.

Due to the audio's sequential nature, the variants of RNN, such as LSTM and GRU, have been proposed for music classification in References [21,22]. In this work the tests were conducted on GTZAN, Emotify, Ballroom, and LastFM by utilizing Mel-spectrograms for feature extraction. This work also compared all datasets and found GRU to be better than LSTM with reliable accuracy. In another work, George Tzanetakis [9] used the timbral texture, rhythm, pitch content, and statistical pattern recognition from the GTZAN dataset as well as real-time music with the accuracy of 60%.

Owing to the immense advancement and effectiveness of neural networks in multiple classification tasks and also to the better forecasting results, a hybridization of CNN and variants of RNN is proposed. By utilizing two prominent features, Mel-spectrograms and MFCC in novel joint architecture are fed into the CNN layers and then to optimized variants of RNN such as LSTM, Bi-LSTM, GRU, and Bi-GRU, for performance evaluation and comparison using the GTZAN dataset.

3. Proposed Hybrid Methodology

This work aims to classify the music genres with the proposed hybrid neural architecture implemented with Mel-spectrograms and MFCC. The proposed hybrid methodology theoretical framework has been presented in a series of steps, including Dataset and Preprocessing, Feature Extraction, and Learning Algorithm. The flow of the proposed architecture, which consists of extracted features such as Mel-spectrogram and MFCC, incorporated with CNN and variants of RNN, is shown in Figure 1.

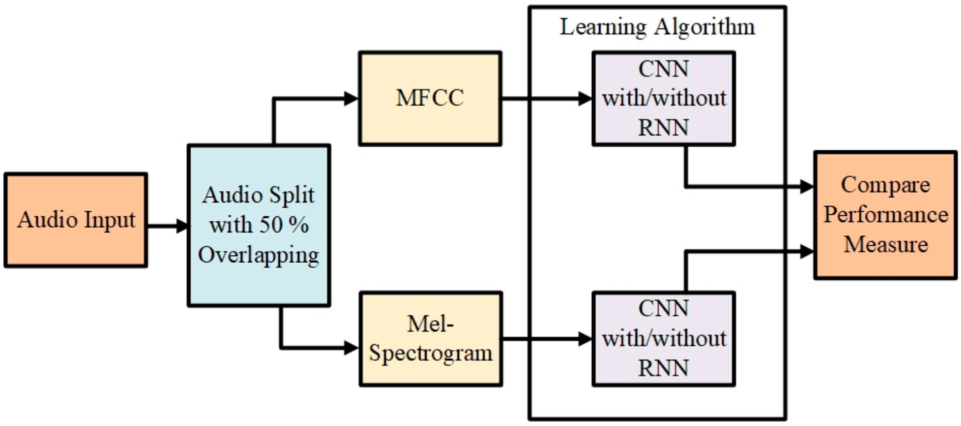

Figure 1. Proposed hybrid architecture with CNN and variants of RNN.

3.1. Dataset and Preprocessing

The experiments were carried out using GTZAN, which is freely available to the public [9]. The dataset contains 1000 music clips divided into 10 genres, each with 100 songs such as "blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock" with a duration of 30 s each as in Reference [23]. With a sample rate of 22,050 Hz, each clip has a size of 16 bits for the mono channel and is encoded in mp3 format. However, during preprocessing, a clip splitting process is used to converts a 30 s music clip into a 3 s duration to meticulously assess the proposed model.

3.2. Feature Extraction

In order to compare how well each method performed, each music clip used a different feature extraction method. These methods are Mel-spectrogram and MFCC. Mel-spectrogram represents an input 2D signal which is also a part of bandpass filters, over the spatiotemporal domain and a digital filter bank that is a subset of bandpass filters, where $X(t)$ refers to the input signal, n represents a filter bank analysis number, and $A(F)$ refers to an analysis filter, as shown in Figure 2. By utilizing this method, a signal can be divided into sub-bands containing part of the original frequency. It looks exactly like an analysis of the digital filter bank to the Mel filter bank in Figure 3. We take the music clip as input and perform a Hann windows operation in Mel-spectrogram processing. Following that, we perform FFT on each block, converting the time-based signal to a frequency-based signal, which is similar to STFT.

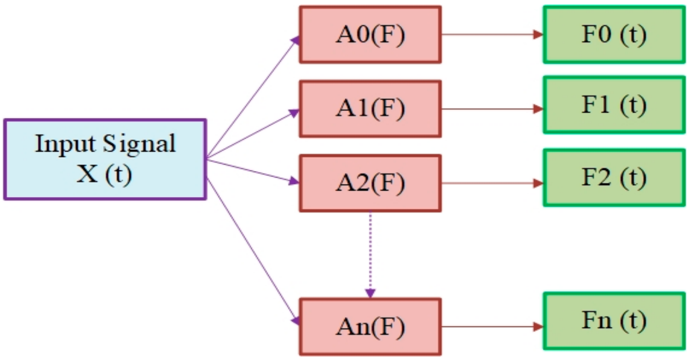

Figure 2. Analysis of digital filter bank.

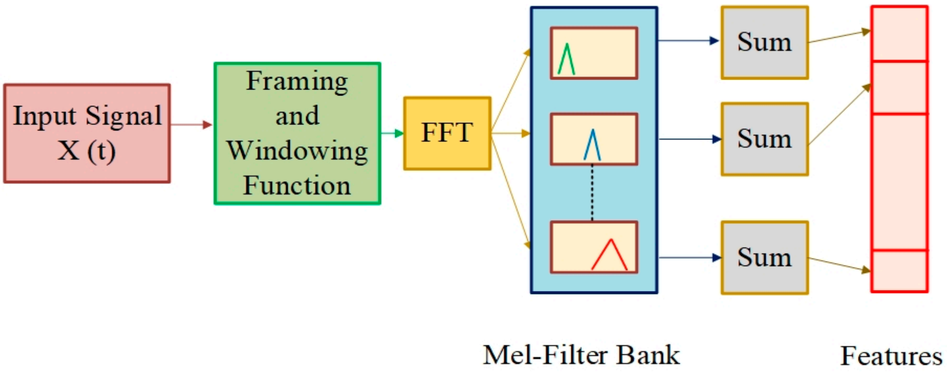

Figure 3. Evaluation of Mel-spectrogram.

Further, Mel-spaced filter bank, also known as analysis filter bank, is used to pass every frequency-based signal, as in Figure 3. In order to determine the bank energy, the product of the frequency and filter bank is calculated, and all coefficients are summed up. Mel-spectrogram vectors create every summed coefficient with n-agreed filters. Every frame of the signal gets a Mel-spectrogram of n-vector. Then, we obtain the resultant in a triangular graph, as shown in Figure 4. Different colors represent different filter bank analyses.

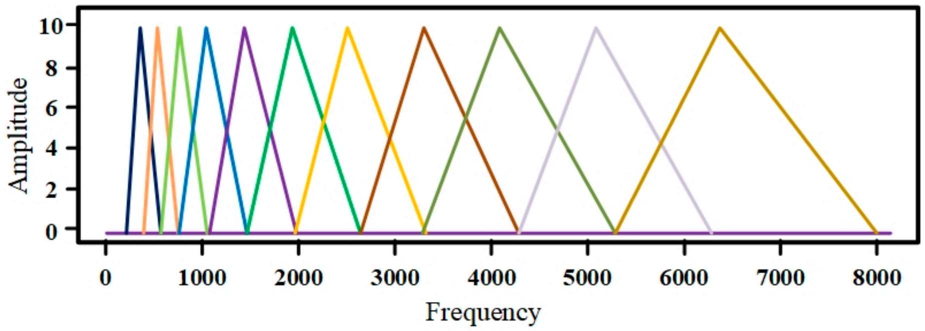

Figure 4. Mel-bank filter with *n* = 10.

A comparison can be made between the Mel-filter bank and filter band, as shown in Figures 3 and 4, respectively, so every triangle filter is considered a distinct block. For instance, the color of the first triangular block is blue, which is the first block. Further, the orange color is taken as the second block until the triangular part is the 'n' block. When Mel-spectrogram is functional to each clip with an FFT window size of 1024 and a hop length of 512, then every clip has dimensions (128,129), as shown in Figure 5.

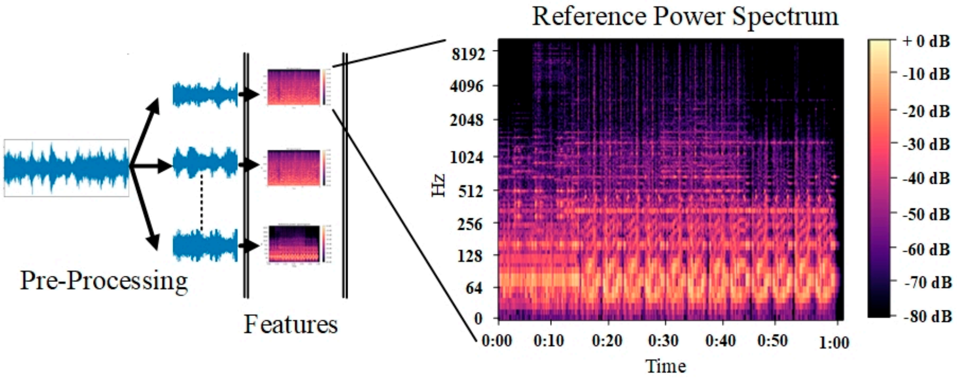

Figure 5. Mel-spectrogram for a piece of blue class music.

MFCC is an alternative form of audio representation after compressing frequency. We calculate the power log and choose 13 to 20 coefficients after performing DCT. The increasing coefficients represent more changes to the energy estimation and show lower amounts of received data. Most of the information gets lost, which is why one uses this technique. Furthermore, DCT is applied similarly to FFT, whose implementation of computation is easy, as is shown in Figure 6, and MFCC for the blues genre can be seen in Figure 7.

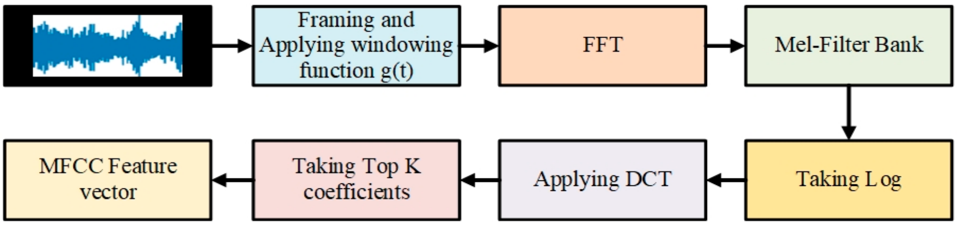
Framing and Applying windowing function g(t)
FFT
Mel-Filter Bank
Taking Log
Applying DCT
Taking Top K coefficients
MFCC Feature vector

Figure 6. Evaluation of MFCC.

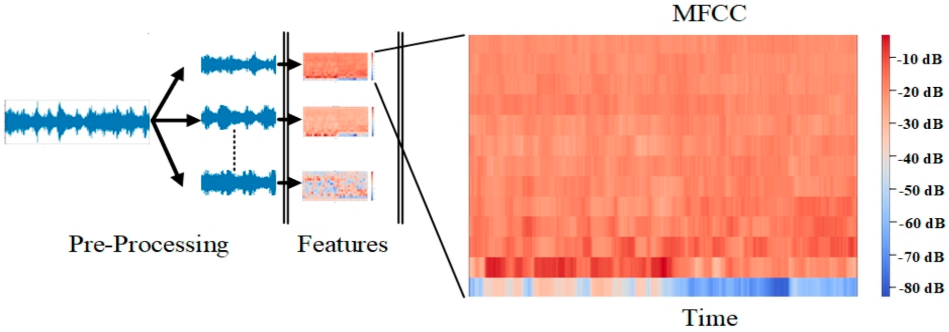

Figure 7. MFCC for a piece of blue class music.

3.3. Learning Algorithm

With an 8:1:1 ratio, we divided the dataset into three categories: training, test, and validation. Our model retained 50% of the previous data and all clips are linked to form a proper chain; the dataset within each genre can be shuffled without losing any information. Following training, the performance of validation samples can be evaluated after one epoch to determine generalized unknown data. The test data are evaluated after the performance of the training and validation data.

CNN has produced impressive results when analyzing image data. To determine the unique patterns for the classification task, all music features are considered image features. During convolution operation, different kernel filters are utilized. A method in [24] used CNN layers to extract features from Mel-spectrogram. As the computations of weight and bias for any hidden layer are diverse, similar values for the bias and weight are therefore considered to accumulate the hidden layers. Further, to address this issue, RNN and its variants play a significant role, using internal memory as mentioned [25–28]. Moreover, a CNN model has been trained both with and without RNN variants, and several hyperparameters have been adjusted throughout the training process.

4. Experiments

For the experiments, we used Spyder 3.3.2, which is integrated with several influential packages in the scientific Python stack, such as NumPy, SciPy, matplotlib, pandas, and other software. This system also includes evaluating the limits of the proposed model by changing parameters.

During the model-building process, music samples are transformed into Mel-spectrograms using the Librosa library. By increasing the window length to 2048 and the hop length to 512, the result becomes scalable by the log function of the music files, yielding the desired shape (640, 128). This technique is based on human perception in the loudness of decibels (dB). It is not advantageous to use the same hyper-parameters for all datasets, because different datasets impact different architectures. As a result, choosing the network size and hyperparameter settings is critical for neural network model training. We ran several experiments to determine the best parameters, including the number of CNN layers, kernel length, number of kernels, neurons (hidden units) in RNN variants, and learning rates. Table 1 displays all optimized parameters.

Table 1. Parameter configuration used to build the proposed architecture.

Parameters	Candidate Set	Optimized
Window Length	-	2048
Hop Length	-	512
Convolutional Block	{3, 5, 7}	5
Kernel Dimensions	{3,5,7, 9, 11}	5
Number of Kernels	{32, 64, 128, 256, 512}	128
Number of LSTM/B-LSTM/GRU/Bi-GRU	-	3
Number of hidden units in RNN Variants	{64, 96, 128, 256, 512}	128, 64
Number of Epochs	{30, 40, 50, 60}	50
Dropout	-	0.25
Window Length	{0.1, 0.01, 0.001, 0.0001}	0.001

In this proposed model, we created 5 layers of convolution block. Each layer included a different-sized kernel filter followed by a maximum 0.25 dropout. Each block layer contains a different size of convolution kernel, followed by a max-pooling layer with a 25% dropout. Afterward, we flatten them using a 1D array and the output layer. This model also consists of three layers of LSTM (or GRU) with two 128 LSTM or GRU units and one 64 LSTM or GRU unit. Further, this outcome is flattened into a 1D array and uses one dense layer followed by an output layer, as shown in Figure 8.

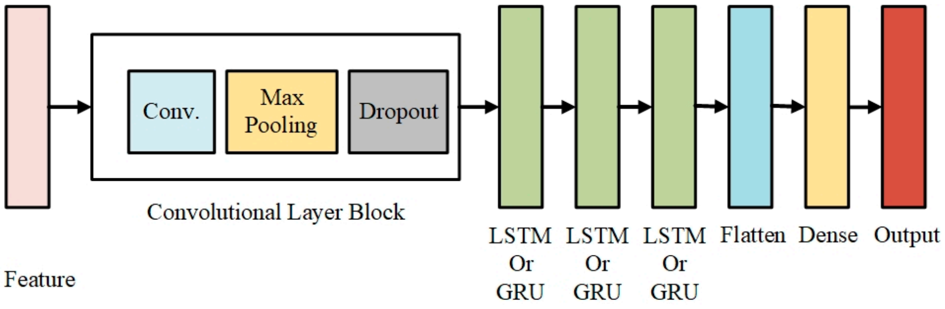

Figure 8. Proposed CNN and variants of RNN (LSTM/GRU).

Similarly, in Figure 9, we used one layer of LSTM or GRU extended with 128 units for CNN incorporated with Bi-LSTM or Bi-GRU. A Bi-LSTM or Bi-GRU was then extended with 128 units of one LSTM or GRU. Additionally, they were flattened into a 1D array followed by one dense layer and an output layer.

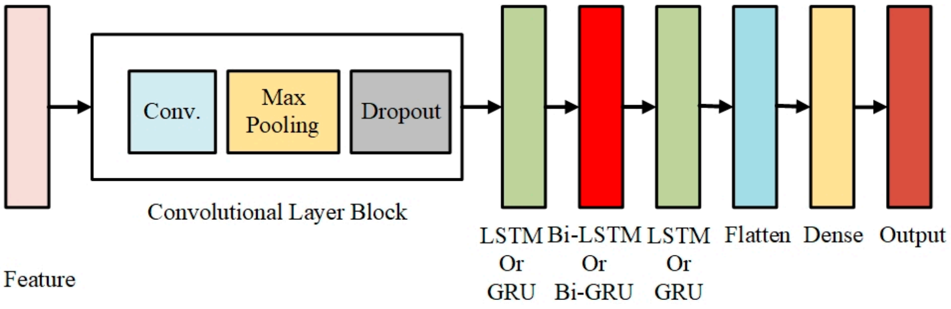

Figure 9. Proposed CNN and variants of RNN (Bi-LSTM/Bi-GRU) architecture.

5. Results and Discussion

We developed and compared the accuracies of four different combinations, including CNN+LSTM, CNN+Bi-LSTM, CNN+GRU, and CNN+Bi-GRU, using Mel-spectrograms and MFCC features, which are mentioned in Table 2 and Figure 10, to evaluate the performance of the proposed hybridization.

Table 2. Results of extracted features with proposed hybrid architecture.

Features	Model	Accuracy
Mel-Spectrogram	CNN+LSTM	87.20%
Mel-Spectrogram	CNN+Bi-LSTM	88.00%
Mel-Spectrogram	CNN+GRU	87.78%
Mel-Spectrogram	CNN+Bi-GRU	89.30%
MFCC	CNN+LSTM	76.40%
MFCC	CNN+Bi-LSTM	73.69%
MFCC	CNN+GRU	71.50%
MFCC	CNN+Bi-GRU	76.30%

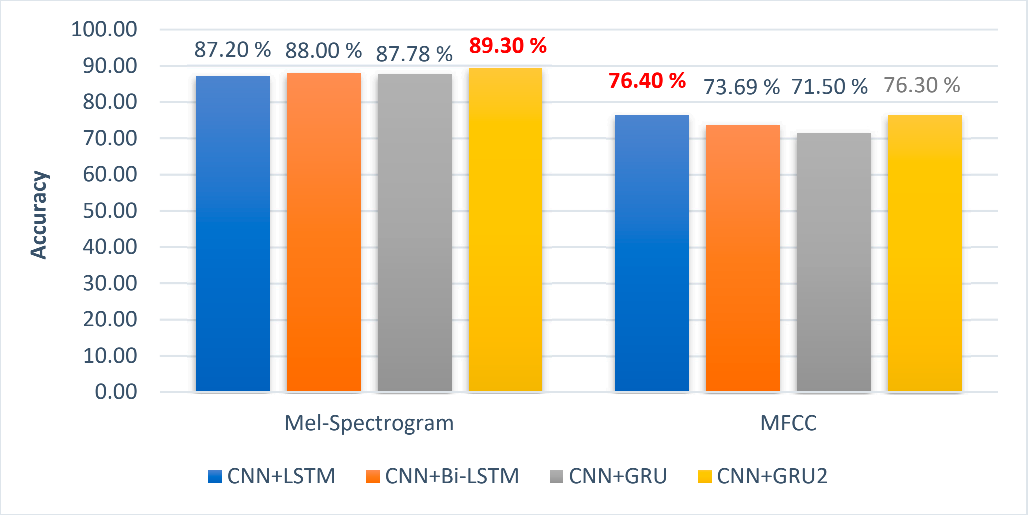

Figure 10. Results of extracted features with the proposed hybrid architecture.

Using Mel-spectrograms and MFCC features from the GTZAN dataset, we compared various combinations. With an accuracy of 89.30%, the combination of CNN+Bi-GRU performed the best for Mel-spectrograms. The CNN+LSTM combination model achieves an accuracy of 87.20%. On the other hand, we evaluated the proposed models' performance in terms of MFCC features. The combination of CNN+LSTM achieved the highest accuracy of 76.40%, while CNN+RNN achieved the lowest accuracy of 71.50%. Table 3 and Figure 11 compare the proposed models to the state-of-the-art models.

Table 3. Comparison between the proposed hybrid model and the other state-of-the-art models.

Method	Accuracy
George Tzanetakis [9]	61.00%
G. Sun et al. [20]	66.40%
A Heakl et al. [18]	70.60%
Nilesh M. et al. [13]	77.78%
Praseneet Fulzeele et al. [16]	89.00%
N. Farajzadeh [19]	86.00%
Pradeep Kumar D et al. [15]	86.00%
Jan Jakubik [21]	87.70%
Proposed Work	89.30%

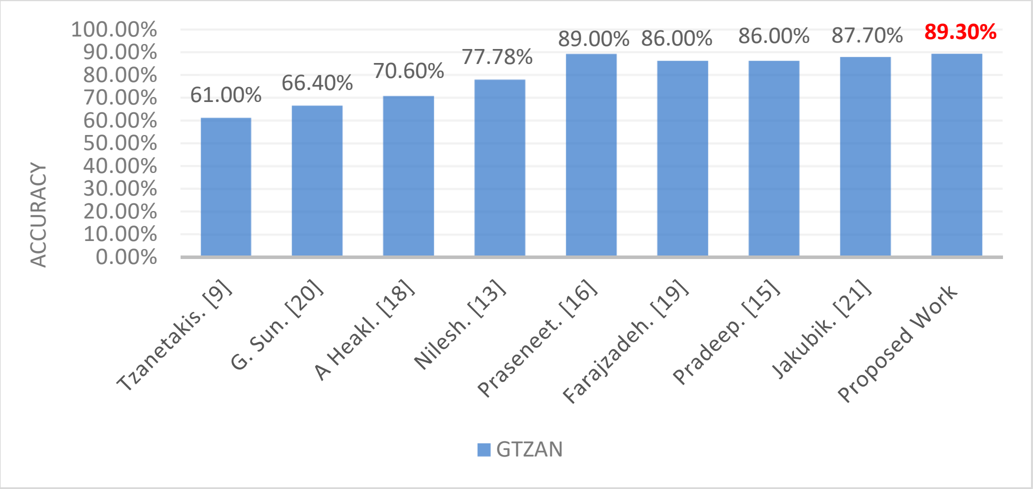

Figure 11. Comparison between the proposed hybrid and the other state-of-the-art models.

The findings indicate that the classification effectiveness of the CNN-BiGRU and CNN-BiLSTM networks using a bidirectional framework is preferred to that of CNN-GRU and CNN-LSTM networks without bidirectionality. Furthermore, acknowledging the pattern of information enables the model to more intelligently recognize the value of the features retrieved by the convolutional neural networks at a specific time. There is initially a convolutional block in the hybridization in terms of CNN and RNN variants. Also, the gradient during backpropagation is not sufficient, restricting the network layers from being proficiently updated, which probably contributed to lower accuracy. The training performance of the model depends upon the hardware used for the experiments. We used a Core i5 Intel 3.2 GHz processor with 10th generation and 32 GB RAM to execute the program on Microsoft Windows. Figure 12 elaborates the comparison between the training and validation accuracy. It can be seen that after 20 epochs, the validation accuracy becomes stable up to 50 epochs. Similarly, Figure 13 shows the validation loss statistics of the model.

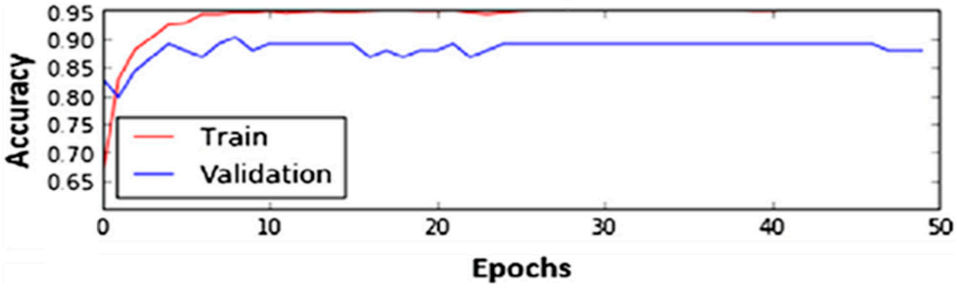

Figure 12. Training and validation accuracy.

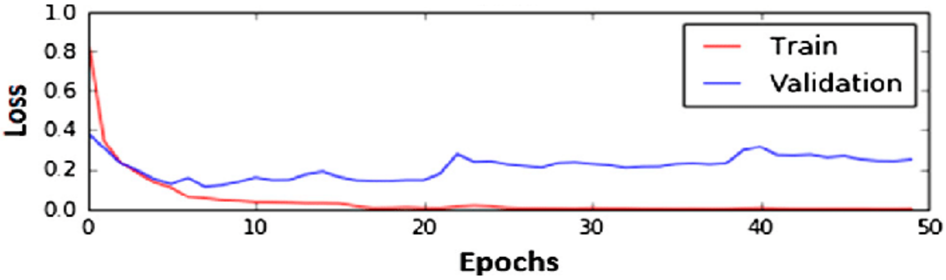

Figure 13. Training and validation loss.

We also calculated the precision, recall, and F1 score to assess how well the model works, as shown in Table 4.

Table 4. Performance evaluation measures.

Performance	Score
Precision	0.85
Recall	0.91
F1-Score	0.88

Furthermore, results are also based on input image size, number of layers, number of kernels, and the size of the network kernels. It can be seen the proposed model obtains the precision, recall, and F1-scores of 0.85, 0.91, and 0.88, respectively, as shown in Table 4. These results confirm the significance of the model when compared with the other models.

6. Conclusions

In this paper, we performed a music classification task on a public dataset called GTZAN by introducing a novel hybridization of CNN and variants of RNN. For feature extraction, we used Mel-spectrogram and MFCC jointly with the four combinations of neural network, namely CNN and LSTM, CNN and Bi-LSTM, CNN and GRU, and finally CNN and Bi-GRU. For the MFCC, using the combination of CNN and LSTM, we obtained the highest accuracy of 76.40%, whereas for the Mel-spectrogram, the combination of CNN and Bi-GRU exhibited the highest accuracy of 89.30%, which was the best among all combinations. We also compared our model with the other state-of-the-art methods and found comparable results. We found that a hybridization is beneficial to music classification using extractive features and temporal aggregation. In future, we will conduct experiments on other datasets, such as FMA, for the classification of music, instruments, or recognition of the artist.

Author Contributions: Conceptualization, M.A., F.A., and J.R.; methodology, M.A., I.U.D., J.R., and M.Y.; software, F.A., M.Y., and S.F.Y.; validation, M.A., I.U.D., M.Y., S.F.Y., and M.T.E.; formal analysis, M.A., F.A., and J.R.; investigation, M.A., I.U.D., and J.R.; resources, F.A., M.Y., and S.F.Y.; data curation, M.A., F.A., and I.U.D.; writing—original draft preparation, M.A., F.A., and J.R.; writing—review and editing, F.A., J.R., and M.T.E.; visualization, M.A., I.U.D., M.Y., S.F.Y., and M.T.E.; supervision, M.A., F.A., and J.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new dataset was created.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nanni, L.; Costa, Y.; Lucio, D.; Silla, C.; Brahnam, S. Combining visual and acoustic features for audio classification tasks. *Pattern Recognit. Lett.* **2017**, *88*, 49–56. [CrossRef]
2. Ashraf, M.; Guohua, G.; Wang, X.; Ahmad, F. Integration of Speech/ Music Discrimination and Mood Classification with Audio Feature Extraction. In Proceedings of the 2018 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 17–19 December 2018; pp. 224–229. [CrossRef]
3. Bisharad, D.; Laskar, R.H. Music genre recognition using convolutional recurrent neural network architecture. *Expert Syst.* **2019**, *36*, 1–13. [CrossRef]
4. Huang, A.; Wu, R. Deep Learning for Music. *arXiv* **2016**, arXiv:1606.04930.
5. Abdoli, S.; Cardinal, P.; Koerich, A.L. End-to-end environmental sound classification using a 1D convolutional neural network. *Expert Syst. Appl.* **2019**, *136*, 252–263. [CrossRef]
6. Murad, A.; Pyun, J.-Y. Deep Recurrent Neural Networks for Human Activity Recognition. *Sensors* **2017**, *17*, 2556. [CrossRef] [PubMed]
7. Wu, W.; Han, F.; Song, G.; Wang, Z. Music Genre Classification Using Independent Recurrent Neural Network. In Proceedings of the 2018 Chinese Automation Congress (CAC), Xi'an, China, 30 November–2 December 2018; pp. 192–195. [CrossRef]

8. Ashraf, M.; Ahmad, F.; Rauqir, R.; Abid, F.; Naseer, M.; Haq, E. Emotion Recognition Based on Musical Instrument using Deep Neural Network. In Proceedings of the 2021 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 17–19 December 2021; pp. 323–328. [CrossRef]
9. Tzanetakis, G.; Cook, P. Musical genre classification of audio signals. *IEEE Trans. Speech Audio Process.* **2002**, *10*, 293–302. [CrossRef]
10. Lau, D. Music Genre Classification: A Comparative Study between Deep-Learning and Traditional Machine Learning Approaches. Available online: https://riteshajoodha.co.za/sitepad-data/uploads/2021/02/2020-Dhiven.pdf (accessed on 30 May 2021).
11. Nasrullah, Z.; Zhao, Y. Music Artist Classification with Convolutional Recurrent Neural Networks. Available online: https://github.com/ZainNasrullah/music-artist-classification-crnn (accessed on 30 March 2020).
12. Kumar, A.; Rajpal, A.; Rathore, D. Genre Classification using Feature Extraction and Deep Learning Techniques. In Proceedings of the 2018 10th International Conference on Knowledge and Systems Engineering (KSE), Ho Chi Minh City, Vietnam, 1–3 November 2018; pp. 175–180. [CrossRef]
13. Patil, N.M.; Nemade, M.U. Music Genre Classification Using MFCC, K-NN and SVM Classifier. *Int. J. Comput. Eng. Res. Trends* **2017**, *4*, 2349–7084.
14. Elbir, A.; Cam, H.B.; Iyican, M.E.; Ozturk, B.; Aydin, N. Music Genre Classification and Recommendation by Using Machine Learning Techniques. In Proceedings of the 2018 Innovations in Intelligent Systems and Applications Conference (ASYU), Adana, Turkey, 4–6 October 2018; pp. 1–5. [CrossRef]
15. Kumar, D.P.; Sowmya, B.J.; Chetan; Srinivasa, K.G. A Comparative Study of Classifiers for Music Genre Classification Based on Feature Extractors. In Proceedings of the 2016 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), Mangalore, India, 13–14 August 2016; pp. 190–194. [CrossRef]
16. Fulzele, P.; Singh, R.; Kaushik, N.; Pandey, K. A Hybrid Model for Music Genre Classification Using LSTM and SVM. In Proceedings of the 2018 Eleventh International Conference on Contemporary Computing (IC3), Noida, India, 2–4 August 2018; pp. 1–3. [CrossRef]
17. Goel, A.; Sheezan, M.; Masood, S.; Saleem, A. Genre Classification of Songs Using Neural Network. In Proceedings of the 2014 International Conference on Computer and Communication Technology (ICCCT), Allahabad, India, 26–28 September 2014; Available online: https://ieeexplore.ieee.org/abstract/document/7001506/ (accessed on 31 January 2022).
18. Heakl, A.; Abdelgawad, A.; Parque, V. A Study on Broadcast Networks for Music Genre Classification. In Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18–23 July 2022; pp. 1–8. [CrossRef]
19. Farajzadeh, N.; Sadeghzadeh, N.; Hashemzadeh, M. PMG-Net: Persian music genre classification using deep neural networks. *Entertain. Comput.* **2023**, *44*, 100518. [CrossRef]
20. Sun, G. Research on Architecture for Long-tailed Genre Computer Intelligent Classification with Music Information Retrieval and Deep Learning. *J. Physics: Conf. Ser.* **2021**, *2033*, 012008. [CrossRef]
21. Jakubik, J. Evaluation of Gated Recurrent Neural Networks in Music Classification Tasks. In Proceedings of the 38th International Conference on Information Systems Architecture and Technology—ISAT 2017, Szklarska Poręba, Poland, 17–19 September 2017; pp. 27–37. [CrossRef]
22. Ashraf, M.; Geng, G.; Wang, X.; Ahmad, F.; Abid, F. A Globally Regularized Joint Neural Architecture for Music Classification. *IEEE Access* **2020**, *8*, 220980–220989. [CrossRef]
23. Jakubec, M.; Chmulik, M. Automatic music genre recognition for in-car infotainment. *Transp. Res. Procedia* **2019**, *40*, 1364–1371. [CrossRef]
24. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks For Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
25. Song, G.; Wang, Z.; Han, F.; Ding, S.; Iqbal, M.A. Music auto-tagging using deep Recurrent Neural Networks. *Neurocomputing* **2018**, *292*, 104–110. [CrossRef]
26. Ashraf, M.; Abid, F.; Atif, M.; Bashir, S. The Role of CNN and RNN in the Classification of Audio Music Genres. *VFAST Trans. Softw. Eng.* **2022**, *10*, 149–154. Available online: https://vfast.org/journals/index.php/VTSE/article/view/793 (accessed on 11 August 2022).
27. Dai, J.; Liang, S.; Xue, W.; Ni, C.; Liu, W. Long Short-Term Memory Recurrent Neural Network Based Segment Features for Music Genre Classification. In Proceedings of the 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP), Tianjin, China, 17–20 October 2016; pp. 1–5. [CrossRef]
28. Abid, F.; Li, C.; Alam, M. Multi-source social media data sentiment analysis using bidirectional recurrent convolutional neural networks. *Comput. Commun.* **2020**, *157*, 102–115. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.