Classification of Music Via Hybrid CNN-LSTM Model

Yilun Sheng
1024040823
Nanjing University of Posts and Telecommunications
School of Computer Science
Nanjing, China

Abstract—This paper presents a hybrid CNN-LSTM-based approach for music genre classification, addressing the limitations of traditional handcrafted feature methods by leveraging deep learning techniques. The system preprocesses raw audio files into mel spectrograms, applies data augmentation, and utilizes a combination of CNN layers for spatial feature extraction and LSTM modules for temporal modeling. The model integrates advanced techniques such as self-attention, CutMix augmentation, and OneCycleLR scheduling to enhance training efficiency and performance. Evaluation on a diverse dataset of ten music genres demonstrates the model's effectiveness, achieving an accuracy of 84.42% and a macro F1-score of 0.8446. These results highlight the robustness of the proposed architecture in capturing complex audio patterns, offering significant improvements in music classification tasks.

Index Terms—Music Genre Classification, Hybrid CNN-LSTM, Deep Learning, Mel Spectrograms, Self-Attention Mechanism

I. Introduction

The rapid growth of digital audio collections has intensified the demand for efficient and accurate music classification systems. As the volume of available music continues to increase, automated methods for organizing and categorizing these collections become indispensable for applications such as recommendation systems, music retrieval, and user personalization. Traditional music classification approaches rely heavily on manual feature extraction, which is time-consuming and often fails to capture the intricate patterns in audio data [1, 2]. Recent advances in deep learning have introduced powerful methods capable of learning complex representations directly from raw or transformed audio signals, thereby revolutionizing the field [3].

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have emerged as two of the most successful architectures for audio-based tasks. CNNs are particularly effective at capturing spatial hierarchies in spectrogram representations of audio [4], while RNNs excel in modeling temporal dependencies, which are crucial for understanding the sequential nature of musical data [5]. By combining these architectures, a hybrid CNN-LSTM (Long Short-Term Memory) model leverages the strengths of both paradigms, providing a robust framework for music classification [6].

This paper presents a comprehensive pipeline for music classification using a CNN-LSTM model. The proposed system preprocesses raw audio files into mel spectrograms [7], applies data augmentation techniques to improve model generalization [8, 9], and trains a hybrid architecture to classify music into predefined genres. The contributions of this work are summarized as follows:

- Data Preprocessing and Augmentation: A detailed preprocessing pipeline converts raw audio files into mel spectrograms and applies various image-based augmentations, enhancing the diversity of the training set.
- Hybrid CNN-LSTM Architecture: A novel architecture combining CNN layers for spatial feature extraction and an LSTM module for temporal modeling, enriched by self-attention mechanisms for capturing global dependencies.
- Training and Optimization Strategies: State-of-the-art techniques such as CutMix, label smoothing, and learning rate scheduling are employed to optimize the training process and mitigate overfitting.
- Evaluation Framework: Comprehensive evaluation on a diverse dataset of music genres demonstrates the effectiveness of the proposed approach.

The rest of this paper is organized as follows. Section 2 reviews related works in music classification and deep learning-based audio analysis. Section 3 defines the problem statement and objectives. Section 4 describes the proposed algorithms and solutions, including the preprocessing steps and model architecture. Section 5 presents experimental results and evaluates the model's performance. Finally, Section 6 concludes the paper with a discussion of findings and future work directions.

II. Related Works

A. Deep Learning for Automatic Feature Extraction

Deep learning has revolutionized the field of music classification by enabling end-to-end feature learning directly from raw or preprocessed audio data. Convolutional Neural Networks have proven particularly effective for extracting spatial features from spectrogram representations of audio. Dieleman and Schrauwen applied CNNs to mel spectrograms and demonstrated their ability to learn hierarchical feature representations for music classification [11]. Similarly, Choi et al. used CNNs for large-scale music tagging, showing their robustness in processing diverse datasets. This shift from manual feature engineering to automated feature extraction has significantly improved classification performance and scalability [12].

B. Attention Mechanisms in Music Analysis

Attention mechanisms have recently become a cornerstone in deep learning models, allowing the network to selectively focus on the most relevant parts of the input. Vaswani et al. introduced the self-attention mechanism in their Transformer architecture, which has since been widely adopted across domains, including audio analysis [13]. In music classification, attention mechanisms are often integrated into hybrid architectures, such as CNN-LSTM models, to enhance their ability to capture long-range dependencies and emphasize critical audio segments. This approach has demonstrated significant improvements in both accuracy and interpretability of classification models [14].

C. Handcrafted Features for Music Classification

Early methods for music classification relied on handcrafted features such as rhythm patterns, pitch histograms, and timbre coefficients. These features attempted to capture key characteristics of music and were widely used in early genre classification systems [10]. Tzanetakis and Cook developed one of the first systems for musical genre classification based on such features, demonstrating the feasibility of automated music analysis. However, handcrafted approaches struggled to adapt to the growing complexity and diversity of modern music, limiting their scalability and effectiveness in large datasets.

III. Problem Statement

Music classification is a challenging task in the domain of data mining and machine learning. The primary objective of this study is to accurately classify music into predefined genres using advanced computational models. Traditional methods for music classification heavily depend on handcrafted features, which require domain expertise and fail to adapt effectively to the increasing complexity of modern music collections. With the advent of deep learning, automated feature extraction from audio data has become a powerful approach, circumventing the limitations of traditional methods.

This work formalizes music classification as a supervised learning task, where the input consists of preprocessed audio data in the form of mel spectrograms, and the output is a discrete set of genre labels. Let represent the dataset, where is the mel spectrogram representation of an audio sample, and is the corresponding genre label, with CCC being the total number of genres. The objective is to learn a function that maps input features X to their corresponding labels Y, optimizing classification accuracy while minimizing computational overhead.

In this research, we propose a hybrid CNN and LSTM architecture to address the dual nature of music data: spatial patterns in spectrogram representations and temporal dependencies inherent in audio signals. To enhance the model's generalizability and robustness, preprocessing techniques, including data augmentation, are employed to simulate diverse scenarios. The system also integrates advanced strategies such as CutMix for data augmentation during training and self-attention mechanisms to capture global dependencies in temporal features.

IV. Solutions

This section describes the comprehensive pipeline for music classification, leveraging CNN and LSTM architectures to handle spatial and temporal features in music data. The approach includes preprocessing, model architecture, and training methodology.

A. Data Preprocessing and Augmentation

In this paper, we work on a dataset with 10 genres, includes 1000 audio files of ten classes: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae and rock. Each class includes 100 au files with a length of about 30 seconds. The data preprocessing pipeline begins by converting raw audio files into mel spectrograms, a representation that effectively captures the frequency and temporal characteristics of audio. Using the librosa library, each audio sample is processed to generate spectrograms with 128 mel bands, normalized in decibels to ensure uniform scaling. These spectrograms are then saved as PNG images to enable image-based data augmentation and easy loading during training. The dataset is divided into training (75%), validation (15%), and testing (10%) subsets using stratified sampling to preserve genre distributions across the splits. To improve model generalization, augmentation techniques such as brightness and contrast adjustments, horizontal flipping, Gaussian noise addition, and random cropping are applied to the training spectrograms. This augmentation increases data diversity by simulating variations in recording conditions and artifacts, which is essential for creating a robust and generalizable model.

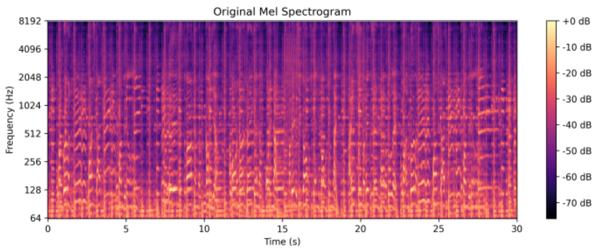

Fig. 1. An Original 30-s Mel-spectrogram sample

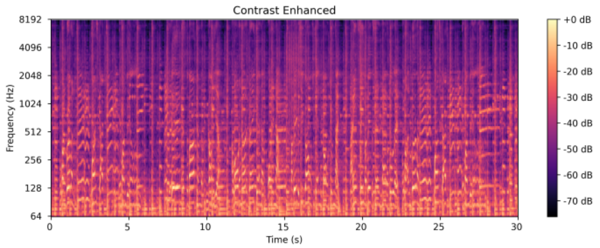

Fig. 2. An Augmented 30-s Mel-spectrogram sample

TABLE I
Model Architecture

Layer	Output
Conv[2D, 3×3] @ 16 - BN - ReLU - MP[2×2]	(128, 16, 64, 64)
Conv[2D, 3×3] @ 32 - BN - ReLU - MP[2×2]	(128, 32, 32, 32)
Dropout[0.5]	(128, 32, 32, 32)
LSTM @ 128 (Bidirectional)	(128, 32, 128)
MultiheadAttention	(128, 32, 128)
LayerNorm	(128, 32, 128)
Dropout[0.4]	(128, 32, 128)
Linear @ 128	(128, 128)
Dropout[0.4]	(128, 128)
Linear @ 10	(128, 10)

B. Hybrid CNN-LSTM Model Architecture

As shown in the proposed architecture, we integrate CNNs and LSTM networks to effectively capture both spatial and temporal features from the mel spectrograms. The architecture consists of two main CNN layers followed by an LSTM network. The CNN sub-blocks perform convolution (Conv[kernel size] @number of filters), ReLU activation, max pooling (MP[kernel size]), and dropout (Dr[dropout percentage]). The first convolutional layer uses a kernel size of 3×3 and 16 filters to extract basic features like edges and textures from the spectrograms, followed by a ReLU activation and max pooling with a 2×2 kernel to reduce spatial dimensions by half. A dropout rate of 40% is applied to prevent overfitting. The second convolutional layer, with 32 filters and the same 3×3 kernel, extracts more complex features, also followed by ReLU activation and max pooling with a 2×2 kernel. After the convolutional layers, the output is reshaped to prepare it for input into the LSTM network, where it is processed to capture temporal dependencies in the music. The LSTM layer is bidirectional with 64 hidden units in each direction, allowing the model to consider both past and future context. Following the LSTM, a multi-head self-attention mechanism is applied to capture long-range dependencies across the temporal sequence, with layer normalization to stabilize training and a dropout rate of 40% to reduce overfitting. The output of the attention layer is then passed through fully connected layers, reducing the feature dimensions to 128 before projecting to the final genre predictions using a softmax activation. This architecture, with its combination of CNNs for spatial feature extraction and LSTMs for temporal modeling, ensures a robust and effective approach to music genre classification.

C. Training and Optimization

The training and optimization process for the CNN-LSTM model was meticulously designed to balance performance, robustness, and computational efficiency. A combination of advanced optimization techniques, data augmentation, and scheduling strategies was employed to achieve optimal results.

The model training employed the AdamW optimizer, which improves over traditional Adam by decoupling weight decay from gradient updates, reducing overfitting. The update rule includes a regularization term:

$$\theta_{t+1} = \theta_t - \eta(\frac{\widehat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} + \lambda\theta_t) \tag{1}$$

where $\widehat{m}_t$ and $\hat{v}_t$ are the first and second moment estimates, ϵ ensures numerical stability, η is the learning rate, and λ is the weight decay parameter.

TA key component of the optimization process was the warm-up learning rate strategy integrated into the OneCycleLR scheduler. In the initial training phase, the learning rate is linearly increased from a small base value to a peak value over a predefined number of warm-up steps. This phase allows the model to gradually adapt to the data, reducing the risk of instability in early iterations. After the warm-up phase, the learning rate follows a cosine annealing schedule:

$$\eta_t = \eta_{max} \cdot \frac{1 + \cos\left(\pi t/T\right)}{2} \tag{2}$$

where t is the current training step and T is the total number of steps. This learning rate schedule accelerates convergence while mitigating overfitting risks.

The loss function employed was Label Smoothing Cross-Entropy, designed to prevent the model from becoming overconfident in its predictions. Label smoothing introduces a controlled amount of uncertainty by slightly softening the target labels:

$$\mathcal{L}_{smoothed} = (1 - \alpha) \cdot \mathcal{L}_{CE} + \alpha \cdot \mathcal{L}_{uniform} \tag{3}$$

where $\mathcal{L}_{CE}$ is the standard cross-entropy loss, $\mathcal{L}_{uniform}$ represents the loss for a uniform distribution, and α is the smoothing factor.

To further improve generalization, CutMix data augmentation was applied, where regions from one spectrogram were replaced with regions from another, and the loss was computed as a weighted sum of the respective labels. The augmentation probability was reduced over epochs, ensuring that the model gradually focused more on clean samples.

During each training epoch, mixed-precision training with PyTorch's GradScaler was utilized to enhance computational efficiency without sacrificing accuracy. Gradient clipping with a maximum norm of 1.0 was applied to stabilize training and prevent exploding gradients:

$$\|\nabla_\theta \mathcal{L}\|_2 \leq max\_norm \tag{4}$$

Validation was conducted after every epoch to evaluate the model's performance on unseen data, tracking accuracy and loss. Early stopping was employed to terminate training if no improvement in validation loss was observed for a defined patience period, ensuring the model did not overfit. The model with the highest validation accuracy was saved as the final model, guaranteeing optimal performance.

These strategies collectively ensured that the CNN-LSTM model achieved high accuracy, robust generalization, and efficient training, making it well-suited for the challenging task of music genre classification.

V. Evaluation

This section presents the evaluation of the proposed music classification system based on the CNN-RNN model. The evaluation includes the experimental setup and results analysis, providing insight into the model's performance in terms of accuracy, loss, and classification effectiveness.

A. Experiment

The training process utilized a batch size of 128, ensuring computational efficiency while maintaining stable gradient updates. A learning rate of 0.001 was scheduled using the OneCycleLR scheduler, which optimized convergence by increasing the learning rate during the initial phase and gradually decreasing it. The AdamW optimizer, with weight decay for regularization, was employed to minimize the loss function. The training spanned a maximum of 400 epochs, with early stopping implemented to terminate training if validation loss showed no improvement for 20 consecutive epochs.

For evaluation, accuracy, F1-score, precision, recall, and a confusion matrix were used as metrics. These metrics comprehensively captured the model's classification effectiveness. The confusion matrix provided a detailed view of genre-specific performance, highlighting areas of strength and opportunities for improvement. Evaluation was conducted using unseen test data, ensuring the model's ability to generalize to new examples.

B. Results

The evaluation results demonstrate the model's effectiveness in music genre classification. The learning rate curve illustrates the OneCycleLR scheduler's effect, with a rise and subsequent decline in learning rate that contributed to effective optimization.

Training accuracy steadily improved, reaching 72.94%, while validation accuracy peaked at 82.87%, indicating minimal overfitting and strong generalization.

The model experienced early stopping at epoch 355 due to a lack of improvement in validation loss over 20 consecutive epochs. This indicates the model had effectively converged before reaching the maximum number of training epochs, ensuring computational efficiency while maintaining performance.

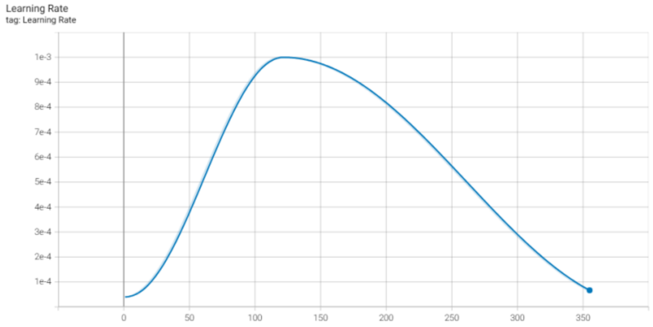

Fig. 3. Learning Rate Curve

The loss curves for both training and validation showed consistent decreases, with training loss reaching approximately 1.2 and validation loss decreasing to 0.9. These trends confirm the model's capability to reduce classification errors over successive epochs.

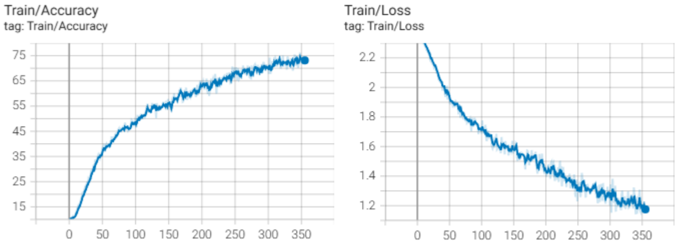

Fig. 4. Train Acc & Loss Curve

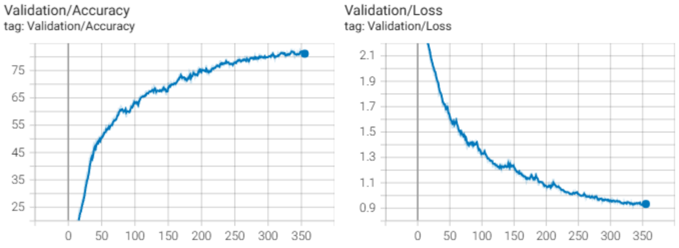

Fig. 5. Validation Acc & Loss Curve

Evaluation metrics further substantiate the model's classification performance. The model achieved an overall accuracy of 84.42%, with a macro F1-score of 0.8446, a macro precision of 0.8474, and a macro recall of 0.8442. The detailed classification report highlights genre-specific results, with classical and jazz achieving high precision and recall values of 0.9365 and 0.9417, respectively. While genres such as rock and country faced challenges, with F1-scores of 0.6695 and 0.7681, this is consistent with the inherent acoustic overlap among certain genres.

TABLE II
Evaluation Metrics

Class	Precision	Recall	F1-score	Support
blues	0.9060	0.8833	0.8945	120
classical	0.9365	0.9833	0.9593	120
country	0.7063	0.8417	0.7681	120
disco	0.7540	0.7917	0.7724	120
hiphop	0.8403	0.8333	0.8368	120
jazz	0.9339	0.9417	0.9378	120
metal	0.9533	0.8500	0.8987	120
pop	0.8571	0.8000	0.8276	120
reggae	0.8966	0.8667	0.8814	120
rock	0.6903	0.6500	0.6695	120
accuracy			0.8442	1200
macro avg	0.8474	0.8442	0.8446	1200
weighted avg	0.8474	0.8442	0.8446	1200

The confusion matrix provides detailed insights into genre-specific classification performance. For instance, classical music achieved near-perfect classification with 118 correctly classified samples out of 120. Jazz and reggae also performed well, with 113 and 104 correctly classified samples, respectively. However, minor misclassifications were observed in genres like country and rock, which are prone to overlapping acoustic features.

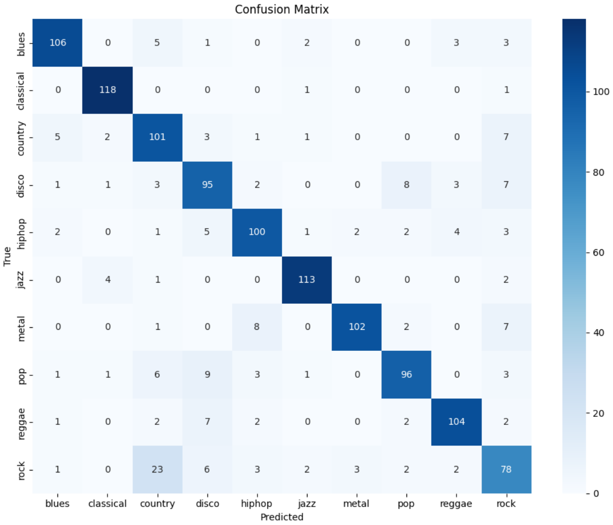

Fig. 6. Confusion Matrix

The provided evaluation curves and confusion matrix figures further substantiate these findings, offering a visual representation of the model's performance. Together, these results validate the proposed CNN-RNN model's efficacy in addressing the music genre classification task.

VI. Conclusion

In this study, we proposed a hybrid CNN-LSTM model for music genre classification, combining the strengths of convolutional networks for spatial feature extraction and recurrent networks for temporal sequence modeling. The integration of self-attention mechanisms further enhanced the model's ability to capture long-range dependencies. By employing advanced techniques such as data augmentation, label smoothing, and adaptive learning rate scheduling, the model achieved a classification accuracy of 84.42% and demonstrated strong generalization capabilities.

The evaluation results, including detailed metrics, learning curves, and confusion matrices, validate the effectiveness of the proposed approach. The system excelled in distinguishing genres with distinct acoustic characteristics while highlighting challenges in classifying overlapping genres such as country and rock. These findings underscore the importance of advanced architectures and preprocessing techniques in addressing the complexities of music data.

This research demonstrates the potential of hybrid deep learning models in tackling complex classification tasks like music genre identification. The approach not only showcases significant improvements in classification accuracy but also highlights how state-of-the-art techniques like self-attention and adaptive learning can mitigate challenges posed by overlapping features and data diversity. Future directions include incorporating transformer-based architectures to further improve long-range dependency modeling, experimenting with larger and more diverse datasets to enhance robustness, and exploring multi-modal frameworks by combining audio features with metadata. These avenues hold promise for advancing automated music analysis and expanding its applicability in real-world scenarios.

References

[1] B. Logan, “Mel frequency cepstral coefficients for music modeling,” in Proc. Int. Symp. Music Information Retrieval (ISMIR), vol. 270, no. 1, pp. 1–11, 2000.

[2] G. Tzanetakis and P. Cook, “Musical genre classification of audio signals,” IEEE Trans. Speech Audio Process., vol. 10, no. 5, pp. 293–302, 2002.

[3] L. Deng and D. Yu, “Deep learning: Methods and applications,” Found. Trends Signal Process., vol. 7, no. 3–4, pp. 197–387, 2014.

[4] Y. Aytar, C. Vondrick, and A. Torralba, “Soundnet: Learning sound representations from unlabeled video,” in Advances in Neural Information Processing Systems, vol. 29, pp. 892–900, 2016.

[5] S. Hochreiter, “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[6] K. Choi, G. Fazekas, M. Sandler, and K. Cho, “Convolutional recurrent neural networks for music classification,” in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), pp. 2392–2396, 2017.

[7] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, “librosa: Audio and music signal analysis in Python,” in Proc. SciPy, pp. 18–25, 2015.

[8] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” J. Big Data, vol. 6, no. 1, pp. 1–48, 2019.

[9] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, “Randaugment: Practical automated data augmentation with a reduced search space,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops, pp. 702–703, 2020.

[10] T. Lidy and A. Rauber, “Evaluation of feature extractors and psycho-acoustic transformations for music genre classification,” in Proc. Int. Symp. Music Information Retrieval (ISMIR), pp. 34–41, 2005.

[11] S. Dieleman and B. Schrauwen, “End-to-end learning for music audio,” in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), pp. 6964–6968, 2014.

[12] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008, 2017.

[14] K. Choi, G. Fazekas, and M. Sandler, “Automatic tagging using deep convolutional neural networks,” arXiv preprint arXiv:1606.00298, 2016.