A Comparative Study of Classical and Robust End-to-End Models for Music Genre Classification

Jiaqi Guan **Evania Cheng** **Luis Bravo** **Salvador Gracia** **Yulin Lin**

University of California, Santa Barbara

Abstract

Music genre classification is a fundamental task in music information retrieval. This project provides a comparative analysis of two distinct pathways for this task using the GTZAN dataset: a classical approach using hand-crafted features and an advanced end-to-end deep learning approach. The classical pathway, which applies models like SVM, Random Forest, and a simple DNN to pre-computed features on a random data split, achieves high but potentially inflated accuracies up to 78.7%. In parallel, a simple CNN trained from scratch on spectrogram segments underperforms at around 66.9%. To address these limitations, we implement an advanced end-to-end pipeline using a ResNet-18 architecture, pre-trained on the FMA dataset with SpecAugment, and fine-tuned on the standardized Sturm splits to prevent data leakage. On this rigorous benchmark, our advanced model significantly outperforms the baselines, achieving a test accuracy of 71.4%. The primary contribution is a novel robustness analysis, which demonstrates our model's superior resilience to simulated audio corruptions like frequency and time masking, proving its superior potential for real-world applications.

1 Introduction

Music genre classification models face challenges due to the complex and subjective nature of music. Genres are usually ill defined, and artists blend elements across styles, which complicates accurate labeling. The task requires distinguishing genres based on subtle and sometimes overlapping features. Specifically, despite its popularity, the GTZAN dataset has documented issues with artist overlap, duplicate tracks, and mislabeling, which can result in overly optimistic performance estimates when using standard random splits Tzanetakis and Cook [2002], Sturm [2013]. To directly mitigate this, our advanced evaluation protocol exclusively uses the fault-filtered Sturm splits, ensuring no artist overlap between training and testing sets.

Research Question: How do classical machine learning models trained on tabular audio features compare to deep learning approaches trained on spectrograms, particularly when evaluated under both standard and rigorous, leak-free protocols? Furthermore, how do these models compare in terms of robustness to audio degradation?

This paper makes the following contributions:

1. We establish baseline performance by evaluating four classical machine learning models (Random Forest, SVM, KNN, Logistic Regression) and a deep feedforward network (DNN) on pre-computed tabular audio features from the GTZAN dataset, using a random data split Li et al. [2003], Flexer [2007], Lee et al. [2009].
2. We implement an advanced end-to-end pipeline using a ResNet-18 architecture. This pipeline leverages **transfer learning** by pre-training on the large-scale FMA dataset and incorporates **SpecAugment** for data augmentation to improve generalization.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

3. We conduct a more rigorous evaluation by fine-tuning and testing our advanced model on the standardized **Sturm splits** of GTZAN, providing a more reliable measure of generalization by directly comparing it against the baseline models.
4. We introduce a novel **robustness analysis**, evaluating our advanced model against simulated audio corruptions (additive noise, time masking, and frequency masking) to assess its potential for real-world performance.

2 Related Work

The task of music genre classification (MGC) has evolved significantly over the past two decades. Early and seminal work by Tzanetakis and Cook Tzanetakis and Cook [2002] introduced the GTZAN dataset and established a baseline using hand-crafted audio features—such as Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, and zero-crossing rate—paired with classical machine learning models like Gaussian Mixture Models (GMMs) and Support Vector Machines (SVMs). This feature-engineering paradigm dominated the field for many years.

With the rise of deep learning, the focus shifted towards end-to-end models that learn features directly from audio representations. Convolutional Neural Networks (CNNs) applied to spectrograms became the de-facto standard, treating audio classification as an image classification problem Lee et al. [2009]. This approach eliminated the need for domain-specific feature design and often yielded superior performance. More advanced architectures, such as Residual Networks (ResNets) He et al. [2016], further improved results by enabling the training of deeper networks.

However, a critical re-evaluation of the GTZAN dataset by Sturm Sturm [2013] revealed significant methodological flaws, including artist overlap between training and test sets, which led to overly optimistic and non-generalizable results. This exposed the need for more rigorous evaluation protocols, such as the artist-filtered splits we use in this work. To improve generalization and robustness, researchers have incorporated techniques from other domains. Park et al. introduced SpecAugment Park et al. [2019], a simple and effective data augmentation method for spectrograms that improves robustness to occlusions in time and frequency. Furthermore, transfer learning from large-scale datasets, such as the Free Music Archive (FMA) Defferrard et al. [2017], has become a common strategy to leverage pre-trained features.

Our work builds upon these advancements. We bridge the classical and modern approaches by directly comparing them on both the flawed random split and the rigorous Sturm split. By integrating a ResNet architecture with transfer learning and SpecAugment, and conducting a novel robustness analysis, we aim to provide a comprehensive and realistic assessment of model performance for MGC.

3 Background

3.1 Datasets

- **GTZAN Dataset:** Our primary dataset, containing 1,000 30-second audio clips across 10 genres (blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock). It is a standard benchmark but is known for data quality issues Tzanetakis and Cook [2002].
- **FMA-small Dataset:** A larger, high-quality dataset of 8,000 30-second audio clips across 8 genres, used for pre-training our advanced model to learn general-purpose audio features Defferrard et al. [2017].

3.2 Audio Feature Representation: Log-Mel Spectrograms

For our end-to-end models, raw audio waveforms are converted into a 2D representation that is more suitable for CNNs. We use log-Mel spectrograms, which represent the spectral power of a signal over time on a Mel-frequency scale. This scale is designed to mimic human auditory perception. For each 30-second audio clip (resampled to 22,050 Hz), we generate a log-Mel spectrogram with 128 Mel bands, using a Fast Fourier Transform (FFT) size of 2048 and a hop length of 512. These spectrograms, generated using `librosa` McFee et al. [2015], serve as the direct input to our CNN models.

3.3 Theoretical Foundations

- **Cross-Entropy Loss:** All deep learning models are trained using the cross-entropy loss function, which is the standard for multi-class classification tasks. To handle class imbalance in the Sturm splits, we apply inverse frequency class weighting to the loss function.
- **SpecAugment:** A data augmentation technique applied directly to spectrograms. It involves masking blocks of consecutive frequency channels (frequency masking) and time steps (time masking). This forces the model to learn more robust and distributed feature representations, improving its generalization and resilience to occlusions Park et al. [2019].

4 Methodology

Our study follows two distinct pathways to evaluate models for music genre classification.

4.1 Pathway 1: Baseline Models

We evaluate two categories of baseline models to establish a performance reference.

- **Classical & Tabular DNN Models:** Four classical machine learning models (Random Forest, SVM, KNN, Logistic Regression) and a 3-layer feedforward DNN were trained on a pre-computed set of tabular audio features provided with the GTZAN dataset.
- **Simple CNN Model:** A baseline CNN with three convolutional blocks was trained from scratch on spectrograms. To handle the large input size, each 30-second track was segmented into ten non-overlapping 3-second chunks, and the model predicted the genre for each chunk. No data augmentation was applied to this baseline.

4.2 Pathway 2: Advanced End-to-End CNN

To address the limitations of the baseline approaches, we developed an advanced end-to-end pipeline.

- **Architecture:** We employ a ResNet-18 architecture as a feature extractor backbone, chosen for its proven effectiveness on image-like data He et al. [2016]. The standard ImageNet-based head is replaced with a custom classifier head suitable for our 10-class problem.
- **Training Strategy:** A two-stage transfer learning approach was used.
  1. **Pre-training on FMA-small:** The model was first trained on the larger FMA-small dataset for 25 epochs. During this phase, we applied SpecAugment (time masking T=70, frequency masking F=27) to learn robust audio features.
  2. **Fine-tuning on GTZAN:** The pre-trained model was then fine-tuned on the GTZAN Sturm splits for 60 epochs, including a brief "freeze" phase (5 epochs) for the backbone before end-to-end training.

4.3 Robustness Analysis Framework

To assess model resilience beyond standard accuracy, we subjected our final advanced model to a series of on-the-fly audio corruptions during inference on the Sturm test set. The corruptions included Additive White Gaussian Noise (AWGN), Time Masking, and Frequency Masking, each applied at various intensity levels. Performance was measured by the degradation in the Macro F1-score relative to the performance on clean data.

5 Experiments

5.1 Data Splitting and Protocols

- **Random Split (Baseline Protocol):** For the initial baseline experiments, a 70%/15%/15% stratified random split of the GTZAN dataset was used. This common approach is known to be susceptible to artist overlap and potential data leakage.

- **Standardized Split (Advanced Protocol):** For our primary model and final head-to-head comparisons, we exclusively use the standardized train/validation/test splits proposed by Sturm Sturm [2013]. This protocol ensures no artist overlap between sets, providing a more reliable measure of generalization.

5.2 Implementation Details

Our pipeline was implemented using PyTorch Paszke et al. [2019]. To optimize training, audio files were pre-processed into log-Mel spectrograms and cached to disk.

- **Preprocessing:** In addition to spectrogram generation, each spectrogram was normalized instance-wise by subtracting its mean and dividing by its standard deviation. This proved essential for robustness.
- **Hardware:** All baseline models were trained on a single NVIDIA T4 GPU, our final model was trained on a NVIDIA 4080 Super.
- **Hyperparameters:** The advanced model was trained using the AdamW optimizer with mixed-precision (AMP) to accelerate computation. During FMA pre-training, we used a learning rate of 5e-4 and weight decay of 8e-3. For GTZAN fine-tuning, we use two learning rates: For the frozen layers, a learning rate of 1e-3 and weight decay of 1e-3, and for the fine-tuned layers, the learning rate was reduced to 3e-5 with a weight decay of 5e-3. A 'ReduceLROnPlateau' scheduler adjusted the learning rate based on the validation F1-score in both stages. We set the patience for the FMA scheduler to 3.

5.3 Evaluation Metrics and Baselines

The primary evaluation metric for all experiments is **classification accuracy**. For the robustness analysis, we use the **Macro F1-score** to better account for class imbalance and performance degradation. Our advanced model is compared against the following baselines: Random Forest, SVM, KNN, Logistic Regression, a tabular DNN, and a simple from-scratch CNN.

6 Results & Analysis

6.1 Baseline Performance on a Random Split

Initial experiments on a random data split yielded high accuracies, as summarized in Table 1. The DNN trained on tabular features achieved the highest accuracy of all baseline models at **78.7%**. The from-scratch CNN underperformed at **66.9%**. While these results are strong, they are evaluated under a protocol known for potential data leakage. Detailed per-genre accuracy plots and training histories for these baseline models are provided in the Appendix (Figures A.1 - A.6).

Table 1: Baseline Model Accuracy on a Random Test Split (Reported from initial experiments).

Model	Feature Set	Test Accuracy (%)
DNN	Tabular (CSV)	**78.7**
Random Forest	Tabular (CSV)	77.5
SVM	Tabular (CSV)	77.0
Logistic Regression	Tabular (CSV)	73.0
KNN	Tabular (CSV)	69.0
Simple CNN	Spectrogram Segments	66.9

6.2 Model Generalization on a Standardized Benchmark

To provide a more rigorous and fair comparison, we evaluated the baseline models and our Advanced CNN on the standardized Sturm test set, which eliminates artist overlap. The results are shown in Figure 1.

The performance of all baseline models dropped significantly, with the top-performing DNN's accuracy falling from 78.7% to 53.5%. This confirms that the initial high scores were heavily influenced

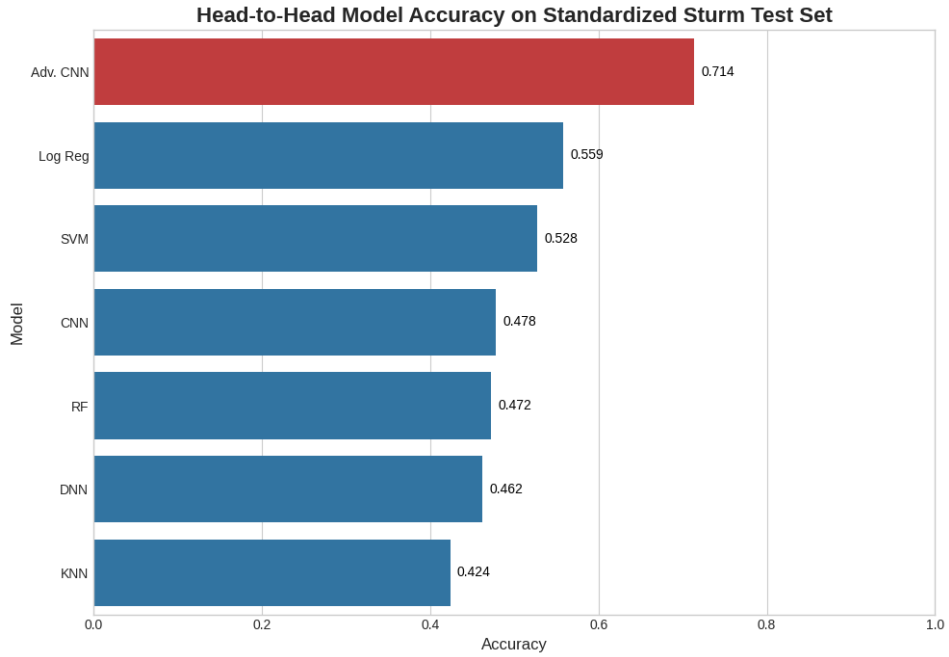


Figure 1: Head-to-Head Model Accuracy on the Standardized Sturm Test Set. Our Advanced CNN is highlighted in red.

by data leakage. In contrast, our Advanced CNN achieves a test accuracy of **71.4%**. This represents a substantial **15.4 percentage point improvement** over the best-performing baseline model (Logistic Regression at 56.0%) on this fair benchmark, demonstrating the clear superiority of our transfer learning and augmentation-based approach.

6.3 Comparative Robustness and Training Dynamics

Having established that our Advanced CNN generalizes better on clean, standardized data, we conducted a final set of experiments to compare its resilience to audio corruption against the from-scratch Baseline CNN. This analysis is critical for understanding how each model might perform in real-world scenarios with imperfect audio.

6.3.1 Robustness to Additive Noise

The first test involved applying additive white Gaussian noise (AWGN) to the input spectrograms. The results, shown in Figure 2, are striking. Our Advanced CNN demonstrates perfect immunity, with its F1-score remaining constant across all noise intensities. This is attributed to the instance-wise normalization in our preprocessing pipeline, which effectively nullifies the statistical impact of the noise. The Baseline CNN, however, proves extremely brittle to this corruption. Its performance completely collapses with even minimal added noise, indicating that its learned features are highly sensitive and not well-generalized.

6.3.2 Robustness to Masking

The analysis of time and frequency masking, which simulates signal dropouts and narrowband interference, reveals the superior quality of the features learned by our advanced model (Figure 3). During FMA pre-training, SpecAugment with a time mask parameter of T=70 and frequency mask parameter of F=27 was applied. For GTZAN, SpectAugment was applied with a time mask parameter of T=40 and frequency mask parameter of F=15.

In the frequency masking test, while both models degrade, our model's F1-score declines gracefully, maintaining a score above 0.50 even with significant masking. The baseline model's performance plummets rapidly, nearing random chance at high corruption levels.

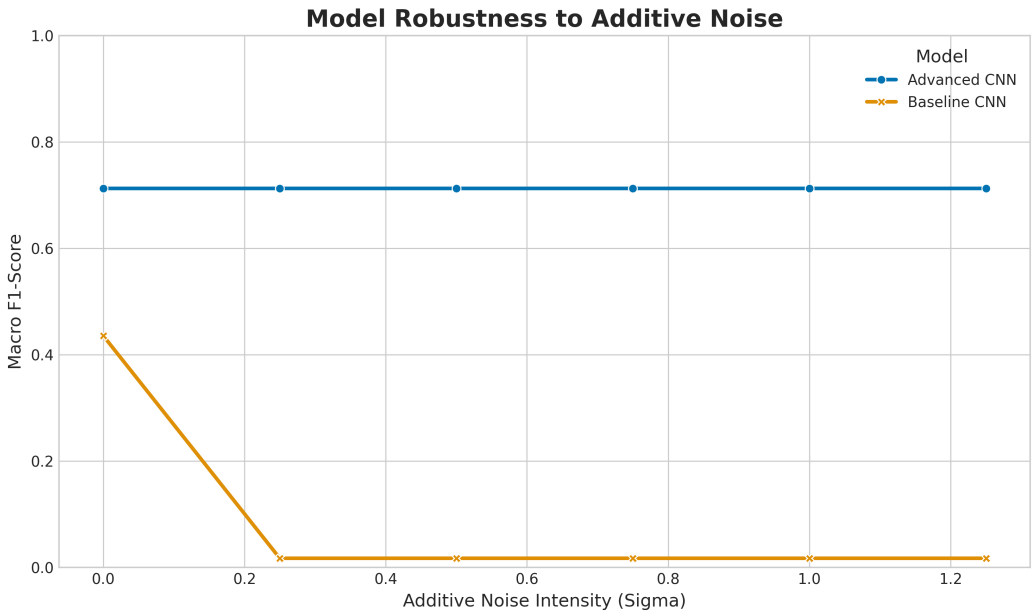


Figure 2: Model Robustness to Additive Noise. Our model's instance-wise normalization provides near-perfect immunity, while the baseline model's performance collapses.

The impact of our training methodology is most evident in the time masking test. Our model is remarkably resilient to temporal occlusions, with its performance remaining stable across all levels. This is a direct validation of using **SpecAugment** during pre-training. The baseline model, lacking this augmentation, fails completely when faced with even moderate temporal gaps. This demonstrates that our model has learned more distributed and fundamentally robust feature representations.

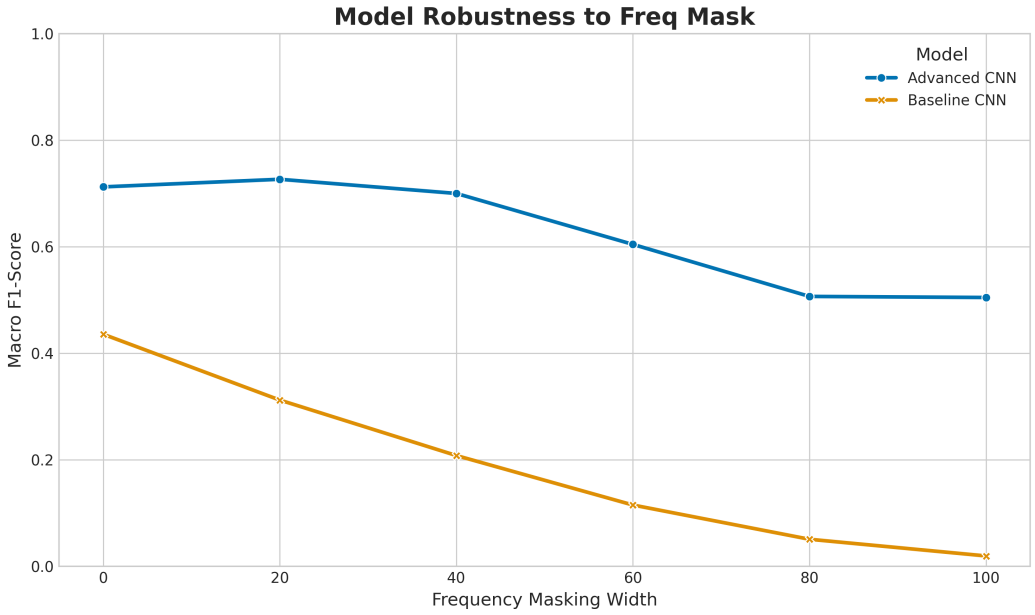


Figure 3: Frequency Masking: comparative robustness of Advanced and Baseline CNN models.

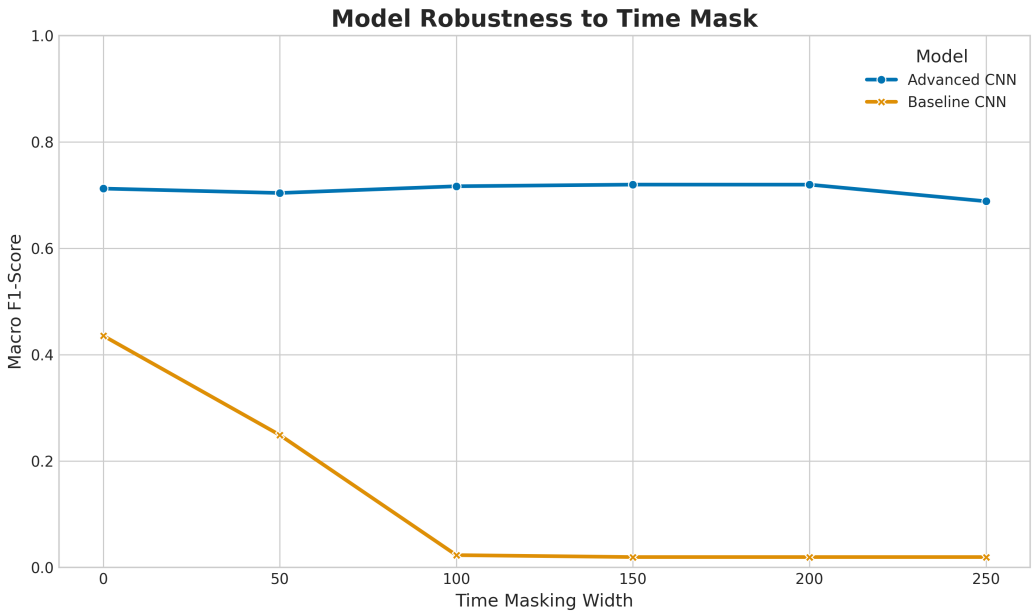


Figure 4: Time Masking: comparative robustness of Advanced and Baseline CNN models.

6.3.3 Training Dynamics of the Advanced CNN

Finally, the training history of our model (Figure 5) shows effective learning. The training accuracy consistently increased while the loss decreased. More importantly, the validation accuracy and loss curves plateau smoothly, indicating successful convergence without problematic overfitting and validating the reported test performance.

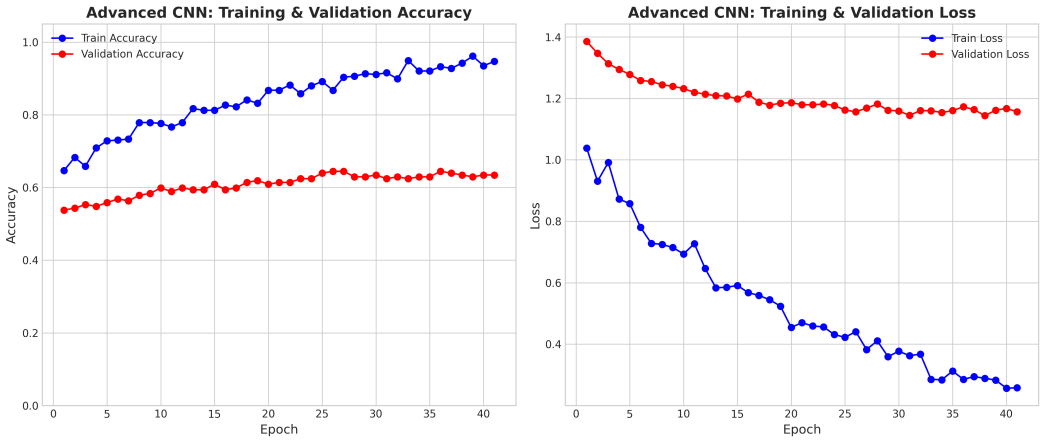


Figure 5: Training and Validation history for our Advanced CNN on the GTZAN Sturm split.

7 Discussion

Our dual-pathway investigation yielded critical insights into music genre classification. The baseline experiments confirmed that on a standard random split, models trained on hand-crafted tabular features can achieve high but misleading accuracy scores. The performance collapse of these models on the standardized Sturm splits provides compelling evidence of the data leakage problem in GTZAN and underscores the importance of rigorous evaluation protocols.

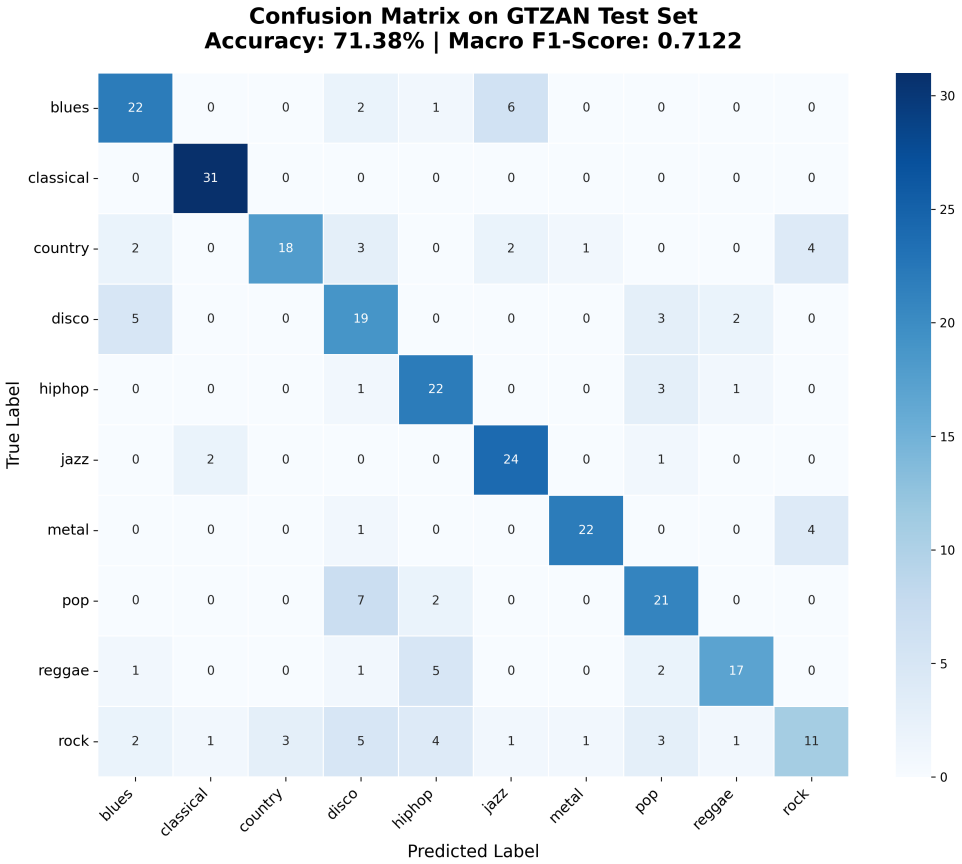


Figure 6: Confusion Matrix for the Advanced CNN on the Sturm Test Set.

It is in this challenging, leak-free environment that our advanced end-to-end model demonstrated its superiority. By achieving a 71.4% accuracy, it not only outperformed all baseline models in a fair comparison but also highlighted the effectiveness of our methodology. The combination of a powerful ResNet architecture, knowledge transfer from the FMA dataset, and SpecAugment data augmentation proved to be a winning strategy. The model's measured robustness to corruption further suggests it has learned more fundamental audio representations, making it the superior choice for practical applications where signal quality is variable.

Qualitative Analysis of Results: An analysis of the confusion matrix (Figure 6) reveals patterns consistent with musicological similarities, suggesting our model has learned acoustically meaningful features. The model achieves near-perfect accuracy for genres with highly distinct timbral signatures, such as `classical` and `metal`. Conversely, the most frequent misclassifications occur between closely related genres. For instance, `blues` is most often confused with `rock` (6 instances), likely due to their shared instrumentation and harmonic structures. Similarly, the confusion between `disco` and `hiphop` (5 instances) can be attributed to their shared rhythmic focus and the historical use of disco samples in early hip-hop. These results indicate that the model's errors often reflect genuine genre ambiguity rather than random failure, further validating the quality of its learned representations.

7.1 Limitations and Future Work

While this study provides a robust framework, we acknowledge several limitations that open avenues for future research. First, our robustness analysis was performed against a baseline CNN, not the tabular models. A direct resilience comparison would require adapting the corruption framework to tabular data, which could be a valuable next step. Second, our work is confined to the GTZAN

and FMA-small datasets; the model's performance on a wider array of genres or production styles remains an open question.

Future work could focus on enhancing feature quality through more advanced pre-training. Exploring self-supervised methods, such as contrastive learning frameworks adapted for audio, could reduce the reliance on labeled data and potentially yield even more generalizable representations. Furthermore, applying the rigorous Sturm evaluation protocol to other state-of-the-art architectures could provide a clearer picture of true progress in the field of music genre classification.

8 Collaboration Statement

The baselines were established by Yulin Lin, who worked on the classical models, and Jiaqi Guan, who trained the baseline deep learning models. The advanced end-to-end pipeline was implemented by Luis Bravo, who also retrained baseline models on the sturm splits. Evania Cheng and Salvador Gracia contributed to the initial project design and report organization. All authors contributed to the final report.

9 Code and Data Availability

The code for this project can be found at:

1. GTZAN/FMA Sturm-Splits Dataset:`https://www.kaggle.com/datasets/bravola/precomputed-mel-spectrograms-fma-small-and-gtzan`
2. Main Model :`https://github.com/brvola/music-classification-ml`
3. Colab:`https://colab.research.google.com/drive/1f-KqB3O6iE2WOhtAU-5XYtqY1aOoeAFO`
4. GTZAN Dataset: `https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification`

References

Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. Fma: A dataset for music analysis. *arXiv preprint arXiv:1708.01826*, 2017.

Arthur Flexer. The problem of limited inter-rater agreement in modelling music similarity. *Journal of New Music Research*, 36(3):195–210, 2007.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, page 770–778, 2016.

Honglak Lee, Peter Pham, Yonatan Largman, and Andrew Y Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in neural information processing systems*, volume 22, 2009.

Tao Li, Mitsunori Ogihara, and Qi Li. A comparative study of music genre classification. In *Proceedings of the 26th annual international ACM SIGIR conference*, pages 282–289. ACM, 2003.

Brian McFee, Colin Raffel, Dawen Liang, Eric Battenberg, Matt McVicar, Eric Battenberg, and Juan Pablo Bello. librosa: Audio and music signal analysis in python. In *Proc. Python in Science Conf.*, page 18–25, 2015.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. In *Interspeech*, page 2613–2617, 2019.

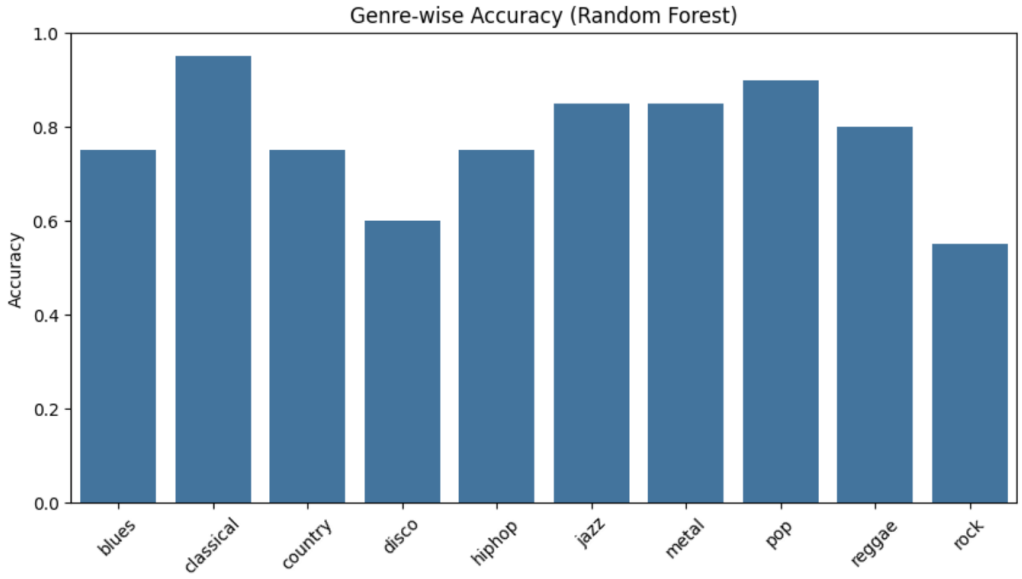
Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, and et al. PyTorch: An imperative style, high-performance deep learning library. `https://pytorch.org`, 2019.

Bob L Sturm. The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*, 2013.

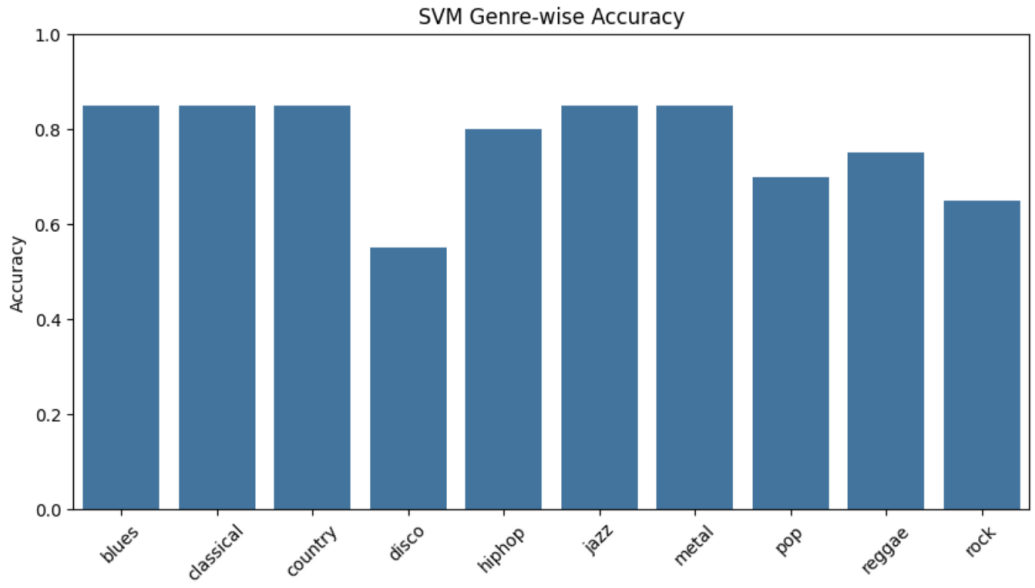
George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

A Appendix: Baseline Model Performance on Random Split

This appendix provides detailed visualizations for the baseline models evaluated on the 70%/15%/15% random split, as referenced in Section 3.1. These results, while high, are potentially inflated due to artist overlap in the GTZAN dataset and serve as a point of comparison against the more rigorous Sturm split evaluation.

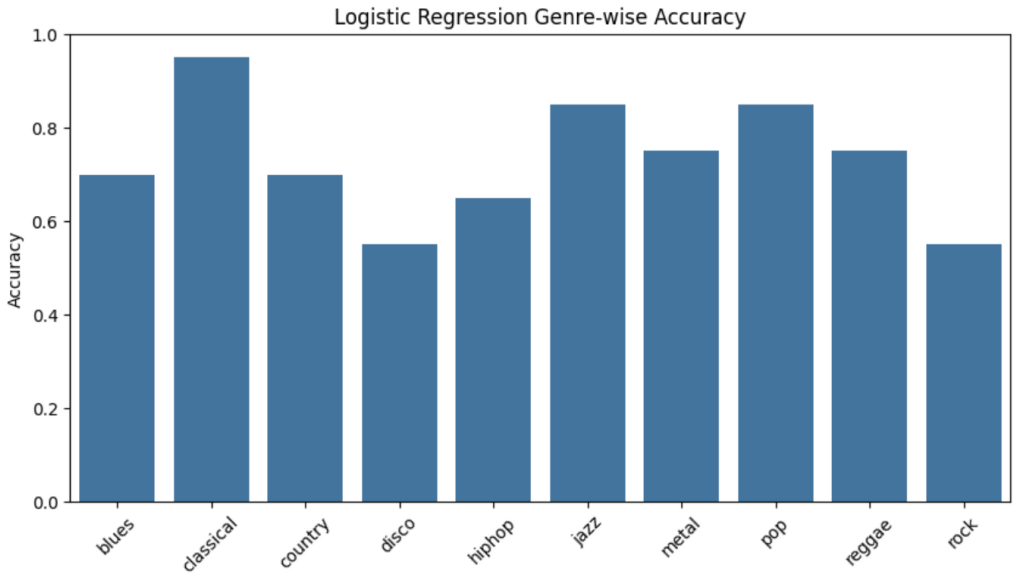


(a) Random Forest

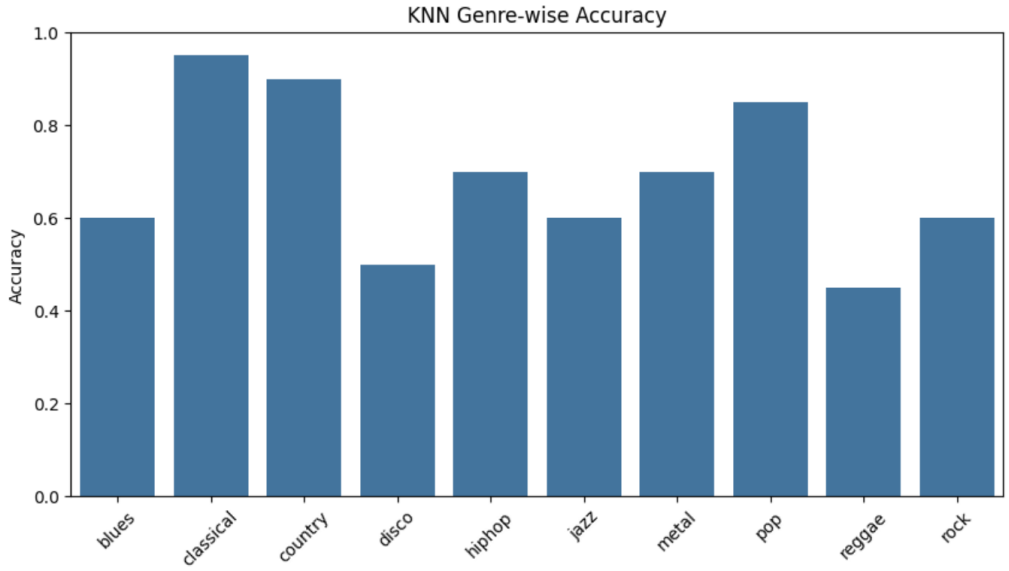


(b) Support Vector Machine (SVM)

Figure 7: Per-genre test accuracies for Random Forest and SVM on the random split.

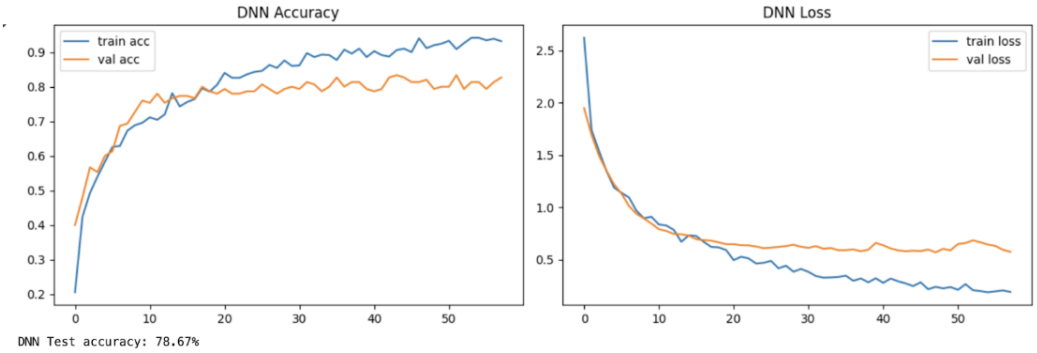


(a) Logistic Regression

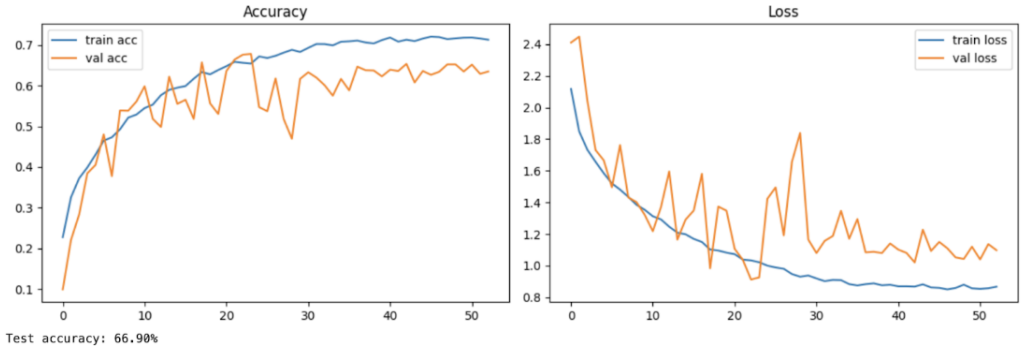


(b) K-Nearest Neighbors (KNN)

Figure 8: Per-genre test accuracies for Logistic Regression and KNN on the random split.



(a) DNN on Tabular Features



(b) Simple CNN on Spectrograms

Figure 9: Training and validation history for the baseline deep learning models on the random split. (a) The DNN trained on tabular features (Test Acc: 78.7%), and (b) the simple CNN trained from scratch on spectrogram segments (Test Acc: 66.9%).