MUSIC GENRE CATEGORIZATION: FINAL REPORT

Aaryaman Singh
Student# 1008731062
aaryaman.singh@mail.utoronto.ca

Mathieu Baudon
Student# 1010806601
mathieu.baudon@mail.utoronto.ca

Yunwang Chen
Student# 1010806488
yunwang.chen@mail.utoronto.ca

ABSTRACT

Music, with its myriad styles and influences, has always been challenging to categorize into neat genre-specific boxes. Recognizing the complexities inherent in this task, our group has united to explore the intricate realm of ”Music Genre Categorization”. Through deep learning, we aim to create a system capable of discerning and categorizing tracks based on their distinct sonic signatures. Our ambition is to simplify music discovery and categorization, intending to enhance the overall musical journey for aficionados across the globe. This document includes our Final Project Report. —-Total Pages: 9

1 INTRODUCTION

In an era where digital music platforms are rapidly evolving, accurately categorizing music by genre has become a critical challenge. Our project addresses this need by harnessing the power of deep learning to develop a sophisticated model for precise genre classification. The significance of this project lies in its potential to revolutionize music recommendation systems, streamline music database management, and enrich music education.

Deep learning is uniquely suited for this task, as it excels in discerning complex patterns in extensive datasets. By leveraging deep neural networks, our model aims to identify the distinct audio characteristics inherent in various music genres, thus elevating genre categorization to a new level of accuracy. This approach not only enhances user experience on digital platforms but also offers valuable insights for artists and educators in understanding and exploring musical genres. The application of machine learning in this context is not just reasonable but essential, given its unparalleled capability in processing and interpreting large-scale audio data.

2 ILLUSTRATION

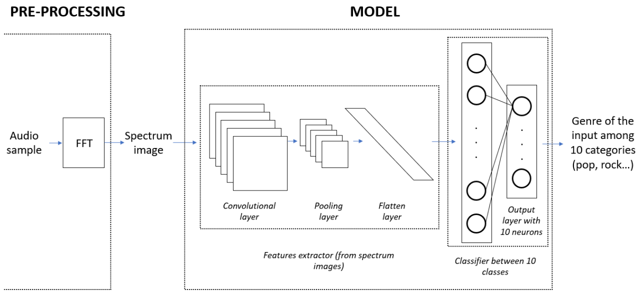

PRE-PROCESSING
MODEL
Audio sample
FFT
Spectrum image
Convolutional layer
Pooling layer
Flatten layer
Features extractor (from spectrum images)
Output layer with 10 neurons
Classifier between 10 classes
Genre of the input among 10 categories (pop, rock...)

Figure 1: Overview of the model used to categorize musics

3 Background and Related Work

K S Mounika (2021) The research paper explores music genre classification using CNN and convolutional recurrent neural networks (CRNN), enhanced by transfer learning. Key methods include fine-tuning, optimizers, and a multi-frame approach for detailed song analysis. The study employs datasets like GTZAN, achieving notable success in genres like Rock (93.8 percent recall), Disco (85.5 percent), and Blues (83.8 percent), but faces challenges with Metal (49.8 percent) and Pop (53.5 percent). A web platform developed for this purpose reported a classification accuracy of 73.2 percent on a dataset of 1000 samples, indicating the potential of these models in practical applications.

Gessle, Gabriel (2022) This study compares CNN and LSTM models using MFCCs for music genre classification. Focusing on GTZAN and FMA datasets, the CNN model showed higher accuracy, especially in distinct genres like classical music. Both models struggled with complex genres such as rock. Despite limitations due to dataset specificity, the research underscores the CNN model's effectiveness in genre prediction and suggests enhancements like using more features and larger datasets for improved classification.

Yu-Huei Cheng (2021) This research utilizes a CNN with a Mel spectrum on the GTZAN dataset, featuring ten music genres. The model, comprising five convolutional layers with specific kernel size and dropout rate, achieved a 77 percent accuracy, which increased to 83.3 with majority voting. The novelty lies in applying CNN and majority voting for music genre classification, addressing the growing need in music streaming. However, the study faces limitations in diverse genre classification, particularly rock, and relies solely on the GTZAN dataset. The model's integration with streaming media and web crawlers is proposed for enhancing music discovery.

Ning CHEN (2020) This study introduces a hybrid model combining Random CNN (RCNN) and Broad Learning (BL) for music classification. This novel approach enhances both accuracy and efficiency, utilizing RCNN for feature extraction from Mel-spectrograms and BL for improved prediction accuracy and reduced training time. Despite potential limitations due to dataset specificity, the model outperforms traditional deep learning combinations on GTZAN, Ballroom, and Emotion datasets. This approach is promising for digital music platforms, offering advancements in music classification, discovery, and recommendation systems.

Wenlong Zhang (2022) This study develops a deep learning method for music genre classification, utilizing data preprocessing and a fully connected neural network with an attention mechanism. The method significantly enhances classification accuracy and efficiency, especially in genres like metal, classical, and blues. However, challenges arise in distinguishing between similar genres like rock and country due to overlapping rhythmic elements. Overall, this deep learning approach demonstrates marked improvements in music genre classification, suggesting its potential utility in digital music platforms

4 Data Processing

In this section, we outline the data processing steps undertaken to prepare the GTZAN dataset for our music genre classification task. Our primary focus was on adhering to copyright regulations and ethical guidelines. To this end, we utilized the GTZAN dataset, as compiled and made available by Olteanu (2019). This dataset is categorized under a license classified as ”Other (specified in the description),” which permits its use for academic and research purposes without infringing on copyright restrictions.

4.1 Visualization in Time Domain

The initial step in our data processing involved visualizing the audio signals in the time domain. This representation offers a fundamental understanding of the waveform characteristics of the music pieces. For instance, a sample music piece from the dataset, visualized in the time domain for a duration of 30 seconds, is depicted below:

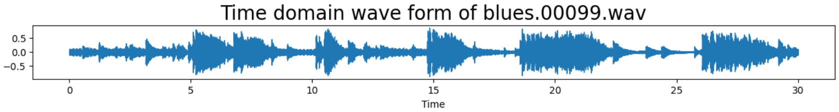


Figure 2: Sample music piece in the time domain over 30 seconds.

4.2 Frequency Domain Analysis

To delve deeper into the signal characteristics, we employed the Fast Fourier Transform (FFT) Wikipedia contributors (2023) technique. This approach allows us to analyze the frequency components of the audio signals over brief intervals, such as 0.01 seconds. By applying FFT, we can unveil the spectral components of the music piece, as demonstrated in the following figure:

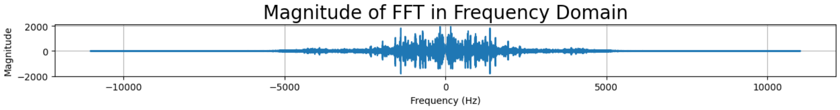


Figure 3: Frequency domain representation of a music piece over a 0.01-second interval.

4.3 Spectrogram Generation

The final step in our preprocessing routine involved generating spectrograms for the audio samples. This is achieved by iteratively applying the FFT over small time steps across the entire length of the audio clip. In our case, we focused on 3-second segments of each track. The resulting spectrogram encapsulates both the variations in frequency and amplitude over time. This format is particularly suitable for input into our convolutional neural network model, as it effectively encapsulates the complex characteristics of musical signals.

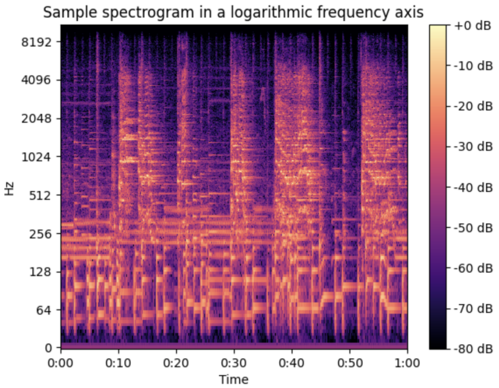


Figure 4: Frequency domain representation of a music piece over a 0.01-second interval.

5 Architecture

The final model architecture can be splitted into two distinct parts: the features extractor and the classifier. The features extractor is a Neural Network which will be able to identify and extract different features from the input, which is an image in our case. Then, the classifier will rely on those features to decide the genre of the audio.

Our best model relies on transfer learning. As the feature extractor, we used AlexNet neural network which is a Convolutional Neural Network (or CNN) and we implemented a fully-connected neural network as the classifier. It will rely on the features extracted by AlexNet.

More precisely, AlexNet takes images as input and it will return computed features of size $256 \times 10 \times 14$ where size is given as *depth* x *height* x *width*. These features are the input of our fully-connected network. The latter uses ReLU activation function and is composed of 2 layers:

- Input has a size of $256 \times 10 \times 14$.
- Hidden layer contains 250 neurons.
- Output layer is composed of 10 neurons because we classify between 10 classes.

As we train a multi-classifier, we used Cross-Entropy loss function. We also used Adam optimizer with a learning rate equal to 7×10^{-5} and a weight decay equal to 1×10^{-2}. Our best performance has been reached with a batch size of 64 and the model was trained over 35 epochs.

6 Baseline Model Evaluation

In evaluating the performance of our machine learning pipeline, we establish a baseline using a Support Vector Machine (SVM) model. The baseline serves as a comparison point for more complex models developed later in the project.

6.1 Input Data Description

The input to the SVM classifier consists of a labeled dataset provided as a CSV file. Each 3s piece in the dataset are divided according to a specific time step, with a granularity of 0.2 seconds, and contains the following features:

1. Amplitude Mean: The average amplitude of the signal during the time step.
2. Amplitude Variance: The variance in amplitude throughout the time step.

6.2 Model Description

Given the non-linear nature of the classification task at hand, we experimented with various kernel settings provided by the scikit-learn package. The kernel that performs best is the sigmoid kernel.

6.3 Model Performance

The performance of the SVM baseline model is summarized in the table below. The model was trained on a subset of the data and evaluated on a separate test set to assess its generalization capability.

MODEL	**TRAIN ACCURACY**	**TEST ACCURACY**
Best baseline model	50.0%	52.0%

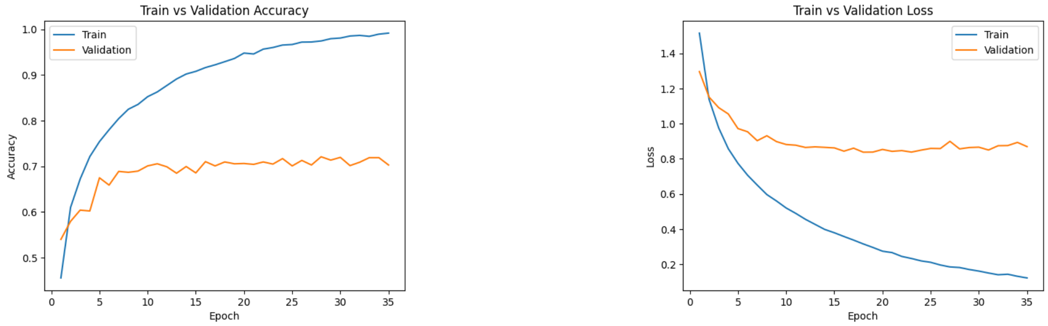
Table 1: Train accuracy and test accuracy for best baseline model

The baseline model achieves a training accuracy of 50.0% and a test accuracy of 52.0%, indicating a slight improvement over random guessing in a balanced binary classification task. This performance sets a preliminary benchmark for subsequent models.

7 Quantitative Results

During training, accuracy and loss curves were plotted for each epoch. Below are the curves for our best model:

Based on Figure 5, we observe the model has reached its limit on validation data as validation accuracy and loss remain steady for at least 15 epochs. Yet, it keeps on learning on training dataset as training accuracy still increase when we stopped the training.



(a) Train vs Validation accuracy

(b) Train vs Validation loss

Figure 5: Training curves for best model

In our model evaluation process, we also set aside a test dataset. It is composed of images our model has never seen during training. We computed the accuracy and the loss of the model on this test dataset.

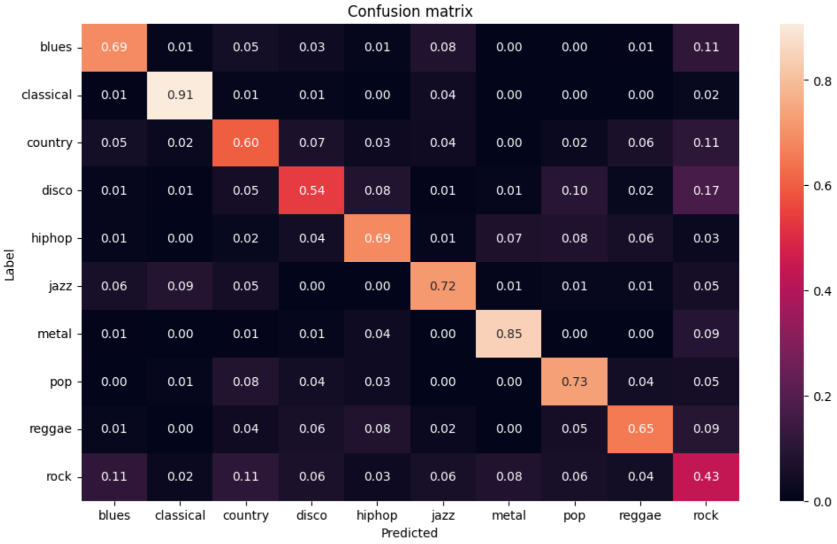
MODEL	TEST ACCURACY	TEST LOSS
Best model	68.0%	0.989

Table 2: Test accuracy and test loss for best model

As reported in Table 2, test accuracy is very close to validation accuracy and is better than the accuracy of previous model which was equal to 64%.

8 Qualitative Results

As seen in previous section, the test accuracy gives us a single value as a measure of our model performance. However, we trained a multiclassifier, meaning this value is not very representative of its performance. In order to have more details on model's evaluation, we can construct its confusion matrix: which is the Figure 6 below. Note that values within the confusion matrix are ranged between 0 and 1.



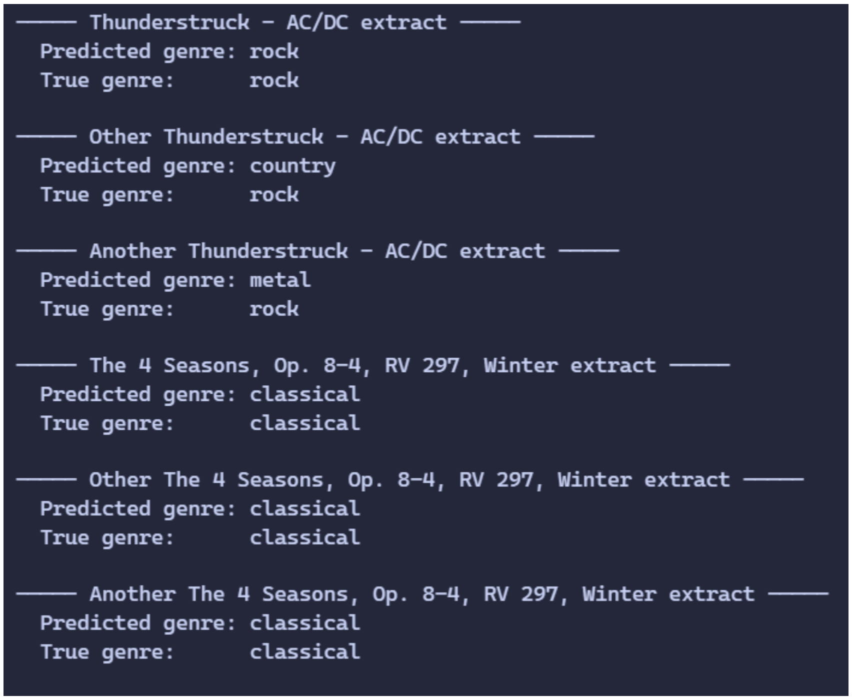
Label \ Predicted	blues	classical	country	disco	hiphop	jazz	metal	pop	reggae	rock
blues	0.69	0.01	0.05	0.03	0.01	0.08	0.00	0.00	0.01	0.11
classical	0.01	0.91	0.01	0.01	0.00	0.04	0.00	0.00	0.00	0.02
country	0.05	0.02	0.60	0.07	0.03	0.04	0.00	0.02	0.06	0.11
disco	0.01	0.01	0.05	0.54	0.08	0.01	0.01	0.10	0.02	0.17
hiphop	0.01	0.00	0.02	0.04	0.69	0.01	0.07	0.08	0.06	0.03
jazz	0.06	0.09	0.05	0.00	0.00	0.72	0.01	0.01	0.01	0.05
metal	0.01	0.00	0.01	0.01	0.04	0.00	0.85	0.00	0.00	0.09
pop	0.00	0.01	0.08	0.04	0.03	0.00	0.00	0.73	0.04	0.05
reggae	0.01	0.00	0.04	0.06	0.08	0.02	0.00	0.05	0.65	0.09
rock	0.11	0.02	0.11	0.06	0.03	0.06	0.08	0.06	0.04	0.43

Figure 6: Confusion matrix for best model's prediction on test dataset

Now we can have a better understanding of how our model truly performs. For instance, we notice it is very good for predicting classical and metal genres but it struggles with rock genre. The test accuracy makes more sense now. We understand this value averages model's excellent performances with its worst ones.

To demonstrate this result, let's test our model on a classical piece and a rock track. We will use 3 extracts for each track: one from the beginning, one from the middle and one from the end of the track.

Figure 7 shows the result of this test below.



```
----- Thunderstruck - AC/DC extract -----
  Predicted genre: rock
  True genre:      rock

----- Other Thunderstruck - AC/DC extract -----
  Predicted genre: country
  True genre:      rock

----- Another Thunderstruck - AC/DC extract -----
  Predicted genre: metal
  True genre:      rock

----- The 4 Seasons, Op. 8-4, RV 297, Winter extract -----
  Predicted genre: classical
  True genre:      classical

----- Other The 4 Seasons, Op. 8-4, RV 297, Winter extract -----
  Predicted genre: classical
  True genre:      classical

----- Another The 4 Seasons, Op. 8-4, RV 297, Winter extract -----
  Predicted genre: classical
  True genre:      classical
```

Figure 7: Model evaluation over isolated samples.

As you can see, our model clearly identified the classical track. However, the model has more difficulty with the rock tracks as it could identify only one extract correctly.

We think the reason behind low accuracy for rock, disco classification is that intuitively the genre of music like rock and disco varies often and include many electrical sound, for which we need more data to capture their features correctly. Also, the reason behind high accuracy for classical genre is because instruments involved in this genre are particularly recognizable in terms of frequency.

9 Evaluation on New Data

For the evaluation phase, we procured copyright-free music tracks representing each genre to test our model's generalizability. These tracks were distinct from those in the GTZAN dataset to ensure a robust assessment. Details on the selected songs are accessible via our group's data repository link.

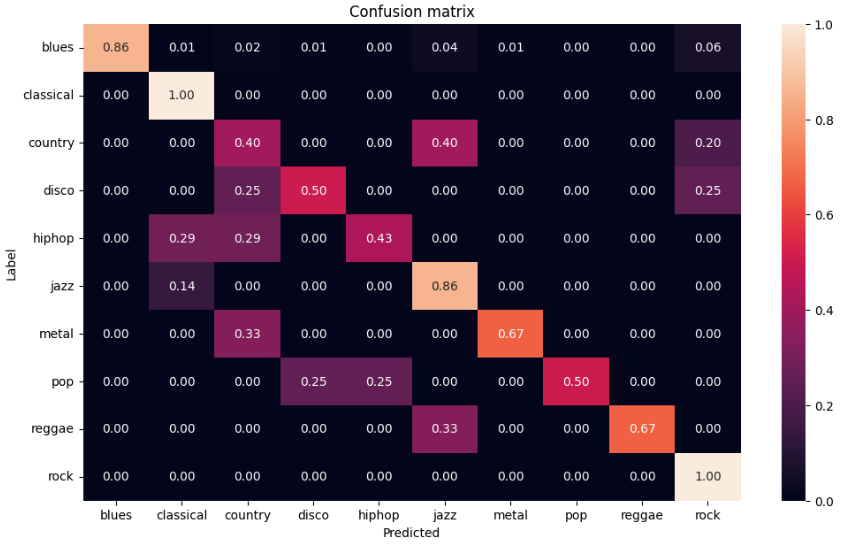


Figure 8: Confusion matrix for the new evaluation dataset, overall accuracy is 81.37%.

The model's performance showed variability across different genres. We posit that this variance is partly due to the prototypical nature of some tracks that align closely with the characteristics learned by our genre classification model. Conversely, genres like country and hip-hop exhibited lower accuracy, which we attribute to their evolving trends over years not being captured within the training dataset, since GTzan is built at 2002, too old for new songs.

10 Discussion

If we refer to the confusion matrix in Figure 6, we can observe most audio are categorized correctly by our model. Indeed, the main confusion comes from the rock genre and results are impacted by this error in two ways:

- If label is Rock, model is likely to predict another class close to Rock genre, such as Blues or Metal.
- If label is not rock but a genre which draws its inspiration from Rock, model is likely to predict Rock.

In other words, if we can fix the confusion over Rock genre, model's performance should skyrocket as you can see on Figure 8. It would limit bad predictions for Rock genre but also for other genres relatively close to Rock.

This result can be surprising at first but there is a very logical explanation behind it. Rock genre is one of the largest genre: it contains a lot of sub-genres. Indeed, Rock's artists draw their inspiration from other genres a lot and this is why we have a large variety within Rock category. For instance, there is Blues Rock, primarily brought by The Rolling Stones which is a mix between Blues and Rock as suggested by its name. But even Metal genre is derived from Rock in a first place. No wonders that the model believes some Rock or Metal tracks are similar.

To sum up, bad performance for Rock genre is totally normal. All the more, other models based on GTZAN dataset noticed the same behavior.

On the other hand, our model performs very well at categorizing classical tracks as they are subjectively distinct from other genres. Indeed, compared to other genres, there are no drums in classical pieces or not much. It means frequency spectrum is less filled.

Another surprising result is for Disco genre. We thought it would be distinct enough but the model predicted a lot of disco tracks to be rock tracks. It seems like there are still a lot of drums in disco tracks. Furthermore, if we compare spectrum diagrams for both disco and rock random tracks, there are not very different. It could be caused by the dataset itself which does not include very distinct disco tracks that could help the model to find the right features for identifying disco genre more accurately.

In conclusion, our model has acceptable performance. Its shortcomings are justified by a similarity between the genres themselves. The model is able to extract features from a spectrum diagram which are useful for categorizing the track.

11 Ethical Considerations

Data Collection and Bias: Our model is only as good as the data it's trained on. If our training dataset lacks diversity, representing primarily Western music genres while underrepresenting or excluding non-Western genres, our model could perpetuate biases. This might inadvertently sideline artists from non-mainstream genres, cultures, or regions. Moreover, it's essential to ensure that the data we utilize respects copyrights and intellectual property rights.

Misclassification and Artistic Integrity: If our model misclassifies certain tracks, it could inadvertently affect an artist's representation and their music's reception. For instance, tagging a Jazzinfused Rock track as purely ”Jazz” could affect its discoverability among Rock enthusiasts, potentially limiting its audience.

12 Project Difficulty / Quality

The Difficulty/Complexity of our project is highlighted by several factors:

Diverse Musical Genres: Our model had to distinguish between a wide range of musical genres, each with its unique sonic signatures.

Advanced Feature Extraction: We implemented sophisticated techniques to accurately extract features from audio data. This involved analyzing various aspects of music, such as tempo, rhythm, and instrumentation, which are not easily discernable.

Data Variability: We faced the challenge of working with datasets that exhibited significant variability. This included differences in recording quality, instrumentation, and genre representation, which added layers of complexity to the model's training and evaluation. Our model's performance in this demanding context was noteworthy. It achieved an impressive test accuracy of 68.0 percent, which is significant considering the intricate nature of the problem at hand. This was further validated by our evaluation on new data, where the model demonstrated an overall accuracy of 81.37 percent, showcasing its ability to generalize well beyond the training dataset.

Innovative techniques and advanced methodologies were employed to address these challenges. Beyond the basic lab requirements, our team implemented a combination of convolutional neural networks (CNNs) and transfer learning using the AlexNet neural network. This approach was chosen for its efficacy in feature extraction and classification, which was crucial for the success of our project.

This project represents a significant learning journey, from the initial concept to the final implementation. Our team adapted and evolved our approach, integrating feedback and learning from early challenges. The result is a model that not only meets expectations in a highly complex and challenging domain.

In conclusion, the high level of difficulty associated with our project and the quality of the results obtained clearly demonstrate our team's ability to create a model that performs better than expected on a challenging project, with learning that goes beyond the requirements of standard labs. This achievement is a testament to our team's dedication, skill, and innovative approach to problem-solving in the field of music genre categorization using deep learning.

All code written and used for this project has been uploaded to this Github repository: `https://github.com/J3y0/music-categorization`.

REFERENCES

Gessle, Gabriel. A comparative analysis of cnn and lstm for music genre classification. `https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1354738&dswid=-5630`, 2022. [Online; accessed 28-November-2023].

K S Mounika. Music genre classification using deep learning. `https://ieeexplore.ieee.org/document/9675685`, 2021. [Online; accessed 28-November-2023].

Ning CHEN. Combining cnn and broad learning for music classification. `https://search.ieice.org/bin/summary.php?id=e103-d_3_695`, 2020. [Online; accessed 28-November-2023].

Andrada Olteanu. Gtzan dataset: Music genre classification, 2019. URL `https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification`. Accessed on 2023-10-10.

Wenlong Zhang. Music genre classification based on deep learning. `https://www.researchgate.net/publication/362835372_Music_Genre_Classification_Based_on_Deep_Learning`, 2022. [Online; accessed 28-November-2023].

Wikipedia contributors. Fast fourier transform — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Fast_Fourier_transform`, 2023. [Online; accessed 3-November-2023].

Yu-Huei Cheng. Convolutional neural networks approach for music genre classification. `https://ieeexplore.ieee.org/abstract/document/9394067`, 2021. [Online; accessed 28-November-2023].