PAPER NAME

C066_C072_MusicGenreClassification_Report.docx

AUTHOR

Pranshu Padia

WORD COUNT

3445 Words

CHARACTER COUNT

19107 Characters

PAGE COUNT

22 Pages

FILE SIZE

1.3MB

SUBMISSION DATE

Mar 30, 2024 9:16 PM GMT+5:30

REPORT DATE

Mar 30, 2024 9:17 PM GMT+5:30

4% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 3% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database
- 4% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 10 words)

MUSIC GENRE CLASSIFICATION USING CNN

NNDL & MLOA Project

Presented by:

Pranshu Padia - C066

Lavhith Pragada - C072

Table of Content

1. Introduction

Process of music genre classification is one of the significant challenges in the realm of audio signal processing. With the vast amount of music available today, automated methods for classifying music into genres can help not only music recommendation systems, but also generating playlists, and music library organization. In this project, we used GTZAN Dataset (a popular dataset used in evaluation for music genre recognition (MGR)) while using CNN (Convolutional Neural Network), it is one of the powerful class of deep learning models, to automatically classify audio excerpts into different music genres which serves as a fundamental aspect in our music taste. Just as leading music streaming platforms like Apple Music, Spotify, YT Music, and others capitalize on genre classification to suggest songs tailored to users' tastes, our approach aimed to streamline this process through advanced neural network technique.

1.1 Overview of the Dataset

GTZAN dataset is one of the most popular dataset used in field of Music Information Retrieval (MIR). The dataset has 1000 audio tracks, with each track being 30s long, organized into ten different genres, with hundred audio clips per genre. The genres included in the dataset are:

- Country
- Classical
- Disco
- Pop
- Hip-Hop
- Jazz
- Metal
- Reggae
- Blues
- Rock

Sample rate of each audio clip is 22050 Hz and stored in uncompressed ".wav" format. This dataset provides a valuable resource for training and evaluating music genre classification algorithms due to its diverse range of genres and standardized format.

2. Problem Statement

The choice of this project was done to understand the music sound and be able to differentiate with another song. There are a lot of aspects that come into the picture when you want to classify a particular music into a genre. Example, melody, pitch, tempo, beats per minute, instruments used, harmonies, etc. These all aspects are very minute in detail which makes it difficult to classify genres in day-to-day life, so we used neural network techniques to try solving this problem.

2.1 Some Possible Challenges in Music Genre Classification

1. **Subjectivity:** Music Genre is highly subjective as in different cultures genres can be perceived differently for each culture. Example, music that one person perceives as “Pop” can be classified as some other genre in some other culture.
2. **Genre Fusion:** While making music there are a lot of elements added, like instruments, beats, etc which can create a lot of confusion while labeling them into a particular genre as the genre boundaries are now blurred.

2.2 Motivation

The motivation behind this project is twofold:

1. **Automating Genre Classification:** We tried building an accurate genre classification model, which can be helpful in automating the process of genre labelling. This can be very helpful to some people like music curators, DJs, and also music enthusiasts saving their time to label music based on genre.
2. **Improve Music Recommendation Systems:** This can also help some streaming platforms like YT Music, Spotify, Jio Saavan, etc who rely on user preferences and listening history to recommend songs. A robust model can help enhance the experience of listening to music by providing a granular insight into user taste.

Our primary objective is to design and train a CNN model which is capable of classifying music genres from a short fragment of the audio.

3. Literature Survey

Sr No.	Title of Paper or Article	Authors	Concepts discussed in paper	Concepts useful in project
1	Music Genre Classification	Haggblade, Hong and Kao	It provides a brief description of feature extraction, like techniques used to extract the information, visualize features like MFCCs, zero-crossing rate, harmonics, etc In this paper it uses machine learning algorithms like KNN, SVM, and neural networks and check performance of the model by predicting genres of the music. It evaluates the performance on basis on different evaluation metrics like classification report, and many more.	This paper helped us in achieving a high accuracy for our model using deep neural networks. It also helped us find the importance of each feature for classifying the genre.
2	Large-Scale Music Genre Analysis and Classification Using Machine Learning with Apache Spark	Mousumi Chaudhury, Amin Karami, and Mustansar Ali Ghazanfar	This paper provides a description of Apache Spark framework which can be helpful in for processing large datasets in parallel to train the genre classification model. It also provides the important features that helps us classify the genres of the music. It also provides us how to tune the hyperparameters and what to tune.	This paper helped us tune the hyperparameters which helped us not only improve performance of the model but also optimize the model, which saved our time. It also gave us an idea to show the most important features using Machine learning model, which are essential to classify the genres.
3	Machine Learning-Based Music Genre Classification with Pre-Processed Feature Analysis	Bojan Bogdanović, Miroslav Milovanović, Veljko Jovičić	This paper provides a series of description for data pre-processing and how to handle problems like class imbalance and much more. This paper uses Machine learning models like SVM, KNN, XGBoost, which could be helpful in some applications like recommendation systems.	This paper helped us in pre-processing the dataset like scaling, normalizing the features, etc. This paper also helped us in future idea, where we can use our model for recommendation systems to predict the genre and provide user their taste of genres.
4	Deep Learning for Music Genre	Jin-wu Xu, Cai-hong Jiang, and	This paper gives insights about different aspects of building a deep neural network model,	This paper helped us build our CNN model with optimization techniques to prevent

	Classification	Yan-feng Zhu	like requires large dataset for training, computational costs, relevant feature extraction, etc. This paper implemented deep learning model for classifying genres, where they build CNN model, add regularization techniques, optimize the model, define the layers, etc, which implements complex model to train.	overfitting the provide the best possible results.

4. Proposed System

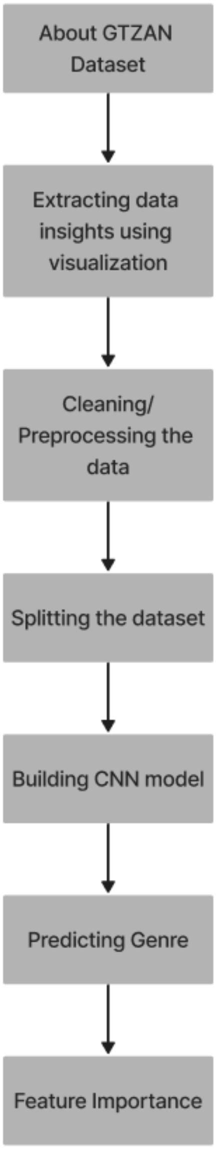

Figure 0 – Flow of Implementation

About GTZAN Dataset: This Dataset is one of the most popular datasets used for Music Genre Classification, the dataset consists length of the audio file, Chroma Frequencies, RMS, Spectral Centroids, Spectral Bandwidth, zero – crossing rate, harmony, perceptual, tempo, MFCCs (Range 1 – 20).

Extracting Features: We will be extracting some of the features which we mentioned above using “Librosa” library and use them to train our CNN Model.

Pre-processing the Data: In terms of pre-processing the dataset, the dataset it almost clean, we are just normalizing them and using label encoders to convert the final prediction into numerical value, to help model classify easier

Splitting the Dataset: We will be splitting the dataset in two parts: Training set (33%) and Validation set (67%).

Building CNN Model: We will be building our CNN model using proper layers, activation function, applying regularization & optimization techniques to perform well while classifying the genre.

Predicting Genre: Then we will be predicting the genre on the validation set and also visualize overall accuracies, loss, etc, to check how model is actually predicting on the new data.

Feature Importance: We also aim to find which are the most essential factors/features that matters when you try to classify the music genre.

5. Implementation

Libraries Used:

a. **Librosa**: It is a Python library that is designed for working with audio files in a simple and intuitive way and seamless manipulation of audio files. Some features of Librosa are:

  i. **Simplified Audio Loading**: It simplifies the process of loading various audio file formats (such as MP3, WAV, etc) with ease directly into your Python environment.

  ii. **Feature Extraction:** It is very useful in extracting countless valuable insights from the audio, like rhythm, pitch, tempo, beats per minute, zero-crossing rates, etc.

  iii. **Dynamic Visualization:** It also has capabilities to craft vivid visual representations of audios, from traditional waveforms to detailed spectrograms and beyond.

  iv. **Enhanced Pre-Processing Tools:** It also provides a variety of tools to refine and prepare the audio for the analysis, like noise reduction, volume adjustments, etc.

b. **IPython:** A robust library which provides an enriched interactive computing experience, In our case we used it for audio file playback directly from the Python shell environment.

2. **Dataset/Genre Information:** In this dataset it consists of 1000 audio tracks, with each 30 seconds long, organized into 10 distinct genres, with 100 tracks per genre.

3. **Sound Wave Visualization for each genre:** Sound waves are a common way to represent audio and to understand frequency distribution of audio signals which is very important for music genre classification.
Sound waves are nothing but a representation of how air pressure changes over time, and it provides a visual depiction of the audio signal.

Genre Specific Sound Waves:

1. **Blues**

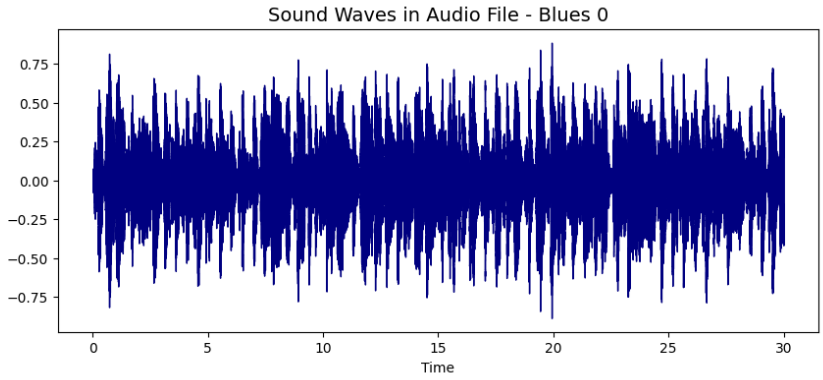

Figure 1.1 - Sound wave of genre “Blue”

2. Classical

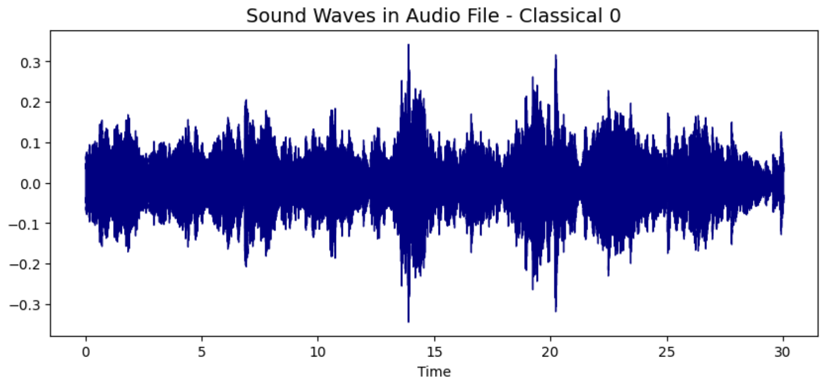

Figure 1.2 - Sound wave of genre “Classical”

3. Country

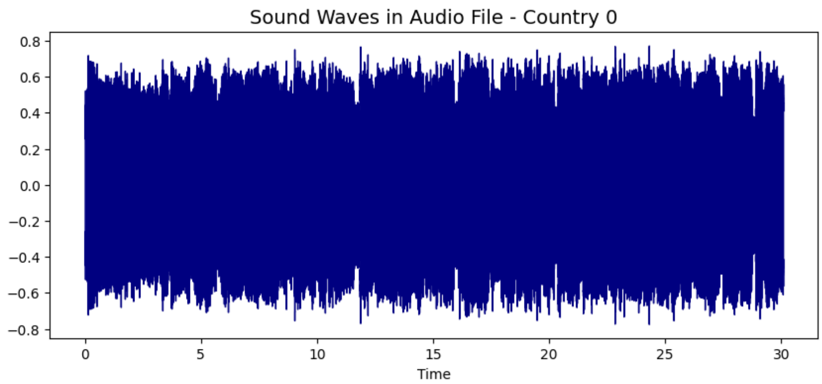

Figure 1.3 - Sound wave of genre “Country”

4. Disco

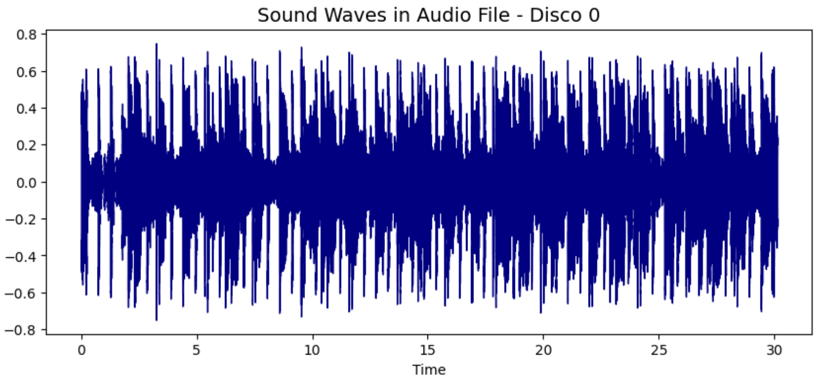

Figure 1.4 - Sound wave of genre “Disco”

5. **Pop**

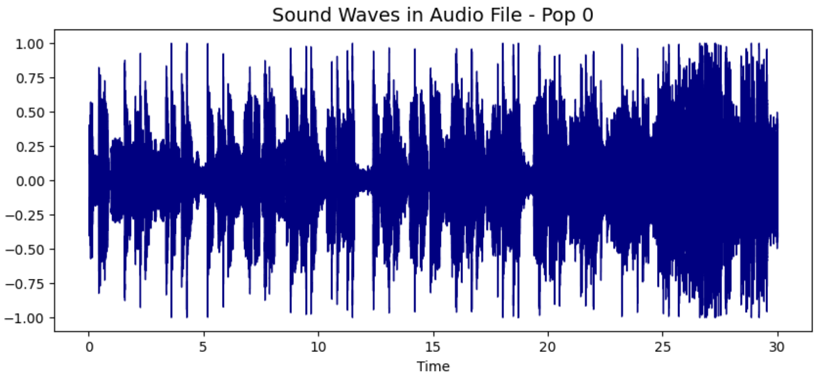

Figure 1.5 - Sound wave of genre “Pop”

6. **Hip-Hop**

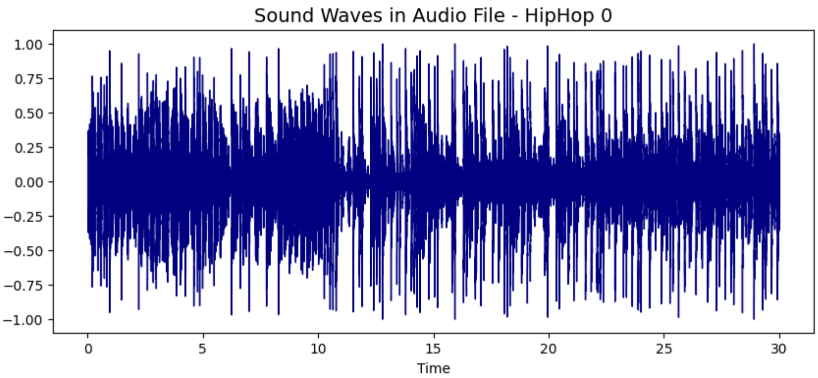

Figure 1.6 - Sound wave of genre “HipHop”

7. **Jazz**

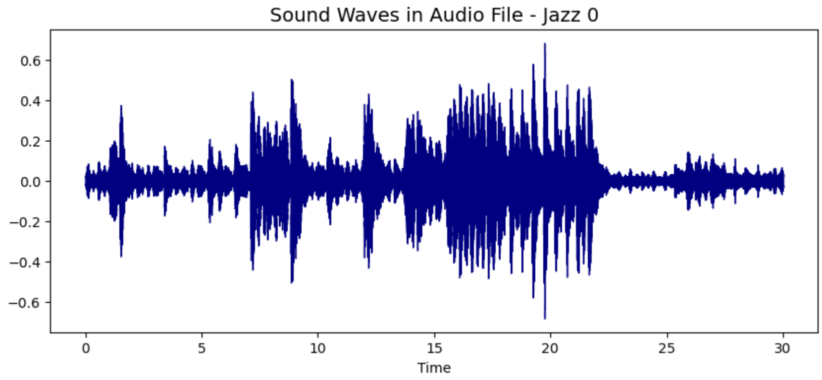

Figure 1.7 - Sound wave of genre “Jazz”

8. Metal

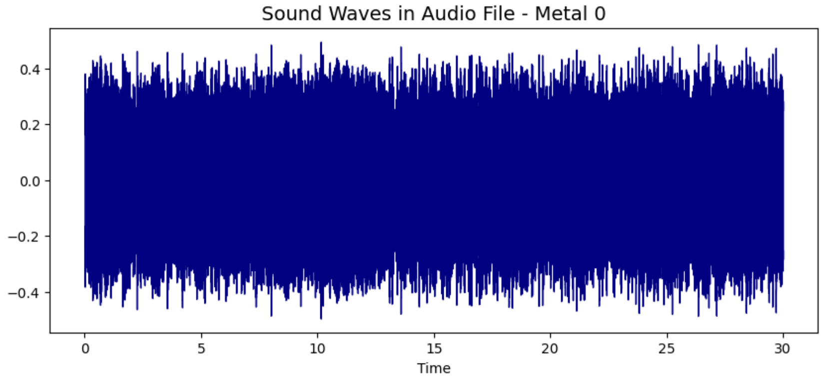

Figure 1.8 - Sound wave of genre “Metal”

9. Reggae

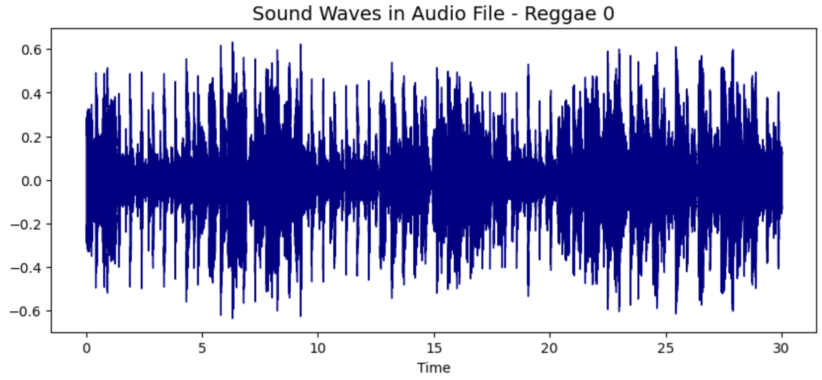

Figure 1.9 - Sound wave of genre “Reggae”

10. Rock

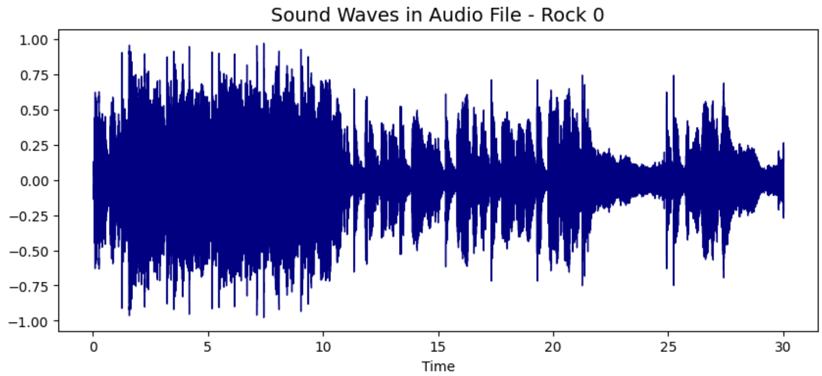

Figure 1.10 - Sound wave of genre "Rock"

As we can observe the unique patterns and structures for each musical style. These visual representations serve as a starting point for extracting relevant features, such as rhythm, tempo, and timbre, which are essential for training our genre classification model.

Now from below we will be visualizing only one of genres to make things easier.

4. **Audio Features Extraction:** Below extracted features provide a numerical representation of various aspects of audio, like frequency content, rhythm, tonal characteristics, timbre, etc.
Note: From now onwards all these features are generated for the "**Blue**" genre.

Features:

1. **Spectrogram**: It shows visual representation of spectrum of frequencies of a signal as it changes with time. It shows how intensity of different frequencies in audio changes over time.

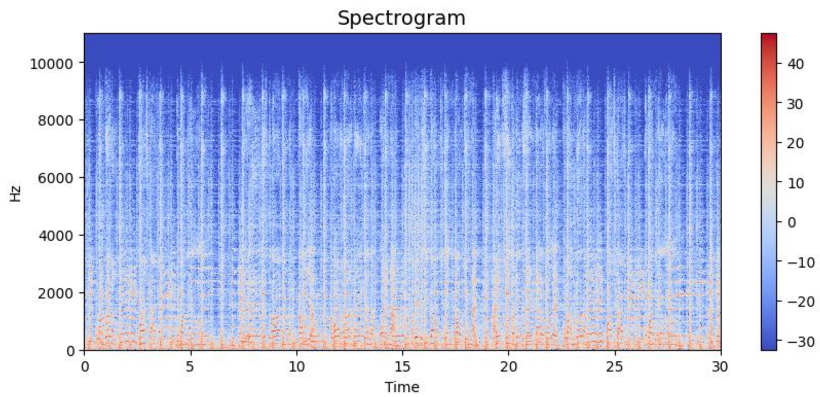

Figure 2.1 - Spectrogram for "Blue"

2. **Harmonics and Perceptual:**

Harmonics - these are multiples of fundamental frequency of a sound. In audio analysis, detecting harmonics helps in understanding the tonal quality and timbre of the sound.

- Harmonics are Multiples: Now, when you create that fundamental tone, the string or the air column also vibrates in other ways at the same time. These extra vibrations create higher-pitched sounds that blend together with the fundamental. These higher-pitched sounds are called "harmonics."

Perceptual - are features based on how humans perceive sound, such as loudness, pitch, and timbre. It's about how our brain interprets the loudness, sharpness and other characteristics of sound.

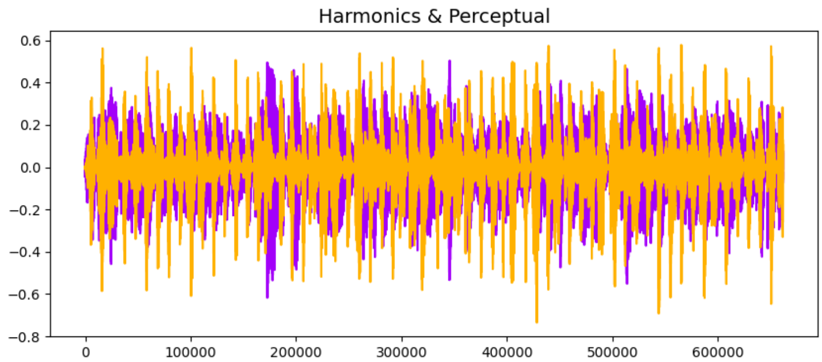

Figure 2.2 - Harmonics & Perceptual

3. **Zero - Crossing Rate:** It shows the no.of times the signal crosses the zero-amplitude axis in a paricular time frame. It provides information about the changes in the waveform and can be indicative of the pitch and timbre of the audio. **Example**, a signal with a lot of zero crossings might be percussive, like a drum hit.

   In our case for “Blue” Total Zero - Crossing were - **55031**

4. **Tempo BPM (beats per minute):** It is the speed/pace of a piece of audio, typically measured in Beats Per Minute (BPM). It reflects how fast or slow the rhythm of the music is and is essential for understanding the genre's energy and mood. **Example** Hiphop will usually have more beats than Classical.

   In our case for “Blue” Tempo was - 123.04 beats per second

5. **Spectral Centroid:** It shows the "center of mass" of spectrum of frequencies present in the signal. It also shows where the "average" frequency of the sound is located.

   It helps us in understanding the distribution of frequency components in a sound, indicating whether it is more dominated by low or high frequencies.

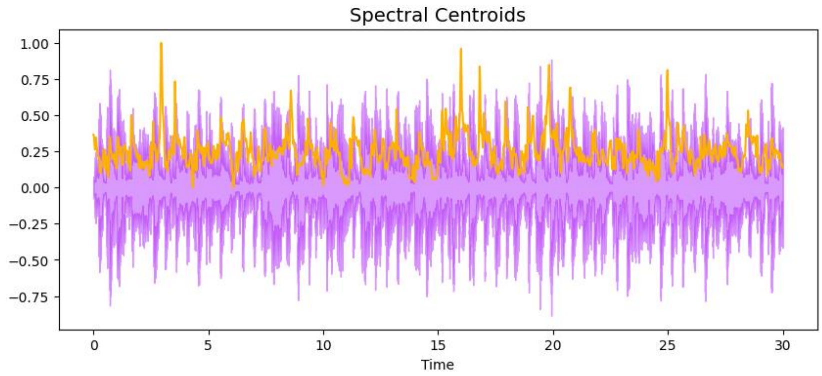

Figure 2.3 - Spectral Centroids

6. **Mel Frequency Cepstral Coefficients (MFCCs):** These are a set of features that shows short-term power spectrum of a sound. It mimics the human auditory system's response to sound by measuring the distribution of energy in different frequency bands. MFCCs are commonly used for speech and music analysis tasks.

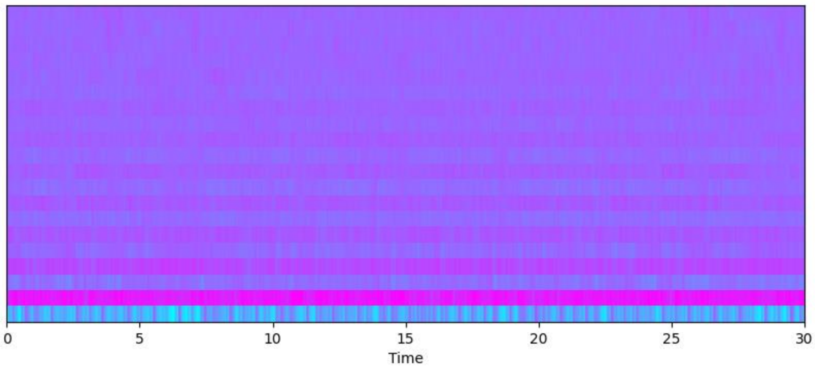

Figure 2.4 – MFCCs

7. **Chroma Frequencies:** These represent the distribution of pitch classes (notes or musical tones) in the audio signal. There are 12 pitches/notes (A, C, F, etc)

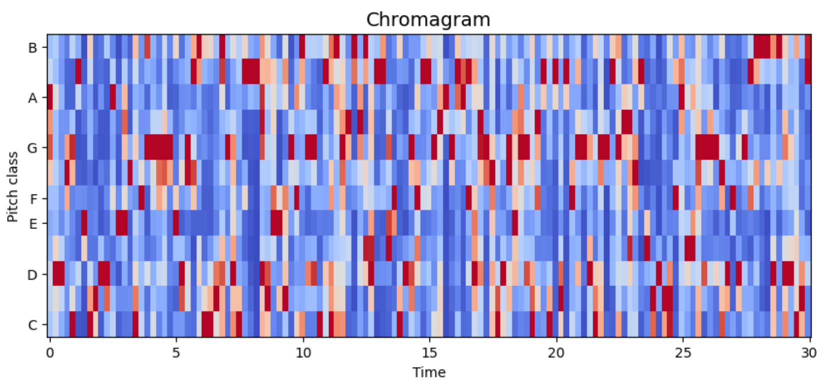

Figure 2.5 - Chroma Frequencies

By analyzing these features, models can learn to distinguish between different music genres based on their unique audio signatures.

Correlation Heat Map:

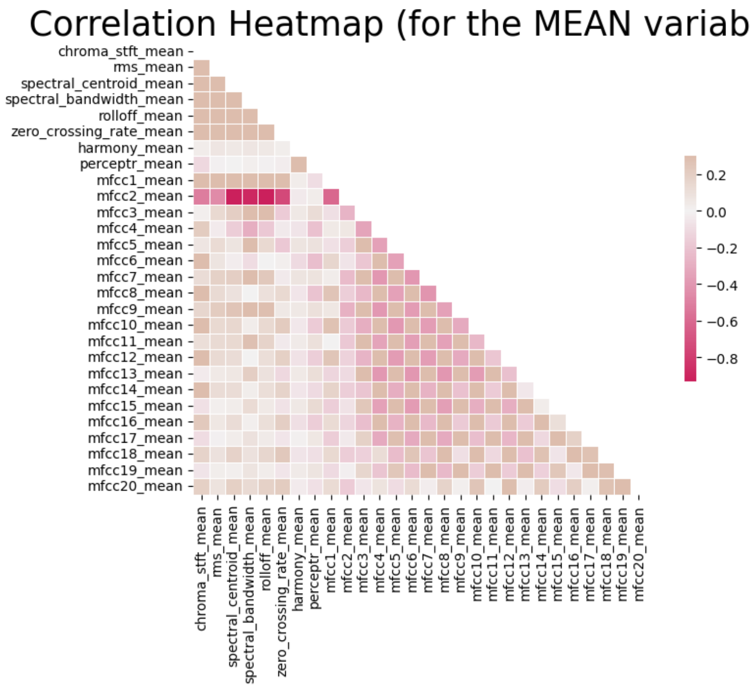

Figure 2.6 - Correlation HeatMap

5. **Splitting the Dataset:** After Pre-Processing the dataset, including scaling numerical features and label encoding the categories of genre, the next step is to divide the data into training and testing sets. This division allows us to train our model on a portion of the data and evaluate its performance on unseen data.

   The splitting is done with a test size of 0.33, meaning **33%** of the data will be used **for testing**, while the remaining **67%** will be used **for training.** This splitting ensures that the model's performance are reliable indicators of its effectiveness in classifying music genres.

6. **Build CNN Model:** We used CNN model for music genre classification, where we tried to create a robust architecture capable of learning intricate patterns from the audio features. We used a series of Dense layers to capture these patterns, along with some optimization techniques to enhance the model's efficiency and generalization on validation set.

   Model Details:

   - **Dense Layers**: we have used multiple dense layers, each with ReLU (Rectified Linear Unit) activation function. These layers help models learn complex patterns and relationships within audio features.

- **Dropout Layers**: we have added a dropout rate of 0.2 which are used to prevent overfitting by randomly deleting/dropping some of the units during training.
- **Final Layer:** The last dense layer consists of Softmax activation function and output of 10 music genres in the dataset.

Hyperparameters & Optimization Techniques:

- **Optimizer:** we have used "**Adam**" optimizer which is a widely used because of its adaptive learning rate properties and efficient convergence.
- **Early Stopping:** It is used to prevent overfitting by monitoring the validation loss. It basically will stop early if validation loss doesn't improve after certain epochs.
- **Reduce LR on Plateau:** this technique dynamically reduces the learning rate when validation loss becomes flat, which helps model navigate flat areas in loss landscape and converge faster.

Why this Approach?

- **Model Complexity**: Multi-layered Dense architecture allows us to learn complex patterns and nuances present in different music genres.
- **Regularization**: Dropout layers mitigate overfitting, ensuring the model performs well on validation set.
- **Optimization**: Adam optimizer offers fast convergence and adapts learning rates for each parameter individually, improving training efficiency.
- **Early Stopping and LR Reduction**: These techniques prevent the model from overfitting and help in fine-tuning the model's performance on the validation set.

7. Predicting Genres

After training our CNN model on the music genre dataset, we evaluated its performance on both sets (training and validation).

The visual representations of accuracy and loss provide us useful information, like model's learning progress and its ability to perform on unseen data.

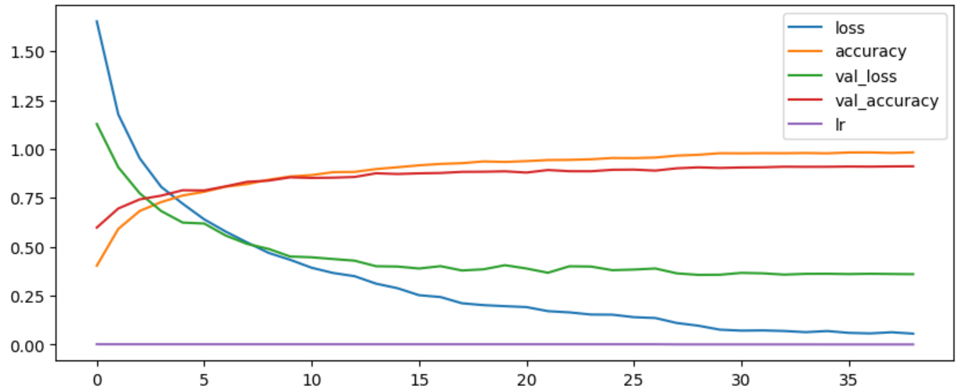

Figure 3.1 - Overall Visualization of the Model

Model Performance Visualization

- **Training & Validation Accuracies (Orange, Red):** In Figure 3.1 – shows training and validation accuracies across epochs. A rising trend in both curves indicates that model is learning effectively from the training data.
- **Training & Validation Losses (Blue, Green):** Figure 3.1 – displays the training and validation loss. As we see decrease in loss it signifies that model is gradually improving, in its ability to make accurate predictions.

Results Showcase

To find the model's predictions, we showcase a sample of the actual music genres, the predicted genres by our model, and provide the audio snippets that were classified

Figure 3.2 - Prediction of Genres

Note:

- The "**Actual Genre**" column represents the true genre labels from the dataset.

- The "**Predicted Genre**" column shows the genres predicted by our trained CNN model.

Interpretation of Results

Our model demonstrates strong performance in accurately predicting music genres across various styles. By comparing the actual and predicted genres alongside the audio snippets, we can observe the model's ability to capture the distinctive characteristics of each genre.

It showcases results which provide a glimpse into how our CNN model effectively learns and classifies music genres based on the extracted audio features.

8. Insights & Analysis of Feature Importance:

Identifying Important Features with Random Forest

To gain insights into the most influential features for predicting music genres, we utilized the Random Forest algorithm. This method allows us to rank the features based on their importance in classifying the genres accurately.

Why did we choose Random Forest?

- **Ensemble Learning**: It is an ensemble learning technique that combines multiple decision trees to make predictions. This approach often leads to more robust and accurate models.

- **Feature Importance**: It also provides a straightforward way to determine feature importance. By analyzing the splits made in the trees, it assigns weights to features based on their contribution to reducing impurity in the nodes.

Interpretation of Feature Importance Plot

- **Top Features**: The bar plot below (Figure) shows the top 10 most important features for predicting music genres.
- **Feature Importance Scores**: Each bar represents the importance score of a feature, indicating its contribution to the model's decision-making process.

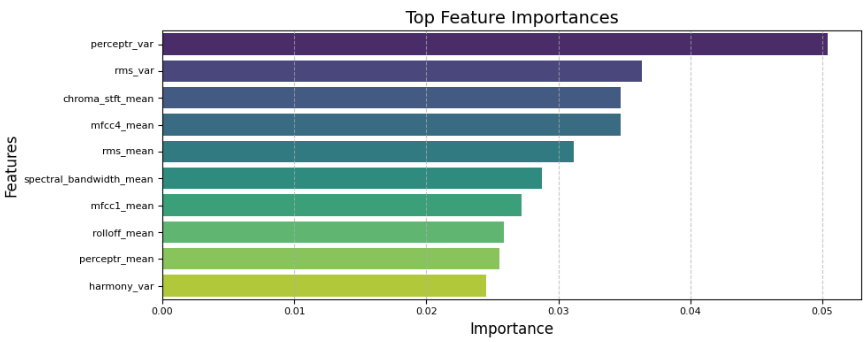

Figure 4.1 – Top Feature Importance

Importance of Feature Analysis

- **Model Understanding**: Feature importance helps in interpreting the model's decisions and provides insights into the genre classification process.
- **Feature Engineering**: Insights from important features can guide further feature engineering efforts, such as selecting relevant features or creating new ones.
- **Music Genre Characteristics**: The identified features reflect key characteristics of music genres, such as tonal quality, rhythm, and melodic structure, etc.

By analyzing feature importances, we gain valuable insights into the significant aspects of audio signals that influence our model's ability to classify music genres accurately.

6. Conclusion and Future Work

6.1 Conclusion

In this project, we embarked on a journey to develop a Convolutional Neural Network (CNN) model for Music Genre Classification using the GTZAN dataset. Here are the key highlights and findings from our work in brief:

- **Dataset and Preprocessing**: We utilized the GTZAN dataset, which contains audio excerpts from 10 different music genres. Preprocessing involved extracting essential features such as spectrograms, MFCCs, and chroma frequencies.
- **Model Development**: Our CNN model architecture, consisting of dense layers with ReLU activation and dropout, proved effective in capturing intricate patterns in the audio features.
- **Optimization Techniques**: We utilized Adam optimizer along with early stopping and learning rate reduction to enhance the model's training efficiency and prevent overfitting.
- **Insights from Feature Importance**: Using the Random Forest algorithm, we identified the most important features for genre classification, such as perceptual, MFCCs, chroma frequencies, etc.
- **Prediction Results**: We visualized the model's performance on training and validation sets, showcasing its ability to generalize well to unseen data.

6.2 Future Work

- **Develop User Interface:** Creating a user-friendly interface where users can upload their audio files for genre classification would enhance accessibility and usability by utilizing frameworks like Streamlit or Flask to build a web application, allowing users to interact with the model seamlessly.
- **Genre Confidence Scores:** Providing users with confidence scores for each predicted genre, indicating the model's certainty in its predictions.
- **Fine -Tuning Model:** Training the model on a larger and more diverse dataset to improve its genre classification accuracy and robustness.

6.3 Final Thoughts

In conclusion, our Music Genre Classification project demonstrates the potential of deep learning models in automatically categorizing music based on its audio features. The insights gained from feature analysis and model performance pave the way for further advancements in this field.

References

Literature References:

[1] Chaudhury, M.; Karami, A.; Ghazanfar, M.A. Large-Scale Music Genre Analysis and Classification Using Machine Learning with Apache Spark. *Electronics* **2022**, *11*, 2567. https://doi.org/10.3390/electronics11162567

[2]Islam, Md & Hasan, Monirul & Rahim, Md Abdur & Hasan, Ali & Mynuddin, Mohammed & Khandokar, Imran & Islam, Jabbarul. (2021). Machine Learning-Based Music Genre Classification with Pre- Processed Feature Analysis. 7. 491-502. 10.26555/jiteki.v7i3.22327.

[3] Zhang, W. (2022) 'Music Genre Classification Based on Deep Learning', Mobile Information Systems. Hindawi, 2022, p. 2376888. doi: 10.1155/2022/2376888.

[4] Haggblade, Michael, Yang Hong and Kenny Kao. "Music Genre Classification." (2011).

Implementation References:

- https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification/data
- https://towardsdatascience.com/music-genre-classification-with-python-c714d032f0d8
- https://towardsdatascience.com/best-practice-to-calculate-and-interpret-model-feature-importance-14f0e11ee660

4% Overall Similarity

Top sources found in the following databases:

- 3% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database
- 4% Submitted Works database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	mdpi.com Internet	<1%
2	CSU Northridge on 2023-04-01 Submitted works	<1%
3	5dok.net Internet	<1%
4	coursehero.com Internet	<1%
5	fastercapital.com Internet	<1%
6	University of Witwatersrand on 2020-12-01 Submitted works	<1%
7	journal.uad.ac.id Internet	<1%
8	Lecture Notes in Computer Science, 2013. Crossref	<1%

9 University of East London on 2023-05-11
Submitted works
<1%