Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2026. Vol. 5. No. 2; e-ISSN: 2808-4519

Music Genre Classification Application Based on Audio Features with Ensemble Learning Algorithm

Fahmi Raditya[1]*, Adisty Ramadhani[2], Syafiq Nabil Assirhindi[3], Riski Annisa[4]

[1,2,3,4]*Informatics Study Program, Universitas Bina Sarana Informatika, Indonesia*
15230265@bsi.ac.id[1*], riski.rnc@bsi.ac.id[2]

Abstract

With the exponential growth of digital music, manual genre-labeling has become ineffective. Consequently, automatic music genre classification is crucial for data management and recommendation systems. This research aims to develop an accurate music classification application by comparing individual machine learning models against advanced **Ensemble Learning** techniques. The methodology involved extracting 26 audio features from the **GTZAN dataset**, followed by training and hyperparameter tuning ten models, including Random Forest, SVM, XGBoost, and LightGBM. The findings demonstrate that ensemble methods significantly outperform individual models. The highest performance was achieved by a **Voting Classifier**, which combines the predictive strengths of SVM, XGBoost, and Logistic Regression, reaching a final test accuracy of **72%**. This superior ensemble model was then successfully implemented into an interactive web application using Streamlit, proving that this approach is not only highly accurate but also functional for real-time, practical applications.

Keywords: *Classification, Music Genre, Audio Features, Machine Learning, Ensemble Learning.*

1. Introduction

Music is a fundamental medium for expressing emotions and cultural identity. With digitization, the volume of music data has grown massively, creating challenges in content organization and discovery. Music genres function as the primary taxonomic framework for navigation and recommendation; however, manual labeling is no longer efficient. Therefore, an automated Music Genre Classification (MGC) system is a necessity. This system utilizes Machine Learning (ML) and Deep Learning (DL) to analyze audio signals and identify genre patterns objectively.

Several previous studies have explored various algorithms for MGC. The use of Convolutional Neural Network (CNN) has been reported to successfully achieve a test accuracy of 81.33% **[1]**. A comparison between Extreme Gradient Boosting (XGBoost) and Decision Tree (DT) showed that XGBoost achieved superior performance with 72% accuracy compared to DT's 51% **[2]**. Furthermore, it was found that Naive Bayes reached its highest accuracy of 58.91% on the Spotify dataset **[3]**.

Despite many algorithms being tested, there is still a need for systematic comparative analysis and practical implementation of the best models. This research aims to compare the performance of several ML and DL algorithms on the GTZAN dataset, which is an industry standard. Furthermore, this study does not stop at model evaluation but implements the best-performing model into a functional web application using Streamlit. Thus, this research bridges the gap between theoretical analysis and practical applications accessible to end-users.

2. Literature Review

This literature review outlines the theories underlying the research, including the concepts of classification, machine learning, and relevant previous studies.

2.1. Classification and Machine Learning

Classification is the process of organizing objects into specific classes based on data patterns [4]. In this context, machine learning is utilized to design algorithms capable of learning from audio feature data to automatically assign genre labels (classes) to a music track (object) [5]. One of the most reliable classification algorithms is the Support Vector Machine (SVM). SVM aims to find the optimal

separating boundary (hyperplane) to distinguish between classes. This algorithm is highly effective in handling high-dimensional feature spaces, which is significantly relevant for complex audio data rich in features such as MFCC, chroma, and spectral centroid [3].

2.2. Ensemble Learning

Ensemble learning is a machine learning technique where several models (referred to as base learners) are combined to produce a prediction model that is stronger and more stable than individual models. The underlying idea is that the "collective wisdom" of many models will reduce errors and overfitting. This research focuses on several key ensemble methods:

a) **Random Forest (RF):** An ensemble method that combines multiple decision trees and utilizes voting to determine the final result, effectively reducing overfitting [3].
b) **Gradient Boosting (GBM, XGBoost, LightGBM):** These methods build models sequentially, where each new model focuses on correcting the errors (residuals) of the previous model. XGBoost and LightGBM are highly efficient gradient boosting implementations known for high performance [2].
c) **Voting Classifier:** This method combines predictions from several different models (e.g., SVM, RF, LR). In soft voting (used in this study), predictions are based on the average probability of each model, which often yields higher accuracy [4].
d) **Stacking Classifier:** A more complex ensemble method where a meta-learner model is trained to combine predictions from several base learner models as its input.

2.3. Support Vector Machine (SVM)

SVM is an algorithm that aims to find the optimal separation boundary (hyperplane) to distinguish between classes. It is highly effective in handling high-dimensional feature spaces, which is relevant for complex audio data, and often shows superior performance in classification tasks.

2.4. Random Forest (RF)

RF is an ensemble method that combines many decision trees to improve accuracy and reduce overfitting. RF works by building a number of trees on different data samples and taking the majority vote for the final prediction.

2.5. Extreme Gradient Boosting (XGBoost)

XGBoost is an efficient and scalable ensemble classification method. This algorithm works sequentially, where each new model is built to minimize the errors (residuals) of the previous model, thus capable of producing high accuracy.

2.6. Convolutional Neural Network (CNN)

CNN is a Deep Learning architecture commonly used for image data. In music classification, audio signals are converted into 2D visual representations such as spectrograms, which can then be analyzed by the CNN to recognize spectral patterns typical for each genre.

2.7. Audio Feature Extraction

The quality of classification is heavily dependent on the features extracted from the audio signal. These features represent the acoustic characteristics of music numerically. Several key features used in this research include:

a) **Mel-Frequency Cepstral Coefficients (MFCC):** The most dominant feature for audio classification, as it captures timbre (sound color) characteristics essential for distinguishing genres by mimicking human hearing [6].
b) **Chroma STFT:** Represents the energy distribution across 12 pitch classes (C, C#, D, etc.), which is useful for analyzing harmonic and melodic aspects.
c) **Spectral Centroid:** Identifies the "center of mass" of the sound spectrum. High values are often associated with brighter sounds (e.g., cymbals in the Metal genre).

2.8. Related Works

Previous research serves as a reference and comparison for this study. A comparison of four algorithms (Naive Bayes, K-NN, Random Forest, and SVM) on the GTZAN dataset showed SVM as the best model with 83% accuracy, followed by RF (73%), K-NN (71%), and Naive Bayes (66%) . Other studies comparing XGBoost and Decision Tree (DT) found that XGBoost was significantly superior with 72% accuracy compared to DT's 51% .

Analysis of CNN usage on music classification with different durations found that 10-second audio clips reached 81% accuracy, higher than 30-second clips at 58% . Furthermore, the implementation of Genetic Algorithms (GA) for feature selection on Random Forest was able to increase classification accuracy to 67% . Research focusing on the Naïve Bayes Classifier found that the Gaussian Naïve Bayes variant provided the best accuracy of 63% on the GTZAN dataset.

Synthesis of the literature review shows significant variability in music genre classification accuracy, even when using the same dataset (GTZAN), with reported ranges from 58% to 83% . This variation indicates that model performance is highly sensitive to factors such as audio duration, feature extraction methods, and validation techniques. Therefore, this research does not aim for the highest absolute

accuracy but to perform a systematic comparison under controlled conditions to identify the most robust model for practical implementation. Accuracy results in the range of 60-72% are considered valid and consistent findings within this research landscape.

3. Research Methodology

The methodology of this research is organized following systematic stages adapted from the Cross-Industry Standard Process for Data Mining (CRISP-DM) framework. This framework ensures a structured research process ranging from data understanding to evaluation and deployment. The research flow is illustrated in Figure 1.

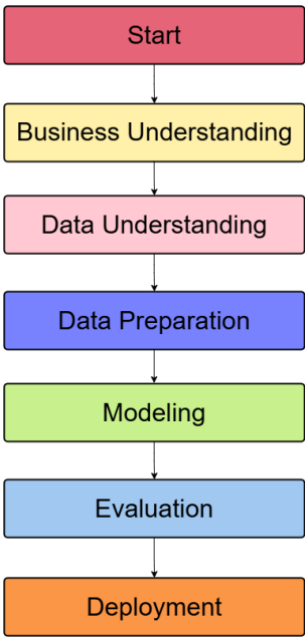

Fig. 1: Research Flowchart

3.1. Research object: GTZAN dataset

The object used in this study is the GTZAN public dataset, which is a de-facto standard for music genre classification tasks. This dataset consists of 1,000 audio samples, where each sample is a 30-second audio clip. The samples are balanced and distributed into 10 music genres: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock. Each genre is represented by 100 audio samples.

3.2. Feature extraction and identification

Every audio file in the dataset was processed to extract a series of acoustic features that represent sound characteristics numerically. Key features used in this research include Mel-Frequency Cepstral Coefficients (MFCC) which capture timbre, Chroma STFT representing harmonic aspects, and various other spectral features such as Spectral Centroid, Spectral Bandwidth, Spectral Rolloff, RMSE, and Zero-Crossing Rate. A total of 26 numerical features were used as predictors for the classification process.

3.3. Data pre-processing

Before the modeling stage, the raw data underwent several pre-processing steps to ensure quality and readiness:

a) **Label Encoding:** Categorical genre labels (e.g., "blues", "rock") were converted into numerical representations (0, 1, 2, etc.) using Scikit-learn's LabelEncoder so they could be processed by algorithms.
b) **Feature Scaling (Normalization):** To equalize the value ranges between features and optimize the performance of algorithms such as SVM and MLP, scaling was performed using StandardScaler. This process transforms data to have a mean of 0 and a standard deviation of 1.
c) **Dataset Splitting:** The cleaned dataset was divided into two proportional parts: 80% for training data and 20% for testing data. This split was performed stratifically to ensure each genre's proportion remains identical in both sets.

3.4. Modeling and testing scenarios

At this stage, ten different machine learning models were trained using the training data. These models include Logistic Regression, Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), Naive Bayes, Gradient Boosting, MLP Classifier, XGBoost, and LightGBM. To obtain optimal performance, each model underwent hyperparameter tuning using GridSearchCV. After evaluating all individual models, the top three with the best cross-validation scores (SVM, XGBoost, and Logistic Regression) were selected to build two advanced ensemble models:

a) **Voting Classifier:** Utilized a soft voting strategy to combine prediction probabilities from the top three models.

b) **Stacking Classifier:** Utilized the top three models as base learners and a meta-learner model (Logistic Regression) to make final predictions.

3.5. Performance evaluation metrics

The performance of each model was evaluated based on its predictions on the unseen test data. Accuracy was the primary metric used for overall comparison. Additionally, an in-depth analysis was conducted on the best model using a Confusion Matrix as well as Precision, Recall, and F1-Score metrics for each genre to understand the model's specific strengths and weaknesses.

4. Results and Discussion

This chapter presents the results of the experiments conducted, ranging from model performance comparison to the analysis of practical application trials .

4.1. Algorithm performance comparison

After an extensive hyperparameter tuning process and evaluation on the test data, the performance comparison of all models was obtained. The accuracy results are presented in Fig. 2 .

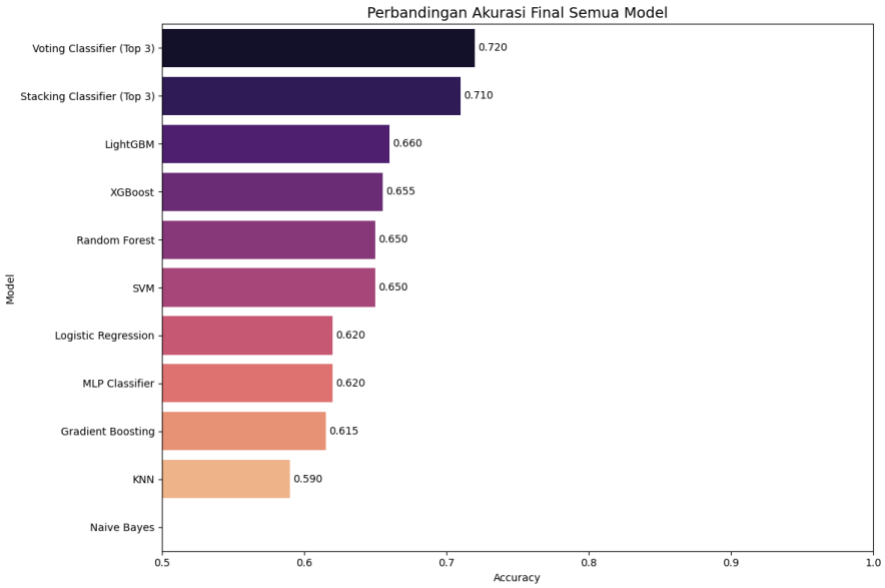

Fig. 2: Comparison of Final Accuracy of All Models

The experimental results shown in Fig. 2 reveal two performance groups. Individual models that have been tuned, such as LightGBM (66.0%), XGBoost (65.5%), SVM (65.0%), and Random Forest (65.0%), demonstrate solid and relatively comparable performance within the 65-66% range. However, a significant breakthrough was achieved by the Ensemble Learning models. The Voting Classifier reached the highest accuracy of 72.0%, followed by the Stacking Classifier at 71.0%. These results prove the research hypothesis that ensemble techniques can significantly outperform any individual model by leveraging their collective predictive power.

4.2. Best model analysis (Voting Classifier)

The experimental results shown in Fig. 2 reveal two performance groups. Individual models that have been tuned, such as LightGBM (66.0%), XGBoost (65.5%), SVM (65.0%), and Random Forest (65.0%), demonstrate solid and relatively comparable performance within the 65-66% range. However, a significant breakthrough was achieved by the Ensemble Learning models. The Voting Classifier reached the highest accuracy of 72.0%, followed by the Stacking Classifier at 71.0%. These results prove the research hypothesis that ensemble techniques can significantly outperform any individual model by leveraging their collective predictive power.

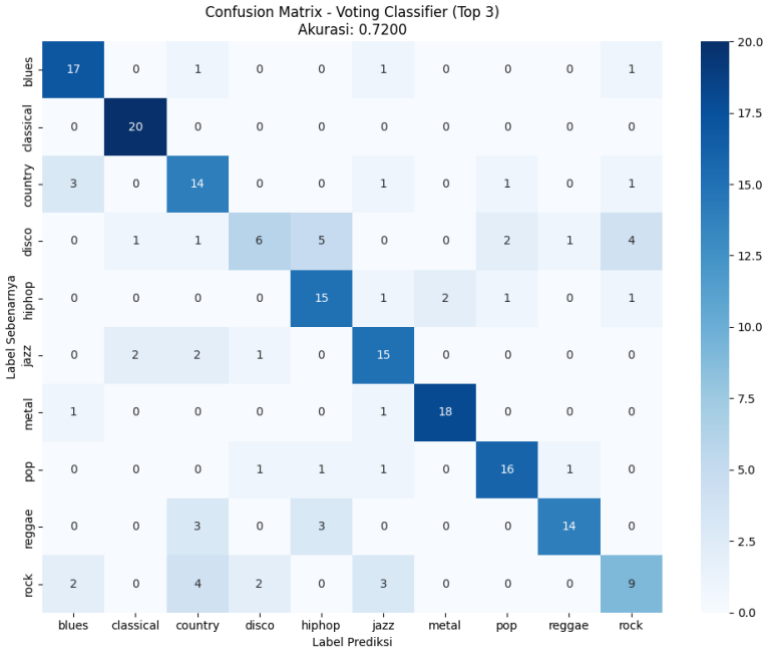

Fig. 3: Confusion Matrix for Voting Classifier Model

Analysis of the data reveals several key findings:

a) **Superior Performance on Unique Genres:** The ensemble approach shows an outstanding ability to identify genres with distinct audio characteristics. Classical was recognized with perfect recall (100%), and Metal also achieved high performance with a 90% F1-score.

b) **Balanced Performance Improvement:** Compared to individual models, the Voting Classifier improved performance evenly across many genres. Genres such as Pop (80% F1-score), Blues (79% F1-score), and Reggae (78% F1-score) now show solid and reliable classification levels.

c) **Challenges in Ambiguous Genres:** The confusion matrix clearly indicates areas of difficulty. The Disco genre was the hardest to recognize (30% recall) and was often misclassified as Country, Hiphop, or Pop, likely due to overlapping rhythmic features and basslines .

d) **Persistent "Rock" Genre Issues:** The Rock genre (45% recall) remains a challenge, even though it performs better than individual models. The model still frequently confuses it with other guitar-based genres like Country and Blues, or high-distortion genres like Metal .

4.3. Implementation and practical application trial

A major contribution of this research is the implementation of the trained Voting Classifier model into a functional web application using the Streamlit framework . As a case study, a test was conducted using the song "All That She Wants" by Ace of Base, which is musically known as a Reggae Pop song. The prediction results are shown in Fig. 4.

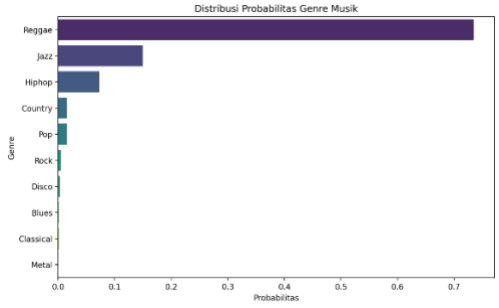

Fig. 4: Application Prediction Results for the song "All That She Wants"

5. Conclusion

This study has demonstrated that ensemble learning techniques significantly improve the accuracy of music genre classification compared to individual machine learning models. Through systematic evaluation and hyperparameter tuning of ten different models using the GTZAN dataset, the highest performance was achieved by the Voting Classifier, which combined the predictive strengths of SVM, XGBoost, and Logistic Regression to reach a final test accuracy of 72%. The model showed exceptional performance in identifying unique genres like Classical (100% recall) and Metal (90% F1-score). Furthermore, the successful implementation of this model into an interactive web

application using Streamlit proves that the proposed approach is not only theoretically accurate but also functional for real-time practical applications.

6. Suggestions

For future development, several approaches can be explored to overcome current limitations:

a) **Deep Learning Integration:** Transitioning from manual feature extraction to automatic representation learning using 2D Convolutional Neural Networks (CNN) with Mel Spectrograms could potentially capture more complex timbre patterns and increase accuracy beyond 72%.
b) **Multi-label Classification:** Addressing genre ambiguity in songs that blend multiple styles (e.g., Disco and Rock) by implementing multi-label classification to provide more nuanced predictions.
c) **Dataset Modernization:** Expanding the research using larger and more contemporary datasets, such as the Free Music Archive (FMA), to improve the model's generalization to modern music landscapes.
d) **Functional Enhancements:** Improving the web application by adding Spotify API integration for official metadata comparison and visualizing genre relationships using t-SNE or PCA plots.

Acknowledgement

The authors would like to express their gratitude to Universitas Bina Sarana Informatika for providing the support and resources necessary to conduct this research. We also thank the creators of the GTZAN dataset for making their data publicly available for scientific advancement.

References

[1.] C. R. Wairata, E. R. Swedia, and M. Cahyanti, "Pengklasifikasian Genre Musik Indonesia Menggunakan Convolutional Neural Network," *Sebatik*, vol. 25, no. 1, pp. 255–261, 2021, doi: 10.46984/sebatik.v25i1.1286 .
[2.] I. Komputer et al., "Perbandingan Metode Extreme Gradient Boosting dan Decision Tree pada Klasifikasi Genre Musik," pp. 373–382, 2023.
[3.] S. Navisa, L. Hakim, and A. Nabilah, "Komparasi Algoritma Klasifikasi Genre Musik pada Spotify Menggunakan CRISP-DM," *J. Sist. Cerdas*, vol. 4, no. 2, pp. 114–125, 2021, doi: 10.37396/jsc.v4i2.162.
[4.] G. Z. Dhamara and A. Nugroho, "Klasifikasi Genre Musik Menggunakan Machine Learning," vol. 6, no. 3, 2025, doi: 10.47065/bit.v5i2.2021.
[5.] Y. V. Via, I. Y. Purbasari, and A. P. Pratama, "Analisa Algoritma Convolution Neural Network (Cnn) Pada Klasifikasi Genre Musik Berdasar Durasi Waktu," *Scan J. Teknol. Inf. dan Komun.*, vol. 17, no. 1, pp. 35–41, 2022, doi: 10.33005/scan.v17i1.3251 .
[6.] T. Nurchaidir, Widodo, and B. P. Adhi, "Klasifikasi Genre Musik Menggunakan Algoritma Naïve Bayes Classifier Untuk Layanan Streaming Youtube," *PINTER J. Pendidik. Tek. Inform. dan Komput.*, vol. 7, no. 1, pp. 1–6, 2023, doi: 10.21009/pinter.7.1.1 .
[7.] N. Amini, T. H. Saragih, M. R. Faisal, and A. Farmadi, "Pada Klasifikasi Genre Musik Menggunakan Metode Random Forest dan Algoritma Genetika," pp. 75–82, 2020.