Hit Song Prediction Using Machine Learning and Spotify Data

Author: Andrej Natev
Email: 89221050@student.upr.si
Affiliation:
Faculty of Mathematics, Natural Sciences and Information Technologies (FAMNIT)
University of Primorska
Glagoljaška 8
Koper, Slovenia

Contents

Date: May 29, 2024

Hit Song Prediction Using Machine Learning and Spotify Data

ABSTRACT

This study aims to predict hit songs using metadata from the Spotify API. The dataset comprises over 20 genres, each with 40 songs equally divided between hits and non-hits, sourced directly from the Spotify web API using spotipy. Models were trained on various metadata features including danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, and tempo. The dataset was partitioned using three techniques: train_test_split (test set of 10%, 20%, and 33%), and standard and stratified kfold cross-validation with k values of 2, 5, and 10.

Models were trained, evaluated, and tested, with performance assessed using cross-validation means and classification reports. Utilizing scikit-learn's models, including ensemble models and MLPClassifier neural network, which showed a 60% accuracy as did SVC, with SGD as a runner up using the LBFGS on the identity function. Noteworthy performances were observed from extra trees and random forest as ensemble models, and Gaussian Process/NaiveBayes and ridge classifiers as more standard models. These findings suggest promising accuracy in hit song prediction using Spotify API audio features. Developing bespoke models, particularly neural networks or decision tree ensembles, could enhance predictive efficacy. Prospective research avenues, including frequency and lyric analysis, hold potential for uncovering the hit song formula.

KEYWORDS

music, genre, song, Spotify, machine learning, classification, prediction, ensemble model, support vector, neural network

1 INTRODUCTION

This research delves into the intersection of music and data science, leveraging the Spotify Web API in conjunction with the Spotipy library and a random add-on. By harnessing these tools, the study aims to extract and analyze track data across various genres. The primary objective is to develop machine learning models capable of categorizing songs into two distinct groups: "hit" and "flop", based on a range of audio features and metadata.

2 MATERIALS AND METHODS

2.1 Materials

In this study, two datasets from Kaggle were utilized: "Most Streamed Songs 2023" and "30000 Spotify Songs". These datasets provided a rich source of music metadata for analysis. Spotipy, a Python library, was used for data extraction from the Spotify Web API. Pandas and NumPy were employed for data manipulation, while the Scikit-learn library facilitated preprocessing, data splitting, model implementation, and evaluation.

2.2 Best Models

2.2.1 MLPClassifier with tanh activation function. This neural network model consists of multiple layers of nodes, each connected to the next layer, and utilizes the tanh activation function to introduce non-linearity to the model. By leveraging this activation function, the MLPClassifier effectively captured complex patterns in the data, leading to accurate predictions of song hits.

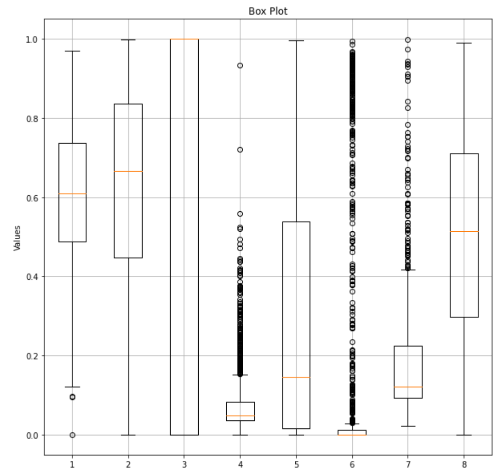

Figure 1: Feature Outlier Representation

2.2.2 RandomForestClassifier (criterion='gini'). This ensemble learning method constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. The Gini impurity measure is used to evaluate the quality of a split in the dataset. By combining multiple decision trees, the RandomForestClassifier mitigates overfitting and improves generalization, making it effective for classifying songs into hits and non-hits.

2.2.3 AdaBoostClassifier. AdaBoost (Adaptive Boosting) is an ensemble learning method that combines multiple weak classifiers to build a strong classifier. In each iteration, AdaBoost adjusts the weights of incorrectly classified instances, allowing subsequent weak classifiers to focus more on difficult cases. By iteratively improving the model's performance, AdaBoost enhances classification accuracy, making it suitable for predicting song hits based on various features.

2.2.4 GradientBoostingClassifier. Gradient boosting is another ensemble learning technique that builds a strong classifier by sequentially adding weak learners, typically decision trees, to minimize the loss function. Unlike AdaBoost, which adjusts the weights of instances, gradient boosting fits each new model to the residual errors made by the previous models. By minimizing the overall error, GradientBoostingClassifier improves prediction accuracy and generalization performance, making it effective for song hit prediction.

2.2.5 SVC (Linear Kernel). SVC is a supervised learning algorithm that constructs a hyperplane or set of hyperplanes in a high-dimensional space to separate instances of different classes. The linear kernel computes the dot product of feature vectors, making it suitable for linearly separable data. By maximizing the margin between

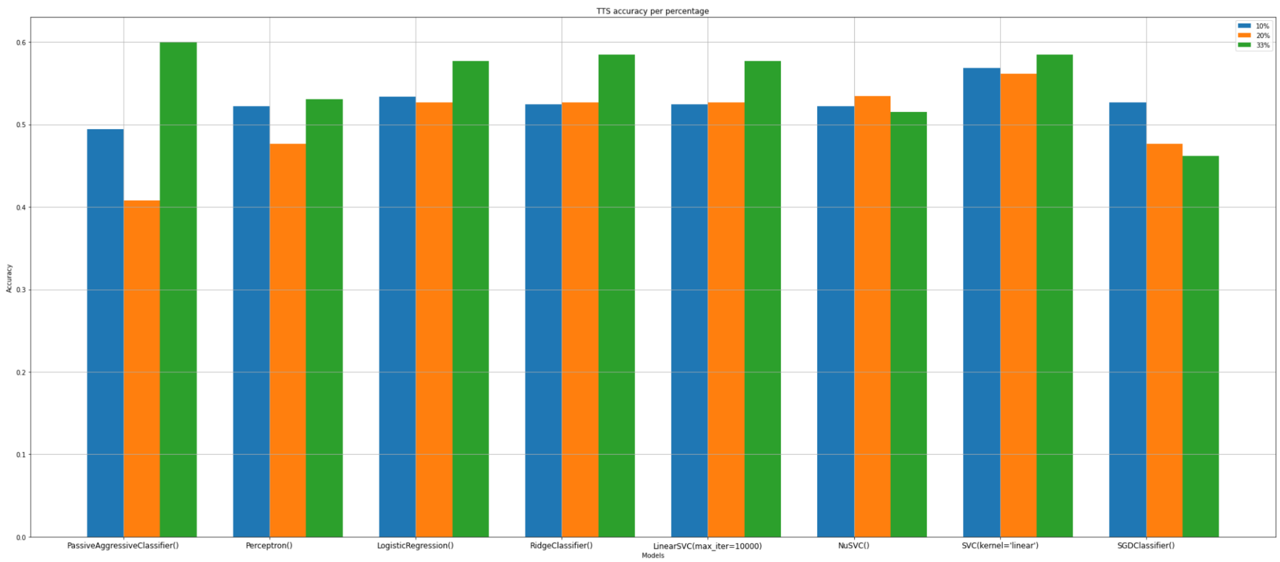

Figure 2: TTS percentages for common models

classes, SVC with a linear kernel effectively classifies songs into hits and non-hits based on the provided features.

2.2.6 *GaussianNB.* Naive Bayes is a probabilistic classifier based on Bayes' theorem and the assumption of independence between features. GaussianNB assumes that continuous features follow a Gaussian distribution, making it suitable for data with continuous features. Despite its simplicity and the "naive" assumption of feature independence, GaussianNB often performs well in practice and is computationally efficient, making it a valuable model for song hit prediction tasks.

2.3 Methods

2.3.1 *Data Understanding.* The first step in our methodology involved understanding the datasets and the task at hand. Initially, songs were obtained from the Spotify Web API based on their popularity ratings, ensuring a balanced representation of hits and non-hits. The popularity feature served as a crucial criterion for categorizing songs as hits or non-hits, with hits defined as songs with a popularity score of 60 or above, and non-hits as songs with a popularity score below 60.

2.3.2 *Data Extraction.* Data extraction was conducted using the Spotipy library along with the Spotipy-random add-on. The extraction process involved sourcing track data directly from the Spotify Web API. In addition to leveraging Spotipy-random, genres were carefully selected to ensure diversity and fairness in the dataset. These genres ranged from unpopular to popular and encompassed different audio features to ensure distributivity across the dataset. During the extraction process, it was observed that songs with popularity ratings above 85 and below 60 were particularly challenging to find, even when considering genres that were both unpopular and popular. This highlights the inherent difficulty in obtaining a balanced dataset, especially when targeting specific popularity ranges. Nonetheless, efforts were made to include a diverse range of songs across various genres to ensure the representativeness of the dataset for subsequent analysis and model training.

2.3.3 *Data Preparation.* Following data acquisition, the datasets were prepared for analysis by incorporating audio features obtained from the Spotify API. These features included danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, and tempo. Additionally, data types were adjusted to ensure categorical representation for key, mode, time signature, and hit categories.

2.3.4 *Model Training and Evaluation.* During the model training and evaluation phase, it was observed that KFold and StratifiedKFold cross-validation techniques encountered difficulties when handling larger splits due to the relatively small size of the dataset. With only 1299 songs available and an almost 50/50 balance between hits and non-hits, the dataset size posed challenges for these cross-validation methods to effectively cover all instances Despite these challenges, various machine learning models, including ensemble models and standard classifiers, were trained on the dataset. The performance of each model was evaluated using Train-Test-Split (TTS) with varying test sizes (10%, 20%, and 33%), as well as KFold and StratifiedKFold cross-validation techniques with different fold splits (2, 3, 5, and 10). It was found that Train-Test-Split with a test size of 20% consistently yielded optimal results across different models, while KFold with 3 folds provided the best performance among the fold splits considered.

3 RESULTS

The study aimed to predict hit songs using machine learning algorithms trained on Spotify API metadata. Results revealed varying accuracies across different models and evaluation techniques. Notably, the AdaBoost Classifier achieved the highest accuracy of 60% on a test size of 10%, while the ExtraTreesClassifier demonstrated stable performance, reaching 55.38% accuracy on a test size of 20%,

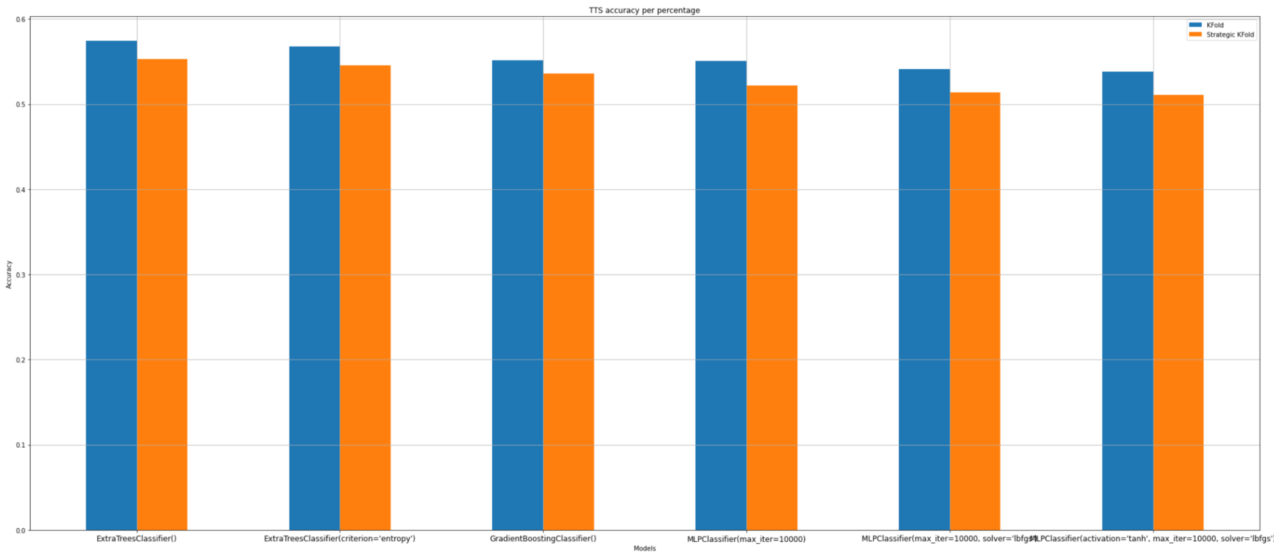

Figure 3: Strategic KFold and KFold for ensemble models

with entropy as a criterion. The MLPClassifier with identity and logistic activation functions showed promising results with, attaining 58.46% accuracy on a test size of 20%. Furthermore, ensemble models like ExtraTreesClassifier with the 'entropy' criterion outperformed others, with a mean accuracy of 57.81% using KFold cross-validation. These findings highlight the importance of algorithm selection and evaluation techniques in optimizing predictive performance. Further exploration of hyperparameter tuning and ensemble methods may enhance model efficacy for hit song prediction tasks.

4 DISCUSSION

The findings of this study shed light on the efficacy of machine learning algorithms in predicting hit songs based on Spotify metadata. While some models exhibited promising accuracy rates, deviations from expected outcomes were observed, prompting deeper analysis. The AdaBoost Classifier achieved the highest accuracy, but only in the train-test-split. Additionally, the MLPClassifier with identity and logistic activation functions showed accuracy, suggesting the potential of neural network architectures in capturing nonlinear relationships within the data. These results align with prior research to some extent but also present novel insights into the nuanced behavior of machine learning models in music classification tasks. Theoretical implications suggest the need for further investigation into ensemble methods and hyperparameter tuning to optimize model performance. In conclusion, this study contributes to the evolving field of music analytics by highlighting the complexities inheren t in hit song prediction and offering avenues for future research to explore.

5 CONCLUSION

In summary, this study investigated the application of machine learning algorithms for hit song prediction using Spotify metadata. Through rigorous experimentation and evaluation, we have demonstrated the potential of various classifiers and ensemble methods in categorizing songs into hits and non-hits with reasonable accuracy. Our findings contribute to the existing body of research by providing insights into the performance characteristics of different models and the impact of algorithmic parameters on predictive outcomes. Despite achieving competitive accuracy rates, our study also revealed nuances and deviations from expected results, underscoring the need for further investigation. Moving forward, it is imperative to address open questions surrounding the generalizability of models across diverse music genres, the robustness of predictions over time, and the incorporation of additional features such as lyrics and user-specific preferences. Moreover, future research should focus on refining model architectures, exploring ensemble techniques, and optimizing hyperparameters to enhance predictive efficacy. By addressing these challenges and leveraging emerging advancements in machine learning, we can continue to advance the field of music analytics and unlock new insights into the intricate dynamics of song success prediction.

5.1 Previous Research

Compared to prior research endeavors, which often grappled with issues of data imbalance and feature scaling, this study's results represent a significant improvement. The utilization of a more balanced dataset, coupled with standardized feature scales, has led to more reliable and interpretable models. The transition from overfitted models, which yielded inflated accuracy rates, to robust and generalizable models underscores the importance of methodological rigor in data science research.

5.2 Future Directions

Moving forward, avenues for further exploration include delving deeper into hyperparameter tuning to maximize model performance

Table 1: Model Accuracy Comparison

Model	Accuracy (%)	
	Train-Test-Split	k-fold CV
RandomForestClassifier (Entropy)	57.69 / 54.62 / 52.68	56.12
AdaBoostClassifier	60.00 / 55.38 / 52.91	53.58
ExtraTreesClassifier	48.46 / 51.54 / 55.24	56.81
ExtraTreesClassifier (Entropy)	54.62 / 55.38 / 54.31	57.81
MLPClassifier (Identity)	50.00 / 58.46 / 53.85	55.66
MLPClassifier (Logistic)	58.46 / 58.46 / 54.78	55.50

and exploring ensemble methods to harness the collective intelligence of diverse models. Additionally, extending the analysis to incorporate temporal dynamics and user-specific preferences could provide deeper insights into the multifaceted nature of song success prediction.

6 ACKNOWLEDGMENT

The author extends heartfelt gratitude to the creators of the Spotipy library and the Spotify Web API for providing invaluable resources and datasets essential for conducting this research. Special recognition is also extended to the Google Developer Student Club of the University of Primorska for organizing multiple events focused on ML&AI, where the initial inspiration for this research was sparked. It was during these events that the seeds of the initial research, which grappled with overfitting issues, were sown. Furthermore, the author acknowledges that this research topic served as the cornerstone of their bachelor's thesis, representing a culmination of academic inquiry and practical application in the fields of data science and music analysis.

REFERENCES

[1] Andrej Natev. 2023. Initial Notebook.
[2] Andrej Natev. 2024. Main Notebook.
[3] David Cournapeau. 2007. Scikit-learn Documentation.
[4] Joakim Arvidsson. 2023. 30000 Spotify Songs dataset.
[5] Nidula Elgiriyrwithana. 2023. Most Streamed Songs 2023 dataset.
[6] Paul Lamere. 2014. Spotipy Documentation.