Predict songs popularity using ML and MIR

TSHABU NGANDU BERNARD

UNIVERSITE NOUVEAUX HORIZONS – LUSHITRAP MUSIC

Introduction

Currently, being able to predict that something might be popular beforehand is an important research subject for every industry. It also has recently became a very important subject for the growing and competitive music industry as well. Since wide use of digital music platforms (Spotify, Billboard, Lastfm), data can be easily reached and the listening behaviors of the listeners can be easily observed. This provides convenience in forecasting techniques and it is also frequently used in recommendation systems. Predicting music popularity can have a significant impact on the music industry, as well as on the lives of individual artists and musicians.

The Lack of Music Recommendation Functionality is an Issue Currently Faced by Lushitrap.com

Lushitrap.com is a new music platform that has been gaining popularity in recent months. However, the platform currently lacks one important feature: music recommendation functionality. This can have a negative impact on user experience in a number of ways.

Music recommendation functionality allows users to discover new music that they might not otherwise find. This can be done by analyzing a user's listening history, their social media activity, or their demographic information. By recommending music that is tailored to a user's individual interests, music platforms can help users to find new music that they love.

The lack of music recommendation functionality can have a negative impact on user experience in a number of ways. First, it can make it more difficult for users to discover new music. If users are not able to find new music that they like, they are less likely to return to the platform. Second, the lack of music recommendation functionality can lead to user fatigue. If users are constantly exposed to the same music, they are less likely to be engaged. Third, the lack of music recommendation functionality can lead to user churn. If users are not satisfied with their experience on the platform, they are more likely to switch to a different platform.

- It can lead to users feeling overwhelmed.
When there is no music recommendation functionality, users are faced with a vast library of music that they may not know where to start. This can lead to feelings of overwhelm and frustration.

- It can lead to users missing out on new music.
If users are not able to discover new music that they might like, they may miss out on discovering new artists and genres.

- It can lead to users spending less time on the platform.
If users are not engaged with the music that they are listening to, they are less likely to spend time on the platform.

Methods

Machine Learning for Music Recommendation

Music recommendation is a challenging task, as there are many factors that can influence a person's enjoyment of music, such as their personal taste, mood, and the context in which they are listening to the music. However, machine learning can be used to improve the accuracy of music recommendations by taking into account a variety of factors, including the user's listening history, the song's audio features, and the music industry's trends.

Audio Features
Audio features are measurements of the sound of a song, such as its tempo, loudness, and pitch. These features can be used to create a fingerprint of a song that can be used to identify similar songs. For example, two songs with similar tempos and pitches are likely to be similar in terms of their overall sound.

Music Information Retrieval (MIR)
Music information retrieval (MIR) is a field of computer science that deals with the extraction of information from music. MIR techniques can be used to extract features from songs, such as the song's tempo, loudness, and pitch. These features can then be used to create a fingerprint of a song that can be used to identify similar songs.

Before Recommending a Song

Before recommending a song, we must first know what makes that song enjoyable or popular by users. This can be done by collecting data on user ratings of songs. This data can then be used to train a machine learning model to predict how likely a user is to enjoy a song based on the song's audio features.

Supervised learning is a type of machine learning where the model is trained on a dataset of labeled data. In the case of music recommendation, the labeled data would consist of song audio features and user ratings. The model would learn to predict the user rating of a song based on its audio features.

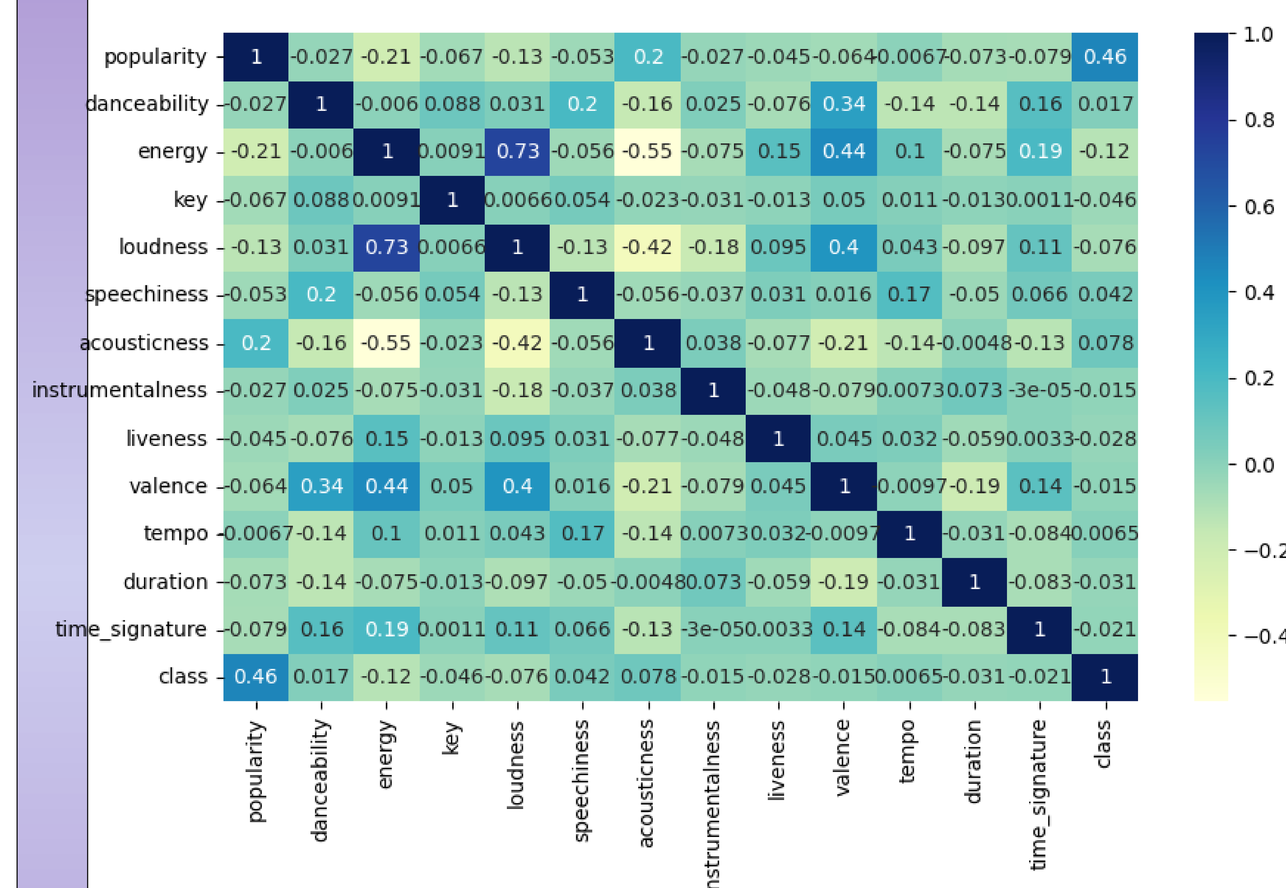
Binary classification is a type of supervised learning where the model is trained to predict one of two possible outcomes. In the case of music recommendation, the two possible outcomes are whether the user will enjoy the song or not. The model would learn to predict whether the user will enjoy a song based on its audio features.

Recommending Songs

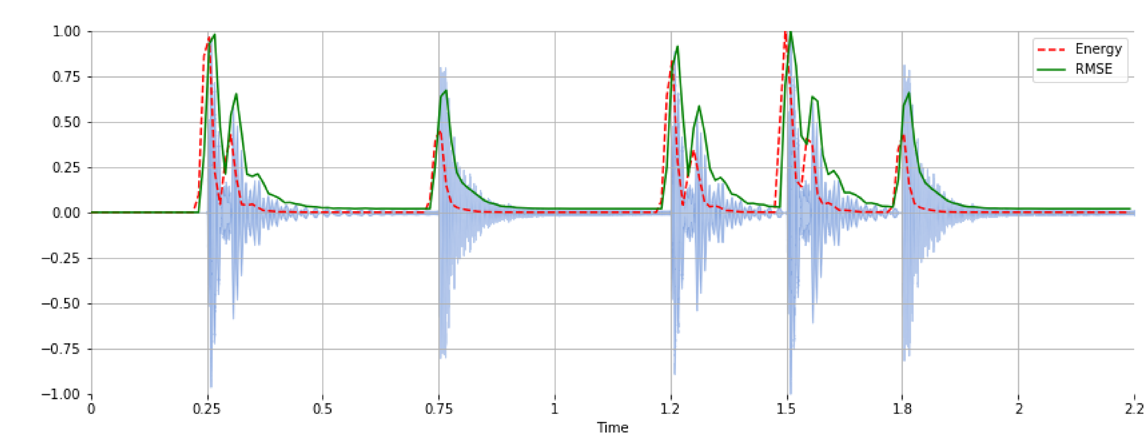
Once the model has been trained, it can be used to recommend songs to users. This can be done by first collecting the user's listening history. The model can then be used to predict how likely the user is to enjoy songs based on their listening history. The model can then recommend songs that the user is likely to enjoy.

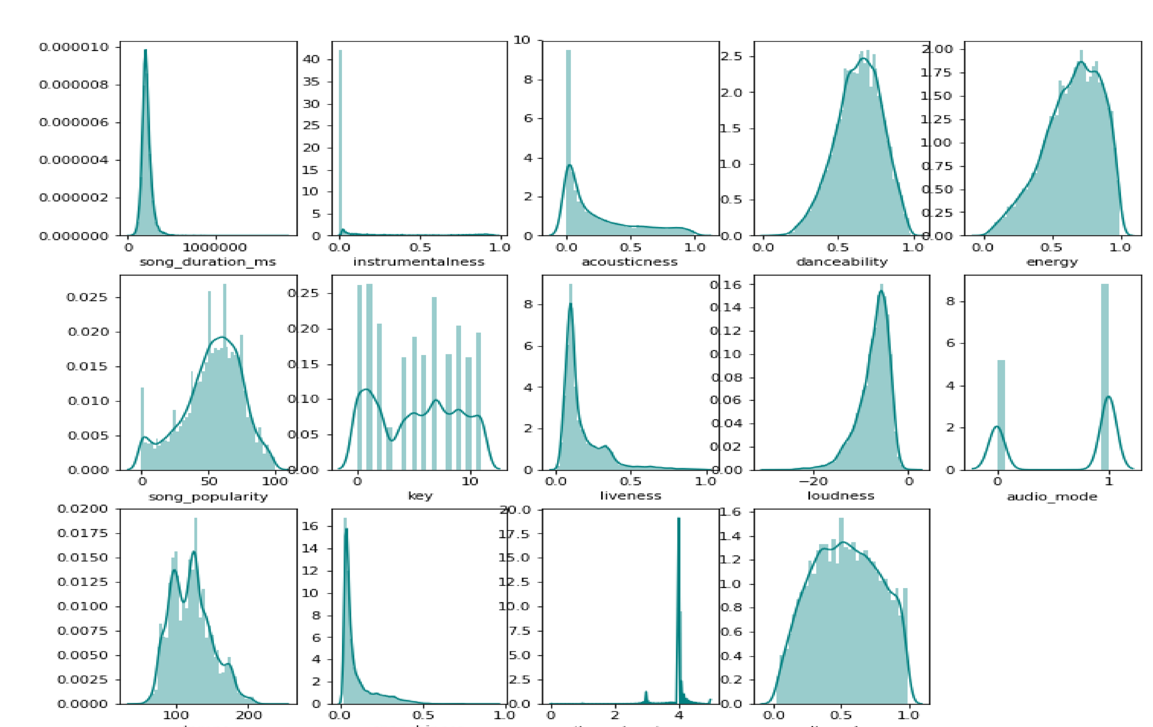
Results

We will start by predicting the popularity of songs on Spotify, using their API to collect data on the song's audio features. We will then use this data to create a dataset for Lushtrap, using sentiment analysis on comments posted on songs, ratings, download and play history. Finally, we will train a machine learning model on this dataset to predict the popularity of songs on Lushtrap



MIR techniques can be used to calculate these features in a variety of ways. One common approach is to use a Fourier transform to convert the audio signal into the frequency domain. This allows us to analyze the song's individual frequencies and their relative amplitudes. Eg: Energy and RMSE





Conclusions

Logistic Regression
Our problem has two possible outputs popular(1) and unpopular(0) which is suitable for binary logistic regression. Since it is a probability value that we want to get from the problem, we obtained a value between [0,1] using the sigmoid function.

Test accuracy: 71.8608972657287 %

KNN Algorithm
Tuned hyperparameter k: {'n_neighbors': 2}
Best accuracy: 0.7598619591186621 %

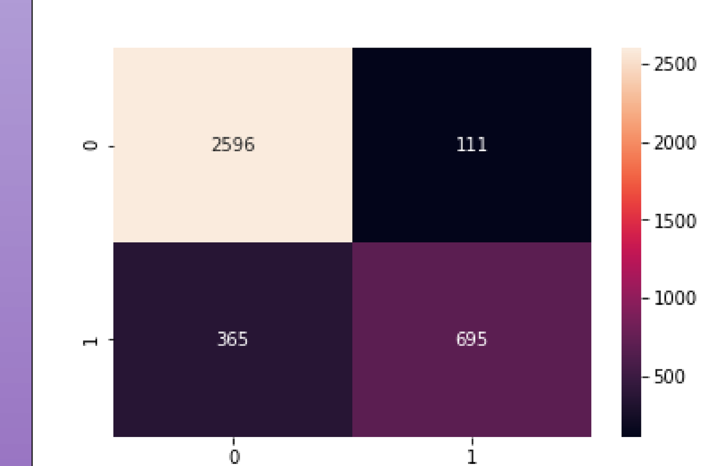
Support Vector Machine
Tuned Model Parameters:
{'SVM__C': 100, 'SVM__gamma': 0.1}
Test accuracy: 0.7990443323599682 %

Naive Bayes
Test accuracy : 0.718608972657287 %

Decision Tree Classifier
Test accuracy : 0.819219538093974 %

Random Forest Classifier
In order to avoid overfitting, random forest models select and train hundreds of different sub-samples (multiple deep decision trees) randomly and reduce the variance. We used 100 estimators and random state 3 gave the about % 89 accuracy.

Test accuracy : 0.8975311919299177 %

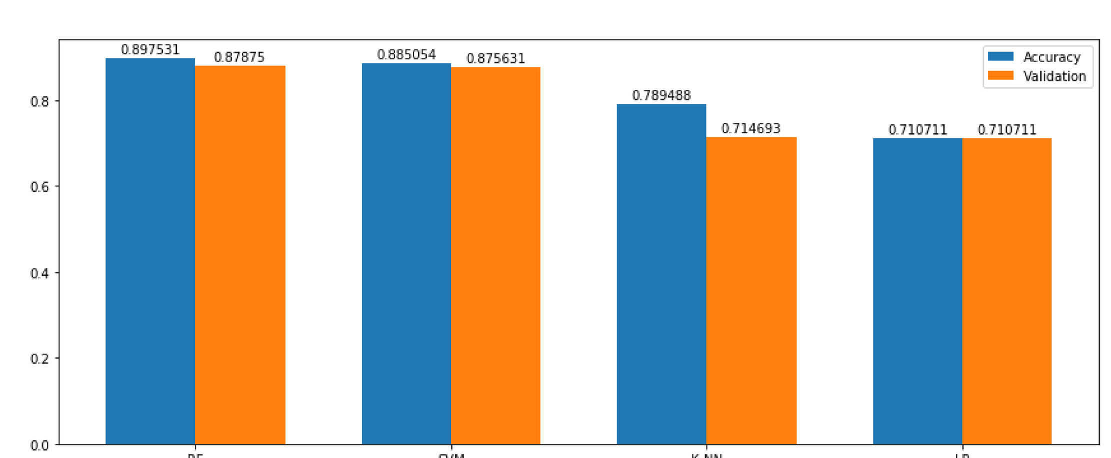


True Positive (TP): The model correctly identified 2596 instances as positive.

False Positive (FP): The model incorrectly identified 111 instances as positive, when they were actually negative.

True Negative (TN): The model correctly identified 695 instances as negative.

False Negative (FN): The model incorrectly identified 365 instances as negative, when they were actually positive.



Bibliography
1. Music information retrieval references and algorithms : https://musicinformationretrieval.com/
2. Spotify API : https://developer.spotify.com/documentation/web-api