Predicting Song Popularity

Sam Xu *samx@stanford.edu*, Joyce Xu *jexu@stanford.edu*, and Eric Tang *etang21@stanford.edu*

***Abstract*—This project aims to predict the popularity of songs by analysing acoustic features such as tempo, duration, mode, loudness, key; artist information such as location and popularity; and metadata such as release title and year. The codebase for the project is located at `https://github.com/inSam/SoundScorer221`**

I. Introduction

The music industry is a multi-billion dollar industry which produces over a million songs each year in the United States alone. With such a large volume of music produced, identifying which songs will become popular is a highly lucrative business. We would like to predict the popularity of a song by analyzing its audio features and metadata. Such a tool would be valuable for record labels, streaming services, and average consumers; it would also help the research community understand what acoustic features are currently popular with the public.

II. Related Work

Myra et al 2018 [1] analyzed over 500,000 songs to predict popularity (appearance on the Top 100 song list) from acoustic features and the ”popstar status” of its artist. Using a random forest classifier, Myra et al. obtained a roughly 70% classification accuracy without using the ”popstar status” feature, and a roughly 80% success rate with the popstar feature.

In a 2015 CS229 project, Pham et al. attempted to predict song popularity using support vector machines, multi-layered perceptrons, and logistical regression. Pham et al. utilized auditory features, such as key, tempo, and volume, as well as metadata like artist name and album name. Their metric of success was a ”hotness” rating from The Echo Nest, an online aggregator of song popularity metrics. Their system performed moderately well with a Mean Squared Error of about 0.15, measured over a popularity index between 0 and 1 [2].

In a 2017 work, Yang et al. used convolutional neural networks (CNNs) to extract complex auditory features from songs and predict their resulting popularity. They found that features extracted using CNNs outperform features solely extracted using simple linear-regression techniques on audio files. [4]

III. Methodology

A. Dataset

Our dataset comprises songs extracted from the Spotify API, which provides acoustic features, author information, and song metadata. More abstract features such as dancebility, energy, and song popularity were pulled from the Echo Nest, whose database is incororated into the Spotify API.

As preliminary data, we pulled 10,000 songs from the Spotify API, which contains a plethora of pre-extracted features that are also provided by the Spotify and the Echo Nest. These features include:

- Acoustic Features: tempo, duration, loudness
- Metadata: artist name and information
- Audio: song preview
- Echo Nest Features: ”danceability”, ”energy”, etc.

The six Echo Nest features are derived from the audio file but use Echo Nest’s proprietary algorithm.

Each song is also labeled with a popularity score between 1 and 100. This popularity score will serve as our ground truth label, and we will use the features above to run our preliminary baseline and oracle tests.

B. Concrete Example of Input and Output

Here you can see an an Input and Output example for the song *Never Gonna Give You Up*:

$$
\begin{array}{l}
''TRAXLZU12903D05F94'' : \{ \\
\qquad ''features'' : \{ \\
\qquad artist\_name : RickAstley \\
\qquad loudness : -7.75 \\
\qquad tempo : 113.359 \\
\qquad duration : 211.69587 \\
\qquad mode : 1 \\
\qquad energy : 48.149 \\
\qquad previewurl : p.scdn.co/mp3/99 \\
\quad ...\}, \\
\quad \}
\end{array}
$$

The output is simply the Spotify popularity rating, a score between 1 and 100 (in this case, 86). Although a song’s popularity fluctuates over time, we predict only the current popularity rating.

C. Implementation

We started by importing 10,000 songs from the public Spotify dataset and extracting features from the Spotify API.[3] For our baseline experimentation, we trained a Linear Regression classifier on this Spotify data.

D. Evaluation Metric

We use Mean Squared Error as our evaluation matric.

$$\frac{\sum_{i=1}^{n}(\text{Predicted}_i - \text{Actual}_i)^2}{n}$$

IV. Preliminary results

A. Baseline

For our baseline, we trained a linear regression classifier on seven features already stored in the Spotify database: ”acousticness”, ”danceability”, ”energy”, ”loudness”, ”speechiness”, ”tempo”, and ”valence”. We used an 85/15 train/test split for our database of 10,000 songs, and our linear regression classifier achieved a mean squared error of 61.1.

B. Oracle

Besides using the direct data from the label, an oracle for this problem is difficult to come by. Spotify doesn't release other relevant oracle data such as number of monthly listens, making it difficult to use other features in our oracle prediction. In addition, the existing body of literature on music popularity prediction is comparatively sparse, leaving few ”state of the art” models to compare our performance against.

The best measure we can determine for our oracle is the previous work of Myra et al., who achieved roughly 80% accuracy in predicting a song's Top 100 status. This is an easier problem than our regression problem, but serves as a useful target for potential success, and illustrates that the problem is feasible. By setting a threshold for our model, we may be able to compare our scores to Myra's work.

V. Next Steps

A. Improve Features

We will extract more features from the Spotify database such as song age, location, audio sections, and signature. We will also try to construct more advanced features from historical data such as musical trends. Using Spotify's provided links to song previews, we may be able to extract more advanced features from the audio recordings of each song. This step will involve a lot more feature tuning and experimentation.

B. Improve Models

For future models, we will try to tackle this problem by using ensemble methods such as gradient boosting regressors and random forests, support vector regressors, and neural networks to predict popularity. In particular, neural networks may be able to extract more meaningful features from the audio tracks of each song.

VI. Challenges

One challenge is to consider the effects of time on the popularity on the song. We may be able to tackle this by simply adding the age of the song as a additional feature.

Another challenge is that there is plenty of noise that influences a song's popularity, beyond its mere auditory features. While we have assumed a 100% success rate with our Oracle, this may be unrealistic to achieve with a classifier.

References

[1] Myra Interiano, Kamyar Kazemi, Lijia Wang, Jienian Yang, Zhaoxia Yu, Natalia L. Komarova, *Musical trends and predictability of success in contemporary songs in and out of the top charts* Royal Society Open Sciences, 2006.

[2] James Pham, Edrick Kyauk, Edwin Park, *Predicting Song Popularity* Stanford University CS 229, 2015.

[3] `https://developer.spotify.com/documentation/web-api/`

[4] Li-Chia Yang, Szu-Yu Chou, Jen-Yu Liu, Yi-Hsuan Yang, Yi-An Chen, *REVISITING THE PROBLEM OF AUDIO-BASED HIT SONG PREDICTION USING CONVOLUTIONAL NEURAL NETWORKS* Academia Sinica, Taiwan, 2017.