Trendy Tunes: Predicting Popularity of Top 200 Spotify Songs

Dhruv Medarametla
[1]Stanford University, CS 230

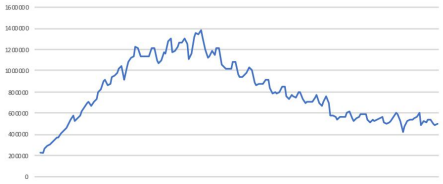
Background

With over 30 million songs available, and billions of hours of music listened to, Spotify is a huge player in the music streaming industry. With all this usage comes data, especially about the most frequently streamed songs. This data is extremely useful when it comes to predicting future Spotify song trends.

This project aims to do precisely that—to predict the number of streams per day in the future for different songs in the Top 200. Ideally, this will not only reveal insight into how the typical Top 200 song behaves on the charts over time, but also could identify patterns that may help artists create songs that are longer-lasting by seeing which factors are the most influential.

I was able to collect data from a Kaggle dataset containing daily song rankings and number of daily streams for Top 200 Spotify songs in several regions. I used this dataset to create United States-centric data for each song that had been in the Top 200 in the past year, and use this to create features and outputs for the rest of the project.

Streams per day for *Bodak Yellow* by Cardi B: 7/15-1/9



Features and Outputs

Using this data, I created an X and Y dataset. The X dataset differed slightly depending upon the model being called; however, the following features were always included in some form.

- Number of daily streams for the past 30 days
- US Charts daily rank for the past 30 days
- Day of the week
- Sign of change in number of daily streams for past 29 days
- Sign of change in US Charts daily rank for past 29 days

For the RNN, the features were only included as one set per day, while for the other two models, all features were given simultaneously. Thus, the first two models had a total of 124 input features, while the RNN had 10 features a day for the past 30 days.

The desired output was the number of streams for the following day.

Linear Regression

The first attempt at modeling this came through a linear regression. I chose a linear regression as the baseline because it wouldn't take into account the time-series nature of the data and because of its simplicity, and thus might provide a basic way of predicting the number of streams without being too difficult to understand.

Our loss function was slightly different than the normal sum of squared differences. Noticing that using such a loss function would bias our model toward better predictions for more popular songs, I decided to use a modified loss function of the following form:

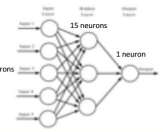
$$\mathcal{L}(\hat{y}, y) = \frac{1}{m}\sum_i \left(\frac{\hat{y}_i}{y_i} - 1\right)^2$$

This choice of loss function ensured that the error was normalized by the actual value, ensuring that there would be no bias toward popular songs.

Neural Network

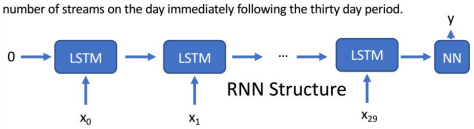
The next attempt at modeling this came through a neural network. I chose a neural network for the second benchmark because I thought it would be a good mix of the simplicity of a linear regression model and an RNN. Because there was a hidden layer, there was an ability for the model to detect more complex patterns, as well as potentially pick up on time-series information.

Our input layer for the neural net had 124 neurons; as a result, we chose to have 15 neurons for the hidden layer, to have a good balance for the final layer, in which there was just one neuron.

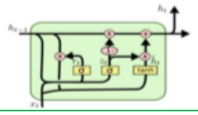


Recurrent Neural Network

We finally used an RNN to predict the number of daily streams. Because we were using an RNN, our input was slightly different. We had a total of 30 days of input, with each day containing 10 data points: the number of streams, the rank on the charts, the sign of change in streams, the sign of change in rank, and 6 variables corresponding to the day of the week. Our output variable was once again the one value, which represented the number of streams on the day immediately following the thirty day period.



LSTM Cell



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Results

As predicted, the Linear Regression did the worst and the Neural Network did the best.

What follows is a chart of our results, both on the training and test set:

Model	Test Performance	Train Performance
Linear Regression	0.002156	0.002178
Neural Network	0.002105	0.001944
RNN	0.001549	0.001212

Each number shown is the training cost, as defined earlier; one way to interpret the results is that on average, the RNN will make a prediction that is about 3.5% off from the actual number of streams.

As can be seen, it seems like while the linear regression model did not face issues of variance, the other two models did significantly better on the training set than on the test set, suggesting that there was some level of variance affecting the results.

Conclusions and Future Work

These three models did about as well as expected; going in, I tried to manually predict the number of streams, and ended up with about 5% error. Thus, all three of these models beat an average human. Using these models on Spotify might reveal some insight as to how popular songs operate.

Given more time, I would try more appropriate regularization on the Neural Network and the Recurrent Neural Network. As shown in the results, there was some variance, and eliminating that could lead to much better performance.

One interesting idea would be to try the RNN model on different genres of songs. That would lead to insight as to how different genres operate, and could help singers of different genres understand exactly what makes their songs popular.

One particularly difficult task would be to take the actual song as input itself. There are functions that transform a song and find its tempo, as well as other significant elements; perhaps passing that in to the RNN as an initial input would yield a stronger prediction.

I want to research reaching out to Spotify with this information, and see if they have already done similar analysis, or if this is novel.

https://www.kaggle.com/edumucelli/spotifys-worldwide-daily-song-ranking/data