Billboard Top 100 Prediction

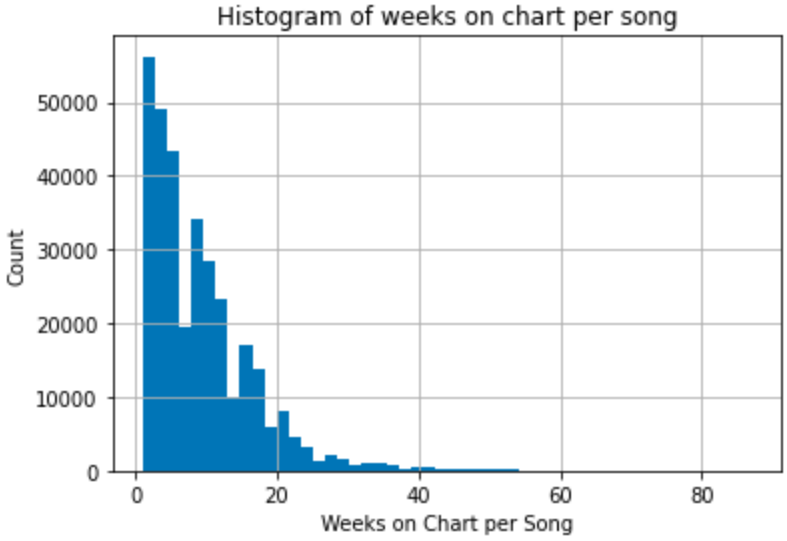
The Problem: What drives the “success” of a song? What factors (such as: title, artist, time, lyrics, emotional score, and prior placement) can be used by artists, producers, or others in the music industry to determine whether or not a song will make it to the top 10 of the Billboard Hot 100?

Context: With the increase and combination of music streaming and social media platforms, now more than ever, artists can connect their music to their fan base seemingly instantaneously. This increased accessibility drives an increased competition between musicians. For over 80 years, Billboard has documented this competition by publishing a weekly chart that ranks song popularity. How can an artist, producer, or other individual in the music industry predict whether or not a new song will be “popular?” Using historical Billboard Hot 100 data we will look at what factors drive a song to the top of the chart.

The Data: Our Top 100 data is imported from this website:
https://data.world/kcmillersean/billboard-hot-100-1958-2017
The emotion data for each song is imported from this website:
https://data.world/tazwar2700/billboard-hot-100-with-lyrics-and-emotion-mined-scores
This data is gathered and cleaned according to this blog post:
https://medium.com/codex/emotion-mining-lyrics-of-all-billboard-hot-100-songs-1958-2020-3007d6963115
The Billboard data includes the week the chart was listed, the song title, the artist, the week position, the previous week position, the peak position, and the instance this song is on the chart. The emotion data includes the number of words in the song, the number of unique words, the year, the decade, and various emotional scores derived from the lyrics.
In our cleaning, we create additional variables, including the total of times an artist appears on the chart, the number of times an artist appears on the the chart in a given year / week, the number of times an artist appears on the chart the year / week prior, and several emotional indicator variables, if an emotion score is above .8, we mark as 1, else 0.

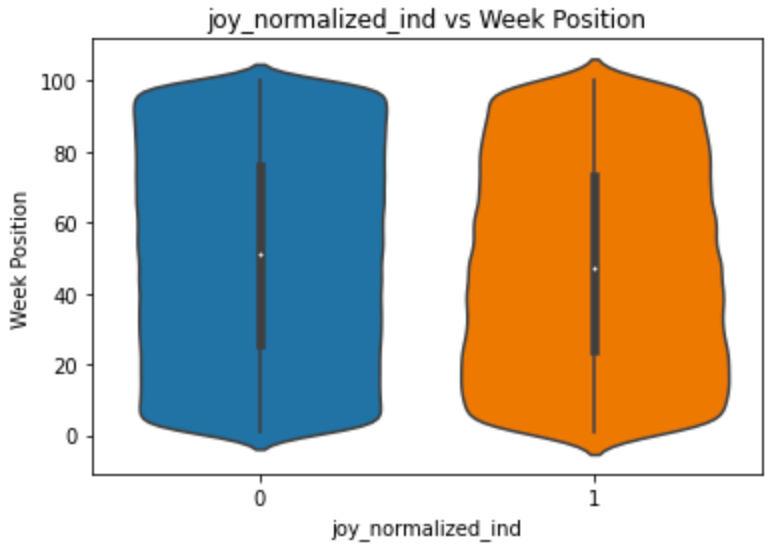
Exploratory Analysis: Some key distributions and relationships we notice in our exploratory analysis are the number of weeks a song is on the chart, the instance a song is on the chart vs its week position, and the week position distribution of joyful songs vs non-joyful songs.

We see below that most songs are only on the chart for 1 or 2 weeks. The distribution of songs that stay on the chart longer than 20 weeks trails off quickly.



Below are comparative violin plots that show the distribution of week position given a song's instance. Instance refers to the amount of times a song has entered the chart after not having been on the chart the week prior. Interestingly, we see that when a song enters the chart for the 7th time, it generally ranks higher.

The violin charts below show the weekly position distribution of songs that are indicated as joyful (1) and not joyful (0). We see that there is a slight increase in higher ranks for joyful songs.



Pre-processing: Our initial method of tackling our question - what features drive the “success” of a song? - was to create a model predicting a Success Indicator - is a song in the top 10 or is it not? As a check, we analyze 3 additional potential response variables: Week Position, Peak Position, and Weeks on Chart.
Through PPS Predictor Analysis, we find that Week Position contains the highest correlation (ppscore) with the most features (see below). We decide to model with Week Position as our response.

	x	y	ppscore	case	is_valid_score	metric	baseline_score	model_score	model
0	Week Position	SuccessInd	1.000000	regression	True	mean absolute error	0.0996	0.000000	DecisionTreeRegressor()
1	Previous Week Position	SuccessInd	0.545172	regression	True	mean absolute error	0.0996	0.045301	DecisionTreeRegressor()
2	WeekID	SuccessInd	0.000000	regression	True	mean absolute error	0.0996	0.155140	DecisionTreeRegressor()
3	Song	SuccessInd	0.000000	regression	True	mean absolute error	0.0996	0.110640	DecisionTreeRegressor()
4	SongID	SuccessInd	0.000000	regression	True	mean absolute error	0.0996	0.105333	DecisionTreeRegressor()

	x	y	ppscore	case	is_valid_score	metric	baseline_score	model_score	model
0	Previous Week Position	Week Position	0.697120	regression	True	mean absolute error	24.9078	7.544077	DecisionTreeRegressor()
1	Peak Position	Week Position	0.563066	regression	True	mean absolute error	24.9078	10.883053	DecisionTreeRegressor()
2	SuccessInd	Week Position	0.178647	regression	True	mean absolute error	24.9078	20.458101	DecisionTreeRegressor()
3	Weeks on Chart	Week Position	0.170591	regression	True	mean absolute error	24.9078	20.658756	DecisionTreeRegressor()
4	Artist_Count_Year	Week Position	0.065553	regression	True	mean absolute error	24.5258	22.918064	DecisionTreeRegressor()

	x	y	ppscore	case	is_valid_score	metric	baseline_score	model_score	model
0	Previous Week Position	Peak Position	0.638250	regression	True	mean absolute error	25.5734	9.251166	DecisionTreeRegressor()
1	Week Position	Peak Position	0.550060	regression	True	mean absolute error	25.5734	11.506494	DecisionTreeRegressor()
2	Weeks on Chart	Peak Position	0.339242	regression	True	mean absolute error	25.5734	16.897834	DecisionTreeRegressor()
3	SuccessInd	Peak Position	0.133766	regression	True	mean absolute error	25.5734	22.152544	DecisionTreeRegressor()
4	Artist_Count_Year	Peak Position	0.095455	regression	True	mean absolute error	25.2626	22.851158	DecisionTreeRegressor()

	x	y	ppscore	case	is_valid_score	metric	baseline_score	model_score	model
0	Peak Position	Weeks on Chart	0.237930	regression	True	mean absolute error	5.4024	4.117006	DecisionTreeRegressor()
1	Previous Week Position	Weeks on Chart	0.145554	regression	True	mean absolute error	5.4024	4.616061	DecisionTreeRegressor()
2	Week Position	Weeks on Chart	0.031534	regression	True	mean absolute error	5.4024	5.232040	DecisionTreeRegressor()
3	DecadeID	Weeks on Chart	0.009313	regression	True	mean absolute error	5.4024	5.352085	DecisionTreeRegressor()
4	decade	Weeks on Chart	0.007149	regression	True	mean absolute error	5.4024	5.363780	DecisionTreeRegressor()

Modeling Results:

We utilize PyCaret to determine which algorithm bet suits our selected response variable. PyCaret provides insight to the time to run and r^2 score. As shown below, we find CatBoost, LightGBM, Random Forest, Extra Trees, and Gradient Boosting to be our top performers.

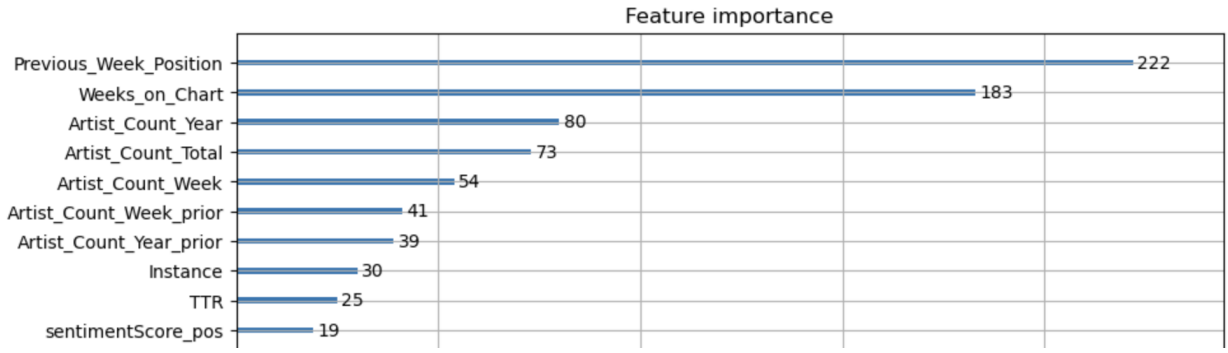
	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
catboost	CatBoost Regressor	5.7055	76.8578	8.7667	0.9079	0.2413	0.2027	7.5700
lightgbm	Light Gradient Boosting Machine	5.7869	78.7547	8.8743	0.9056	0.2406	0.2064	1.3560
rf	Random Forest Regressor	5.9214	82.1829	9.0654	0.9015	0.2434	0.2038	54.5460
et	Extra Trees Regressor	6.0936	86.1485	9.2816	0.8967	0.2490	0.2075	66.0480
gbr	Gradient Boosting Regressor	6.1121	87.5695	9.3577	0.8950	0.2565	0.2267	14.6020

We decided to investigate Light GBM and Random Forest for our modeling. After splitting our data into Training (70%) and Test (30%) sets, we run a 5-fold cross validation and random search to hyper-parameter tune. Based on our results below, we select LightGBM as our optimal algorithm.

For LightGBM all Factors:
- r-square: 0.9015
- n_leaves: 1815
- n_estimators: 147
- max_depth: 3
- learning_rate: .82
- time_to_run: 1 h 10 m

For Random Forest all Factors:
- r-square: 0.888
- n_estimators: 400
- max_depth: 7
- time_to_run: 10+ h

We looked at feature importance of our Light GBM model to determine which factors are influential.
We find:



Since these are mostly uncontrollable, we reconstruct our model to evaluate only controllable features.

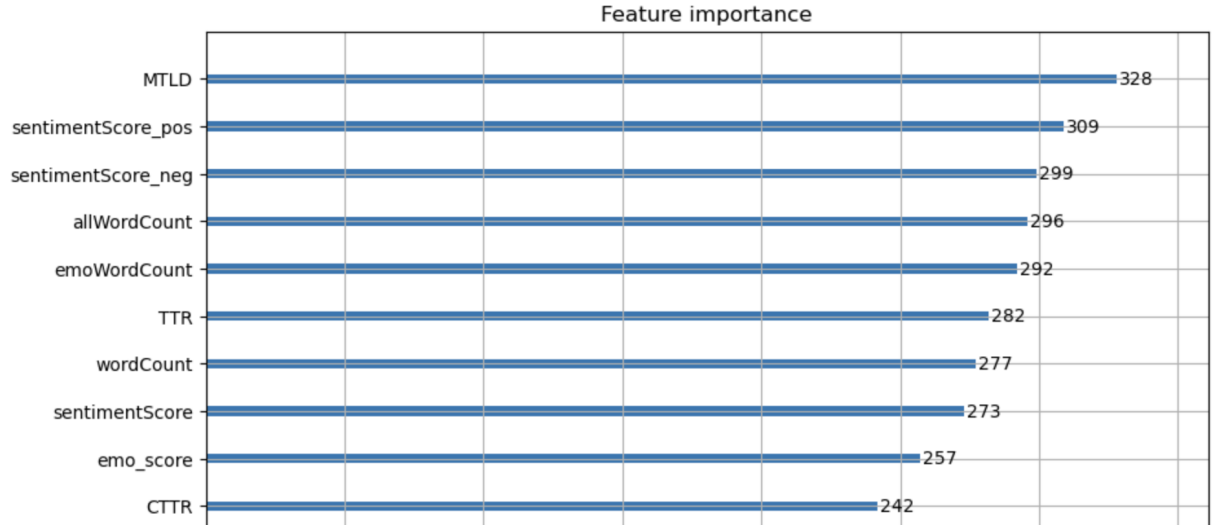
Tuning hyperparameters for Light GBM with controllable factors only, we find the following:

LightGBM Controllable Factors:

MAE: 17.9598
MSE: 513.9297
RMSE: 22.67
R_Sqr: 0.382

Our MAE, MSE, and RMSE are much higher and our r^2 is much lower than the Light GBM with all factors.

Regardless, we look at which controllable features are most important.



It appears that MTLD (the average sentence length), Sentiment Score, Word Count, and TTR (the ratio of unique words to total words) are the most influential factors.

Average of Controllable Features:

Average MTLD for top 5 songs is: 32.93

Average MTLD for bottom 5 songs is: 35.95
Average Positive Sentiment Score for top 5 songs is: 0.17
Average Positive Sentiment Score for bottom 5 songs is: 0.16
Average Negative Sentiment Score for top 5 songs is: 0.0785
Average Negative Sentiment Score for bottom 5 songs is: 0.0789
Average Total Work Count for top 5 songs is: 189.21
Average Total Work Count MTLD for bottom 5 songs is: 166.07
Average TTR for top 5 songs is: 0.32
Average TTR for bottom 5 songs is: 0.35

Recommendation:
Most of the features that predict a Billboard top performing song are out of our control, specifically the amount of time an artist has entered and stayed on the chart. That being said, when accounting for features in our control, focus on the following:

- MTLD (the average sentence length). Shorter sentences perform better.
- Sentiment Score. Top songs are more positive.
- Word Count. Top performing songs have more words.
- TTR (ratio of unique words to total words). Top songs include word repetition.

Future Improvements:
From this analysis we unravel a larger scope to investigate. The data we acquired only runs through 2020, an updated model could include more recent data and/or look specifically at recent years.
External data could provide further insights to song performance. Streaming and marketing information, for example.
This analysis only includes music that actually made it onto the Billboard Top 100 chart. Further analysis using all songs could determine how a song makes it onto the Billboard 100 chart in the first place. Due to the large amount of data, this analysis could be conducted for a smaller amount of time (i.e. one year).