Spotify Song Collection Analysis

Thanveer Ahamed
Faculty of Information Technology University of Moratuwa Moratuwa, Sri Lanka
174005R@uom.lk

Manusha Chethiywardhana
Faculty of Information Technology University of Moratuwa Moratuwa, Sri Lanka
174024A@uom.lk

Nethmee Sellahewa
Faculty of Information Technology University of Moratuwa Moratuwa, Sri Lanka
174151J@uom.lk

Abstract- Songs defiantly hold a special place in the hearts and souls of people as one of the most popular means of entertainment. Music is considered as a universal language where people can enjoy irrespective of their ability to understand the lyrics. In the modern era many music streaming platforms have come to market to fulfill the demand for streaming songs and other media. Platforms such as Spotify, Apple Music and Google Music are known to have millions of active users along with millions of songs. Spotify s one such leading music streaming platform with over 70 million songs and 356 million active users per month. Analyzing the data on songs provided by the Spotify Web API can be useful to derive many insights that can be crucial for not only the general audience but also musicians, music producers and music streaming platforms as well. This paper is dedicated to such analysis conducted on the Spotify dataset to provide meaningful insights on song features over time, using big data analytic approaches.

Keywords- Big Data, Music, Spotify Data, Apache Sparks, Pyspark, Pandas.

I. INTRODUCTION

Songs defiantly holds a special place in people's heart and soul as a major source of recreation. Undoubtedly music is a universal form of entertainment, and an important form of Art. Psychological studies provide important insights to prov that people who enjoy listening to music are less likely to suffer from depression and anxiety.

Music is considered as a universal language, which means as opposed to other forms of entertainment such as reading a book, it does not only serve for a particular language [1]. Even the people who do not understand the language that the lyrics of the song is written in, can enjoy the music and convey the message simply through the musicality of the song which can be stated as the reason behind the universal reception to music.

In the modern era, due to obvious demand there are many platforms which enables millions of users worldwide to stream songs and music as a service through internet. According to Hypebot.com, thousands of songs are uploaded to music streaming platforms like Spotify, Apple Music, Google Music, Napster, Dezzer every hour. Of these platform Spotify holds major portion of the market.

Spotify is a Swedish audio and music streaming platform and a media service company which has over 70 million song tracks from a wide range of record labels and media production houses. They currently own a market share of over 356 million of active users all around the world and has evidently become one of the largest music stream platforms in the world alongside giant competitors such as Apple Music and Google Music. As a result of the enormous number of active users and song tracks as well as the various user actions that take place in an hour Spotify has the ability to generate vast amount of data through their services and business operations.

Spotify provides developers with various data that are generated and collect throughout their operations which are very much beneficial for to conduct valuable analysis which can be used in generating insights for musicians, general audience as well as the streaming platforms and the rest of the industry. One of the popular datasets that Spotify provides, is the dataset with various features of music such as loudness, speechiness ,acousticness etc. For this analysis we have taken a dataset from Kaggle that is implemented obtaining data from the Spotify Web API. We have comprehensively analyzed the dataset considering the timelines as well as taking the features itself.

Spotify provides developers with various data that are generated and collect throughout their operations which are very much beneficial for to

conduct valuable analysis which can be used in generating insights for musicians, general audience as well as the streaming platforms and the rest of the industry. One of the popular datasets that Spotify provides, is the dataset with various features of music such as loudness, speechiness, acousticness, etc. For this analysis we have taken a dataset from Kaggle that is implemented obtaining data from the Spotify Web API. We have comprehensively analyzed the dataset considering the timelines as well as taking the features itself. pandemic. In our approach we have targeted to perform an analysis on text contained in tweets, time and hashtags.

The rest of the paper is structured as follows. Section 02 describes the related work based on the same context while Section 03 describes the methodology adopted for analysis followed by the data set and the tools, techniques that have been used in the analysis. a discussion of the results obtained from the analysis process and the conclusion in Section 05 respectively.

II. RELATED WORK

The songs industry has to evolve with the rapid growth of digital platforms like Spotify, Apple Music, and Google Music. So, it has gained the attention of many researchers and Data Scientists to analyze track data collections that collect by those previously mentioned platforms to extract new information and perceptions from those data.

In a study done by Ian A. Grant from the University of New Hampshire titled 'Are there Differences in Music Preferences Following Major Events?' he has done study research examines and discusses this potential influence of major events such as terrorism, war, or changes in laws on people's emotions and consequently the type of music theychose to listen to. Michael Whittle popular medium article writer has done Exploratory data analysis (EDA) to deigns a model for Spotify Artist Recommendation he use cosine similarity approach to find similar the artist is with the other artists [2]. Sunku Sowmya Sree a Python Developer has written a blog on his experiment on the Spotify dataset with the title of What makes a song popular? Analyzing Top Songs on Spotify in that blog he features presented in track dataset and done correlation analysis and then select set of features to create a model to predict popularity of song using Decision Tree Regressor with Grid search CV and Random Forest then he also had deigned a recommendation system where it recommends similar songs for any given song using Neighborhood Collaborative Filtering using the similarity metrics method [3].

III. METHODOLOGY

This section describes the data set that was used for the analysis, followed by the tools and techniques that were used to carry out the work, and finally the analysis with stages that are completed in the project.

After considering multiple datasets for performing meaningful analysis on song features, the Spotify dataset, which contains over 600,000 tracks released between 1922 and 2021, was chosen for the project. This dataset was obtained from Kaggle. The dataset was gathered with the help of the Spotify Web API and a Python script. This dataset is divided into two sections: one for users of the old version and one for users of the new version. We went with the new version, which includes the tracks.csv and data_by_artist_o.csv files. The collection script is available here: *https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks?select=tracks.csv* [4]

Track dataset contains 586672 unique records in csv format, with a total of 20 columns such as track id, track name, popularity, duration_ms, explicit, artists, id_artists, release_date, danceability, energy, and so on. This dataset has 1922-01-01 to 2021-04-16 temporal coverage and worldwide spatial coverage. A single track from 1900 is also included. data_by_artist_o dataset contains 28680 unique records in csv format and 16 columns. Because the data is well organized and documented, this dataset has earned a gold badge from Kaggle.

The analysis was carried out using the following tools and techniques: Google Colab, Pyspark, pandas, pyplot, Python, and Java.

a)Apache Spark 3.1.1

Apache Spark is a free and open-source analytics engine that can handle massive amounts of data. Spark provides a programming interface for entire clusters with implicit data parallelism and fault tolerance.

b)Pyspark

PySpark is an Apache Spark and Python collaboration. Python is a general-purpose, high-level programming language, whereas Apache Spark is an open-source cluster-computing framework focused on speed, ease of use, and streaming analytics [5].

c) pyplot

pyplot is a set of functions that makes matplotlib behave like MATLAB. Each pyplot

function alters a figure in some way, such as creating a figure, creating a plotting area in a figure, plotting some lines in a plotting area, decorating the plot with labels, and so on [6].

d) Python

Python is a high-level, general-purpose programming language that is interpreted. Python's design philosophy prioritizes code readability, as evidenced by its extensive use of significant indentation.

e) Java 8

JAVA 8 is a major feature release of JAVA programming language development. Its initial version was released on 18 March 2014. With the Java 8 release, Java provided support for functional programming, new JavaScript engine, new APIs for date time manipulation, new streaming API, etc [7].

f) Pandas

Pandas DataFrame is a two-dimensional, size-mutable tabular data structure with labeled axes (rows and columns). A data frame is a two-dimensional data structure in which data is aligned in rows and columns in a tabular fashion. Pandas DataFrame is made up of three major components: data, rows, and columns [8].

Throughout the project, the analysis is carried out in seven major stages.

1. **Analysis of songs based on Popularity, Danceability, and Valence**

   To gain a better understanding, the analysis was divided into two temporal sections.

   - From 1979 to 1999 released songs
   - From 2000 to 2020 released songs

This analysis was carried out to determine the relationships between a song's popularity and its danceability and valence. Line charts were created to see how the mean value of each feature varies over the years.

2. **Clustering songs based on Energy and Danceability.**

   This analysis was carried out using the k-means clustering algorithm to categorize songs into six groups based on their energy and danceability. The analysis was performed on songs released from 2020-06-01 to 2021-01-01.

3. **Analysis of how features of a songs like energy, loudness, speechiness, acousticness, liveness, tempo and instrumentalness changes with the time**

   This analysis was carried out using by using track dataset and by extracting features like energy, loudness, speechiness, acousticness, liveness, tempo and intrumentalness and taking 2 copies of dataset with above features.

   First copy of dataset was group by release year and calculate mean value for each above features and Line charts were created to see how the mean value of each feature varies over the years then calculated year with highest mean value for each feature. Then second copy of dataset was prepressed by grouping above features and dataset by release date and calculate mean value. Then we plot line chart for each feature that varies with release date for the year with highest mean value for each graph and then the Stationarity of variation of mean value of each feature for each year is discussed.

4. **Time Series forecasting of danceability feature by using Facebook prophet and ARIMA Model.**

   Time series data is a series of data points measured at consistent time intervals. The specialty of time series data is each data point in the series is dependent on the previous data points.

   Time series forecasting uses data with historical values and associated patterns to predict future values of that data. There are several methods to do Time series forecasting in this paper we discuss about using ARIMA Model and Facebook prophet to do Time series forecasting.

   ARIMA is made up with 3 terms as Auto-Regression (AR), Integrated (I) and Moving-Average (MA). In AR past values used for forecasting the next value and AR term is defined by the parameter ‘p’ in ARIMA. The value of ‘p’ is determined using the PACF plot. In MA it defines number of past forecast errors used to predict the future values. The parameter ‘q’ in ARIMA represents the MA term. ACF plot is used to identify the correct ‘q’ value [9].

   Assumptions used in ARIMA model,

   - The data series of feature use is stationary.
   - The data as input must be consist of single characteristic and time series.

Stationary means mean and variance of a data series are not change with time. A data series can be made stationary by using log transformation or differencing the series. In this paper we use AIC statistical model to compute p value to detect whether a data series is stationary or not. If p value of data series is less than 0.05 it is a stationary data series [10].

Facebook Prophet is an open sources library created and publish by Facebook. It provides ability to predict time series predictions with high accuracy by labeling data into ‘y’ as time series data and ‘ds’ as Datetime in data frame and it is capable to include impact of custom seasonality, holidays and trend [11].

In This Analysis we use both techniques to forecast

future value of danceability. We consider the mean danceability values of within the period of 2010/01/01 to 2021/01/01 we use at least 10 years of historical data to do future predictions. In our dataset, there are many danceability values record for a one unique release date, so we had to group by data from release date and take the mean value of danceability for that release date. So, in our forecast will discuss future values of mean danceability of songs that will release in future.

5. Classification of the level of popularity of a song

In this Analysis we design a classification model that able to classify the level of popularity of a song by using Genre, liveness, mode, danceability features that precent in data_by_artist_o dataset. When we check the dataset, we could observe Genre attribute was a list of strings and both liveness, danceability and popularity were continuous column and mode was a binary value that describes whether the songs is major (1) or minor (0) track. So, we had to convert those numeric values to levels. both liveness, danceability features were varies from 0 to 1 range so we use algorithm to convert those values to Nominal as levels of Low, Medium and High same kind of algorithm was designed for converting popularity to levels of Low, Medium and High as it was varying from range of 0 to 100.

We use the Naive Bayes algorithm that was supervised learning algorithm, which is based on the Bayes theorem to solve this classification problem. We use VectorAssembler transform data that could use as input to Naive Bayes chaffier model and use randomSplit to split training and testing data sets. Finally, we used MulticlassClassificationEvaluator to evaluates the Naive Bayes classifier model performance and accuracy.

6. Correlation Analysis for the features of the songs

Correlation Analysis is conducted as a Frequent Pattern Mining task that determines whether a certain two phenomenon have any kind of a relationship with one another. The main target of this analysis is to establish an insight to possible connections between various song features which will not only for the direct use but also as a fundamental supporting analysis which can be used as the first step for many complex analyses. In here, the following 9 features namely energy, acousticness, danceability, loudness, tempo, speechiness, instrumentalness, valence and liveliness of songs were taken to recognize the correlations between each other. Out of the four different techniques of Correlation Analysis in statistics (Pearson Correlation, Kendall Rank Correlation, Spearman Correlation and the Point-Biserial Correlation) Pearson Correlation Analysis has been chosen because all the features/attributes taken for the analysis are continuous variables [12]. In order to calculate the Correlation Coefficient, we have taken the data since 1920.

7. Regression analysis to predict the Loudness.

After doing the initial Correlation analysis, judging the output of each Correlation Coefficient it was decided to further analyze the behavior of the feature "loudness" and the feature "energy" since they have a Correlation Coefficient of 0.76, which indicate that aforementioned variables are highly correlated and when one variable increases the other variable also increases. As the first step a scatter plot with energy as the independent variable and loudness as the dependent variable was drawn. To increase the credibility and the usability of the analysis, only the data of the tracks from 01-04-2021 to the latest were taken to draw the scatter plot. After that considering the shape of the scatter plot as well as the Correlation Coefficient of the aforementioned features, a linear regression model was decided to build to predict the loudness of the song when the energy level is given. For this analysis also, the data from 01-04-2021 to the latest were obtained. The model was trained using 1660 data points and tested with 554 data points [13], [14].

IV. RESULTS AND DISCUSSION

In this section, the results of each analysis are discussed using graphical representations.

1. Analysis of songs based on Popularity, Danceability, and Valence

The results of this analysis can be shown in following charts.

A. 1979 to 1999 Songs

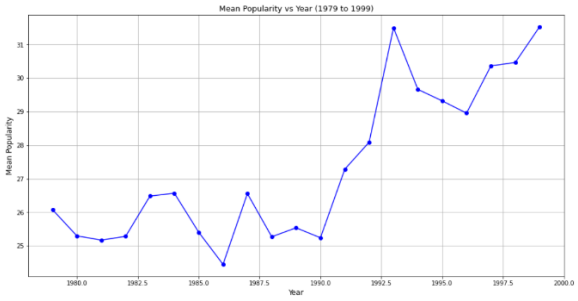


Figure 1. Mean Popularity vs Year graph.

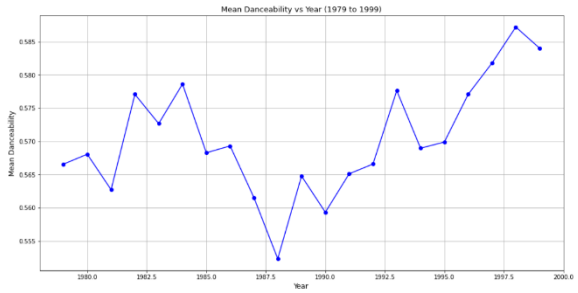


Figure 2. Mean Danceability vs Year graph.

From the above two graphs we can see that the peaks and valleys of song popularity and danceability are mostly similar. Therefore, we can determine that song popularity and danceability have a close relation. Overall, popularity and danceability of songs has increased over time.

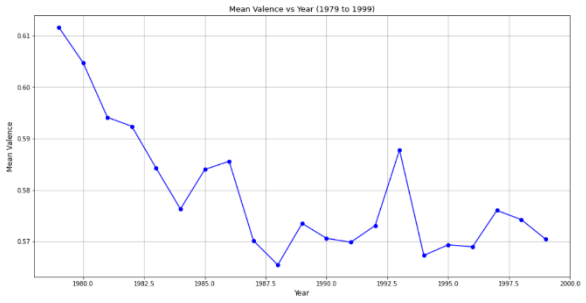


Figure 3. Mean Valence vs Year graph.

In contrast to popularity and danceability, the third graph shows that valence has decreased over time. Valence describes the musical positivity conveyed by a track. A song with a high valence is happy and cheerful, whereas a song with a low valence is sad, depressed, or angry. This chart shows that songs have lost their cheerfulness over time.

B. 2000 to 2020 Songs

The results of this analysis are depicted in the charts below.

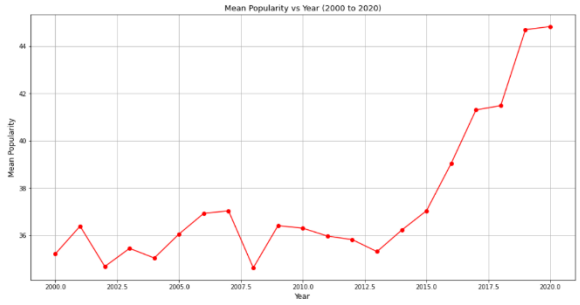


Figure 4. Mean Popularity vs Year graph.

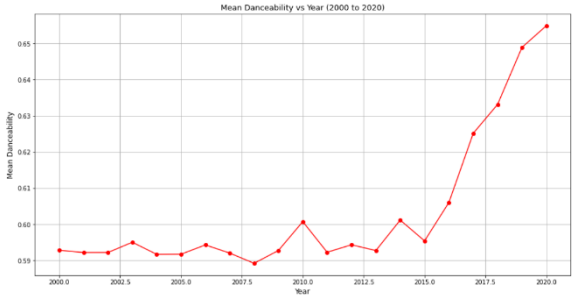


Figure 5. Mean Danceability vs Year graph.

Songs from 2000 to 2020 have fewer variations in popularity and danceability charts than songs from 1979 to 1999. The popularity of songs has been steadily increasing since 2013, and the danceability of songs has been increasing since 2015.

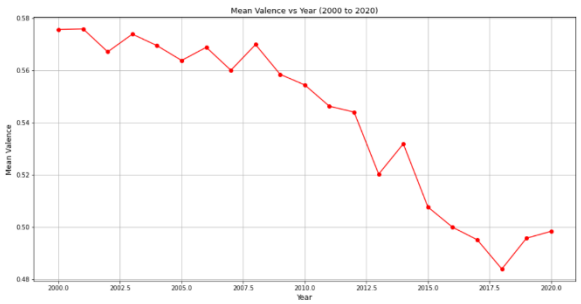
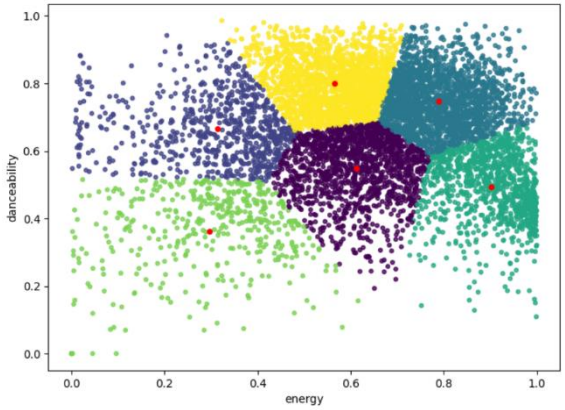


Figure 6. Mean Valence vs Year graph.

In 2018, the lowest mean value for valence was recorded. Valence of songs was falling until 2018, when it began to rise.

2. Clustering songs based on Energy and Danceability.

The clusters are created by taking energy for X axis and danceability for Y axis. Tracks released between 2020-06-01 and 2021-01-01 were chosen for this analysis. Below figure shows the six clusters of tracks that are plotted in different colors. The clusters' centers are denoted by a red dot.



A higher proportion of songs have a danceability of 0.4 to 0.9 and an energy of 0.4 to 1. Using the clusters created, we can categorize songs as having high, average, or low danceability and energy. These song clusters can be used by Spotify to create song albums with different danceability and energy levels.

3. Analysis of how features of a songs like energy, loudness, speechiness, acousticness, liveness, tempo and instrumentalness changes with the time

The results of this analysis can be shown in following charts. In this analysis we did not consider 1900/01/01 as valid point as only one single data has been recorded regarding that date. We use all the data recorded in period of 1900 to 2021.

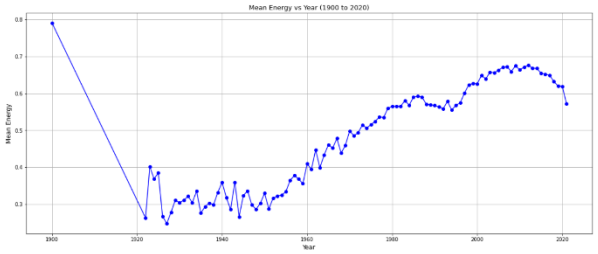


Figure 7 Mean Energy vs Year graph.

According to chart highest mean value of energy is after 1900 is in between 2000 to 2020.When we consider maximum value of mean energy it recorded in 2012. This time series plot 1900 to 2020 shows trend in between 1920 to 2012. Let look at how mean energy varies in the period of 2012.

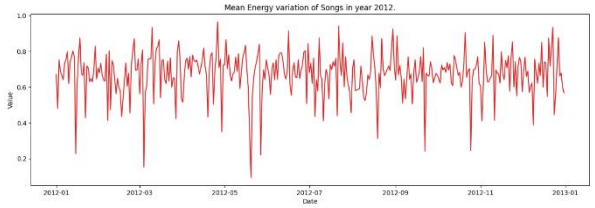


Figure 8 Mean Energy variation in year 2012.

When we look at the mean energy that was computed by grouping by release date show a stationary type of plot. We use AIC statistical model to compute p value to detect whether a data series is stationary or not in this scenario we got 4.482404352983931e-23 as the p value. This p value is less than .05 so we can say that mean energy variation of year 2012 is stationary.

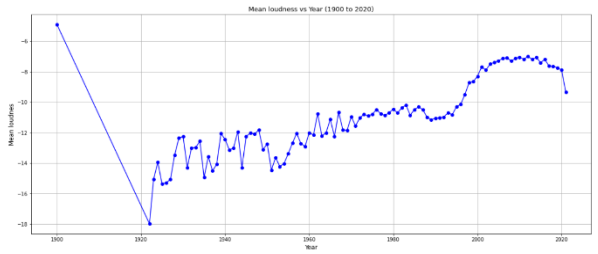


Figure 9 Mean Loudness vs Year graph.

According to chart highest mean value of loudness is after 1900 is in between 2000 to 2020.When we consider maximum value of mean loudness it recorded in 2012. This time series plot 1900 to 2020 shows trend in between 1920 to 2012. Let look at how mean loudness varies in the period of 2012.

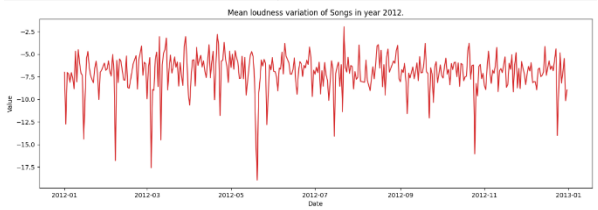


Figure 10 Mean Loudness variation in year 2012.

By the shape of the plot, we can say that data series of mean loudness is stationary in year 2012.when we compute p value from AIC statistical model, we got p value as 4.8724601998798374e-30 which is less than 0.05 which means statistically we can confirm this plot is stationary.

And we can observe that both loudness and Energy shows similar variation timeseries plots and same year of highest mean value and very similar p values. We can discuss the correlation too.

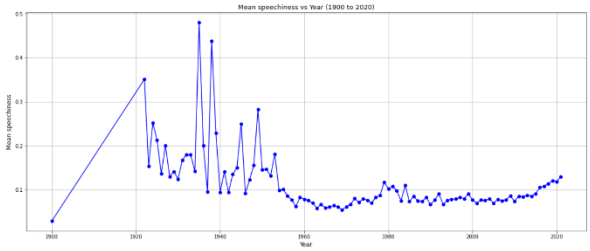


Figure 11 Mean Speechiness vs Year graph.

According to chart highest mean value of speechiness is in between 1920 to 1940.When we consider maximum value of mean speechiness it recorded in 1935. This time series plot from 1960 to 2020 has slow growth of mean speechiness of songs. Let look at how mean speechiness varies in the period of 1935.

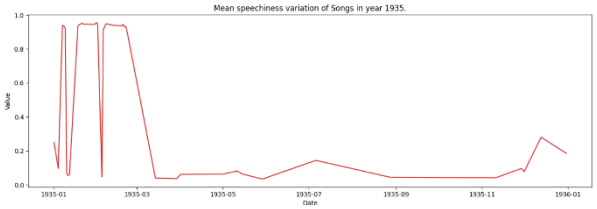


Figure 12 Mean speechiness variation in year 1935.

By the shape of the plot, we can't clearly say anything about stationarity of data series of mean speechiness in year 1935.when we compute p value from AIC statistical model, we got p value as 0.08429768164374285 which is higher than 0.05 which means statistically, we can confirm this plot is not stationary.

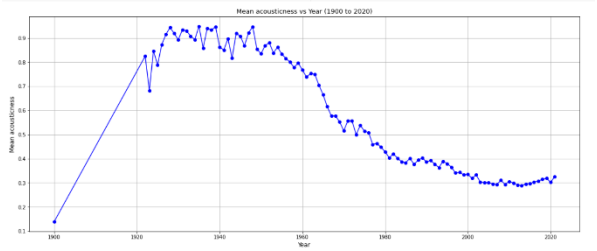


Figure 13 Mean acousticness vs Year graph.

According to chart highest mean value of acousticness is in between 1920 to 1940.When we consider maximum value of mean acousticness it recorded in 1935. From 1920 to 1935 has high values of mean and after 1940 it starts to reduce and 2000 to 2020 it has a very slow growth. Let look at how mean acousticness varies in the period of 1935.

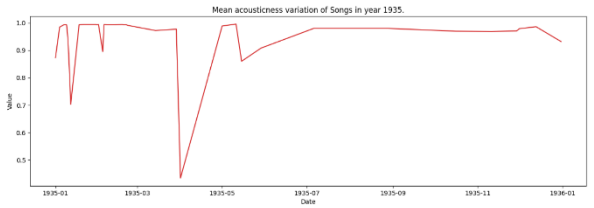


Figure 14 Mean acousticness variation in year 1935.

By the shape of the plot, we can say data series of mean speechiness in year 1935 is stationary. when we compute p value from AIC statistical model, we got p value as 3.5544180147014064e-07 which is lesser than 0.05 which means statistically, we can confirm this plot is stationary.

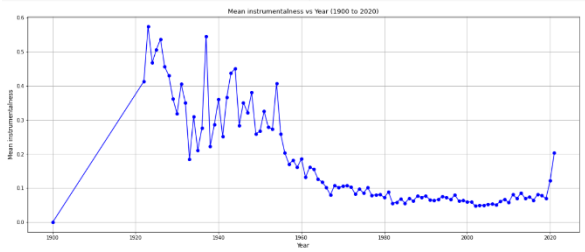


Figure 15 Mean intrumentalness vs Year graph.

According to chart highest mean value of intrumentalness is in between 1920 to 1960.When we consider maximum value of mean intrumentalness it recorded in 1923. After 1960 it starts to reduce and in 2020 it has started growth again. Let look at how mean intrumentalness varies in the period of 1923.

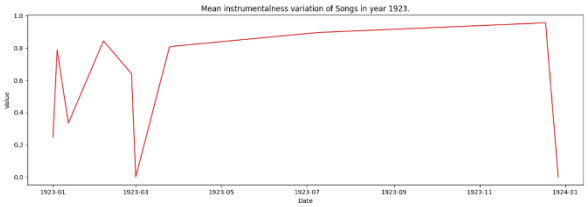


Figure 16 Mean intrumentalness variation in year 1923.

By the shape of the plot, we can't clearly say anything about stationarity of data series of mean intrumentalness in year 1923.when we compute p value from AIC statistical model, we got p value as 0.06198968773525048 which is higher than 0.05 which means statistically, we can confirm this plot is not stationary.

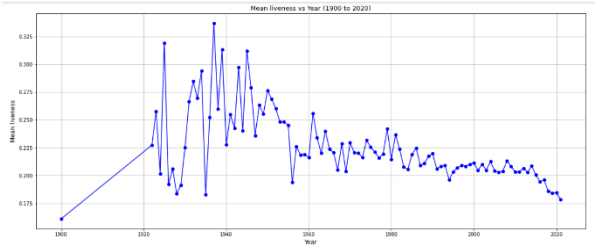


Figure 17 Mean liveness vs Year graph.

According to chart highest mean value of liveness is in between 1920 to 1960.When we consider maximum value of mean liveness it recorded in 1937. This time series plot 1920 to 1960 has high values of mean variation and after 1960 it shows reduction. Let look at how mean liveness varies in the period of 1937.

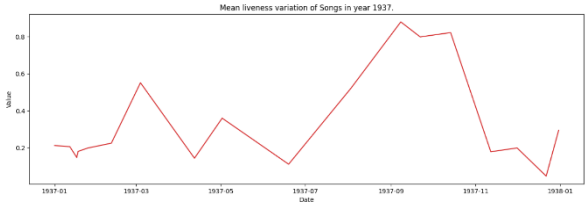


Figure 18 Mean liveness variation in year 1937.

By the shape of the plot, we can't clearly say anything about stationarity of data series of mean liveness in year 1937.when we compute p value from AIC statistical model, we got p value as 0.9308122070074111 which is higher than 0.05 which means statistically, we can confirm this plot is not stationary.

According to chart show in figure 19 highest mean value of tempo is in between 2000 to 2020 middle half after 1900. This time series plot 1920 to 2011 has shown an increase of a mean tempo value and tempo and plot show reduction in later part 2000 to 2020 period. When we consider maximum value of mean liveness it recorded in 2011.

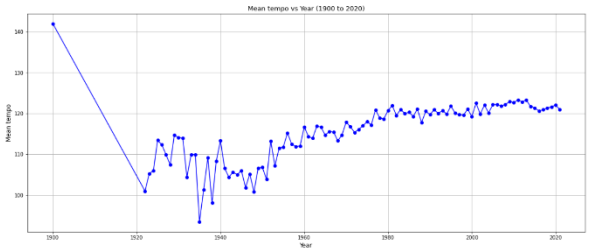


Figure 19 Mean tempo vs Year graph.

Let look at how mean tempo varies in the period of 2011.

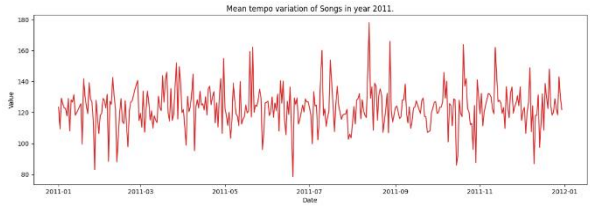


Figure 20 Mean tempo variation in year 2011.

By the shape of the plot, we can say data series of mean tempo in year 2011 is stationary. when we compute p value from AIC statistical model, we got p value as 2.4812592076772166e-30 which is lesser than 0.05 which means statistically, we can confirm this plot is stationary.

4. Time Series forecasting of danceability feature by using Facebook prophet and ARIMA Model.

a) Facebook prophet

```
phropetFocastDf.head(5)
```

	y	ds
229896	0.595826	2010-01-01
371569	0.568000	2010-01-02
74219	0.440804	2010-01-03
334604	0.659222	2010-01-04
334630	0.637316	2010-01-05

Figure 21 preprocessed Dataset

As per the first step we must preprocess our data according to conditions of Facebook prophet Model for that we have to convert release date attribute a Datetime object and label it as 'ds' and label danceability as 'y' as per Figure 21 shows.

yhat
0.591200
0.593264
0.592751
0.590848
0.585413

Figure 22 y hat prediction values

After fitting data Facebook prophet Model, we can read forecast data from predict method it returns a dataset of 100 new predicted values in label of y hat.

```
plot1 = m.plot(forecast)
```

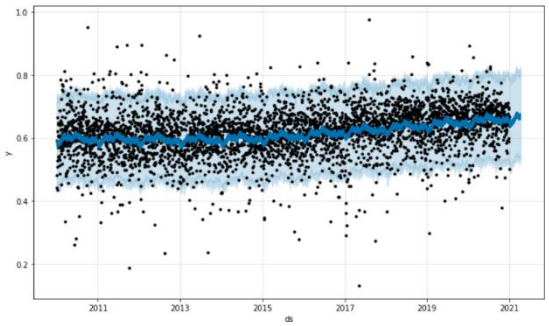


Figure 23 plot of danceability with forcast

In Figure 23 it shows plot of danceability with scatter distribution in period of 2010/01/01 to 2021/01/01 and it also extends up to 100 more future dates until 2021/04/10. The thick light blue line indicates the prediction done by Model.

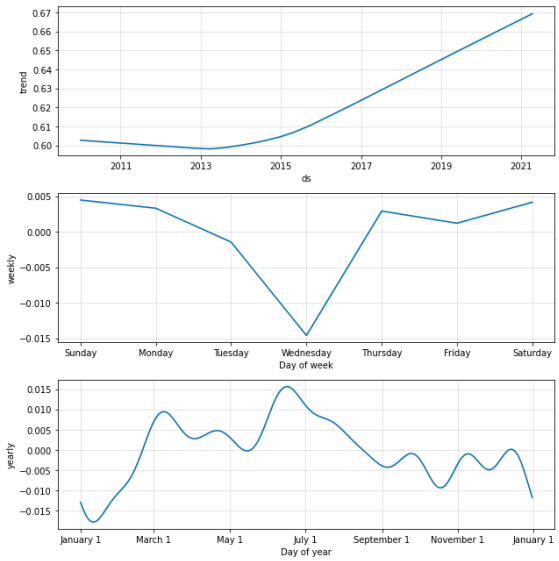


Figure 24 plots on trends, yearly and weekly prediction.

Facebook prophet Model provides methods to compute plots on trends, yearly and weekly prediction and it has no issue with stationary data when doing forecasting related to time series data, but data series should be historical.

b) ARIMA Model

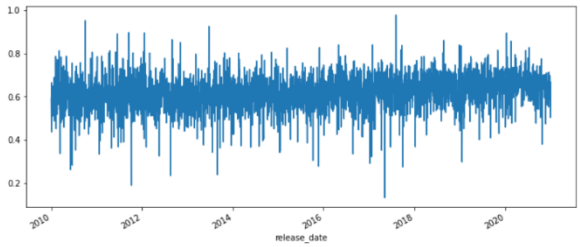


Figure 25 Mean Danceability variation in 2010 to 2020

As we discussed in methodology section ARIMA model support only features with stationary type. To detect stationarity of a feature we use statical model called AIC. we have computed p value of danceability, and result was 8.592894407507869e-07 which is less than 0.05 so danceability feature can use in ARIMA model.

SARIMAX Results

Dep. Variable:	y	**No. Observations:**	3962
Model:	SARIMAX(4, 1, 1)	**Log Likelihood**	4799.122
Date:	Fri, 11 Jun 2021	**AIC**	-9584.244
Time:	15:55:15	**BIC**	-9540.254
Sample:	0	**HQIC**	-9568.643
	- 3962		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	9.533e-06	1.22e-05	0.784	0.433	-1.43e-05	3.34e-05
ar.L1	0.0419	0.015	2.858	0.004	0.013	0.071
ar.L2	-0.0299	0.017	-1.717	0.086	-0.064	0.004
ar.L3	-0.0024	0.016	-0.149	0.882	-0.034	0.029
ar.L4	0.0043	0.016	0.272	0.786	-0.027	0.035
ma.L1	-0.9899	0.003	-386.426	0.000	-0.995	-0.985
sigma2	0.0051	7.48e-05	68.839	0.000	0.005	0.005

Ljung-Box (L1) (Q):	1.87	**Jarque-Bera (JB):**	1842.87
Prob(Q):	0.17	**Prob(JB):**	0.00
Heteroskedasticity (H):	0.71	**Skew:**	-0.50
Prob(H) (two-sided):	0.00	**Kurtosis:**	6.19

Figure 26 Result of auto_arima function

Then we have to find the best ARIMA model to predict the future values of mean danceability for that we use a function called auto_arima function which provide by pmdarima library. auto_arima calculate a AIC score to judge how good a particular order model is then it select the order of model with minimum AIC score and suggest us the order as per figure 25 it highlight order suggest for danceability feature is 4,1,1.

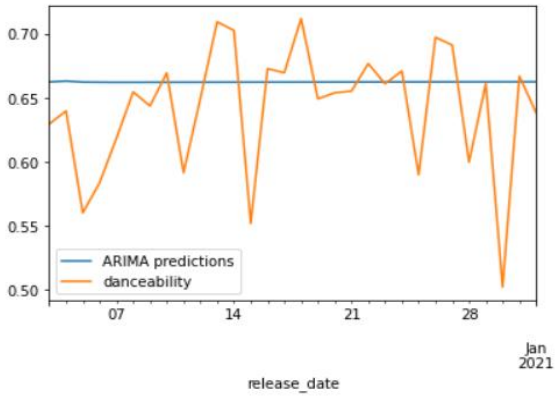


Figure 27 comparison of model prediction and real danceability variation

After finding the order we have to fit the data with mode for that we create 2 models 1st model is to evaluate our performance model and 2nd model is to predict the future values of danceability. In 1st model we divide our data to train data and test data. we use 3932 data points to train and 30 data point to test.

When evaluating the model, we 1st find the mean value of test data set as 0.6425765622746 and root mean squared error of prediction as 0. 0524486588856.So average error of our ARIMA model with order of 4,1,1 going to be roughly 8.162242752823356%.

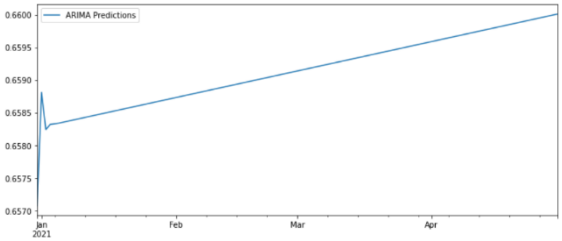


Figure 28 prediction of 2nd Model

In 2nd model we predict future dancabilty valuse for 120 days.

2021-01-02	0.656787
2021-01-03	0.656325
2021-01-04	0.654476

Figure 29 Prediction from Facebook prophet

2021-01-02	0.658239
2021-01-03	0.658319
2021-01-04	0.658328

Figure 30 prediction from ARIMA Model

When observing Figure 29 and Figure 30 we can say both ARIMA and Facebook prophet Models provides a similar level of prediction values.

5. Classification of the level of popularity of a song

```
Test set accuracy = 0.7144404052904324
Test set f1 measure = 0.6117950515867921
Test set precision = 0.5889440037487235
Test set recall = 0.7144404052904324
```

Figure 31 Result of MulticlassClassificationEvaluator

We use Naïve Bayes method to classify the level of popularity of song as High, Medium and Low by using Genre, liveness, mode, danceability features. After encoding and fitting data to Naïve Bayes model we evaluate the classification model using MulticlassClassificationEvaluator.

6. Correlation analysis of song features.

After preprocessing the relevant data columns, a matrix of the correlation coefficient between 9 features were obtained using the PySpark correlation calculation function where the results are as follows.

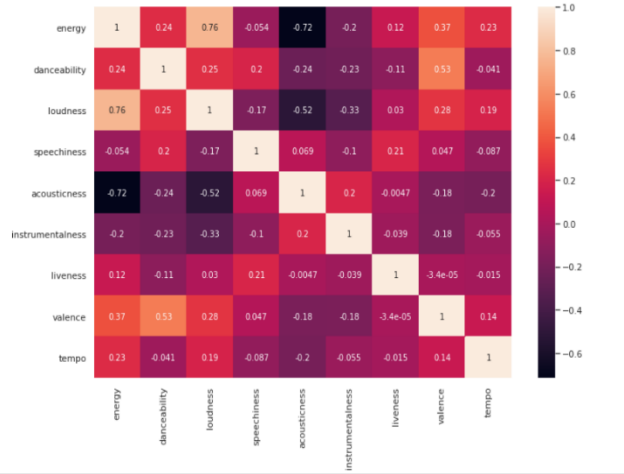


Figure 32: Result of correlation analysis

According to the results shown in the above heatmap, lpoudness and the energy has the highest positive correlation which is of 0.76 whereas the highest negative correlation of -0.72 was recorded between acousticness and energy.

On a sidenote it is evident that danceability of a song also increases when energy and the loudness increase. Furthermore, is it clear that valence of the song has a negative correlation with liveliness and acousticness although the strength of the relationship is very low. In a nutshell, all the features that have a positive value as their Correlation Coefficient has a positive relationship with one another and vice versa. Also, of the Correlation Coefficient is near 1(more than 0.8) it indicates a string relationship between the variables.

7. Regression analysis for loudness vs. energy

The results of the linear regression model between energy vs. loudness was measured by using Absolute Mean Error and the Mean Squared Error values.

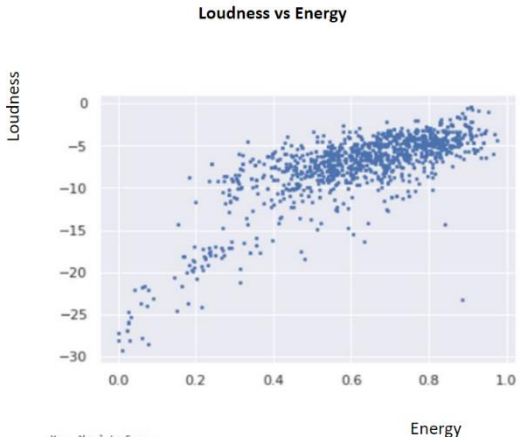


Figure 33: Scatter diagram for Energy vs. Loudness

The Mean Absolute Error (MAE) for the linear regression was recorded as 1.511 and the Mean Squared Error (MSE) was reported as 4.572 [15].

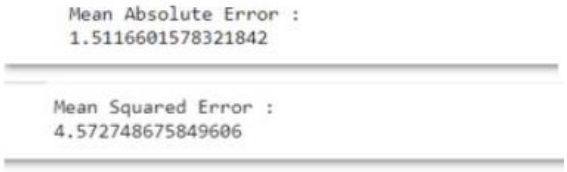


Figure 33: Evaluation of Regression model

Using this model musicians or producers can predict the loudness of a song when the energy level is measured.

V. FURTHER WORK

Other than the methods presented in this paper, there are numerous other ways to analyze the dataset. It is possible to recover more hidden patterns among data by properly analyzing the dataset. Can use different types of classification and clustering techniques and compare the results. We can use random forest classification techniques to classifications.

VI. CONCLUSION

In this paper, a large-scale global song dataset, Spotify Dataset, is analyzed. The dataset contains more than 600,000 tracks that are released between 1922 and 2021.

Earlier research has revealed that several studies have been conducted to analyze song data for a variety of purposes. These studies did not conduct in-depth analyses of song features and future trends, nor did they employ machine learning models to make predictions.

In this paper, the analysis has been done with the use of following tools and techniques which are namely, Apache Spark 3.1.1, Pyspark, Python and Java 8 and the Analysis is performed under 7 stages throughout the project. An in-depth analysis of the dataset was performed in order to discover relationships between song features, cluster songs based on song features, and forecast future trends using machine learning models such as ARIMA and Facebook prophet.

In this paper, according to observations of time seirs analysis both mean energy vs year and mean loudness variation were very similar to each other reason for that is both of feature has positive correlation of .76 and another observation related time series analysis is with the time more songs in Spotify have less lyrics this was shown by spechiness variable. And when come to time series forecasting both Facebook prophet and ARIMA model gives us similar level of prediction. But to fit a feature to ARIMA the feature should be

stationary in relevant timeline. When Genre, liveness, mode, danceability features precented we can classify a level of popularity of song with 71.44% accuracy by using Naïve Bayes method.

Finding frequent patterns among the song features can be very much useful for various entities, especially for music producers, artists, music production houses as well as researchers who are in the academia of music. Investigating the relations between various song features using Correlation analysis has led us to discover some strong crucial correlations such as the positive correlation between loudness and energy as well as the negative correlation between acousticness and energy. Extending the discoveries of the Correlation analysis we have built a linear regression analysis where the users can predict the loudness of a song when the energy level is given to the model with a Mean Absolute Error of 1.51 which will be beneficial, especially for music producers.

VII. REFERENCES

[1] "8 reasons why music is important to us," *Mitch de Klein*. https://www.mitchdeklein.com/blog/201688-reasons-why-music-is-important-to-us (accessed Jun. 12, 2021).

[2] "Spotify Artist Recommender | by Michael Whittle | Apr, 2021 | Towards Data Science." https://towardsdatascience.com/spotify-artist-recommender-7950af1fe20a (accessed Jun. 12, 2021).

[3] "What makes a song popular? Analyzing Top Songs on Spotify - KDnuggets." https://www.kdnuggets.com/2021/04/song-popular-analyzing-top-songs-spotify.html (accessed Jun. 12, 2021).

[4] "Spotify Dataset 1922-2021, ~600k Tracks | Kaggle." https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks?select=tracks.csv (accessed Jun. 01, 2021).

[5] "Quickstart — PySpark 3.1.1 documentation." https://spark.apache.org/docs/latest/api/python/getting_started/quickstart.html (accessed May 27, 2021).

[6] "Pyplot tutorial — Matplotlib 3.4.2 documentation." https://matplotlib.org/stable/tutorials/introductory/pyplot.html (accessed May 27, 2021).

[7] "Java Platform Standard Edition 8 Documentation." https://docs.oracle.com/javase/8/docs/ (accessed May 27, 2021).

[8] "pandas documentation — pandas 1.2.4 documentation." https://pandas.pydata.org/docs/ (accessed May 27, 2021).

[9] "Time Series Forecasting With ARIMA Model in Python for Temperature Prediction | by Nachiketa Hebbar | The Startup | Medium." https://medium.com/swlh/temperature-forecasting-with-arima-model-in-python-427b2d3bcb53 (accessed Jun. 12, 2021).

[10] "Time Series Forecasting Methods | Arima In Python and R." https://www.analyticsvidhya.com/blog/2018/08/auto-arima-time-series-modeling-python-r/ (accessed Jun. 12, 2021).

[11] "Time Series Forecasts using Facebook's Prophet," *Analytics Vidhya*, May 10, 2018. https://www.analyticsvidhya.com/blog/2018/05/generate-accurate-forecasts-facebook-prophet-python-r/ (accessed Jun. 12, 2021).

[12] "Pearson Correlation Assumptions," *Statistics Solutions*, Jan. 30, 2013. https://www.statisticssolutions.com/pearson-correlation-assumptions/ (accessed Jun. 12, 2021).

[13] "Regression Techniques in Machine Learning," *Analytics Vidhya*, Aug. 13, 2015. https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression/ (accessed Jun. 12, 2021).

[14] "sklearn.linear_model.LinearRegression — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html (accessed Jun. 12, 2021).

[15] Stephanie, "Absolute Error & Mean Absolute Error (MAE)," *Statistics How To*, Oct. 25, 2016. https://www.statisticshowto.com/absolute-error/ (accessed Jun. 12, 2021).