Hit song prediction using Spotify data

Hemankith Reddy M
Department of Computer Science and Engineering
PES University
PES2UG19CS147
hemankith.pes@gmail.com

Jayant Harwalkar
Department of Computer Science and Engineering
PES University
PES2UG19CS160
jayanthharwalkar@gmail.com

Skanda S
Department of Computer Science and Engineering
PES University
PES2UG19CS391
skanda2594@gmail.com

Vijay Murugan A S
Department of Computer Science and Engineering
PES University
PES2UG19CS454
vijaymurugan1457@gmail.com

I. Introduction

If we could predict product sales before they are released on the market, business would be much easier. As the cost of failure in new product development is very high, researchers and product developers are looking for good product success/failure prediction models. Research that aims to answer these questions has established many approaches to creating success prediction models. To give examples, we define success based on organizational and industry factors, social data, and predictions from test markets. Predicting the popularity of electronic household products, however, seems a lot more straightforward than predicting cultural products such as music. Their success and popularity seem related to taste and more subjective measurements which makes prediction all the more complex.

Music is an important type of online information. Amounts of money earned by the music industry indicate its importance. In 2020, the total revenue of the recorded music industry amounted to 23.1 billion U.S. dollars. In 2019, revenue of industry was 11.1 billion U.S. dollars were up 13% versus 9.8 billion U.S. dollars the prior year.[13]

Technological advancements have seen the rise of streaming services. In 2020, 54% of the total revenue was generated by streaming services. Streaming services like Spotify have had a positive influence on the revenue growth however controversial they may be on the profitability for the artists.

There are two main branches in music industry scientists and engineers are working on: Recommendation and Prediction.

The growth in streaming and related technologies have enabled users to experience a new flow of services like search-able music collections, automatic playlist suggestions, music recognition systems and more. They can do so because of the (user generated) big data and their digital song database.

Next to the importance of recommendation, there is the importance of prediction of music popularity. All parties in the music industry have an interest in finding content consumers will want to buy, and it remains one of the biggest mysteries in the industry why some songs are successful, while others do not.

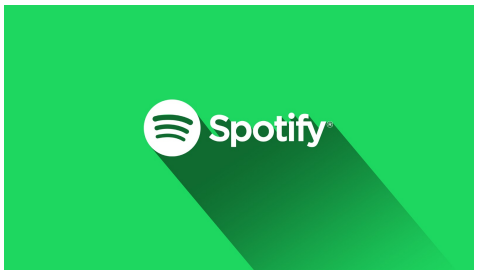

Fig. 1. Spotify

A relatively new approach focuses on the data generated by streaming services like Spotify. Hit

song science (HSS) aims to predict whether a given song will become a chart-topping hit. The underlying assumption in HSS is that hit songs are similar with respect to their features. Hit Song Science is an active research topic in Music Information Retrieval (MIR).

The insights gained in this field can provide huge benefits for the industry and all parties involved in the music content life-cycle. Beneficial examples include that artists can work reversely the process of HSS and focus on characteristics that make their songs more popular and that record companies, aiming at maximum profit, could benefit by selecting the most promising works for publication and marketing goals.

Moreover, music streaming services are always looking ways to diversify their revenue channels. The data they hold can open new possibilities in doing so. For example, they can sell prediction models to record companies and artists and also us e them to fine-tune and improve their own services. It illustrates the importance of product success prediction and the emerging research field of MIR and HSS in the field of business.

II. Similar Approaches

A. Paper I

The name of the first paper is "Dance Hit Song Prediction". In this research, accurate models are built to predict if a song is a top 10 dance hit or not. For this purpose, a dataset of dance hits including some unique audio features was compiled. Based on this data different efficient models are built and compared. A total of five models were built for each dataset using diverse classification techniques.

- C4.5 tree
- RIPPER ruleset
- Naive Bayes
- Logistic regression
- Support vector machines

The study describes two experiments. The first one builds models for all of the three datasets (D1, D2 & D3), both with and without feature selection. The evaluation is done by taking the average of 10 runs, each with a 10-fold cross-validation procedure. During the cross-validation procedure, the dataset is divided into 10 folds. Nine of them are used for model building and one for testing. This procedure is repeated 10 times. In the second experiment, the performance of the classifiers on the best dataset is compared with an out-of-time test set.

AUC	D1		D2		D3	
	–	FS	–	FS	–	FS
C4.5	*0.53*	*0.55*	*0.55*	*0.54*	*0.54*	*0.53*
RIPPER	*0.55*	*0.56*	*0.56*	*0.56*	*0.54*	*0.55*
Naive Bayes	**0.64**	**0.65**	*0.64*	**0.63**	**0.6**	*0.61*
Logistic regression	**0.65**	**0.65**	**0.67**	**0.64**	**0.61**	**0.63**
SVM (Polynomial)	*0.6*	*0.59*	*0.61*	*0.61*	*0.58*	*0.58*
SVM (RBF)	*0.56*	*0.56*	*0.59*	*0.6*	*0.57*	*0.57*

FS = feature selection, $p < 0.01$: italic, $p > 0.05$: bold, best: bold.

Fig. 2. Results for 10 runs with 10-fold validation

Although decision trees and rulesets do not always offer the most accurate classification results, their main advantage is their understandability. Support vector machines do not perform very well on this particular problem. The overall best technique seems to be the logistic regression, closely followed by naive Bayes. A second experiment was conducted with an out-of-time test set. The instances were first ordered by date, and then split into a 90% training and 10% test set. An interesting observation from this experiment is that the model seems to be able to predict better for newer songs.

An interesting future expansion would be to improve the accuracy of the model by including more features such as lyrics, social network information and others. One disadvantage of this model is that accuracy decreases as the songs get old. This is because of the evolution of music with time. The songs which were hit, for example 30 years ago, might not come out hit in this age. This goes the other way too.

B. Paper II

The second paper "Machine Learning approach for genre prediction on Spotify top ranking songs" focused on analyzing the daily song ranking on Spotify. This study analyzed characteristics of top ranking songs with respect to their audio features. The data was collected from multiple sources and merged together into the final dataset. There are 13 audio features including acousticness, danceability,

duration time (in milliseconds), energy, instrumentalness, key, liveness, loudness, mode, speechiness, tempo, time signature, valence.

This research paper used OneVsRestClassifier which is a multi-label classifier based on the idea of Support Vector Classifier. Its strategy is to fit one classifier per class. For each classifier, the class is fitted against all the other classes. Since each class is represented by one and one classifier only, it is possible to gain knowledge about the class by inspecting its corresponding classifier. The main advantage of this classifier is its interpretability.

The songs should be categorized into genres based on their audio features. The labels are in ten music categories: Electronic, Folk, Funk/Soul, Hip Hop, Jazz, Latin, Pop, Reggae, Rock, and Stage & Screen. The model was given the song's audio features in the test set. The model predicted their genres as an output for songs in the test set. The OneVsRestClassifier implements Support Vector Classifier with the kernel function set to linear. This study is a multilabel classification, each song can have any number of labels. The genre label with the highest probability is the final predicted genre.

TABLE I
PRECISION AND RECALL FOR EACH GENRE [4]

	Precision	**Recall**
Electronic	0.2945	0.3992
Folk	0	0
Funk/Soul	0	0
Hip-Hop	0.6215	0.8213
Jazz	0	0
Latin	0.1954	0.3667
Pop	0.3060	0.0978
Reggae	0	0
Rock	0	0
Stage & Screen	0	0

The machine learning method did not achieve a high accuracy in this study. The model was not sensitive to distinguish these three genres well. One possible reason is that the data is not very distinguishable. For example, some genres have similar audio features so that it might confuse the model in predicting the correct genre. The accuracy of the model is 46.9%. Another reason is that some songs stayed among daily top 200 for days, weeks, even months. If the model predicted these songs as an incorrect genre, it would repeat as many times as it stayed among the top 200.

C. Paper III

The third paper "What makes for a hit pop song? What makes for a pop song?" tries to classify the songs to their respective genres and also tries to predict if a given song will become popular or not. The genres included were: classic pop and rock, folk, dance and electronica, jazz and blues, soul and reggae, punk, metal, classical, pop, and hip-hop. Features for each song include loudness, tempo, time signature, key, mode, duration, as well as average timbral data and average timbral variance. In order to measure the popularity of a song, for each song the number of view counts was collected. The classification models that were used are:

- Linear Classification using Support Vector Machines
- String Kernel

The paper concluded that from the audio features extracted there does not seem to be the information relevant in making the song popular. This could be a result either of feature selection, or of popularity being driven by social forces, i.e. the inherent unpredictability of cultural market. Furthermore once an artist is popular, they may later produce works which are musically different from those that made them popular, yet the new tracks will become popular simply by virtue of being created by the popular artist.

Using the million song genre subset, several classification algorithms including Support Vector Machines with tenfold cross validation, k-nearest neighbors, and random forests were tested. These were run on the entire data set, on a uniformly distributed subset, and also on a four-genre subset consisting of classical, metal, soul and reggae, and pop. In addition, Support Vector Machines were trained on individual pairs of genres, with n songs from each genre.

Results for various methods on different subsets are reported in the table. Because the dataset was not uniformly populated, a uniform dataset was used. In all three classification scenarios, Random Forests perform approximately as well as K-means in combination with Support Vector Machines.

TABLE II
ACCURACY FOR DIFFERENT MODELS USED [4]

	Precision	Recall
All 59600 Songs	SVM	49.09
All 59600 Songs	K-Means + SVM	54.15
All 59600 Songs	Random Forest	56.80
All 59600 Songs	10-Nearest Neighbors	43.84
Uniform Genre	SVM	47.60
Uniform Genre	K-Means + SVM	52.93
Uniform Genre	Random Forest	51.18
Uniform Genre	10-Nearest Neighbors	19.72
4 Genre	SVM	76.00
4 Genre	K-Means + SVM	83.35
4 Genre	Random Forest	83.19
4 Genre	10-Nearest Neighbors	74.58

D. Paper IV

This paper focuses on solving the HSS (Hit Song Science) problem and aims to predict which songs will become a hit. The assumption in HSS is that hit songs are similar with respect to their features. Predicting Hit songs would prove to be useful to Artists, Music Labels and productions to generate a larger revenue and increase profits by producing the music that are likely to become a hit.

They used the Spotify API to create a dataset with around 1.8 million songs and then reduced the size of the dataset by considering the songs between 1985 and 2018. A dataset with unique songs in Billboard Top 100 in the years 1985 and 2018 was created using Billboard API which consisted of around 16k songs. After merging the above mentioned datasets, the new dataset contained about 12k hit songs. In order to balance the data, another 12k non-hit songs were added and a new dataset was created which consisted of 12k hit and non-hits songs. Each track consisted of 27 attributes. The train, test and validation sets were also created after the processing of SpotifyBillboard features and made sure that there was no overlapping between these sets. In addition to the audio features, they also considered the artist's history and past tracks to make it as practical as possible.

They used four different models to predict.

- Logistic Regression
- Random Forest
- Neural Network
- Support Vector Machine

Each model had different accuracy on test and validation sets.

Models	Accuracy		Precision		Recall	
	Test	Val	Test	Val	Test	Val
Logistic Regression	0.8151	0.8065	0.7526	0.7457	0.9391	0.9298
Neural Network	0.8214	0.8305	0.8235	0.8233	0.7913	0.7671
Random Forest	**0.877**	**0.887**	**0.86**	**0.87**	**0.9**	**0.89**
SVM	0.839	0.828	0.995	0.993	0.704	0.706

Fig. 3. Results for 10 runs with 10-fold validation

E. Paper V

The name of the paper is 'Cluster Analysis of Musical Attributes for Top Trending Songs' which aims to understand what attributes make certain songs trendy and help services, artists and labels to better user experience and gain marketing profits in the music industry by performing cluster analysis.

The dataset used by them was the 'Top 100 Trending Spotify Song of 2017' which had approximately 12-13 attributes. They then checked correlation between the attributes which came out to be consistent. There was a high correlation between loudness and energy which was not an issue in this study as it focuses on clustering which measures distance.

They then cleaned the dataset and filtered out all the unwanted features which wouldn't be effective for cluster analysis. Some features which had low variance and the ones which were nominal proved no use in cluster analysis and hence were removed. Later, the non-categorical features were normalized so that each attribute has equal weight. The type of clustering they used was K-means Clustering.

The optimal value for K had to be calculated. For this, they used the Silhouette method and Agglomerative clustering to decide that 2,4,5 values for K were optimal and further study of overlaps resulted in the optimal value for K being 4. Hence 4 was selected and K-means clustering was performed to identify groups of trending songs.

After clustering, multiple scatter plots were plotted with one dimension being a song attribute and the other being one of the cluster labels. The results showed that all the clusters consisted a majority of Pop and Dance tracks from the trending list and hence concluded that the genres of Pop and Dance contained a successful, chart topping musical structure that are high in loudness and low in speechiness.

F. Paper VI

The goal of this paper is to predict the popularity of a song before it is actually released, to identify emerging trends and understand the factors that affect the popularity of a song. In speech and music signal processing, CNN(Convolutional Neural Network) Models have exhibited remarkable strength in learning task-specific audio features from data and outperforming other models in many predicting tasks.Hence they decided to use the CNN models for feature learning and for extracting high level audio features. Therefore, they formulated hit song prediction as a regression problem and test how they can predict the popularity of Chinese and Western Pop music among Taiwanese KKBOX users. So the main aim is to apply CNN and check whether it can prove to be effective in doing this task.

They obtained a dataset of user listening records contributed by Taiiwanese users over a period of one year and it included very popular Chinese Pop songs and the Western songs. They were able to obtain this data in collaboration with KKBOX Inc. In the initial steps of data preprocessing, they checked whether they had to compensate for the bias due to the songs being released at different times. After some plotting, they came to the conclusion that they didn't have to compensate for the time bias. They did not use only the play counts of the songs for determining its success as it is possible that only few users contributed to the play counts of a song and hence they took the product of log of play counts and the number of users who listened to the song. After that they sampled 10k songs for both the subsets-Chinese Pop and Western and then split each subset into training , validation and test sets. They used 5-6 models:

- Linear Regression: They used a 256-dim feature vectors per song which is the input to a single layer shallow neural network model.
- CNN: Their CNN model consisted of 2 early convolutional layers(128 by 4 and 1 by 4 layers) and 3 late convolutional layers(all 1 by 1 convolutional kernels).
- Inception CNN: It uses multi-scale vectors to learn features.
- JYnet + Linear Regression: JYnet is a CNN model trained to predict tag-set and the output of this model acts as an input to the LR model which predicts hit scores.
- Combination of [(4) and (2)] and [(4) and (3)]: Here they optimise the model parameters of both the models jointly.

On comparing the first three models, it was found that better results were shown when deep and complicated models and structures were used showing the effectiveness of deep structures for this task.They also inferred that audio-based hit prediction is easier in Mandarin subset.

Comparing the first and the fourth models, it was found that the tag based method outperforms the simple LR model for the Western subset.

They concluded that deep and complicated neural structures are very effective in hit prediction and are very important for the Western songs which are very diverse.

G. Paper VII

Many researchers agree that Spotify is a well-known music streaming application for younger generations (Cummings, 2016; Riesewijk, 2017; Swanson, 2013). However, those researchers do not elaborate further on the use of Spotify in English teaching. The Curriculum 2013 emphasizes to build students' characters, developing relevant skills based on students' interests and needs, and developing a thematic approach that benefits students' cognitive abilities (Gunawan, 2017).

This research is to aim to explore which kinds of content words frequently appear in the song lyrics. The first step to analysis is to identify the content words used in the songs. Abrusán,Asher and Van de Cruys (2018) say that content word is a reflex of knowledge words. It means that content word is a word that conveys information in a text or speech act in English grammar and semantics (Nordquist, 2018). On the other words, content words cover nouns, verbs, adjectives, and adverbs as a type of content words to be able to take a part in productive compounding and derivation (Schmauder, Morris & Poynor, 2000; Nordquist, 2018). It means that, each types of content words found in the song lyrics represent the song's meaning. In specific, this paper is therefore to answer the following questions:

- How are content words categorized based on meaning?

- How frequent is each content words appeared on the song's lyrics?

The method used classified content words used in the songs under dissection into classes, record the frequencies of these words, and then draw conclusions from this data. However, the sample size was far too small to make accurate conclusions like whether the song was suitable for teens based on parameters such as the amount of swear words that it contains, the words that have meanings that align with the goal of Curriculum 2013.

H. Paper VIII

The name of the paper is "Collaborative Filtering in Spotify". Collaborative filtering is the most common method for recommender. It uses the K-nearest neighbour technique (KNN) on to a friend's recommendation. It assumes the fact that friends have similar tastes in music. Collaborative Filtering can further be divided into three types:

- Memory Based
- Model-Based
- Hybrid

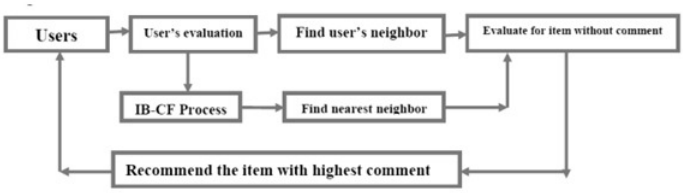

Fig. 4. Statistic to show whether the recommended music matches the taste of the user or not.

Collaborative filtering uses feedback control by error. clicking "like" is widely accepted. The service for recommendation is based on the history of clicking the "like" record. It also uses what songs are "liked" by the users, and use that information to recommend songs , from which it collects data on whether the users liked the recommended songs or not.

Only less than 20 per cent are fully satisfied with the recommendation service. So there is still progress to make for the feedback system. Moreover, every new user experiences a cold-start. The personalization would be weakened as a result of popular songs which will be recommended. A perfect recommender should not involve too much human effort, because users are not always willing to rate. The data is not always representative due to ratings growing towards those who rate.

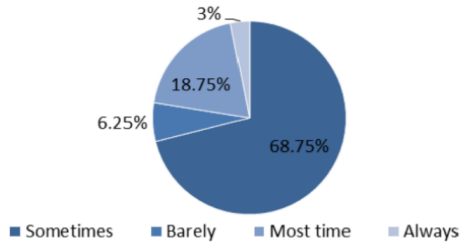

Fig. 5. Statistic to show whether the recommended music matches the taste of the user or not.

I. Paper IX

This research investigates the relationship between audio features of songs from the Spotify database and song popularity measured by the number of streams a song has on Spotify. The attribute-approach was used to explore whether song attributes have an explanatory power on stream count. It will address the identified gap in the existing product success prediction field of HSS by analysing stream count on Spotify instead of Spotify's popularity metric, defining popularity as hits or non-hits or by chart position.

They used the Spotify database for collecting data. They used the Spotify Search API to obtain the audio features for each song and also wrote a python script using the Spotipy Web API to automatically pull data from the Spotify Database. To make their data balanced, they included 10 most popular genres on Spotify and each song was labelled with one of these genres. The data extracted was for the year 2017 so as to eliminate bias towards songs that were released earlier since they had more time to solicit the streams. They selected 100 most popular songs from each genre adding up to a total of 1000 songs.

Data preprocessing was performed to gain insights and clean data. Some of the features were normalized to give equal weights and some nominal variables were encoded. They approached the problem with a Linear Regression Model and made sure all the assumptions of regression such as no autocorrelation, no multicollinearity,etc, were met. They

plotted the distribution of dependent variables and heteroscedasticity was observed. So they performed data transformation which optimises linear correlation between the data. Then correlation analysis was done between stream count and each feature to determine whether the feature has an effect on the prediction of stream count or not.

Later, a hypothesis test for each correlation or variable was done and a linear regression model was built to check whether a combination of independent variables contribute significantly to the stream count. They then constructed the model summary which is as shown below.

Model	R	R^2	Adjusted R^2	Std. Error Estimate
9	,458[a]	,210	,202	1,67013

Fig. 6. Model Summary

Then the regression coefficients for each feature was calculated using which they drew many inferences. The increase or decrease in stream counts for increase in features was obtained. The results of the correlations showed that there were significant relationships and that the directions of the relationships seemed to fit their hypothesis.

Some of the cons of this paper is that there are some reasons that likely limit the explanatory power of their model. For example all genres were included for measuring popularity and it is possible that different genres do not share the same attributes resulting in noise in the hit prediction model. The model also showed R2 value to be 20.2% which is a little weak. Hence the model is not as effective.

J. Paper X

This paper addresses " deep content based music recommendation". Music can be recommended based on available metadata: information such as the artist, album and year of release is usually known. There is a large semantic gap between the characteristics of a song that affect user preference, and the corresponding audio signal. Extracting high-level properties such as genre, mood, instrumentation, and lyrical themes from audio signals requires powerful models that are capable of capturing the complex hierarchical structure of music. The Million Song Data-set (MSD) is a collection of metadata and precomputed audio features for one million contemporary songs. The authors of the MSD provide precomputed features instead of raw audio. The Taste Profile Subset contains play counts per song and per user, which is a form of implicit feedback. This method used the weighted matrix factorization (WMF) algorithm, proposed by Hu et al., to learn latent factor representations of all users and items in the Taste Profile Subset. Let rui be the play count for user u and song i. For each user-item pair, we define a preference variable pui and a confidence variable cui (I(x) is the indicator function, alpha is a hyper-parameter):

$$\begin{aligned} p_{ui} &= I(r_{ui} > 0), \\ c_{ui} &= 1 + \alpha \log(1 + \epsilon^{-1} r_{ui}). \end{aligned}$$

Fig. 7. Preference and confidence

If the song has never been played, the confidence variable will have a low value, because this is the least informative case. The WMF objective function is given by:

$$\min_{x_\star, y_\star} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i)^2 + \lambda \left(\sum_u ||x_u||^2 + \sum_i ||y_i||^2 \right)$$

Fig. 8. Weighted Factorization Matrix

where lambda is a regularization parameter, xu is the latent factor vector for user u, and yi is the latent factor vector for song i. For each song, similar songs were identified by measuring the cosine similarity between the predicted usage patterns. Then compare the usage patterns predicted using the latent factors obtained with WMF (50 dimensions), with those using latent factors predicted with a convolutional neural network. When the predicted latent factors are used, the matches are mostly different, but the results are quite reasonable in the sense that the matched songs are likely to appeal to the same audience. The paper concludes that predicting latent factors from music audio is a viable method for recommending new and unpopular music, which

solves the issue of slow start in collaborative filtering. It showed that recent advances in deep learning translate very well to the music recommendation setting in combination with this approach, with deep convolutional neural networks significantly outperforming a more traditional approach using bag-of-words representations of audio signals.

III. Paper XI

This research paper mainly focuses on the three main functions which Spotify uses to recommend music to a user namely related artists, discover and browse functions. The research is based on analysis of a single artist from the population as a sample and making respective conclusions for each function performed. The intent behind this research was to explore the idea that "music is connected to other music" is the core of Spotify's ordering of music through recommendations in related artists and discovery. The results obtained off each function is as follows:

- The related artist's function was analyzed by considering a specific artist and concluded that these recommendations were based on various factors like race, gender, and nationality.
- The discovery to function as a result of the user's national position and the IP. It was also time-sensitive and picked up on the recent habits of the individual listener.
- The browse function contained a banner with announcements on top then content with an overview of popular content based on differed by countries and serve multiple audiences at the same time.

This research paper dealt with analyzing an individual artist which might not have been applicable to other artists and might give false conclusions. There haven't been any models used to give a clear idea of the result that can be obtained. A lot of trial and error has gone into the analysis in this paper.

IV. Paper XII

In this paper, a next-song recommendation system for runners has been proposed. The system makes personalized recommendations to increase runners' motivation and performance. The intent behind this was to keep runners motivated while running with music.

The DJ-Running is a research project that monitors the runners' emotional and physiological activity during the training sessions, to automatically recognize their feelings and to select, in real-time, the most suitable music to improve their motivation and performance. It predicts the next song to be played considering the user's location and emotions.

The next song recommendation systems used collaborative filtering and hybrid approaches. The friendship between social network users is included in the models of these collaborative approaches. The factors that were usually involved were the songs' order and popularity, the most listened to artists, and the users' response to those recommended songs analyzing the user's dislikes. The users' profiles were determined by utilizing the users' past interactions or processing the messages published by the users on social media.

The DJ-Running technological infrastructure allows a runner to configure his/her profile in order to listen to personalized music during the training sessions. It also interacts with the Spotify services to access the musical preferences of the user, and with different geographic systems that offer relevant information related to the runner's current location.

This system takes all the information available and tries to determine the next song to play in a personalized way. This model also includes the songs that are skipped by the runner as part of the musical preference in personalizing the next track for the runner. This system also determined the next track from the geographic and environmental data related to the runner location and aimed to make context-aware recommendations to the user.

When there is no data about a new user, it isn't possible to make effective recommendations. There has been low accuracy and efficiency with the proposed model which could be looked to improve with time and future versions of it.

V. Problem Statement

The possibility of a hit song prediction algorithm is both academically interesting and industry-motivated. Exploratory data analysis and predictive statistics of what makes a song popular are what we intend to explore. To do so we preprocess the data set and make inferences from the same.

VI. Inferences

There are many things we can infer from the dataset chosen after some pre-processing. Some of them are listed below:

- As expected popularity is highly correlated with the year released. This makes sense as the Spotify algorithm which makes this decision generates it's "popularity" metric by not just how many streams a song receives, but also how recent those streams are.
- Energy also seems to influence a song's popularity. Many popular songs are energetic, though not necessarily dance songs. Because the correlation here is not too high, low energy songs do have some potential to be more popular.
- Acousticness seems to be uncorrelated with popularity. Most popular songs today have either electronic or electric instruments in them. It is very rare that a piece of music played by a chamber orchestra or purely acoustic band becomes immensely popular (though, again, not impossible).
- The popularity is also related to duration of the song. Shorter songs have relatively more chance of becoming popular.
- Key of a song and it's popularity are negligibly correlated. This is peculiar because musicians and composers consider the key of a song to be an important part of the composition process.

Thus, from this data, it would be better for an artist to create a high energy song with either electric instruments or electronic songs to have the best chance at generating the most popularity.

VII. Our Work

Our work for this cycle included:

- Data collection. This data was collected from Kaggle. It includes 19 columns and 11845 rows.
- Data cleaning. We cleaned the data before proceeding. We checked for null, duplicate and inconsistent values.
- Manual feature engineering. Some features require to be transformed and modified. Attributes such as duration of the song was changed to minutes from milli-seconds. Date attribute was modified to reflect only the year.
- Dataset Inspection. This is the phase where we inspect the data, and visualise the data by plotting various graphs between the different attributes of the dataset and drawing conclusions from the same. This is the part where we conduct the EDA.

References

[1] Kehan Luo. Machine Learning Approach for Genre Prediction on Spotify Top Ranking Songs. A Master's Paper for the M.S. in I.S degree. April, 2018. 37 pages. Advisor: David Gotz

[2] Dance Hit Song Prediction by Dorien Herremansa, David Martensb & Kenneth Sörensen
DOI: 10.1080/09298215.2014.881888

[3] A SEMANTIC ANALYSIS ON SPOTIFY TOP SONGS FOR TEENS by Siwi Isnuhoni1 , Chandraswari Swastya Respati2 1State University, Jl. Colombo No. 1, Karang Malang, Caturtunggal, Depok, Daerah Istimewa Yogyakarta 55281, Indonesia https://doi.org/10.31002/jrlt.v2i2.521

[4] Music Recommendation System Spotify - Collaborative Filtering MUS-17 Mithun Madathil https://hpac.cs.umu.se/teaching/sem-mus-17/Reports/Madathil.pdf

[5] WHAT MAKES FOR A HIT POP SONG? WHAT MAKES FOR A POP SONG? by NICHOLAS BORG AND GEORGE HOKKANEN

[6] Cluster Analysis of Musical Attributes for Top Trending Songs Zayd Al-Beitawi NASA JPL
Mohammad Salehan Cal Poly Pomona
Sonya Zhang Cal Poly Pomona

[7] Song Hit Prediction: Predicting Billboard Hits Using Spotify Data Kai Middlebrook, Kian Sheik https://arxiv.org/pdf/1908.08609

[8] Revisiting the problem of audio-based hit song prediction using convolutional neural networks 10.1109/ICASSP.2017.7952230

[9] Deep content-based music recommendation Aäron van den Oord (UGent) , Sander Dieleman (UGent) and Benjamin Schrauwen (UGent) http://hdl.handle.net/1854/LU-4324554

[10] PREDICTION OF PRODUCT SUCCESS: EXPLAINING SONG POPULARITY BY AUDIO FEATURES FROM SPOTIFY DATA Nijkamp, Rutger (2018) Prediction of product success: explaining song popularity by audio features from Spotify data. http://purl.utwente.nl/essays/75422

[11] Organizing music, organizing gender: algorithmic culture and Spotify recommendations https://doi.org/10.1080/15405702.2020.1715980

[12] DJ-Running: An Emotion-based System for Recommending Spotify Songs to Runners https://pdfs.semanticscholar.org/8ed3/8ed685f1baafc93b911df8a6c0db600107d8.pdf
[13] https://www.statista.com/statistics/272305/global-revenue-of-the-music-industry/