CBS COPENHAGEN BUSINESS SCHOOL
HANDELSHØJSKOLEN

Exploring the Dynamics of the Music Industry based on Genius and Spotify Data

Pascal Essig (133348), Luca Mamoli (139037), Jan-Niklas Wolters (133816)

Data Science for Business Applications (CDSCV1003E)

29 pages with ⊘ 1,861 characters per page

Examiners: Daniel S. Hain and Roman Jurowetzki

Copenhagen, 04.01.2021

Find all documents relevant to this paper here.

Abstract

This project aims at examining the music industry to provide actionable insights for involved artists. It is becoming increasingly difficult for them to be discovered by listeners and the costs of being promoted are high. To address this problem, this analysis investigates what makes a song popular, how the industry is organized, and which themes predominantly occur in the lyrics. The analysis' underlying data is fetched from the APIs of Spotify and Genius and thereafter analyzed with methods from the areas of machine learning, network analysis, and natural language processing. Based on the analysis, it is proposed that (I) a track's danceability, instrumentalness, and loudness positively influence its popularity, (II) artist's collaboration behaviour differs across genres, and (III) that there are considerable differences between genres with regards to what they are singing about. By providing those insights, this study helps artists to make a more informed decision about their song-production and collaboration strategy.

Keywords: Network Analysis, Machine Learning, NLP, Text Analytics, Data Science, Music Industry, Lyrics

Contents

1 Introduction

In the last two decades, technological progress led to enormous transformations of the music industry that changed their revenue models. From the 1970s till 1990s, the music business was based on record music sales. This model radically changed in 1999 when Napster was launched. The illegal peer-to-peer file-sharing platform changed the industry forever, spurring the switch towards digital music and awakening the industry of the inevitability of digitalization and the need for novel revenue models. Another milestone came in 2003 when Apple launched the iTunes Music Store, the first legal market for online music. In 2008, more than four billion songs had been sold on iTunes, accounting for more than 70% of global digital music sales [1]. This figure has grown to more than 25 billion in 2018 [2]. Nonetheless, sales started declining from 2013 when a new model emerged: streaming services.

Since 2013 streaming services have shown constant revenue growth, increasingly gaining market shares. In 2019 they represented the largest revenue share in the music industry, accounting for 56% of the recorded music market [3]. By 2030 the streaming market is estimated to reach a value of 75 billion dollar and 1.22 billion paying subscribers [3]. These growing figures draw a promising future for streaming services.

The Swedish company Spotify registered a 36% share of paying subscribers in 2019, making it the leading streaming platform [3]. With services such as Apple Music and Amazon Music the competition has grown in the last years. Nonetheless, Spotify's leadership is predicted to be maintained which led to the decision to base the following project on Spotify data [3].

Streaming platforms gather and supply an unprecedented amount of data from listeners that can be exploited to get a better understanding of their taste. The music industry is not new to employing data-driven solutions to understand listeners better. An early example was Polyphonic HMI, a company that developed an algorithm capable of predicting a song's success, such as for Norah Jones' debut album "Come Away with Me" and OutKast's single "Hey Ya!", before they topped the charts and become global hits [4]. Such predictions are economically crucial since 'hit songs are worth a fortune' [4].

Although streaming services represent a promising opportunity, the unprecedented number of newly available tracks also poses challenges. It is becoming increasingly difficult for new artists to be discovered by listeners and the costs of promoting and launching new artists are high. Without the proper

tools, talented artists could remain undiscovered. Data-driven tools can help overcome such challenges by providing insights on which listeners enjoy a specific artist in order to target them [5].

2 Research Objective

The objective of the project is to leverage opportunities for the increasingly data-driven music industry and to develop use cases that can be employed to illuminate the industry from different perspectives. Specifically, the following three questions shall be answered.

Question 1: What makes a popular song and how can it be predicted?

As stated above, hit songs can be highly lucrative. Creating a hit, however, is a notoriously difficult task and requires a fine sense for the current taste of the listeners. In this paper, this challenge is tackled by investigating the effect of different audio features on a track's popularity in order to determine what it is that people like to listen to at the moment.

Question 2: How is the music industry organized and who are its central artists?

Many current songs emerge from collaborations. Fans can thereby find out about artists previously unknown to them, who in turn can expand their own reach. Therefore, the artist network is examined in order to identify the most sought-after artists and to gain further insights into how collaborations are organized in the music industry.

Question 3: What are popular artists singing about and how does that differ by genre?

In addition to a song's audio features and its performers, lyrics are another key feature through which the song's message is communicated to listeners. An examination of current lyrics helps to understand what topics are important in the music industry and in the respective genres.
The remaining report is dedicated to answering these questions. In the following, the applied framework and the steps taken to generate insights will be presented.

3 Conceptual Framework

The Cross-Industry Standard Process for Data Mining (CRISP-DM) model is suitable framework for this project [6]. In the following it will be explained and mapped to the project.

3.1 CRISP Model

The CRISP-DM has been around for more than two decades and was originally developed for data mining. However, due to its generalizability it is also suitable for other analytical processes and developed into the most widely-used model [7]. The process consists of six major steps which are applied in an iterative manner; Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation and Deployment (see Figure 1).

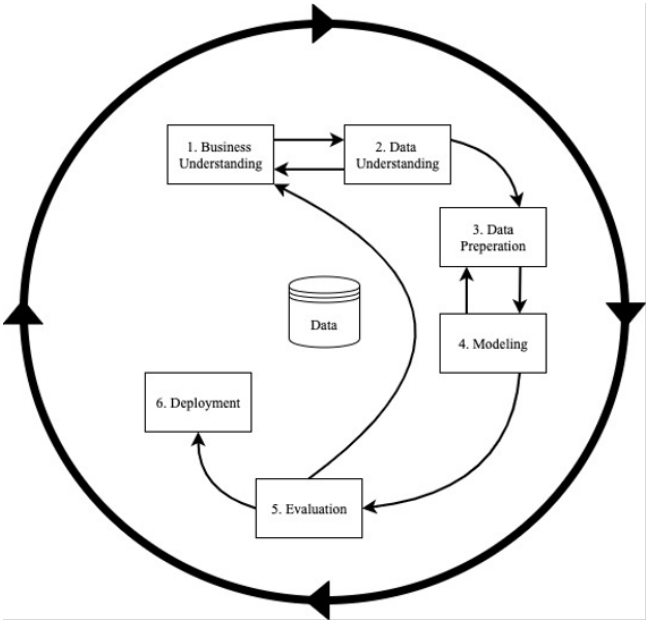

Figure 1: Phases of the CRISP-DM Reference Model (own illustration based on [6, p. 14])

3.2 Project Outline

As a foundation for analysis, an understanding of the business is crucial. By taking a business perspective and assessing the situation, suitable objects can be determined to solve a problem [7; 6]. This part is already covered in section 1 and 2. In the data understanding, data is collected, described (4) and prepared for use (5). After that, the data is modeled with network, machine learning and text analytics techniques (6). All the taken steps can be observed in the data flow diagram (see Figure 2). These techniques are then evaluated, discussed and future work is outlined (7). Since the deployment phase is not applicable to the case at hand, it is left out in this paper. Finally, a conclusion is drawn (8).

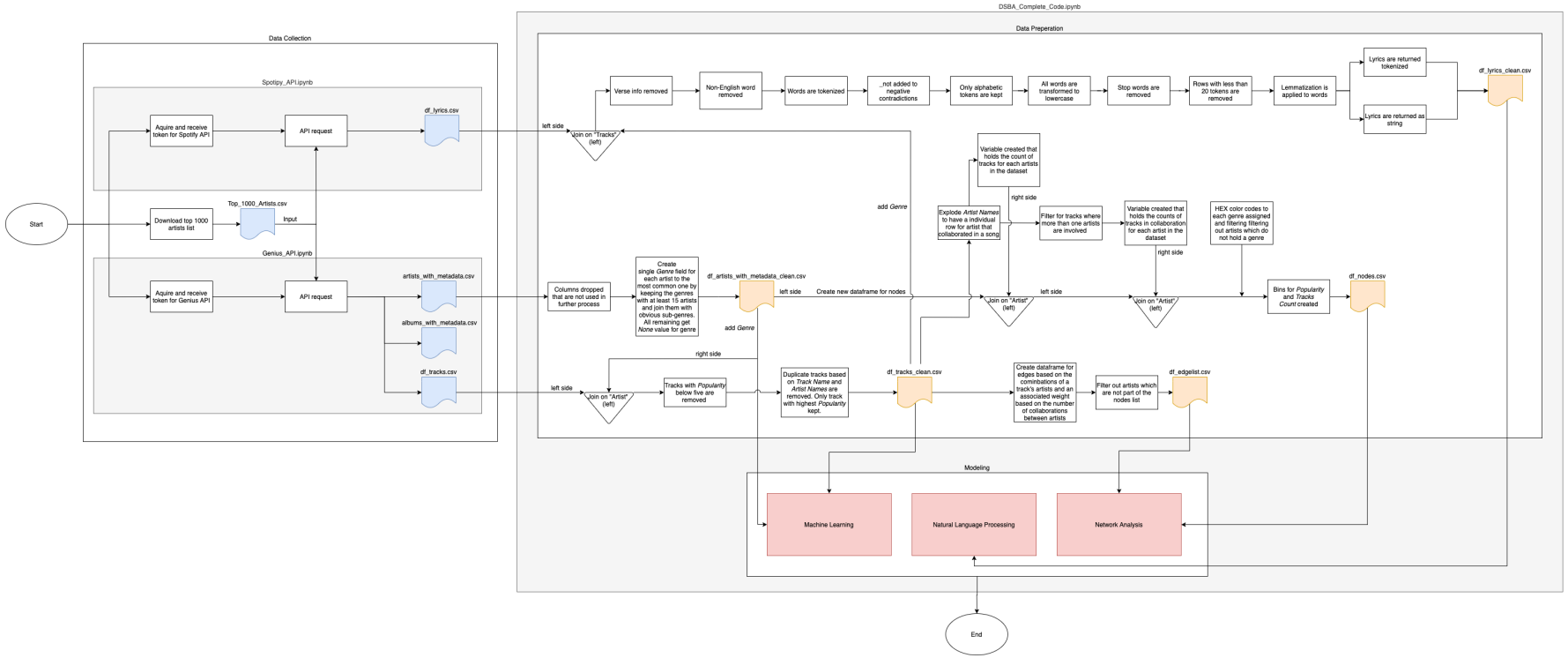

Figure 2: Data flow diagram

4 Data Understanding

With a target in place, suitable data can be collected. This phase further includes steps to describe, explore and verify the quality of the pulled data [6].

4.1 Data Collection

The data used in this paper is fetched with the application programming interface (API) from Spotify and Genius. APIs allow the communication between different applications. In this case, data is requested from a web-server which returns a JSON-file in response.

The choice of the API data that is consumed is based on (I) the companies' relevance in the industry as described in section 1 and (II) the APIs' quality and ease of use. There are more than 60 million tracks available on Spotify, which makes it a comprehensive data source [8]. Spotify data is available through a web API called *Spotipy*. Spotipy enables users to fetch various metadata from the Spotify data catalogue including information on artists, albums, and tracks [9].

Genius is the self-proclaimed "world's biggest collection of song lyrics and musical knowledge" featuring more than 25 million songs [10]. For a better user experience in Spotify, a selection of its songs hold the lyrics from Genius. However, Genius' lyrics data cannot be retrieved via the Spotify API. Instead, it provides its own propriety API to fetch lyrics which is called *lyricsgenius* and is used as an additional data source for this analysis.

Using the Spotify API requires a multi-step approach, as each API endpoint only provides a specific set of data. In addition to this, certain data fetches, such as getting a song's metadata, requires information (e.g. track ID) that only becomes available after being fetched itself. Hence, the data is collected sequentially as outlined below.

In order to get a sample of the most popular artists on Spotify, a list was identified in which the 1,000 most popular artists are available [11]. The artists' names from this list are fed to Spotipy's search-endpoint, which allows identifying Spotify's internally stored artist IDs and additional metadata. In the next step, the artist IDs are used to get all albums of each artist. The albums are not an essential part of the analysis but build a necessary intermediate step as the artists' tracks can only be identified via their albums. Using the obtained album IDs, a set of all track IDs that belong to the respective albums is created which constitute the entirety of tracks used in this paper. Lastly, the track IDs are

used to get additional metadata on the tracks as well as their audio features.

For the lyrics, the list of the artists also provides the basis for the API call. With the artists as input the API returns the list of songs stored. To limit the amount of data, only the 25 most popular songs of each artist are fetched. Based on this list, the actual request is made to retrieve the lyrics of the songs.

4.2 Data Description

The data collection results in three data frames - artists data, track data, and lyrics data - based on which the analysis is conducted. To facilitate the understanding of the analysis that follows, these data sets and their respective features are explained. A detailed overview of all the data sets including their feature descriptions can be found in the appendix (10).

Artists Data The artists data encompasses all relevant information regarding bands and singers from Spotify (see Table 10). Besides the artists' name, their follower count, popularity, and their assigned genres are provided. No data point in the table is missing and there are no other major irregularities. The only preprocessing step required for this dataset is the determination of one definitive genre for each artist. The dataset comprises 614 different genres, with each artist having between zero to twelve genres associated to them. Besides the most popular genres - such as pop (278 times), dance pop (210), or rap (124) there are lots of niche genres - such as "early music", "bedroom soul", or "el paso indie" - that appear only once in the data. Since no generalizable information can be derived from those niche genres, in the data preprocessing the list of genres is narrowed down.

Tracks Data The tracks data is the central dataframe of this analysis (see Table 10). It contains 445,578 tracks which constitute all tracks from all albums of all 1,000 artists. Besides general meta information about the track such as its artist(s), its album, its length, its release date, and whether the track contains explicit content, twelve different audio features are available. The audio features contain elements such as acousticness, danceability, or valence (positiveness) of a song, providing a notion of what a song is like. Most of them are in a range between zero and one which indicates the confidence that a track fulfills the respective feature. The audio features are calculated by Spotify and

no further information on how those measures are estimated is given. Nevertheless, they are a useful way to get a deeper understanding of the tracks. The audio features are missing for 56 tracks.
One important variable in the analysis is popularity which ranges between 0 and 100. It is largely, based on the amount and recency of plays of the track. Accordingly, a song that is currently played a lot is considered to be more popular than a song that used to be played a lot in the past. The popularity mentioned before in the artists data is mathematically derived from the track popularity. However, the exact algorithms for determining the popularity scores are not made public by Spotify.
For the track names, a potential challenge arises as several names appear more than once, indicating potential duplicates in the data (Table 11). Out of 211,905 different track names, 69,131 are not unique and 181 of those tracks even appear more than 50 times. While it is conceivable for some track titles such as "Intro" that they are from different tracks and artists, it unlikely for titles such as "Partita No.1 in B flat, BWV 825: 1. Praeludium". Therefore, handling these duplicates is part of the preprocessing.

Lyrics Data With up to 25 lyrics per artist available, the lyrics data consists of 24,260 rows (see Table 13). Each row contains the title of a track, its artist and the corresponding lyrics. The lyrics themselves are quite unstructured and contain tags from Genius, such as "Intro" or "Chorus". Hence, this data also requires substantial preprocessing.

5 Data Preparation

The phase of data preparation includes all necessary steps to bring the initial raw data into the proper format for the analysis. This includes selecting relevant data from the data available, cleaning it to achieve the required data quality and creating new attributes, if required [6].

5.1 Genre Determination

In the artists data multiple genres are assigned to each artist. As several genres per artist make the analysis more complex, the genre count per artist should be reduced to one. In order to achieve that, the occurrence of each genre is counted and each artist gets assigned the first genre of the most common genre that the artists is part of. The cutoff value for a genre is set to at least holding 15 artists. Additionally, genres that are expected to have a strong overlap such as “dance pop” and “pop

dance", "modern rock" and "rock", and "country" and "contemporary country" are merged. This results in nine genres. Artists that do not hold any of those genres get assigned a None value.

5.2 Duplicate Removal

As observed in section 4.1, several tracks are potential duplicates. Since duplicates negatively influence the analysis by giving over-proportionate weight to the respective tracks, a duplicate removal strategy is applied. The goal of this strategy is to remove as many duplicates as possible without risking to lose too many relevant tracks that occur only once. Looking at track titles occurring multiple times, it can be seen that many of them are very unpopular. Therefore, all tracks with a popularity of five or less are removed. This reduces the number of redundant tracks titles by over 20,000. The next step is undertaken for artists that have several songs by the same name. Whenever this is the case, only the most popular track is kept and all others are deemed to be duplicates and get removed. This method does not guarantee that all removed songs are indeed duplicates (e.g. an artist can have two different albums with unique tracks called "Intro"), however, the number of removed duplicates is considered to outweigh the number of mistakenly removed songs. After the procedure is applied, the duplicate count is reduced by 58,334, to a total of only 10,797 titles appearing more than once.

5.3 Creation of Nodes and Edgelist

The building elements of a *network* (also known as *graph*) are *nodes* (or *vertices*) and connected through *edges* (or *links*). The data needs to be provided in relational form.

First, an *edgelist* with at least a source and target column for each row is required. If the graph is *directed*, the source and target node are assigned deliberately to indicate the direction, for an *indirect* graph they are interchangeable. Further columns can provide information regarding the weight of each connection, such as the number of interactions.

Second, a *nodelist* can be provided that holds further information and enables network grouping based on these characteristics. Creating a network in graphical form instead of tabular form can make it easier to identify inter-dependencies within the network, but can also get messy in bigger networks [12].

For the case at hand, the tracks data and artists data provide the input for the edgelist and nodelist.

For the creation of the edgelist, all tracks are considered to which more than one artist is attributed. All pairings between artists are identified and their frequency counted. The frequency, acts as the weight of the edgelist. The resulting edgelist contains around 38,000 edges, which also includes collaborations where one artist is not part of the artists data. To reduce clutter in the network later on, only those collaborations are kept where both artists are in the artists data and have a genre assigned. This reduces the number of edges to 4,387 which is suitable for the analysis.

The "Popularity" variable of the artists data ranges from 57 to 100, with approximately a normal distribution. As feature for the nodelist for later grouping of the network the variable is too granular, since it is continuous (see Figure 10). Therefore, the artists are split into three equally sized popularity bins, containing less popular (57-72), popular (73-85) and very popular (85-100) artists.

5.4 Lyrics Pre-Processing

The pre-processing of textual data is commonly referred to as text normalization and is a process to convert textual data into a standardized form to improve the results of later modelling [13]. The following steps are taken:

To enable a consistent analysis, the language of each text is identified and all non-English texts are removed. This is done using a probabilistic language detection function that operates based on Google's language-detection tool. Of the 24,260 tracks in the data, 19,145 are classified as English and the remaining ones are removed.

Next, a series of preprocessing steps are applied, partly with the help of the *nltk* package. Since sections like "[intro]" and "[chorus]" do not add value, they are removed from the lyrics.

In the following step, the strings are broken up into the words, which is known as tokenization. Elements such as "\n", the denotation for a new line, are automatically removed as well by this applied method. To keep negative contradictions, commonly written as "n't", they are replaced with "_not". Afterwards, all symbols except of alphabetic and an underscores are removed and subsequently all tokens are converted to lowercase.

Next, stop words are removed. Those are commonly used words - such as "the", "to" or "in" - that add little or no content and can take up lots of storage and therefore increase the processing time drastically [14]. The nltk stop words list is used as foundation and through an iterative process it is

extended by music-related stop word, such as “huh”, “ooh” and “lala”.

For the normalization of words, lemmatization was used. This normalization method unifies words holding the same root such as “sang”, “sung” and “sings” that are all lemmatized to “sing”. Thereby, lemmatization consolidates words belonging together by reducing unnecessary variance [13]. Due to different packages used later on that require either tokens or a string as input, the preprocessing function provides both options as output. After the lyrics preprocessing, lyrics containing less than 20 words are removed due to the limited information they provide. In the end, 16,154 track lyrics remain for the analysis.

6 Modeling

In the modeling phase a number of models are selected, set up and applied to the data. Along the way the results are assessed and the parameters are optimized to achieve optimal results [6].

6.1 Supervised Machine Learning

Supervised learning is a subcategory of machine learning with the task of learning a function that maps an input to an output, based on sample input-output pairs [15]. Supervised learning problems can be simplified into two main categories (I) classification problems and (II) regression problems. Predicting popularity falls under a regression problem.

The final goal of creating estimators is minimizing the model’s prediction errors by refining features, and applying different algorithms. This is a process of repetitive adjustments for step-wise improvement. Due to this paper’s spatial constraints, only the most relevant results are presented.

Supervised machine learning for popularity prediction is performed utilizing *Sklearn* and *Statsmodels*, two of Python’s most utilized machine learning packages. Statsmodel is a Python package to estimate different statistical models and conduct statistical tests and data exploration [16], which can be used to understand the impact of different features.

To maximize the model’s prediction capabilities, the data is cleaned, removing unnecessary variables. Firstly, all the non-quantitative features are discharged. Next, EDA is conducted to understand which independent variables are related to popularity the most. This is done by looking at the correlation.

The main correlation coefficients in the tracks data are shown in Table 1.

After observing correlations, EDA is conducted on the most relevant variables to better understand their influence on popularity. The findings can be summarized as follows: (I) the less acoustic a song is, the more popular it is likely to be. (II) Almost all the songs with popularity higher than 80 have a Loudness value between -10 and 0. (III) On average, the more instrumental songs are, the less they are likely to be popular. The mean instrumentalness of songs with popularity over 40 is in between 0 and 0.1. (IV) Higher values of energy and danceability, have a positive impact on popularity.

Value	**Correlation**
Loudness	0.38
Acousticness	-0.30
Instrumentalness	-0.27
Energy	0.26
Danceability	0.26

Table 1: Top correlation coefficients in the track data

Ultimately, the variables for training the models are created. Regarding the independent variables, two approaches are considered: (I) all the musical features are considered, (II) only the features with correlation over 0.2 are kept. Such a choice is made to test if the features less correlated to popularity have a lower effect on predicting popularity. Finally, the data is divided into a train and test set.

6.1.1 Songs from all years

The first attempts in popularity prediction consider songs from all the years present in the data. Two linear regression models are created. One model is fitted with all the music features, while the second only contains the features most correlated to popularity. The results show that both models can explain around 20% of the dependent variable. Consequently, prediction results have a considerable error value. Notably, none of the models can predict popularity values over 50.

Although other models and variables combinations have been tried, the results could not be improved. A possible explanation for these results is that music features alone are not sufficient to predict popularity. Another possibility comes from how the Spotify API calculates popularity. As explained in Section 4.2, popularity is highly related to the number of plays a song recently had. Therefore, it could be argued that the more recent a song is, the more likely it is to be popular. Such an assumption is

strengthened by analysis conducted on another dataset obtained using Spotify Web API [17], which found a strong correlation between popularity and year. However, the referenced dataset presents an equal distribution of songs over the last century, while the tracks used in this project come from the most streamed artists of 2020. Consequently, the vast majority of songs are from after 2009, as shown in Fig 3. Therefore, a viable option is considering only songs from 2020. Indeed, the number of songs from 2020 is big enough to have a sample of different popularity values to better understand the features characterizing songs that are currently popular.

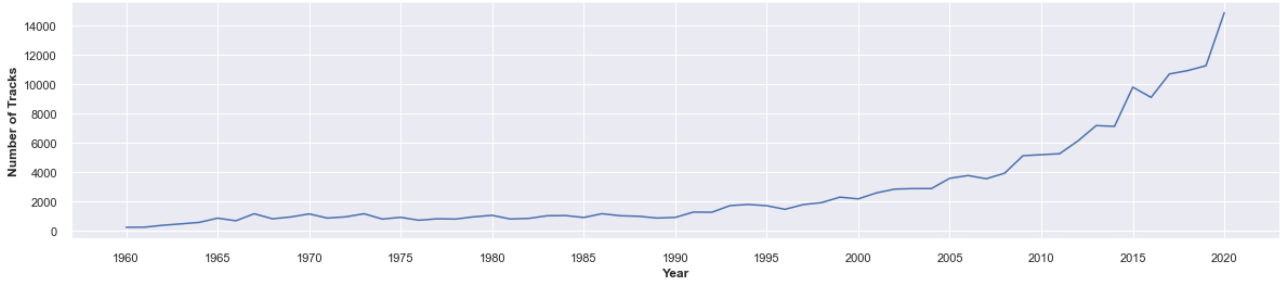

Figure 3: Number of tracks over the years

6.1.2 Songs from 2020

To predict the popularity of songs from 2020, the same procedure adopted in the previous section is followed. All the variables are filtered to contain only tracks from 2020 and, as in the previous section, two linear regressors are fitted. The results are considerably better, as shown in Table 2 , with the model explaining more than 50% of the variance of the dependent variable. Despite the improvements, predictions are still not accurate with predicted popularity values only below 65. Further, it is possible to notice how multicollinearity is present when all features are fed into the model. To limit this issue, two regularized linear regressors are used – Lasso and Ridge regression. Lasso and Ridge regression alter the loss function to penalize large coefficients during training, encouraging simpler models with smaller coefficient values. The difference between Ridge and Lasso regression is that the latter can set the coefficients to zero for those input variables that contribute too little to the prediction task. On the other hand, Ridge regression limits to shrinking coefficients without setting them to zero [18]. The results obtained with both regularized linear regressors are shown in Table 2. As can be seen, their adoption does not reduce errors significantly.

As all the estimators used so far assume a linear relationship between the dependent and independent variables, it is worth verifying that this assumption is true. To do so, the models' outputs are analyzed. If a linear relationship is not present, other models should be used [19]. One way to detect non-linearity is to inspect the following two plots: (I) observed vs predicted values or (II) residuals vs predicted values. If a linear relation is present, points are symmetrically distributed around a diagonal line in the former plot and around a horizontal line in the latter [20]. Firstly, linearity analysis is conducted on the outputs from the statsmodels' predictions. As shown in Fig 4, the points follow the desired behaviour only over predicted popularity values of 40.

Figure 4: Plots for linearity detection

Then, the Ridge and Lasso regression output is analyzed, using the *Yellowbrick* package, which allows to visualize the distribution of residuals and how normalized the distribution of errors is. Similarly to what was observed in the Linear Regression estimators, also Lasso and Ridge do not follow a linear relationship for values of popularity less than 40.

As the findings seem to suggest that a non-linear model may be more appropriate, following Scikit-Learn documentation [21] an ensemble method is employed, namely Random Forest Classifier. Parameter hyper tuning is applied to the model to improve the prediction capabilities. As shown in Table 2,

the Random Forest Classifier outperforms the linear regression models.

	Linear Regression	**Lasso Regression**	**Ridge Regression**	**Random Forest**
MSE	190.97	192.51	193.68	149.87
RMSE	13.82	13.87	13.92	12.24
$\mathbf{R}^2$	0.56	0.54	0.53	0.59

Table 2: Prediction results for regressors considering all music features

To gain maximum insight from the random forest classifier, the results are analyzed with the *ELI5* package, which helps to explain the results from machine learning classifiers. The package helps to examine which variables affect popularity the most. Such insight is considered valuable as it allows to identify the music features that could allow artists to maximize their chances of producing a popular song. Table 3 shows how each variable affects the score. For this prediction, it seems the most important factors are loudness, instrumentalness and danceability.

Feature	**Contribution**	**Value**	**Weight**
Loudness	+14.481	-2.211	0.5742 ± 0.0168
Instrumentalness	+4.614	0.000	0.0824 ± 0.0080
Danceability	+1.921	0.6719	0.0905 ± 0.0107
Energy	-1.603	0.786	0.0900 ± 0.0106
Acousticness	-2.410	0.343	0.0947 ± 0.0100

Table 3: Features contributions and weights

6.2 Network Analysis

Based on the interconnectedness of many real-world phenomena, there are different kinds of relationships between elements. Networks can be found among others in chemistry, biology, computer science and social science, with a strong rise in social network analysis due to platforms such as Instagram or Twitter [12]. For the case at hand, the network consists of music collaborations between artists.

The analysis is divided into four sections. First, the data is investigated based on conventional network measures. Second, the network is clustered with the Louvain algorithm and subsequently by a genre-based grouping approach to achieve more human interpretable results. Lastly, a hypothetical case is presented that shows how an artist could strike a collaboration with a sought-after artist.

The network is created using the *NetworkX* package. It is based on the edgelist and enriched with the nodes data (see Section 5.3). The network only holds nodes that have a genre assigned to them. Additionally, 174 of the artists from the artists data are not included into the network since they do not have a single collaboration. This decision is made to reduce the noise in the network and to put the focus on those artists that are actually collaborating with other artists.

6.2.1 Global Network Measures

Fist, various global network measures are calculated. *Density* is amount of edges in the network divided by the amount of edges a network would have if all nodes were connected [12]. With 0.02 the value of the network is quite low, but reasonable, since only a few artists of the overall network collaborate with each other.
Transitivity, also known as *clustering coefficient*, is the share of closed triplets and shows how a network is clustered locally [12]. The networks shows a transitivity measure of 0.29 and indicates that there are local clusters of artists in place that tend to collaborate more strongly.

To identify artists that collaborate most with each other and therefore have a high relevance in the music industry, the sum of weights per artist is calculated. Furthermore, different centrality measures are calculated. To calculate the degree centrality, the number of nodes to which a given node is connected is divided by the total number of nodes in the network. [12; 22]. The second centrality measure applicable to the data is the *eigenvector centrality*. The eigenvector centrality of a node is calculated by weighing the centrality of the node's neighbors [12]. The five most relevant artists in the music industry grouped by the different measures are shown in Table 4. They exhibit a strong overlap with Gucci Mane and Lil Wayne ranking in the top five of each measure.

Artist Name	Weight Sum	Artist Name	Degree Centrality	Artist Name	Eigenvector Centrality
Gucci Mane	479	Lil Wayne	0.145367	Gucci Mane	0.335215
Rick Ross	427	Future	0.140575	Rick Ross	0.320372
Lil Wayne	398	Ty Dolla $ign	0.138978	Lil Wayne	0.279134
Future	348	Nicki Minaj	0.137380	Future	0.247427
Young Thug	313	Gucci Mane	0.135783	DJ Khaled	0.241401

Table 4: Top five artists by weight sum, degree centrality and eigenvector centrality

Collaborative artists can also be identified in visual manner, for example through the circos plots (see Figure 5) available in the *nxviz* package. The plot is modified to only show edges with a weight higher than five. The same artists highlighted through thicker lines (which shows the weight the of edge) are the same as identified before by centrality measures, such as Rick Ross, Lil Wayne, DJ Khaled and Nicki Minaj.

6.2.2 Network Clustering

In order to detect different communities within the network, the *Louvain Algorithm*, which identifies clusters based on graph topology, is applied [12]. The clusters are created by assigning nodes to a cluster that the modularity is minimized. Modularity is a measure that determines the density of edges within their communities relative to circumjacent communities of the network [12; 23]. This minimization is repeated by the algorithm until no further improvement can be achieved [12]. This process results in 13 communities.

The network is inspected in a visual manner with several plots (see Figure 11, Figure 12 and Figure 13) from the previously mentioned *nxviz* package, as well as *NetworkX* and *pyvis*. Additionally, various measures are calculated, such as density and transitivity per cluster and similarities to other characteristics examined. The *assortativity coefficient* (which measures the probability that nodes that share similar characteristics are connected) of 0.46 suggests that reasonable clusters are created, since groupings based on other characteristics such as the genre result in lower values [24]. However, the problem that algorithm-based communities are hardly human interpretable remains and makes it difficult to derive valueable information about the music industry from it.

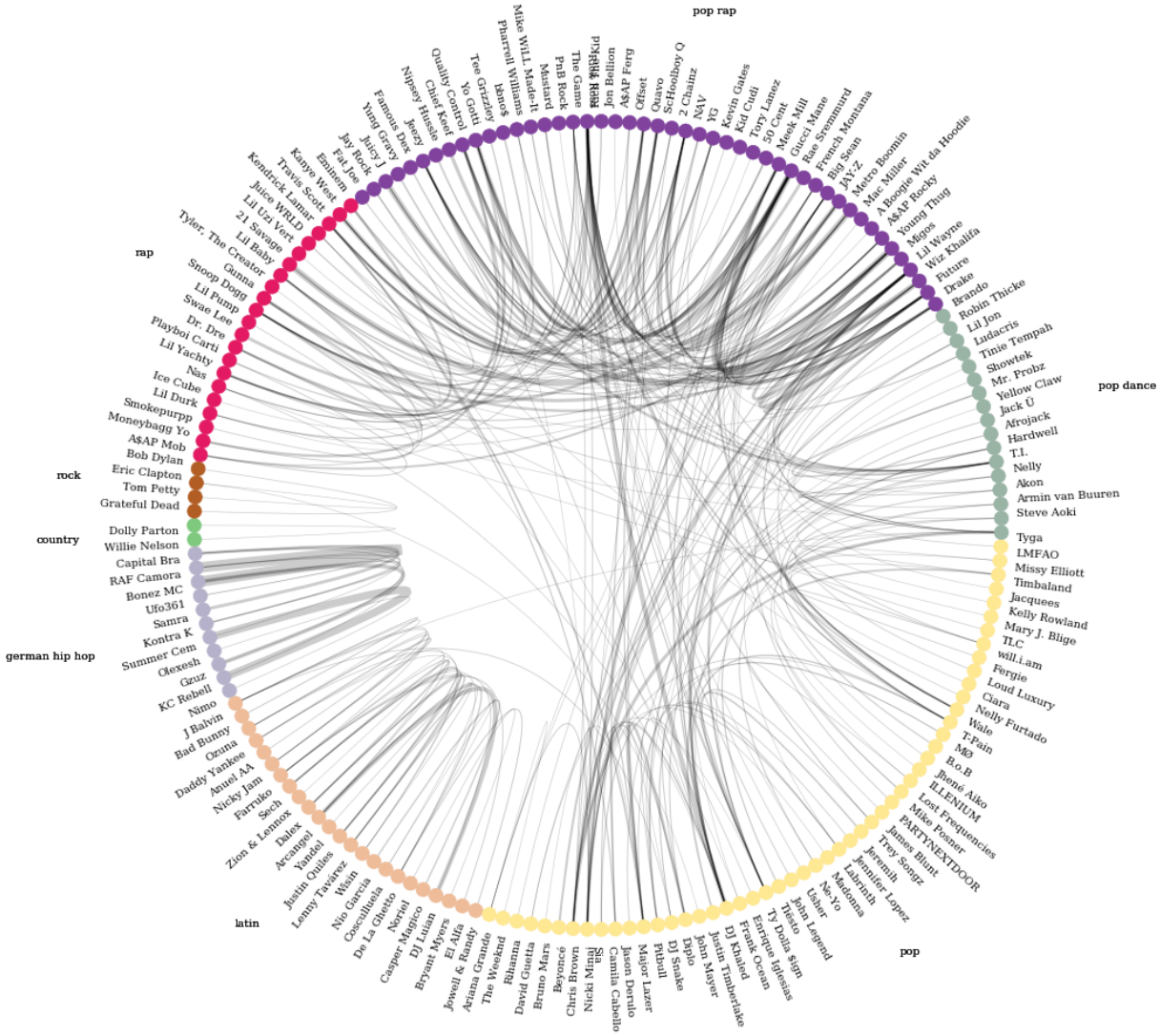

Figure 5: Circos plot with weights more than five and grouped by the genre

6.2.3 Genre-based Network Analysis

Next, the network is grouped based on the artists' genres. As the graph shows (see Figure 6), the degree centrality (depicted by the size of the nodes) of artists reveals that the most relevant artists for collaborations are from the pop rap genre (green) and that within the genre collaborations are very common, as its dense structure indicates. Also, the pop genre (red) holds many of the most relevant artists for music collaborations and is highly connected; not only in itself, but also with pop rap, pop dance (light blue), rap (dark green) and latin (light grey). This can be seen by its spread-out structure. German hip hop (yellow) artists seem to promote strongly each others success, since they almost exclusively work within their genre, which is indicated by the distance to the remaining

network. Also, they collaborate very frequently as the weight (thickness of edges) shows.

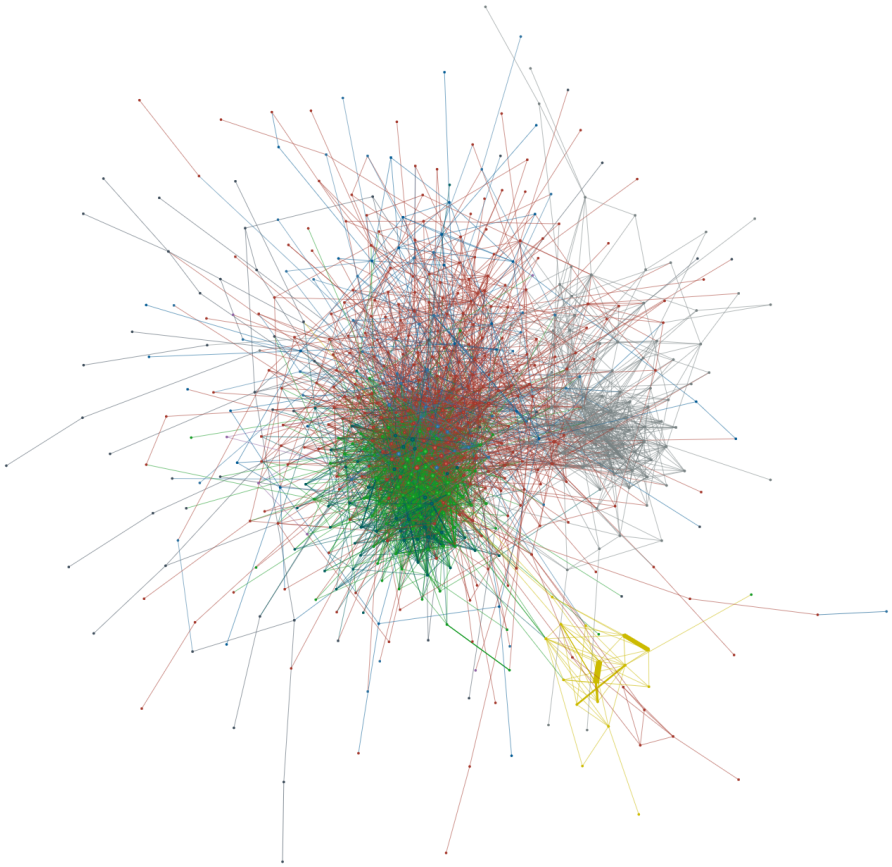

Figure 6: Pyvis network graph grouped by the genre

The circos plot in Figure 7 confirms these findings and additionally shows that in rock, alternative metal and country collaborations between artists do not seem common.
The small amount of collaborations might be due to the fact that coordinating bands to make a feature is more difficult.

The genres' independent density and transitivity (see Table 14) confirms most of the visually-based findings. For example, German hip hop is highest in terms of density and transitivity, followed by pop rap. The pop genre has a much lower density and transitivity, which can be explained by the large numbers of artists in the genre and cross-genre collaborations. However, country artists are much more connected and locally clustered as it seemed in the visual observation.

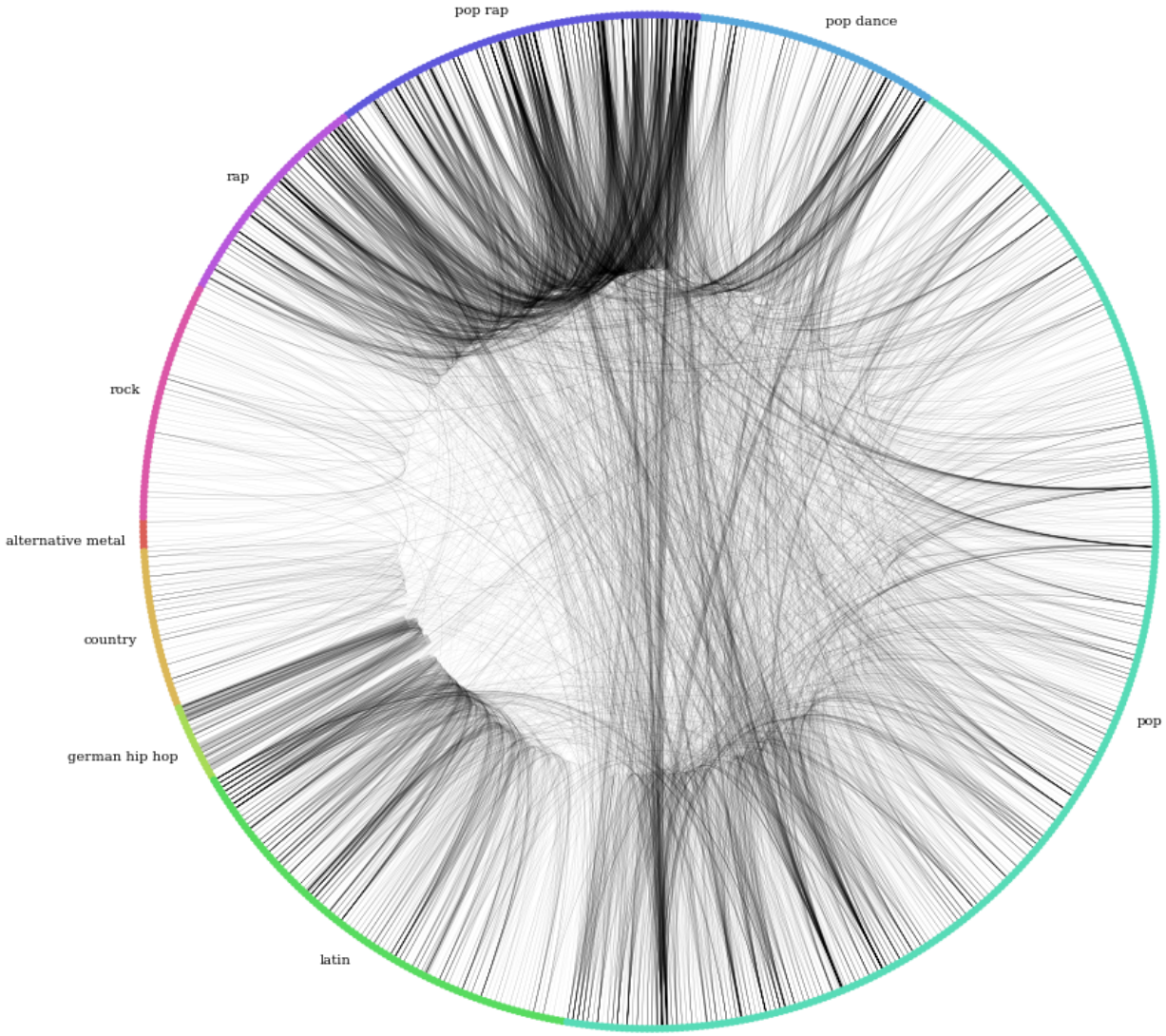

Figure 7: Circos plot grouped by the genre

Lastly, the assortativity coefficient of 0.33 shows that overall artists have a strong tendency to collaborate with artists within their genre [12].

6.2.4 Connecting with Popular Artists

The network grouped by popular artists in form of a circos plot (see Figure 14) indicates that very popular artists have a higher probability to collaborate with other very popular artists. The assortativity coefficient of 0.08 confirms this finding, but the tendency is weaker than visually observed. This

could be explained by the higher number of edges in this area which deceives the perception.

Even though this tendency is in place, it could be worthwhile for newcomers to strive for a collaboration with sought-after artists to expand their listener base and to get recognized, especially in a genre where collaborations are very common.

Taking the German hip hop scene as an example, identifying one of the most popular collaborators based on degree centrality suggests to strive for collaborations with Ufo361, Capital Bra or RAF Camora.

Assuming now in hypothetical case that a newcomer already knows an artists from the genre, who can then possibly establish the connection to the desired artist for the collaboration. Additionally, the assumptions are made that artists that are less popular are more accessible and artists that collaborated more frequently together have a better relation, so both factors can increase the chance for a successful arrangement.

This means for the german hip hop case, if a striving artist would already known Kollegah and wants to reach out to Ufo361 as the desired collaborative artist, Kollegah could introduce the artist to KC Rebell and he in turn to Ufo361. The second-best option is over RAF Camora, but the chance of a positive outcome is less likely since RAF Camora is a lot more famous than KC Rebell.

6.3 Natural Language Processing

In this analysis NLP is used for a better understanding of the lyrical content of popular music. The central focus, hereby, lies on the examination of similarities and differences between the genres. For this purpose, at first there is an examination of general text data followed by the creation of a text classifier and a topic model.

Looking at the song length, which is determined on the basis of the cleaned lyrics without stopwords, clear differences between the genres can be seen (Table 5). Lyrics from the two "longest" genres - pop rap and rap - are on average more than twice as long as country or rock songs. Even the longest country song is barely longer than an

Genre	Mean	Median	Max
Pop Rap	241.2	226.0	1426
Rap	239.9	224.0	5039
Latin	154.9	139	366
Pop	137.7	126.0	871
Pop Dance	136.6	117.0	523
Alternative Metal	127.3	119.0	477
Country	116.5	117.0	241
Rock	100.9	94.0	630

Table 5: Song length by genre

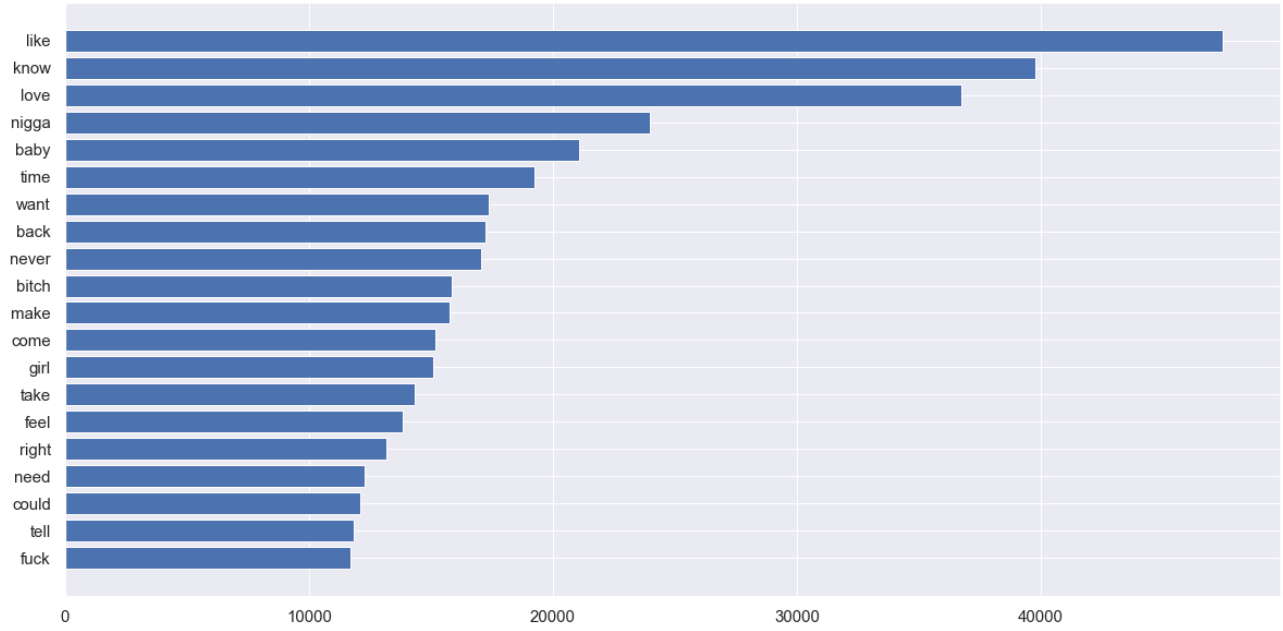

Figure 8: Most frequently occurring words in popular music

average song from the two rap genres.

Differences between the genres can also be observed with regard to the words used. Looking at the words that appear most frequently overall, two key observations are made (Figure 8). The first observation is that many of the words are among those frequently used in everyday life, such as "like" (1st most popular), "know" (2nd), "love" (3rd), and "baby" (5th). However, it can also be observed that words along the lines of "nigga" (4th) and "bitch" (11th) play a prominent role in popular music. Investigating word frequencies by genre uncovers that the origin of those rather profane words lies in the rap and pop rap genres, where also terms such as "shit" and "fuck" are prominently featured. The other genres have strong overlaps regarding their most frequent terms, only differing in the order in which they are ranked. To find out by which words the respective genres are characterized and how they can be distinguished from each other, simple lists of the most common words are consequently not sufficient.

6.3.1 Text Classification

A more precise understanding of the genres and their characteristics can be achieved by training a classifier. This classifier fulfills a double purpose. On the one hand, it serves to see how reliably the genre of a song can be determined computationally on the basis of its lyrics. On the other hand,

however, the classifier is also intended to be used for the extraction of genre-specific features that are interpretable by humans. Since the goal of a classifier is to detect differences between the lyrics used in different genres, it fits such a task very well. The classifier of choice for this analysis is the logistic regression due to its capability to discover links between the outcome of a model and the features that cause it [13]. Since genre prediction constitutes a multi-class classification problem, a multinomial logistic regression is used.

Different model configurations are tested to determine the one that produces the best classifications. In all cases, largely the same procedure is used. First, the lyrics are defined as the independent and the genres as the dependent variable. Second, the split into train and test data is performed. Third, a feature representation of the lyrics is created using a Tfidf-vectorizer. Fourth, the logistic regression model is trained. Fifth, the model is evaluated based on a confusion matrix and performance measures. Lastly, the feature importances are investigated and interpreted with the help of the *eli5* library.

To get a first impression of the model performance, a logistic regression is trained without any further specifications. Based on the results of this analysis, a few observations can be made that inform steps to subsequently improve the model. The precision, i.e. the times a genre is correctly predicted by the model divided by the times it is predicted in total, exceeds 50% in all but two cases. Only latin and alternative metal can't reach this threshold which is likely caused by them being underrepresented in the data. On the flip side, pop, which exhibits the highest frequency in the data, is predicted disproportionately often constituting roughly 67% of guesses while "only" constituting 44% of samples. Additionally, pop rap and rap seem to be frequently mistaken for each other, which is also reflected in the largely identical features by which they are predicted. For that reason, latin and alternative metal are dropped for all subsequent models and pop rap is subsumed under the rap genre.

	Model 1 All Genres default model	**Model 2** Reduced genres default model	**Model 3** Reduced genres grid search	**Model 4** Reduced genres oversampled classes
Accuracy	0.57	0.63	0.61	0.58
F1-Score	0.52	0.59	0.57	0.58
Precision	0.59	0.63	0.63	0.59
Recall	0.57	0.63	0.61	0.58

Table 6: Classification performance of logistic regression models (measures are macro average)

In order to improve the performance of the classifier - and thus the informative value of the most

important features - various model configurations are tested with grid search. Moreover, due to the class imbalance of the genres, a model is trained in which the number of samples of all minority classes was adjusted to the majority class by random oversampling. An overview of the respective model performances can be found in Table 6.

Since model 2 shows the best performance, it is taken as the basis for the feature analysis of the genres, which is conducted using the *eli5* library (Table 7). It can be seen that each genre has its own set of features which are - with the exception of "bitch" appearing in both pop and rap - not overlapping. Country and rap music in particular exhibit features that have an intuitive fit to the genres. They show that country music seems to largely revolve around the consumption of alcohol and life on the road, whereas rap is mainly characterized by its explicit content. Pop music seems to primarily promote an outgoing lifestyle whereby it shows a certain overlap with pop dance. Rock's main features are more ambiguous to interpret and no final judgement can be made about them. The genres can further be distinguished by what they are not. For example, the features that negatively affect country are congruent with the most positive features of rap music. In a similar manner, many of the features negatively influencing pop are strong predictors of country music.

Position	**Country**	**Pop**	**Pop Dance**	**Rap**	**Rock**
1	whiskey	giving	cause	nigga	well
2	beer	body	chick	shit	monty
3	country	club	worry	fuck	child
4	road	bitch	give	bitch	dead
5	town	piece	shaggy	drug	strange

Table 7: Top five most influential features per genre

6.3.2 Topic Modelling

Despite noticeable differences between the genres, the misclassified samples, which remain even after extensive training of the classifier, indicate potential overlap in content between the genres. This also seems logical from an intuitive point of view - in different genres artists can sing about the same topics, making a clear distinction difficult. To identify potential cross-genre topics, topic modeling in the form of Latent Dirichlet Allocation (LDA) is conducted. LDA is a generative probabilistic model that was introduced by Blei et al. [25]. It works by assigning words that frequently occur together in text documents to joint topics which, in turn, can be interpreted.

To prepare the text corpus for modeling, the 0.01th percentile of the most frequently occurring tokens and all tokens occurring less than ten times are removed. In addition, terms that have a large impact but low interpretive value in modelling, such as "really" or "without" are removed.

The number of targeted topics has to be manually specified in the LDA. To identify an optimal number of topics, models with topic counts between 2 to 15 were developed. These models were then evaluated based on the coherence score, i.e. the semantic similarity of the most frequent words in each topic [26].

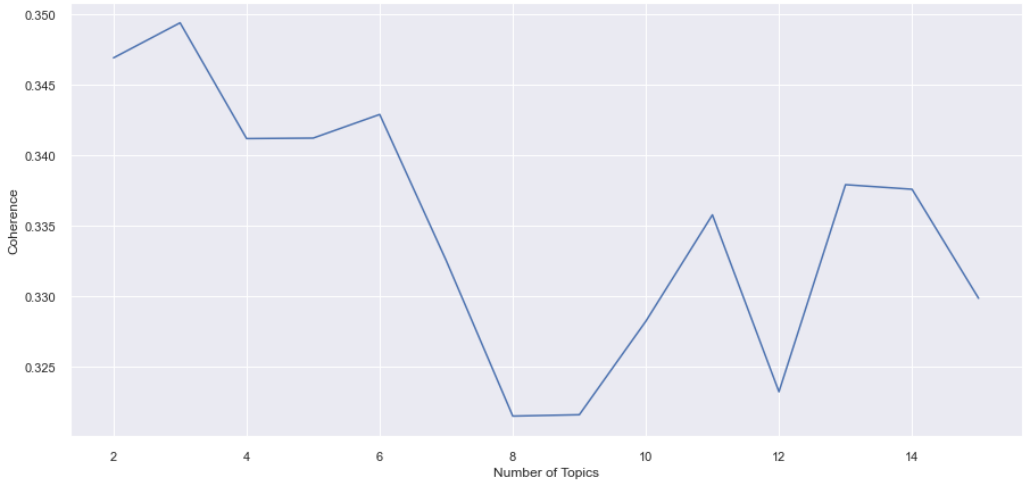

Figure 9: Coherence scores for different numbers of topics

As Figure 9 shows, the most promising candidates for the number of topics based on the coherence scores are three, six, and 13. In order to find the most suitable model among the three, they are all visualised and investigated using the intertopic distance map provided by the *pyLDAvis* library (see Figure 15 for 3 topics, Figure 16 for 6 topics, and Figure 17 for 13 topics). Based on the observations, the model with six topics is found to be the most suitable one, offering more topics to work with than the three topic model and less overlap than the 13 topic model.

For the naming of the topics, human intervention is again required. Based on the most important words of each topic, topic names are determined that are deemed most appropriate according to the authors' subjective interpretation. Accordingly, the six topics into which songwriting in the music industry can be subdivided are Feelings, Gangster, Rebel, Party, Rage, and Relationship, which are described in more detail in Table 8.

Based on the intertopic distance map, it is also possible to make assertions about the relationship of the individual topics to each other. For example, Gangster, Rebel and Rage are located relatively

Topic	Topic Name	Frequency	Terms
1	Feelings	37.8%	home, always, eye, light, hold, feeling, long, find, not_know, tonight
2	Gangster	20.3%	pull, hoe, ride, work, pussy, diamond, rich, throw, club, body
3	Rebel	13.5%	real, dumb, crazy, going, bring, side, game, gang, problem, feeling
4	Party	11.7%	fire, high, party, black, young, rain, dance, move, body, town
5	Rage	9.2%	check, hell, beat, hate, fucking, name, turn, please, made, minute
6	Relationship	7.5%	friend, not_you, talk, walk, mine, wish, not_need, babe, pretty, bang

Table 8: Summary of the LDA model

closely to each other, therefore sharing a certain semantic similarity. The same is true for Feelings and Relationship. The only exception is Party, which has no clear proximity to any of the other topics and can therefore be considered a bit of an outlier.

Finally, the identified topics are put into context with the genres. To achieve this, each song lyric is assigned the topic to which it most closely corresponds, based on the lyric-specific topic distribution. From the songs possessing both a genre and a topic, a cross-table can be created from which the distribution of the genres on the respective topics can be derived (Table 9). It can be seen that in all genres, with the exception of rap, the majority of songs fall into the Feelings category. Rap, in turn, mainly deals with the topics of Gangster and Rebel. Furthermore, country and to a lesser extent rock are quite strongly represented in Party music, whereas pop dance shows a certain closeness to the Gangster area.

	Feelings	Gangster	Party	Rage	Rebel	Relationship
Country	0.63	0.01	0.24	0.03	0.05	0.04
Pop	0.65	0.09	0.07	0.05	0.07	0.06
Pop Dance	0.58	0.12	0.11	0.04	0.08	0.07
Rap	0.17	0.48	0.04	0.07	0.22	0.02
Rock	0.69	0.03	0.15	0.05	0.02	0.06

Table 9: Topic-distribution of genres

7 Evaluation

The evaluation phase contains the three steps of evaluating the results, reviewing the process and determining the next steps [6].

7.1 Contributions

The contributions of this analysis are intended to answer the research questions and put them into a business context. For the sake of clarity, the questions are reiterated here before being addressed.

Question 1: What makes a popular song and how can it be predicted?

Among the different predictors, the Random Forest Classifier identified Loudness, Instrumentalness and Danceability as the most relevant music features in popularity prediction. This insight could be shared and further explored with music experts to understand their effects on a song and how to transform this into actionable insights to consider in the production phase.

Further, a non linear behaviour in songs with lower popularity values was identified. This could imply that, despite the songs having the proper 'musical formula' to be popular, they don't turn out to be a success anyway. A possible reason for this could be that among the songs included into an album, only a few are released as 'singles' and on the radio, becoming more popular as they are generally more listened and referenced to from playlists and movies, for example. Therefore, songs with similar musical features could reach considerably different levels of popularity. A viable strategy to turn this into an advantage could be releasing more songs as 'singles' before releasing the entire album to maximize the reach of each song, avoiding cannibalization of tracks in the same album. Such a strategy is adopted already by different artists [27].

Question 2: How is the music industry organized and who are its central artists?

Through the network analysis heterogeneous collaboration behaviour between genres was identified. While some genres are highly connected and remain within their genre, others seem to be well suited to collaborate also with other genres. Lastly, there are genres where it is more common to produce songs independently.

If an artist wants to succeed in a collaborative genre, reaching out and striving for a track collaboration with one of the most sought-after artists can be highly beneficial. These artists have been identified based on centrality measures.

Succeeding in striking such a collaboration can help to be recognized in the industry, expand the fan base and thus boost the career. However, in the music industry there is the tendency that already popular artists prefer to collaborate with other popular artists. This can make it challenging to put such a collaboration into action.

Question 3: What are popular artists singing about and how does that differ by genre? From the text analysis it has emerged that, in addition to tonal differences between the genres, also clear content-related differences can be observed. Rap music in particular is a clear outsider in this regard since, unlike the other genres, it does not focus on Feelings-related lyrics, but rather deals with Gangster topics.
For artists who feel that they belong to a certain genre and want to start their career there, the analysis can provide orientation by showing them which topics are currently being sung about and which words are predominantly used. This allows artists who want to fit into a genre to adopt the current terminology. For instance, a conformity-minded pop singer might be advised to write Feelings-related lyrics and avoid the Gangster and Rebel-related ones. At the same time, artists who want to add new impulses to a genre can do so by consciously turning away from the current style of songwriting and thereby break new ground. For example, while the reference to Gangster themes would not necessarily promise success in country music, it would definitely be a radical departure from the norm that could catch listeners' attention.

7.2 Limitations and Future Work

While this analysis has shown how far different audio features can influence a track's popularity, it cannot provide any information on how these features are obtained. Spotify only vaguely outlines how they calculate the features and no effort has been undertaken to reverse-engineer them in this paper. Hence, it can only be recommended to make a track "danceable", without being able to state what this term actually implies.

Furthermore, popularity is a multi-faceted problem that involves various variables not considered in this research. Such variables are not directly related to a track's audio features but rather depend on the reach of the artists, their media exposure and labels (e.g. marketing efforts, labels push, social media followers).

This study is further limited by the fact that it can only speak for the present in terms of song popularity. The music industry is fast-moving, and what is popular today may no longer be of anyone's interest in a year's time. Therefore, the generated insights should be considered with respect to the current time and not be extrapolated into the future.

Another weakness of this analysis lies in the determination of genres which forms an essential part of our analysis. By always taking only the most popular genre associated with an artist, many nuances are lost and the genres that are already strongly featured tend to be over-represented. As a result, an imbalanced data set is created, which decreases the analysis' meaningfulness to a certain degree. In future studies about this subject, the application of more elaborate genre-determination procedures that take the co-occurrence of genres into account would be advised for. Also a network analysis that looks at which genres frequently occur together could provide more clarity in this regard.

Finally, even though Spotify provides very rich data, it still only offers a window to the music industry rather than the full picture. This view could be widened by incorporating different data sources into the analysis, such as data from other streaming services or even traditional music outlets.

8 Conclusion

Based on data from Spotify and Genius, this study applied data science methods to provide insights into the music industry. Supervised machine learning methods were used to identify factors that can predict the popularity of a song. Through a network analysis, key players in the music industry were identified and differences between genres were discovered. In addition, NLP was used to analyze the content of song lyrics and to identify central themes of songwriting. Ultimately, the results of the analysis were transferred into proposals that artists and decision makers can incorporate into their decision making process. Therefore, this study contributes to a more comprehensive understanding of the music industry.

9 References

[1] G. Kot, *Ripped: How the Wired Generation Revolutionized Music.* Scribner, 2009.

[2] P. Wikström and R. DeFillippi, *Business Innovation and Disruption in the Music Industry.* Edward Elgar Publishing, 2016.

[3] L. Yang, P. Mubayi, P. Terry, Heath, and H. Bellini, "Music in the air - The show must go on," Goldman Sachs, Tech. Rep., 2020. [Online]. Available: https://www.goldmansachs.com/insights/pages/infographics/music-in-the-air-2020/report.pdf

[4] C. Duhigg, *The Power of Habit: Why We Do What We Do in Life and Business*, 2012.

[5] B. Marr, *The Amazing Ways Artificial Intelligence Is Transforming The Music Industry.* Accessed on 2021-01-03, 2019. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2019/07/05/the-amazing-ways-artificial-intelligence-is-transforming-the-music-industry/

[6] C. Shearer, "The CRISP-DM Model: The New Blueprint for Data Mining," vol. 5, no. 4, pp. 13–22, 2000.

[7] M. S. Brown, *What IT Needs To Know About The Data Mining Process.* Accessed on 2020-12-13, 2015. [Online]. Available: https://www.forbes.com/sites/metabrown/2015/07/29/what-it-needs-to-know-about-the-data-mining-process/

[8] Spotify, *Company Info.* Accessed on 2020-11-20, 2020. [Online]. Available: https://newsroom.spotify.com/company-info/

[9] Spotify for Developers, *Web API.* Accessed on 2020-11-20, 2021. [Online]. Available: https://developer.spotify.com/documentation/web-api/

[10] Genius.com, *Genius | Song Lyrics & Knowledge.* Accessed on 2021-01-26. [Online]. Available: https://genius.com/

[11] ChartMasters, *Most streamed artists ever on Spotify.* Accessed on 2020-11-08, 2020. [Online]. Available: https://chartmasters.org/most-streamed-artists-ever-on-spotify/

[12] D. S. Hain, *Introduction to Network Analysis.* ccessed on 2021-01-09, 2020. [Online]. Available: https://sds-aau.github.io/SDS-master/M2/notebooks/network_analysis_theory.html

[13] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 4 2008.

[14] GeeksforGeeks.org, *Removing stop words with NLTK in Python.* Accessed on 2021-01-08, 5 2017. [Online]. Available: https://www.geeksforgeeks.org/removing-stop-words-nltk-python/

[15] J. F. Trevor Hastie, Robert Tibshirani, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 6 2013, no. 536.

[16] Statsmodels.org, *Statsmodels, statistical models, hypothesis tests, and data exploration.* Accessed on 2021-01-11. [Online]. Available: https://www.statsmodels.org/stable/index.html

[17] Y. E. Ay, *Spotify Dataset 1921-2020, 160k+ Tracks.* Accessed on 2021-01-08, 2020.

[18] GeeksforGeeks.org, *Lasso vs Ridge vs Elastic Net | ML.* Accessed on 2021-01-18, 2020. [Online]. Available: https://www.geeksforgeeks.org/lasso-vs-ridge-vs-elastic-net-ml/#:~:text=LassoregressionstandsforLeast,termtothecostfunction.&text=Thedifferencebetweenridgeand,ofcoefficienttoabsolutezero

[19] Q. Lanners, *Choosing a Scikit-learn Linear Regression Algorithm.* Accessed on 2021-01-15, 2019. [Online]. Available: https://towardsdatascience.com/choosing-a-scikit-learn-linear-regression-algorithm-dd96b48105f5

[20] E. Lewinson, *Verifying the Assumptions of Linear Regression in Python and R.* Accessed on 2021-02-02, 2019. [Online]. Available: https://towardsdatascience.com/verifying-the-assumptions-of-linear-regression-in-python-and-r-f4cd2907d4c0

[21] Scikit-learn.org, *Choosing the right estimator.* Accessed on 2021-01-12. [Online]. Available: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

[22] NetworkX 2.5 documentation, *networkx.algorithms.centrality.degree_centrality.* Accessed on 2021-01-20, 2020. [Online]. Available: https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.centrality.degree_centrality.html

[23] Wikipedia, *Louvain method*. Accessed on 2021-01-15, 12 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Louvain_method&oldid=995883113

[24] R. Jurowetzki, *SDS-master/M2_Directed_Networks_hands_on_Python.ipynb at master · SDS-AAU/SDS-master · GitHub*. Accessed on 2021-02-03, 2020. [Online]. Available: https://github.com/SDS-AAU/SDS-master/blob/master/M2/notebooks/M2_Directed_Networks_hands_on_Python.ipynb

[25] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," vol. 3, pp. 993–1022, 2003.

[26] S. Kapadia, *Evaluate Topic Models: Latent Dirichlet Allocation (LDA)*. Accessed on 2021-02-01, 2019.

[27] E. Leight, *Why Your Favorite Artist Is Releasing More Singles Than Ever*. Accessed on 25.01.2021, 2018. [Online]. Available: https://www.rollingstone.com/music/music-features/why-your-favorite-artist-is-releasing-more-singles-than-ever-629130/

10 Appendix

Feature	Description	Data Type
Artist Name	String	The name of an artist.
Follower Count	Integer	Number of followers of an artist.
Genres	List[String]	A list of genres an artist is associated with.
Popularity	Integer	The popularity of an artist calculated from the popularity of all the artist's tracks.

Table 10: Description artist data

Track Name	Count
Intro	472
Aria mit 30 Veränderungen, BWV 988 "Goldberg Variations": Aria	155
Outro	155
Violin Concerto No.2 In E, BWV 1042: 3. Allegro assai	138
Partita No.1 in B flat, BWV 825: 1. Praeludium	116

Table 11: Top five of track names that appear multiple times

Feature	Description	Data Type
Track Name	String	The name of the track.
Artist Names	List[String]	The names of the artists who performed the track.
Album Name	String	The name of the album on which the track appears.
Duration in ms	Integer	The track length in milliseconds.
Explicit	Boolean	Whether or not the track has explicit lyrics. (true = yes it does; false = no it does not or unknown.
Popularity	Integer	The popularity of the track.
Track ID	String	The Spotify ID for the track.
Artist IDs	List[String]	The IDs of the artists who performed the track.

Album ID	String	The ID of the album on which the track appears.
Acousticness	Float	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
Danceability	Float	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
Energy	Float	Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.
Instrumentalness	Float	Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.
Liveness	Float	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.

Loudness	Float	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typical range between -60 and 0 db.
Speechiness	Float	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.
Valence	Float	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).
Tempo	Float	The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.
Time Signature	Integer	An estimated overall time signature of a track. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure).
Key	Integer	The key the track is in. Integers map to pitches using standard Pitch Class notation . E.g. 0 = C, 1 = C♯/D♭, 2 = D, and so on.

Mode	Integer	Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0.
Release Date	String	The date the album - on which the track is located - was first released.
Genres	List[String]	A list of genres that the artist of the track is associated with.

Table 12: Description track data

Feature	**Description**	**Data Type**
Artist	String	The name of an artist.
Title	String	The title of the track.
Lyrics	String	The lyrics of the track.

Table 13: Description lyrics data

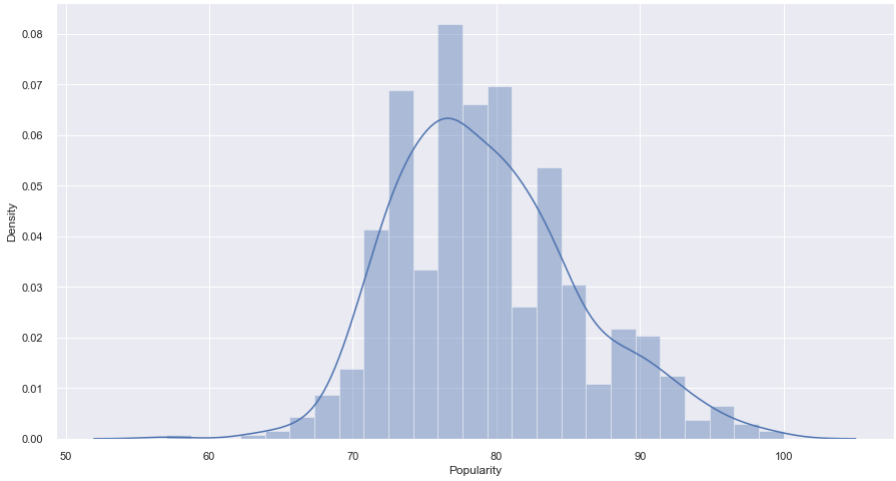

Figure 10: Popularity distribution plot

Genre	**Density**	**Transitivity**
pop rap	0.1813	0.4945
pop	0.0229	0.1523
rap	0.1354	0.3082
latin	0.1301	0.4595
rock	0.0186	0
alternative metal	0.0667	0
pop dance	0.0442	0.2233
country	0.1129	0.3529
german hip hop	0.3833	0.5612

Table 14: Density and Transitivity by Artist.

Figure 11: NetworkX network graph grouped by the Louvain partition

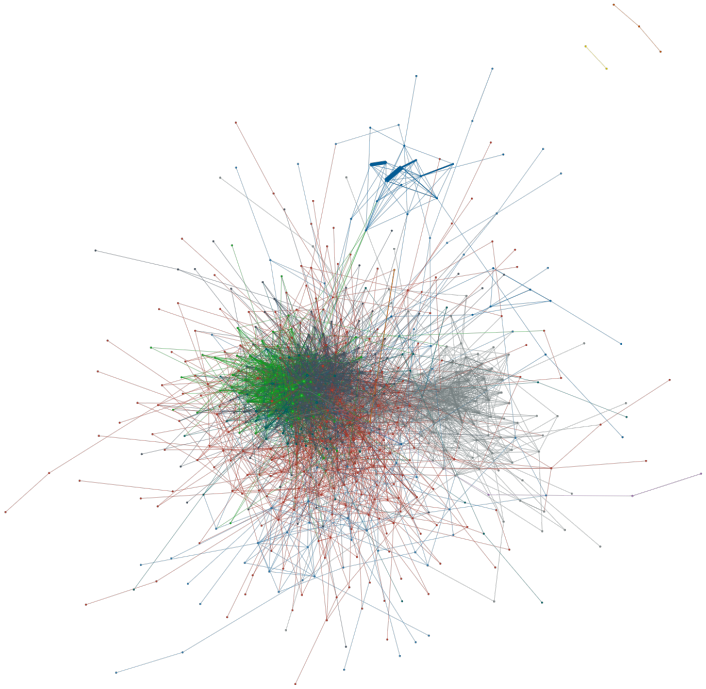

Figure 12: Pyvis network graph clustered with the Louvain algorithm

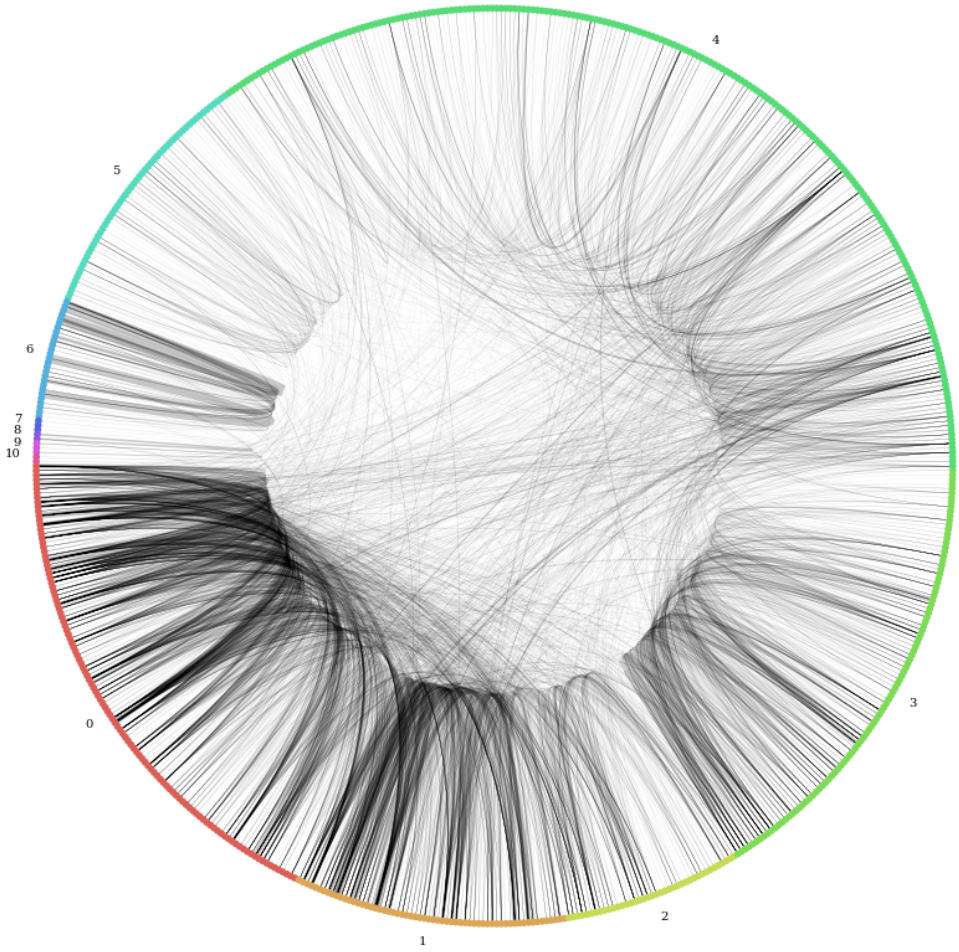

Figure 13: Circos plot grouped by the Louvain partition

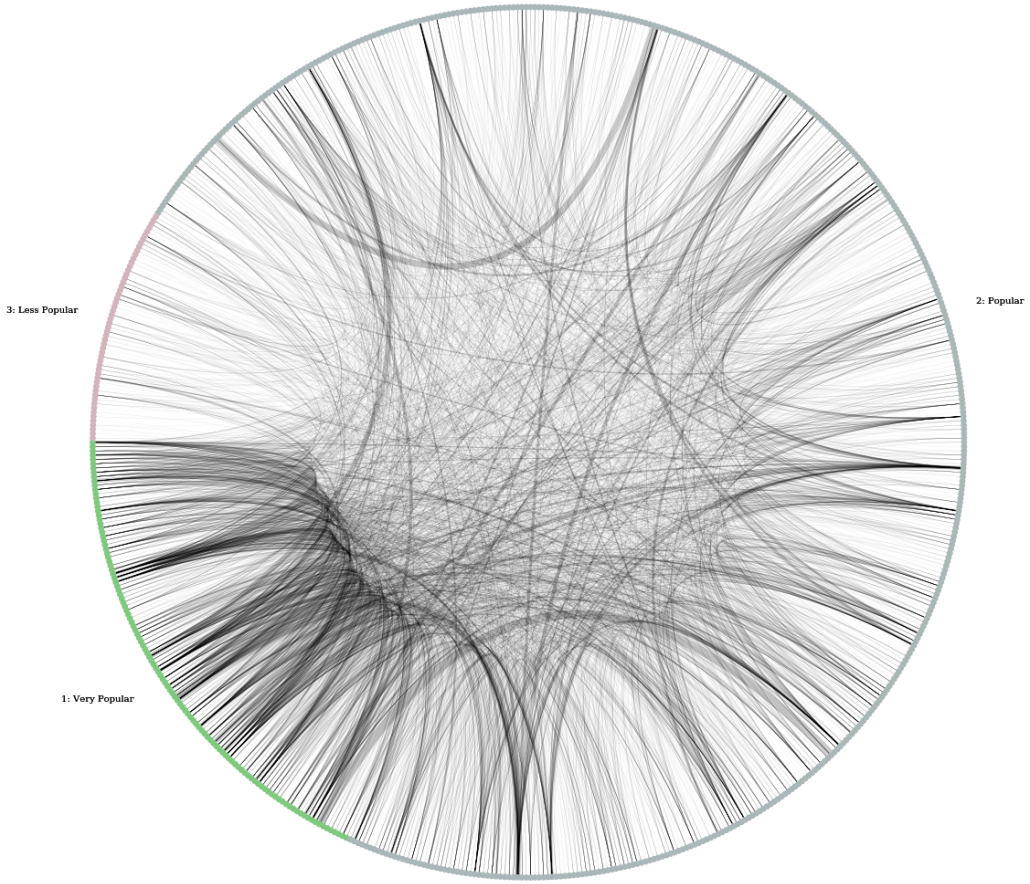

Figure 14: Circos plot grouped by popularity

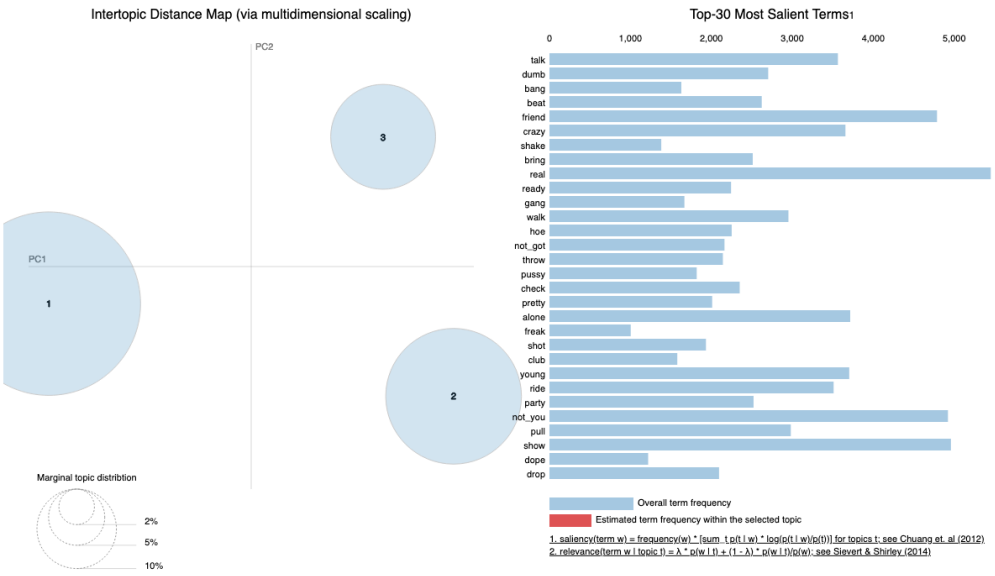

Figure 15: LDA model with 3 topics

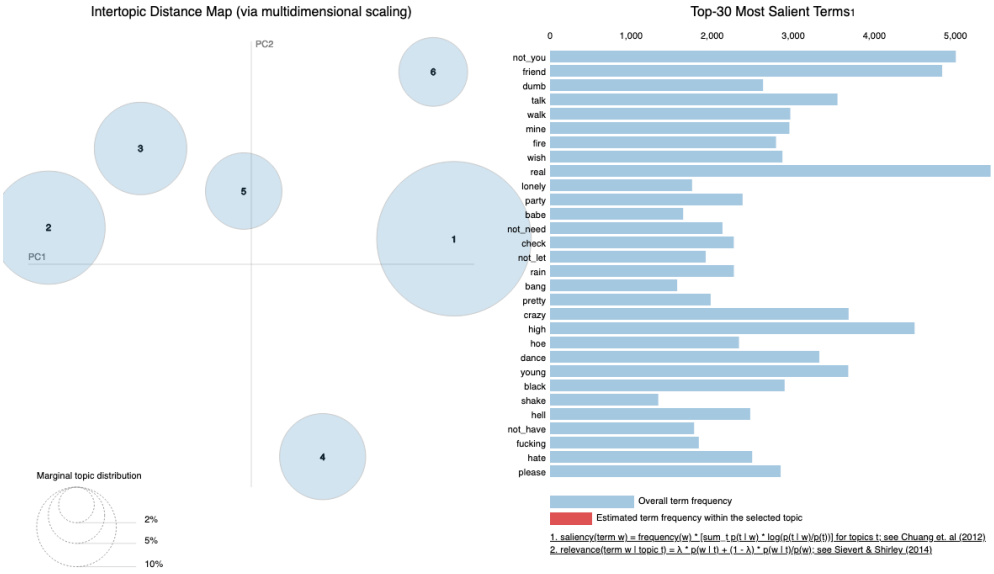

Figure 16: LDA model with 6 topics

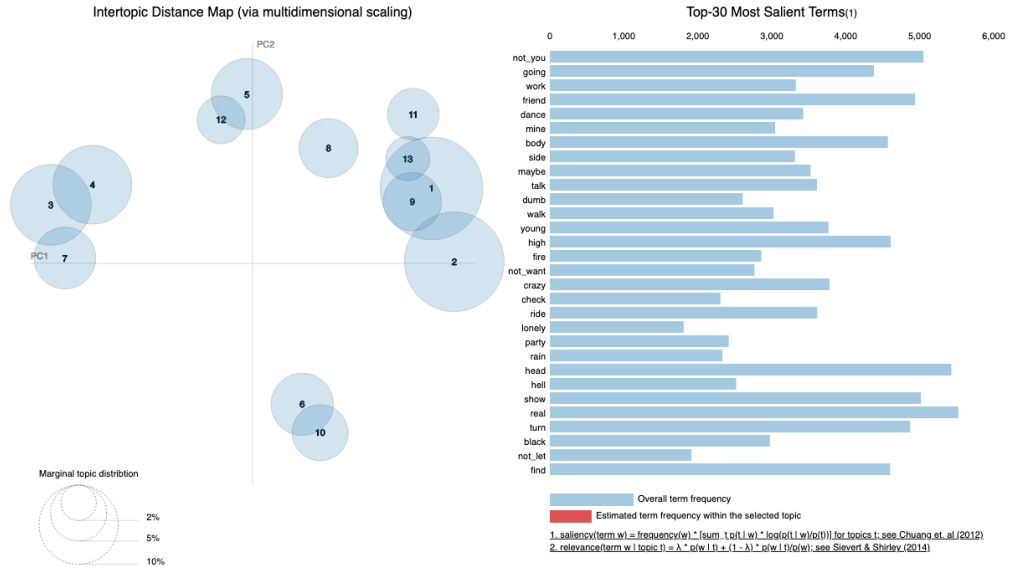

Figure 17: LDA model with 13 topics