Department I – Economics and Social Sciences
Business Administration – Digital Economy
Management Information Systems in Enterprises

BHT Berliner Hochschule für Technik

Bachelor Thesis

Lyrics Mining

Investigating the Influence of the Coronavirus on 2020s Music Charts

Felix Siegfried Riedel
Matriculation-Nr. 878301

January 31, 2022

Supervised by Prof. Dr. Klaus-Peter Schoeneberg
Reviewed by Prof. Dr. Peter Weimann

Abstract

This paper presented the advances and techniques of Natural Language Processing. Its application in the music industry was in particular focus. It was shown how NLP-algorithms can be used to extract keywords, sentiment and topics from song lyrics of the 2019 and 2020 Billboard Weekly Hot 100 charts and insights were given on the influence and presence of COVID-19 in these U.S.-American charts. It showed that 10% of the songs that were released after March 2020 addressed coronavirus and reduced the average sentiment by 8%. After the outbreak in March explicit language was 16% more frequent and the topic love was 7% less frequent in the lyrics of the songs.

***Keywords*—** NLP, music, charts, coronavirus

Zusammenfassung

In dieser Arbeit wurden die Fortschritte und Techniken von Natural Language Processing präsentiert. Deren Anwendung innerhalb der Musikindustrie lag in besonderem Fokus. Es wurde gezeigt, wie NLP-Algorithmen verwendet werden können um Keywords, Stimmung und Themen der Songtexte aus den Billboard Weekly Hot 100 Charts auszulesen und es wurden Einblicke gegeben in den Einfluss von COVID-19 auf die U.S.-amerikanischen Musikcharts. Es wurde gezeigt, dass 10% der nach März 2020 veröffentlichten Songs den Virus addressieren und die durchschnittliche Stimmung um 8% gesenkt haben. Nach dem Ausbruch war explizite und vulgäre Sprache zu 16% häufiger und das Thema Liebe zu 7% seltener in den Songtexten vertreten.

***Keywords*—** NLP, Musik, Charts, Coronavirus

Contents

Abbreviations

AI Artificial Intelligence. 7, 15, 50

ALPAC Automatic Language Processing Advisory Committee of the National Academy of Science. 9

API Application Programming Interface. 5, 14, 24, 27, 30

BB-AS dataset with unique artists and songs featured in BB-T100. 29, 30

BB-L dataset with unique artists, songs and lyrics featured in BB-T100. 30, 31

BB-L-EN BB-L with only English songs. 32, 50

BB-T100 Billboard Weekly Hot 100 dataset with songs featured in 2019 and 2020. 29, 30, 33

BB-T100-EN BB-T100 with only English songs. 33, 50

COVID-19 Coronavirus Disease 2019. 3, 18–20, 22, 23, 33, 49

CRISP-DM Word Embedding Association Test. 21

DL Deep Learning. 10, 12

DNN Deep Neural Network. 11

DOM Document Object Model. 24

G19-BB Diego Guzmán's Billboard Weekly Hot 100 dataset with songs featured in 2019. 28, 29

IAT Implicit Association Test. 17

IBM International Business Machines Corporation. 10, 11

LDA Latent Dirichlet Allocation. 15, 38, 39, 47, 52

ML Machine Learning. 8, 10–13, 48, 50

MT Machine Translation. 8, 9, 11, 13, 48

NLP Natural Language Processing. 3, 4, 6–14, 23, 24, 32, 43, 46, 50, 52

NLU Natural Language Understanding. 7, 9, 11

NN Neural Network. 10–12

R20-BB Billboard Weekly Hot 100 dataset with songs featured in 2020. 28, 29

WEAT Word Embedding Association Test. 17

1. Introduction

Music is one of the biggest mediums for entertainment and is often used by its creators to express certain feelings and opinions. Changes in our world and society are commonly addressed in public media and the entertainment industry such as the music industry. As songs are released daily, lyrics are also published that express the context of the given songs. With the growing importance of *Natural Language Processing* (NLP) to aid voice assistants and chat bots, text mining algorithms and techniques become better and more reliable.

This thesis explores NLP and shows how it can be used for the classification of songs based on their lyrics. The analysis shows if a thematic and sentimental change in the U.S-American charts is noticeable by potentially changed listening and songwriting habits that were influenced by the ongoing pandemic.

1.1. Motivation

NLP has been researched since the early days of computing (Hancox 1996) and became part of our daily lives with voice assistants being present in almost every modern phone. Still algorithms struggle to perfectly interpret written and spoken language. This thesis is not trying to improve text mining techniques nor developing new ones. Instead it shows how existing techniques can be used to classify songs based on their lyrics.

The rapid change in today's society caused by the outbreak of the *Coronavirus Disease 2019* (COVID-19), often referred to as coronavirus, in late 2019 will presumably be noticeable in a thematic change of songs the population is listening to and that is tried to be shown by this research paper.

There are many analyzes on song lyrics already done by data scientists and enthusiasts that inspired this thesis (P.-H. Chen 2020; Kulp 2020), but an analysis on COVID-19's influence on the American music charts is yet to be found. Though there is a study published by Liu et al. which examines how the

listening habits of humans changed during the outbreak: “PANDEMICS, MUSIC, AND COLLECTIVE SENTIMENT: EVIDENCE FROM THE OUTBREAK OF COVID-19”.

1.2. Objective

The main goal of this thesis is to investigate how well lyrics can be classified by NLP-algorithms. Coronavirus’ influence on the charts will be in special focus of the paper.

1.3. Research Questions

How can songs that address the coronavirus be classified through text mining? is the research question the main part is based on. It shows that the thesis focuses on the practical approach and its results rather than the exploration of theories. Additionally it is attempted to answer the question: *How did the virus influence the US-american music charts?*

1.4. Target Audience

The thesis is targeted towards the general public and aims to be comprehensible to people with a secondary school degree.

Due to its focus on music charts, text mining and the current pandemic, many groups can profit from the results: The classification of songs and the identification of trends in songwriting can be particularly interesting for the music industry. Politicians and sociologists can learn how the media consumed by the general public reacts to controversial political decisions regarding the coronavirus. Data scientists and computer engineers may find interest in the application of the text mining techniques and learn about their boundaries and opportunities.

1.5. Methodic Procedure

The empirical research is split into two parts. A quantitative analysis of the charts lyrics of 2019 and 2020 to receive an understanding what trends are present and how they change with the outbreak of the coronavirus; and a qualitative analysis of songs published after the outbreak.

Only songs with the biggest appeal to the public are analysed. The song selection is based on the Billboard Top 100 Weekly Charts[1] which feature the most played songs during one week in the USA. Kaggle user Guzmán already provided a dataset that contains the charts of 2019 (Guzmán 2020). The charts of 2020 are scraped from the Billboard website (Riedel 2021) with the use of the Scrapy[2] package. With the help of LyricsGenius[3], a python framework for the genius.com *Application Programming Interface* (API), all available lyrics of the songs are gathered. A song is not considered in the analysis, if it happens to have no lyrics or if the major part of the lyrics happens to be in a language other than English.

The preparation and analysis of this data is executed in Jupyter Notebooks that run Python and common data science packages are utilized such as NumPy, Pandas and Seaborn. The spaCy API is used for processing the lyrics.

All data and code that is related to the project can be found in the GitHub repository[4]. The order in which the Jupyter Notebooks and Python programs are executed is described in the repository and in the appendix B.2.

1 https://www.billboard.com/charts/hot-100

2 https://scrapy.org/

3 https://lyricsgenius.readthedocs.io/en/master/

4 https://github.com/FelixSiegfriedRiedel/chart-lyrics-analysis

2. Origin & State of Natural Language Processing

To give an overview of the technology the empirical research is based on, the following sections are dedicated towards NLP. Similar definitions and interpretations of NLP from different times are presented and past and current advancements reveal how the field became what it is known for today and why it has such a high importance for our daily lives.

2.1. Fundamentals

2.1.1. Definition by Elizabeth DuRoss Liddy

Although the execution of NLP has greatly changed in the last twenty years, a definition proposed in 2001 is still relevant today:

> "Natural Language Processing is a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications."
>
> Liddy 2001, p. 2

Liddy's thoughts on NLP are further displayed in the following paragraphs, based on her chapter in the book "Encyclopedia of Library and Information Science".

Depending on the goal of the analysis, multiple approaches and techniques are used. Texts of any kind can be processed as long as they are of a natural origin and are understandable by mankind. NLP systems are based on various linguistic methods subconsciously used by humans to communicate with each other. Analysis are supposed to have results similar to humans reaction to natural text, therefore

Artificial Intelligence (AI) plays a big role in the field. NLP can have many applications, its use in a musical context is schown in the third chapter: "Application of Text Mining Techniques on Songs". Liddy emphasizes on differing between NLP and *Natural Language Understanding* (NLU). The latter was mainly used during the beginnings of AI and gives a false impression as NLP systems struggle to draw conclusions from texts thus being unable to fully understand natural language. (Liddy 2001, pp. 2–3)

Liddy states three pillars NLP stands on: linguistics, computer science and cognitive psychology. Linguistics provides the theoretical framework for languages, Computer Science translates the models into working technical representations and Cognitive Psychology examines how human brains process and understand text. She separates between two distinct tasks: NLP and Natural Language Generation, similar to how humans receive and emit language. Although many techniques are used for both purposes, the computer assisted creation of text especially needs certain models to define a message or meaning that needs to be delivered. Distinctions between written and spoken language can also be made and the interpretation of spoken language in particular can be considered a field of its own (Liddy 2001, pp. 3–4), but I will mainly focus on written texts in this bachelor thesis.

2.1.2. Definition by Kevin Cohen

A more recent definition has been composed by Cohen in 2014:

> "Natural language processing is the study of computer programs that take natural, or human, language as input. Natural language processing applications may approach tasks ranging from low-level processing, such as assigning parts of speech to words, to high-level tasks, such as answering questions. Text mining is the use of natural language processing for practical tasks, often related to finding information in prose of various kinds. In practice, natural language processing and text mining exist on a continuum, and there is no hard and fast line between the two."
>
> Cohen 2014, p. 3

He differences between two types of NLP systems: knowledge-based systems and statistical systems.

The knowledge-based approach is based on psychology, linguistics and empirical data. Knowledge about the language, the format of the text and the concrete application is used to develop certain rules and knowledge databases. Experience plays a big role in setting up a knowledge-based NLP system

and the setup of said system can be very complicated and time consuming says Cohen. (Cohen 2014, pp. 5–6)

According to Cohen, statistical systems, often referred to as *Machine Learning* (ML), are able to form their own classification of the given data. The Algorithms are trained with unknown data and are able to work with outliers as long as they are present in the training data. Due to their self-learning capabilities, no specific knowledge about the domain is required. Though creating useful training data can be very challenging and as stated by Zipf's Law (Chao and Zipf 1949), some cases are too rare to be represented in a meaningful way for the algorithms. In many use cases, both systems are combined depending on time and financial resources (Cohen 2014, pp. 5–6).

Based on the definitions by Liddy and Cohen I want to introduce my interpretation of the field:

> *Natural Language Processing is the algorithmic analysis of language and texts produced by humans. It is compromised of mathematical, computational and linguistic techniques and its ultimate goal is to achieve interpretation on a level similar to and beyond the way humans understand and interpret language. Modern-day NLP is executed using machine learning and artificial intelligence.*

Whenever NLP is mentioned in this paper, I am referring to the definition above.

2.2. Historical Milestones

I am presenting the early stages of NLP based on the listings by Liddy and Hancox to show the different influences on the field. (Liddy 2001, pp. 4–6, cf. Hancox 1996)

1949 – Cryptography Research on NLP started with the memorandum written by Warren Weaver. Code breaking gave Weaver the idea, that languages could be translated using methods of cryptography. (Weaver, "Translation" reproduced in: Locke and Booth 1955, pp. 15–23)

During the cold war, researchers, usually mathematicians, began developing early *Machine Translation* (MT) systems. Developing such systems happened to be a greater challenge than first expected and barely returned valuable results. (Hancox 1996)

1957 – Linguistics In the late 50s MT researchers started cooperating with linguists. Chomsky had a great influence on the future of NLP with his study on syntactic structures. (Chomsky 2002) It was agreed on, that systems needed to pre-process the data, especially to work with equivocal words. During this period new concurring NLP application areas spawned: language processing with the use of grammatical theories and speech recognition with the help of statistics. (Liddy 2001)

1966 – ELIZA Weizenbaum introduced his influential computer program ELIZA, simulating conversations between a psychologist and a patient. The program composes questions based on the previous inputs using defined rules. (Weizenbaum 1966)

1966 – ALPAC Report In that same year a report by the *Automatic Language Processing Advisory Committee of the National Academy of Science* ALPAC stated that MT is too complicated and funding shall stop (*Language and Machines: Computers in Translation and Linguistics - A Report by the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council* 1966). This resulted in many scientists discontinuing their research on MT. (Liddy 2001)

1972 – SHRDLU One of the early examples of NLU is SHRDLU, a computer program developed by Winograd. It is capable of describing and rearranging a virtual room filled with objects and can react to tasks and messages input by the user. SHRDLU is able to memorize and debate its current and past behavior and can ask the user to elucidate his or her intentions. (Winograd 1972)

1985 – Discourse Strategy Model In her paper, McKeown introduces a discourse strategy model for text generation systems. Based on predetermined communication goals such as comparison, definition, or description, the model guides how the responses are generated. Her model is based on natural text and its functionality is presented with TEXT, a program, that can give extensive answers on questions regarding database structure. (McKeown 1985)

1988 – Rhetorical Strucuture Theory The Rhetorical Structure Theory is proposed by Thompson and Mann. It says that texts consist of multiple parts that have various relations to each other instead of being a collection of clauses. The relations of the sentences are influenced by the writers intentions for the text and the writers premises on the reader. Neighboring parts form relations and can be divided in main- and ancillary-information. Analysis in accordance with RST build a hierarchy of the rhetoric structure of the text. Those mechanism and their application are featured in their research paper. (Thompson and Mann 1987)

Liddy has multiple explanations for the quick growth of NLP research during the 90s: A lot of texts have been digitized, the performance of computers greatly improved and more people had access to the internet. Meanwhile, NLP systems have been developed that are reasonably good at processing standard text.

In the modern era of the field, *Deep Learning* (DL) is the main driver of successful NLP systems. As a matter of fact, languages are too complex and ambiguous to develop valuable solutions with traditional programmatic approaches. DL algorithms on the other hand are capable of forming their own decisions and identifying suitable inputs and outputs. (Johri et al. 2021, p. 369) During the 2000s NLP models were build on *Neural Networks* (NN) beeing able to make predictions on upcoming words of a text. (Louis 2020)

2003 – Neural Probabilistic Language Model Word sequences used for training NLP models often differ from the actual passages the models are applied on. This is a big challenge of statistical language modeling which was tackled by the approach Bengio et al. introduced: Broad representations of words inform the model of other sentences with a similar semantic structure. Various versions of words are learned simultaneously with an additional probability function for word chains. The proposed technique turned out to be more reliable, having less prediction errors than state of the art methods of that time. (Bengio et al. 2003)

2007 – Watson In 2007 *International Business Machines Corporation* (IBM) started development on a question-answering system called Watson. Four years later the program managed to win the American quiz show Joepardy beating two former human champions. It was developed by circa 25 engineers and researchers and was required to answer never seen clues in real-time; and independently evaluate possible answers. Watson underlies the DeepQA architecture, that scores possible evidences to an answer based on various factors and decides if Watson should use the buzzer. The architecture included ML algorithms that were trained on previous games. Similar to human contestants, Watson

had to be self-contained, meaning no connection to the internet or other individuals was allowed. Therefore resources were build using parsing, information extraction and statistics on encyclopedias and other texts. A unique scoring system was developed by IBM to make answer candidates comparable. (Ferrucci 2012, pp. 1, 3–6, 11)

2008 – Multitask Learning In their study Collobert and Weston applied multitask learning, a ML technique where multiple aspects are learned simultaneously, on an NLP system that runs different language processing predictions. The NLP system was build upon a NN. As the research turned out, models using multitask learning are not only quicker but also more reliable when adapting to new unseen data, giving NLP researchers a new approach for their models. (Collobert and Weston 2008)

2013 – Word2Vec Mikolov, Sutskever, et al. introduced *Word2Vec* which underlying functionality is explained in two research papers written by them. The system spawned a new more efficient way of representing words in vectors which is useful when working with large datasets of words. The vectors represent the semantical and syntactical relation between words and due to similar words being closer in the vector realm, a certain level of NLU was achieved. Instead of having an extensive vocabulary, new words and phrases are represented by the combination of existing word vectors. The accurate and performance-saving system is able to produce high quality vector representations for datasets with more than 1.5 billion words in less than 24 hours. NLP applications like automatic knowledge bases or MT heavily profit from this new approach. (Mikolov, K. Chen, et al. 2013; Mikolov, Sutskever, et al. 2013)

2014 – Sequence to Sequence Learning Working with sequences of unknown limits is a big challenge for *Deep Neural Networks* (DNN), but has great importance for NLP sub-fields like MT or question answering. Sutskever, Vinyals, and Le came up with a new approach to allow sequence to sequence mapping. MT systems based on statistics were outperformed by the NN approach with a smaller vocabulary and more precise translations. (Sutskever, Vinyals, and Le 2014) The framework turned out to be the new state of the art for MT as Google implemented it into their translator in 2016. (Wu et al. 2016, p. 3)

2015 – Semi-Supervised Sequence Learning NLP systems that have been pretrained happen to learn better and perform better with unknown data. This was the result of the research on Semi-Supervised Sequence Learning by Dai and Le. They proposed two methods of pretraining sequence learning algorithms and achieved great performance classifying text data from movie databases like IMBD. (Dai and Le 2015) This approach was further utilized and improved by other scientists and researchers like Howard and Ruder. Their method tries to address the weaknesses of previous systems and is applicable to any NLP task. The improved fine tuning techniques for the NN managed to prevent the loss of key information. It was tested on similar text classification tasks. (Howard and Ruder 2018)

2015 – spaCy The introduction of spaCy, developed by Honnibal has special significance for this thesis as it is used for mining the lyrics during the empirical research. It is among the most utilized NLP libraries for Python being targeted towards small companies. SpaCy renders a faster and more accessible alternative to other commonly used packages in the field and has a profound documentation that is well structured and easy to understand for professionals and newcomers. (Honnibal 2015)

2.3. State of the Art

The actual relevance of NLP becomes aware when modern use cases are examined: Current applications featuring NLP are discussed in "Natural Language Processing: History, Evolution, Application, and Future Work" written by Johri et al.

Voice assistants like Siri or Google Assistant would not be able to react to orders without the help of DL and NLP. (Johri et al. 2021, pp. 372–374)

Another common use nowadays is sentimental analysis: businesses can adapt to the current zeitgeist by retrieving information from comments and thoughts posted on social media. Instead of manually reading and analyzing long passages of text, doctors can give quicker diagnoses if they use NLP on the medical records of the patient. (Johri et al. 2021, pp. 372–374)

Spam filtering technology, which is very common in email applications, improved greatly with the help of NLP. Established ML algorithms like decision trees, a decision process that calculates a target value based on a series of tests that are dependant on the results of the prior tests (Sammut and Webb

2011, p. 508), are used to differentiate between wanted and unwanted, or more precisely legitimate and illegitimate e-mails. (Johri et al. 2021, pp. 372–374)

In modern medicine, radiology is used to assist doctors with their diagnosis. Additionally to the images, extensive reports are provided, that describe the disease pattern seen on the image. With the help of ML and especially NLP, the unstructured reports can be evaluated much quicker with creditable precision. In "Essential Elements of Natural Language Processing: What the Radiologist Should Know", P.-H. Chen describes the origins and techniques of NLP in the medical realm. The technology advanced to a point, where it is possible to extract meaning and realistic concepts from unstructured text.

Grammarly is another great example of how NLP is used in modern day applications, say Johri et al. It is an assistant software for writers and can not only check grammar but also suggest different wordings based on the overall theme of the text.

NLP also has great importance for MT which is used by search engines like Google to translate websites. (Johri et al. 2021, pp. 372–374)

Johri et al. further go on explaining that many modern companies use chat bots to quicker communicate with their customers and adapt to user-specific problems without the need of human resources. NLP assisted chat bots are usually faster than humans processing the messages.

Text Mining also has a lot applications in education that are revealed by Bahja. For example NLP can assist teachers in evaluating the language skills of their students by the use of an automatic scoring system applied on the texts. (Bahja 2020, p. 59)

3. Application of Text Mining Techniques on Songs

As shown in the previous chapter, there are many tasks Text Mining can be used for and depending on the goal, various underlying techniques of NLP are being implemented. This chapter is going to be focused on how NLP is being used within the realm of music as similar techniques are going to be applied during the empiric research.

3.1. AI Songwriting

A lyrics generation algorithm running on OpenAI's text generator called GPT-2 is able to generate lyrics similar to the songs written by Kendrick Lamar, Taylor Swift or The Beatles. The tool was developed by Zyro, a website builder from the United Kingdom and acts as an advertisement for their text generation capabilities (Kulp 2020). The generator is available on Zyro's website[1]. Though lacking musicality and sometimes logical connections, the lyrics are able to represent the inspired artists writing style. I generated lyrics to a fictional Beatles song that show, how the algorithm creates new songs based on passages of already existing ones. In the result, that can be found in the appendix (A.1), certain similarities to their song "All You Need Is Love" are noticeable.

Meanwhile, OpenAI updated their Text Mining API to GPT-3 which functionality and improvements are explained in the paper written by Brown et al. Modern day NLP tasks are tackled with the help of pre-training on large datasets and additional fine-tuning. The updated technology of GPT-3 shows better performance than traditional approaches without the use of extensive fine-tuning. It is better at many common tasks in NLP such as translating or unscrambling words and could generate articles that humans were unable to differentiate from man-made texts. (Brown et al. 2020)

1 https://zyro.com/ai-festival

A more musical approach to AI assisted lyrics generation was presented by Y. Chen and Lerch. The model is build upon SeqGAN, a sequence generation framework that can train itself by using a discriminator which constantly evaluates the generations (Yu et al. 2017).

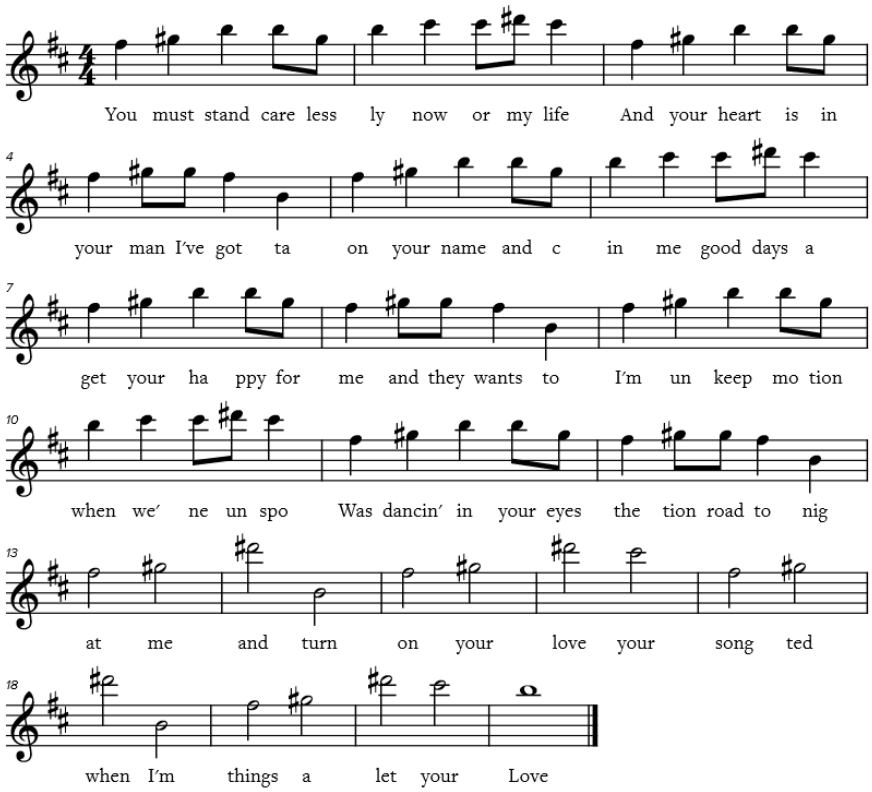


Figure 3.1.: Lyrics Generated by *Lyrics Lab* for a Melody Written by F. S. Riedel.

As seen in figure 3.1, the lyrics generation is influenced by the melodic changes in the song, therefore words and syllables are mapped towards specific notes of the melody, similar to how songs are written by musicians. Y. Chen and Lerch's system consists of N-Gram language models. These are statistical approaches on predicting words by considering already existing words in the sentence (Jurafsky and Martin 2020). With supervised learning the generator is taught to predict the outcome of the sequence based on its previous elements. Additionally the lyrics can be generated related to observable moods in the songs. With the help of *Latent Dirichlet Allocation* (LDA), an algorithm that classifies the topics of texts (Blei, Ng, and Jordan 2003), the representative sentiment of each line of the lyrics in the training dataset was extracted. As a result of the adjustments on the LDA algorithm, five themes were present in the lyrics the model was trained on: *Relationship*, *Patriotic*, *Love Theme*, *Gospel* and *Party*. (Y. Chen and

Lerch 2020, pp. 1–4) A working example of the model can be found in their web-application Lyrics Lab[2]. However, the results, similar to the one that can be seen on figure 3.1, show that the generated text indeed is connected to the lyrics, as the syllables match the rhythm. still a logical or thematic connection between the words and sentences is missing and some parts show clear grammatical errors. Y. Chen and Lerch had similar observations and stated that interjections and cut-off sentences are among the reasons for meaningless lyrics (Y. Chen and Lerch 2020, p. 8).

3.2. Style- & Bias-Decoding of Song Lyrics

In contrast to the mostly manual examination of song lyrics, Barman, Awekar, and Kothari came up with a method to analyze the style and the bias of thousands of songs published in the last 50 years (Barman, Awekar, and Kothari 2019, p. 1).

Up to 500.000 distinct songs represented in two datasets were used for the analysis. The popular songs were featured in a dataset including the Billboard Hot 100 Charts from 1965 to 2015, whilst all other songs published during that time were retrieved from the Million Song Dataset (Bertin-Mahieux et al. 2011). Both datasets were populated with the songs lyrics that were scraped from sites like LyricsMode[3] (Barman, Awekar, and Kothari 2019, p. 1).

(a) Top 100 Words in 1965

(b) Top 100 Words in 2015

Figure 3.2.: Comparison of Top Words Used (Barman, Awekar, and Kothari 2019, p. 2)

2 https://lyrics-lab.herokuapp.com/

3 https://www.lyricsmode.com/

While analyzing the style, Barman, Awekar, and Kothari found a thematic shift in the music industry during that period: The word "love" was significantly less popular in modern songs compared to 1965 (see fig. 3.2) and a comparison between mentions of "rock" and "blues" showed a decline in the popularity of "blues" although "rock" stayed relevant. Even though songs featured in the Billboard charts tend to use less explicit language, an increase in swear word usage is noticeable since the beginning of the 90s. (Barman, Awekar, and Kothari 2019, p. 2)

To gain information on the sentiment of the text, the *Word Embedding Association Test* (WEAT) was used. WEAT, introduced by Caliskan, Bryson, and Narayanan in 2017, is based on the *Implicit Association Test* (IAT). The IAT measures the bias of individuals on certain categories by choosing between two possible associations (Greenwald, McGhee, and Schwartz 1998). Caliskan, Bryson, and Narayanan applied this method on word embeddings, a representation of words in vectors, where the distance between vectors represents the intensity of the connection between those words. The bias towards categories is retrieved by determining their distance in the vector realm instead of surveying humans (Caliskan, Bryson, and Narayanan 2017). For example if the categories (*metal*, *jazz*) and the associations (*loud*, *calm*) are given, WEAT would associate *metal* with *loud*, if its vector is closer than *calm* in the given text database.

Test no.	Target Words	Attribute Words	w2v	FT	CA	IAT
1	Flowers v/s Insects	Pleasant v/s Unpleasant	1.04	1.34	1.5	1.35
2	Instruments v/s Weapons	Pleasant v/s Unpleasant	1.59	1.62	1.53	1.66
3	European-American v/s African-American names	Pleasant v/s Unpleasant	0.71	0.6	1.41	1.17
4	Male v/s Female names	Career v/s Family	1.47	1.72	1.81	0.72
5	Math v/s Arts	Male v/s Female terms	1.10	0.98	1.06	0.82
6	Science v/s Arts	Male v/s Female terms	1.03	1.15	1.24	1.47
7	Mental v/s Physical disease	Temporary v/s Permanent	0.89	1.15	1.38	1.01
8	Young v/s Old People's names	Pleasant v/s Unpleasant	0.37	0.97	1.21	1.42

Table 3.1.: Comparison of Effect Size (Barman, Awekar, and Kothari 2019, p. 4)

To test how well lyrics represent the human bias on different things, Barman, Awekar, and Kothari compared the calculations of WEAT on the lyrics to the results of the IAT and WEAT how it was originally applied by Caliskan, Bryson, and Narayanan. To create the word embeddings for the lyrics two different algorithms were used: Word2Vector and fastText. The results can be seen in table 3.1. Columns four and five show the WEAT score based on the two different word embedding methods, column six shows the original WEAT score that was calculated on a large text dataset scraped from the internet and the last column shows the result of the IAT conducted on human subjects. Positive scores mean that the first target word is more likely to be associated with the first attribute word, negative scores mean the opposite and scores close to zero indicate little to no bias. Some tests could not be sufficiently recreated: Specific names and scientific terms needed for tests 3, 7, 8 were not present in

large quantities in the lyrics. However the results show that songs do in fact represent public bias, especially test 4 showed strong similarities in gender stereotypes. (Barman, Awekar, and Kothari 2019, pp. 3, 4)

3.3. Sentiment Analysis on Songs Listened to During the Outbreak of COVID-19

Similar to the empirical part of this paper, Liu et al. had the idea to examine how the rise and spread of COVID-19 cases in a country influence that countries music sentiment (Liu et al. 2020, pp. 157–158).

To analyze the sentiment of the music that has been listened to before and during the outbreak, they made use of the LFM-1b dataset containing music streaming data of *last.fm*[4] users (Schedl 2016). The data they used displays listening activity between November 1, 2019 and March 27, 2020 of more than 12.000 users from 40 different countries. It is assumed that public sentiment on artists is validly represented by the user generated tags. Those tags were scraped from the corresponding artist pages on last.fm. The data on the coronavirus cases is based on the numbers provided by *Our World in Data* (Ritchie et al. 2020). For every country the first reported case of COVID-19 determined the start of the pandemic (Liu et al. 2020, p. 158).

For their sentiment analysis on the artist tags, Liu et al. followed the approach by Zangerle et al. (Zangerle et al. 2021) and made use of sentiment lexicons as described in "Lexicon-Based Methods for Sentiment Analysis" (Taboada et al. 2011). Each word of the tag gets a sentiment score from the dictionary, in case it matches to multiple lexicons, the arithmetic mean is calculated. The mean value of every word's sentiment of a tag forms the overall sentiment of that tag. Then the artist sentiment is calculated as weighted average of all sentiment scores of the specific artist's underlying tags (Liu et al. 2020, pp. 158, 159).

To avoid adulteration based on changing listening trends unrelated to the pandemic, the researchers made use of the *Difference in Differences* (DD) approach. Given two groups of individuals where only one group is receiving treatment, DD is used to determine the actual influence of the treatment without outside influences that may have occurred on both groups (Angrist and Pischke 2009, p. 169). In this case, countries that already had COVID-19 cases were compared with countries that were unaffected

4 https://www.last.fm/

by the virus in that time frame. Liu et al. build their models using two dependent variables: *USE* and *POS*. USE hereby refers to the average polarity score of the artists a last.fm user listened to. It is a floating point number that can range between 0 and 1. Whenever the USE score is equal or above the 90th percentile of all values, POS, the second dependent variable, gets set to 1 otherwise its 0. (Liu et al. 2020, pp. 159, 160)

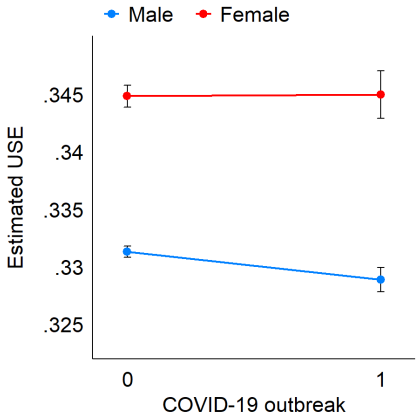


Figure 3.3.: Estimated Interaction Effect Between COVID-19 Outbreak and Gender (Liu et al. 2020, p. 161).

The study shows the possibility of collecting public sentiment by examining listening habits of a wide range of users. After the first case of COVID-19 was reported, especially male users show a more negative artist sentiment. However, female users did not change their interest towards more negative connoted artists and songs as seen in columns three and four of the first panel in table 3.2. This fact is also illustrated in figure 3.3. As seen in column 6 panel 1, a decreased probability of 2.8% of users listening to artists with extremely positive sentiment scores (above 0.9) was caused by the outbreak. The probability, that males show extremely positive listening habits declined by 3.16%, almost twice as much seen with female users (see columns 7, 8 in table 3.2, panel 1).

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Panel1								
Model	**USE**				**POS**			
	All	All	Female	Male	All	All	Female	Male
COVID19	-0.0020***	-0.0024***	0.0008	-0.0028***	-0.0284***	-0.0290***	-0.0161***	-0.0316***
	(0.0006)	(0.0006)	(0.0015)	(0.0007)	(0.0023)	(0.0024)	(0.0062)	(0.0025)
Gender	0.0141***	0.0136***			0.0335***	0.0329***		
	(0.0005)	(0.0005)			(0.0018)	(0.0020)		
COVID#Gender		0.0025**				0.0036		
		(0.0012)				(0.0047)		
Age	-0.0000**	-0.0000**	0.0001**	-0.0000***	-0.0001	-0.0001	0.0003*	-0.0001**
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0001)	(0.0001)	(0.0002)	(0.0001)
Constant	0.3321***	0.3322***	0.3428***	0.3328***	0.0999***	0.1000***	0.1199***	0.1025***
	(0.0005)	(0.0005)	(0.0013)	(0.0005)	(0.0020)	(0.0020)	(0.0053)	(0.0020)
Observations	184,277	184,277	31,894	152,383	184,277	184,277	31,894	152,383
R-squared	0.030	0.030	0.036	0.025	0.014	0.014	0.020	0.011
Panel 2								
Model	**USE**				**POS**			
	All	All	Female	Male	All	All	Female	Male
COVID19 cases	-0.0005**	-0.0005**	-0.0003	-0.0006**	-0.0067***	-0.0065***	-0.0069***	-0.0068***
	(0.0002)	(0.0002)	(0.0006)	(0.0002)	(0.0008)	(0.0009)	(0.0023)	(0.0009)
Gender	0.0141***	0.0141***			0.0340***	0.0351***		
	(0.0005)	(0.0005)			(0.0018)	(0.0020)		
Case#Gender		0.0001				-0.0014		
		(0.0003)				(0.0010)		
Age	-0.0000**	-0.0000**	0.0001**	-0.0000***	-0.0001	-0.0001	0.0003*	-0.0001**
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0001)	(0.0001)	(0.0002)	(0.0001)
Constant	0.3321***	0.3322***	0.3432***	0.3327***	0.0996***	0.0995***	0.1224***	0.1017***
	(0.0005)	(0.0005)	(0.0013)	(0.0006)	(0.0020)	(0.0020)	(0.0055)	(0.0021)
Observations	184,277	184,277	31,894	152,383	184,277	184,277	31,894	152,383
R-squared	0.030	0.030	0.036	0.025	0.013	0.013	0.020	0.011

Table 3.2.: Estimated Effect of the Outbreak of COVID-19 (Panel 1) and Number of New COVID-19 Cases (Panel 2) on Users' Music Sentiment. [...] (Liu et al. 2020, p. 162).

Results of the analysis on the influence of rising coronavirus cases in a country on that countries music sentiment can be seen in the second panel of table 3.2. It is prooven that a negative association between the number of cases and the users' sentiment of music is present. Every integer of growth in the number of coronavirus cases leads to an increase towards negative sentiment by 5% (see table 3.2, panel 2, column 1). Liu et al. also discovered, that countries with more reported COVID-19 cases also show a stronger direction towards negative artist sentiment in their listening habits (Liu et al. 2020, pp. 160–162)

4. Lyrics Mining Concept

4.1. Empiric Structure

As the name suggest, mining lyrics can be described as a data mining task and therefore the structure of the empiric research was oriented on Shearer's model: *CRoss-Industry Standard Process for Data Mining* (CRISP-DM) (Shearer 2000).

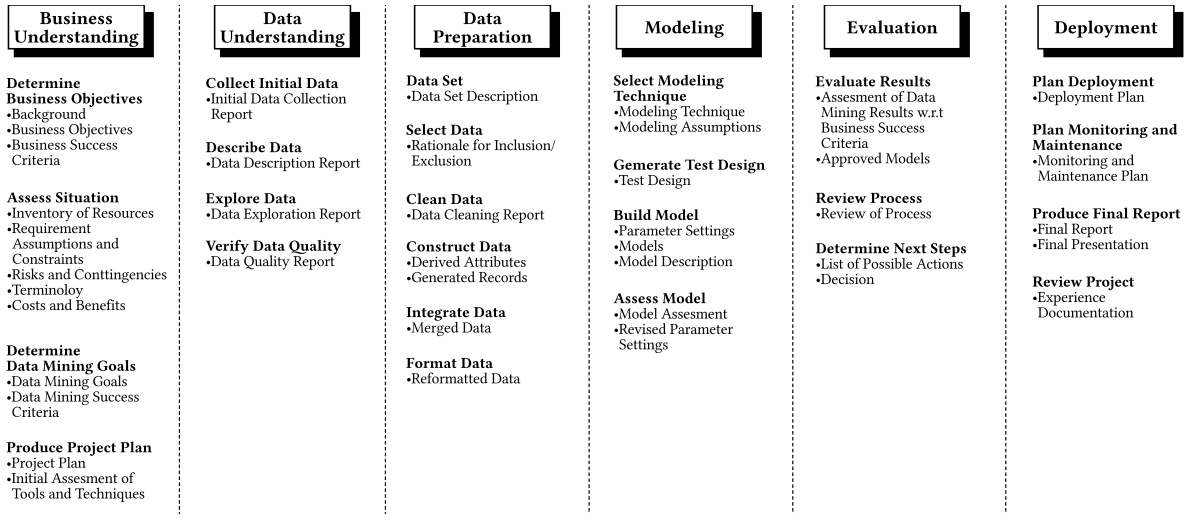


Figure 4.1.: Tasks and Outputs of the CRISP-DM Reference Model (Shearer 2000, p. 14)

CRISP-DM is more targeted towards commercial data mining projects, therefore a project like this study is not considering every step of the model so the structure has been slightly modified. As the current section acts as the conceptional part of the research, only the first two steps of CRISP-DM are featured. The other stages of the lyrics mining process are covered in the 5th chapter. Background, objective, situation and tools and techniques of the project are covered in "Business Understanding". Understanding the business in this case means understanding the coronavirus situation, how the charts work and the project in general. "Data Understanding" describes which data is being used and how it is

organized and a rough exploration is given. The "Modeling" section includes tasks of "Evaluation" and "Deployment" and shows how the three different models were developed. The results of the modeling process are discussed in chapter 5.2.4.

4.2. Business Understanding

4.2.1. Situation

Although first early cases of COVID-19 haven been reported in November 2019 in China (Ma 2020), the virus had its real outbreak in the United States in early 2020 as seen in figure 4.2. Being the first pandemic happening in the digital era, coronavirus spawned exceptional circumstances for the research.

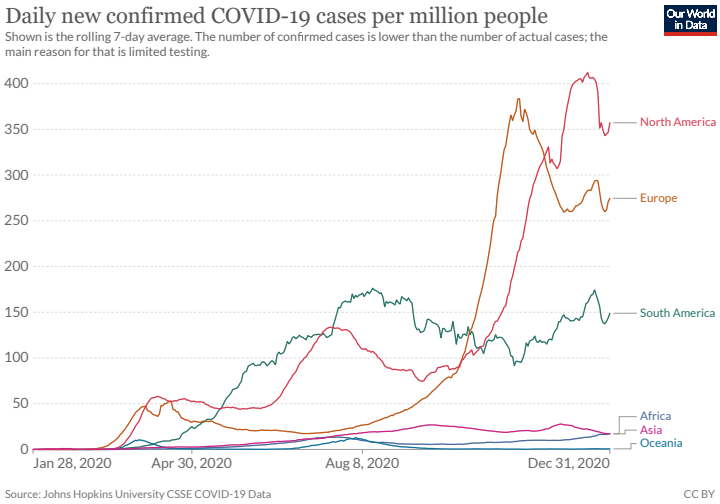


Figure 4.2.: Daily New Confirmed COVID-19 Cases Per Million People (Ritchie et al. 2020)

The article “Prevalence of stress, anxiety, depression among the general population during the COVID-19 pandemic: A systematic review and meta-analysis” says, that psychological disorders like depression and anxiety can be caused due to different factors that are accelerated by the pandemic: misinformation and missing coping strategies, economic problems, problems falling asleep, concern for own and other’s well-being, consumption of social media and isolation (Salari et al. 2020). To the day the virus has not been defeated. However, for the analysis only 2020 was taken to account, as during that time many

countries still struggled to adapt to the new circumstances and presumably people were more heavily affected by it and therefore had stronger changes in their sentiment.

The data for the analysis was retrieved and consolidated from different resources: First the Billboard Weekly Hot 100 Charts are considered, more precisely the charts for every week from January 1, 2019 to December 31, 2020; second the lyrics of each song that was featured in the charts were retrieved from Genius.com, a website with a great array of user generated and confirmed lyrics.

4.2.2. Goal & Assumptions

The main data mining goal was to determine whether it is possible to use algorithms to find interpretations for song lyrics and to see if the lyrics had a thematic change caused by the COVID-19 outbreak. The project was oriented on the earlier proposed research questions: *How can songs that address the coronavirus be classified through text mining? How did the virus influence the US-american music charts?*

I had different assumptions about the result of the study: Regarding the application of NLP for analyzing lyrics, I supposed using similar techniques as presented in chapter 3 would be sufficient to extract the overall orientation of the songs and coronavirus-specific songs should be easily identified based on their keywords and topics. As already shown by Liu et al., the coronavirus does in fact influence public listening sentiment so I presumed that artists with more negative songs are more likely to be charting on Billboard which would lead to a decline in the sentiment within the music charts. I predicted that with a descending sentiment the topic distribution would change with a trend towards more negative topics. Almost every individual is affected by COVID-19, no matter wealth, origin or gender, therefore popular artists might start publishing songs that deal with the current situation and these songs might even be more likely to rank higher on the charts, as the public can easier relate to them. I expected that it would take a few months until songs that are directly addressing coronavirus appear in the charts. Even though the speed of writing, recording and producing a song greatly varies I think it is unlikely that a *coronavirus song* would be released a few weeks after the outbreak.

4.2.3. Tools & Techniques

The complete data mining task was done on a Lenovo Thinkpad E495 with a 2.1 GHz AMD Ryzen 5 3500U quad-core processor and 8.00 GB RAM running Windows 10. The average specs of this laptop render the project reproducible on common PCs but may lead to longer calculation times when it comes to running the NLP models because the integrated graphics card is not as powerful as discrete graphics cards. All data mining tasks were done with Python 3 but Microsoft Excel was used occasionally for additional cleaning and exploration of the data. The different packages with their versions that were used for the tasks can be found in the appendix: B.3.

Billboard does not provide an official dataset containing the charts. Therefore, the Billboard data for 2019 is taken from Guzmán's dataset on Kaggle (Guzmán 2020).

To gather the data for 2020 the web scraping tool Scrapy was used and the text of each relevant element of the *Document Object Model* (DOM) of the different weekly charts was saved to a JSON-file. The source code for the scraper can be found in the appendix: B.4. I uploaded the resulting dataset on kaggle.com (Riedel 2021).

Lyrics platform genius.com has an API (*Genius API* 2021) that can be used to search the website for songs, artists and lyrics. W. Miller developed the LyricsGenius framework (W. Miller 2021) that is used to access the API within Python and export lyrics and other information from the site.

4.3. Data Understanding

4.3.1. Billboard Weekly Hot 100 Charts

As seen in figure 4.3 below, different information can be extracted from the charts.

The number on the left indicates the position on the chart and an arrow below that number indicates if the song kept its position (→), climbed up (↑) or climbed down (↓) compared to the week before. Songs that have not been on the charts yet are indicated by the tag *New*. The title of the song is written in bold text and the smaller text below represents the artists that were involved. In the column *AWARD* different symbols can be found, if the song was given an award. *LAST WEEK* displays the position the song had in the prior week, *PEAK* shows the highest rank the song achieved and

Figure 4.3.: Top 5 Songs of the Weekly Hot 100 Chart of July 25, 2020 (Billboard 2020)

WKS ON CHART says how many weeks the song was featured in the Billboard Weekly Hot 100 charts.

Billboard has an in-depth explanation on how their charts work: The Hot 100 charts are influenced by sales, airplay and streaming data. Sales are based on the U.S music retail market and the data is provided by MRC Data and includes online and offline sales. The radio charts are generated by MRC Data by monitoring mostly commercial radio stations across the United States of America. They are dependent on active listeners. The streaming charts rank the most streamed songs and music videos on popular music streaming platforms. Every Tuesday a new chart is published that is based on the data between Monday and Saturday. A descending song below rank 25 is removed after a year or if it is below rank 50 it is removed after 20 weeks. (Billboard 2021)

billboard_id	rank	artist	song	rank_last_week	peak_rank	weeks_on_chart	date
8100	1	DaBaby Featuring Roddy Ricch	Rockstar	1	1	13	2020-07-25
8101	2	Juice WRLD x Marsh-mello	Come & Go	0	2	1	2020-07-25
8102	3	Jack Har-low Featuring DaBaby, Tory Lanez & Lil Wayne	Whats Pop-pin	3	2	23	2020-07-25
8103	4	The Weeknd	Blinding Lights	2	1	33	2020-07-25
8104	5	Juice WRLD	Wishing Well	0	5	1	2020-07-25

Table 4.1.: Top 5 Songs of the Weekly Hot 100 Chart of July 25, 2020

Table 4.1 shows how the same data is found in the dataset after scraping. I decided to not use the awards for the analysis and therefore they have not been scraped. The arrows and tags below the ranking were skipped as well but the change indication of the rank compared to last week could be recreated with the data.

4.3.2. Genius Lyrics

Even though many lyrics on genius.com are approved by artists, a majority of the content is user generated and therefore wrong lyrics can always be present. I assumed that most of the lyrics that were used for this project were accurate representations of the artists words because songs of the charts receive more attention and presumably have more editors on the site. The posted lyrics all share a similar structure, the key components can be seen in figure 4.4 below.

Different sections of a song are usually marked with square brackets, eg. [Intro]. All other words are the actual lyrics that are noticeable in the song and even interjections like *Yeah* are present. Background vocals are often marked with round brackets, eg. (Oh). Genius also allows users and artists to give annotations to certain passages of the song. Lines with annotations are grey and turn yellow if clicked on.

In addition to the raw text, the release date, the description, annotations and comments have been scraped as seen in table 4.2. The ID of the lyrics on genius and the URL were saved as well to

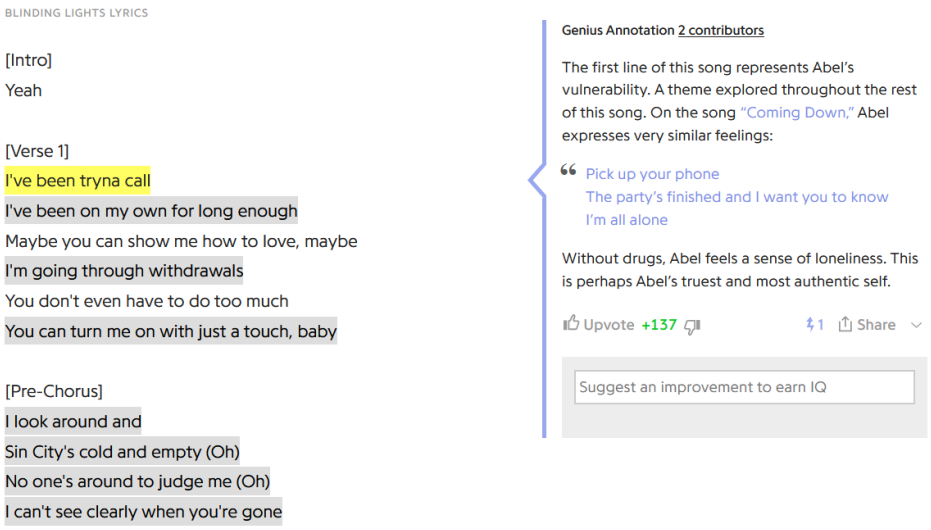


Figure 4.4.: Excerpt of the Lyrics to The Weeknd's song "Blinding Lights" published on genius.com (Tesfaye 2020a)

billboard_id	lyrics	genius_description	genius_annotations	genius_comments	release_date
8103	Yeah I've been tryna ca...	"Blinding Lights" is the s...	[('You can turn me on with...	['This gave so much "False...	2019-11-29

Table 4.2.: Result of Scraping *Blinding Lights*

improve future requests on the API. See B.5 in the appendix for the procedure that was used to scrape the information. A more in-depth description of the resulting datasets is given in the following section.

Views of the datasets similar to table 4.2 do not have citations for the individual lyrics as they are just used for exemplary purposes.

5. Empirical Research

5.1. Data Preparation

On the intersection of my dataset (Riedel 2021) and Guzmán's dataset (Guzmán 2020) featuring Billboard charts between January 5, 2019 and January 2, 2021, different operations needed to be executed to create a usable dataset that was used to extract the lyrics from genius.com eventually and that has enough features to provide useful information for the analysis. For improved readability I am referring to the datasets as *R20-BB* and *G19-BB*. The files in the project directory related to the datasets have the same names.

5.1.1. Dataset Description

The differing shapes of the two datasets can be seen in tables 5.2 and 5.1.

	title	artist	peakPos	lastPos	weeks	isNew	rank	date	year
0	Mood	24kGoldn Featuring iann dior	1	1	15	False	1	2020-11-28	2020
1	Therefore I Am	Billie Eilish	2	94	2	False	2	2020-11-28	2020
2	Positions	Ariana Grande	1	2	4	False	3	2020-11-28	2020

Table 5.1.: Header of R20-BB

	rank	artist	song	rank_last_week	peak_rank	weeks_on_chart	date
0	1	Mariah Carey	All I Want For Christmas Is You	1 Last Week	1 Peak Rank	37 Weeks on Chart	2020-01-04
1	2	Brenda Lee	Rockin' Around The Christmas Tree	2 Last Week	2 Peak Rank	32 Weeks on Chart	2020-01-04
2	3	Bobby Helms	Jingle Bell Rock	9 Last Week	3 Peak Rank	30 Weeks on Chart	2020-01-04

Table 5.2.: Header of G19-BB

To consolidate both datasets the names of the columns and the structure of the date had to be adjusted. Columns *isNew* and *year* in Guzmán's table were removed and the remaining columns were renamed to fit to the columns of R20-BB. The values of the columns *rank_last_week*, *peak_rank* and *weeks_on_chart* were transformed into integers by removing the text. G19-BB also featured several weeks of 2020 that were removed to avoid redundant entries in the final dataset. The resulting dataset *BB-T100* has a total of 5300 rows and 7 columns featuring 733 artists and 1291 songs.

The descriptive statistics seen in table 5.3 give a rough overview on the data. It can be observed that songs on average peak on rank 36. The maximum number of weeks a song has been on the charts is 61 which is more than four times of the average and 44 weeks above the 75th percentile.

	rank	peak_rank	rank_last_week	weeks_on_chart
count	10500.00	10500.00	10500.00	10500.0
mean	50.50	36.16	39.34	12.1
std	28.87	28.39	30.36	10.8
min	1.00	1.00	0.00	1.0
25%	25.75	10.00	11.00	4.00
50%	50.50	32.00	36.00	9.00
75%	75.25	59.00	65.00	17.00
max	100.00	100.00	100.00	61.00

Table 5.3.: Descriptive Statistics on the BB-T100 dataset

The dataset *BB-AS* (see table 5.4) that only contains the unique songs and artists was created to avoid scraping lyrics to the same song multiple times.

	artist	song	weeks_on_chart	peak_rank	first_appearance
0	2 Chainz Featuring Ariana Grande	Rule The World	2	94	2019-03-16
1	2 Chainz Featuring Kendrick Lamar	Momma I Hit A Lick	1	100	2019-03-16
2	2 Chainz Featuring Travis Scott	Whip	1	75	2019-03-16
3	21 Savage	1.5	1	86	2019-01-05
4	21 Savage	A Lot	23	12	2019-01-05

Table 5.4.: Header of BB-AS Subset

During aggregation the column *weeks_on_chart* got the maximum value and the column *peak_rank* got the minimum value to represent the overall performance of the song. The column *first_appearance* indicates the first time a song appeared in the charts. Based on different separators that were used to list the different artists that were responsible for the song such as *ft.* and *&*, a new column was created to obtain better search results on genius.com as the site often only shows the first artist. The column *billboard_id* shows the id of the songs first occurrence in the billboard charts dataset and *lyrics_id* acts as the primary key of the lyrics table.

Said table was populated with the genius.com data, such as lyrics, annotations, descriptions and comments, as previously described in section 4.3.2. The extended table is referred to as *BB-L*.

5.1.2. Data Cleaning

As cleaning data is an important task in data science, I am emphasizing which problems I came across while gathering the data and how I tackled them.

Fortunately the BB-T100 dataset is an accurate representation of the data on the Billboard website. Multiple randomly selected entries were compared with the source and proofed to be valid. Receiving the right lyrics for BB-AS on the other hand was more complicated.

LyricsGenius (W. Miller 2021) is a great tool to access the genius.com API, but its search functionality is only as good as the search on genius.com and produced many invalid and missing values and there are multiple reasons for that:

1. Artist names differ between Billboard and genius.com. Billboard does not separate the main artist from other contributing artists and instead lists all of them in one name whereas genius.com lists the other artists separately. For example searching for "34+35" by Ariana Grande Feat. Doja Cat & Megan Thee Stallion will return translations and a Spotify New Music Friday list[1]. The correct lyrics can only be found if the contributing artists are not included in the search[2].
2. Explicit titles are censored on billboard but sometimes uncensored on genius.com. The search for "LIGHTSKIN SH*T" by DaBaby does not produce valid results[3] but searching for "LIGHTSKIN SHIT" instead does[4].
3. Sometimes songs are named differently on both platforms for instance "How Bout Now" by Drake as found on genius.com is called "How About Now" in the billboard dataset. That minor difference is enough to cancel out the genius search engine[5]

Even though some of the issues were avoided by separating the artists as stated earlier and removing censoring still many lyrics in BB-L were invalid.

1 https://genius.com/search?q=Ariana%20Grande%20Feat.%20Doja%20Cat%20%26%20Megan%20Thee%20Stallion%2034%2B35

2 https://genius.com/search?q=Ariana%20Grande%2034%2B35

3 https://genius.com/search?q=LIGHTSKIN%20SH*T%20DaBaby

4 https://genius.com/search?q=LIGHTSKIN%20SHIT%20DaBaby

5 https://genius.com/search?q=How%20About%20Now%20Drake

Invalid lyrics were present in three different forms like table 5.5 shows. The procedure that uses the LyricsGenius package automatically generates *!NoSong!* or *!Error!* in the dataset if the search query returns no song or receives an error. The third type of invalid lyrics can not be identified as easily as the others because the search returned lyrics strings that turned out to be wrong when examined.

billboard_id	lyrics_id	artist	song	lyrics
6822	1321	twenty one pilots	Level Of Concern	!Error!
8299	506	J. Cole	the.climb.back	!NoSong!
9707	97	Ariana Grande Feat. Doja Cat & Megan Thee Stallion	34+35	Mmm\n\nPensarás que estoy loca\nPor la forma en que he lo estado deseando\nSi lo pongo bien claro\nSolo dame bebés\nEntonces, ¿qué harás esta noche...

Table 5.5.: Examples for Different Types of Invalid Lyrics

These wrongfully identified lyrics were usually lists, translations or genius.com user listening records. A quick way of identifying wrong lyrics was looking at their URLs. URLs with the correct lyrics always start with the artist name, for example the URL for "34+35" by Ariana Grande start with "Ariana-grande"[6] the URL to the lyrics page that got originally crawled for that song starts with "Genius"[7] instead.

LyricsGenius offers to find lyrics by their URLs or GeniusIDs. So eventually the links and IDs to the missing lyrics were passed to LyricsGenius and the lyrics were exported to the dataset. Lyrics that were scraped this way always end with a number and the string "EmbedShare URLCopyEmbedCopy". Therefore the newly generated lyrics were cleaned using Python-integrated string operations.

5.1.3. Data Construction

Besides the additional column for the first artist, columns with information on the lyrics were added to BB-L. As seen in the code snippet below, columns were created for the string length based on the number of characters in the lyrics and for the number of words.

6 `https://genius.com/Ariana-grande-34-35-lyrics`

7 `https://genius.com/Genius-traducciones-al-espanol-ariana-grande-34-35-remix-ft-doja-cat-and-megan-thee-stallion-traduccion-al-espanol-lyrics`

```
lyrics['length'] = lyrics['lyrics'].apply(lambda string: len(str(string)))
lyrics['word_count'] = lyrics['lyrics'].apply(lambda string:
↪  len(str(string).split()))
```

Listing 5.1.: Lyrics Length Calculation

Thanks to the LanguageDetector package build on spaCy (Balaji 2019), I was able to automatically detect the main language of the individual lyrics (see listing 5.2). I created a function *detect_language* that utilized the package and returned the detected language by the NLP-algorithm and its confidence as a score. This function was applied to generate the columns *language* and *language_score*.

```
import spacy

nlp = spacy.load("en_core_web_lg")
nlp.add_pipe('language_detector')

def detect_language(text, nlp):
    DetectorFactory.seed = 0
    if type(text) == str:
        doc = nlp(text)
        result = doc._.language['language'], doc._.language['score']
    else:
        result = {'-': -1}
    return result
```

Listing 5.2.: Lyrics-Language Detection

All the extended code and data that was used for this project can be found in the GitHub repository[8].

5.1.4. Data Selection & Integration

Only songs that are written in English were considered for the analysis. Therefore, the column *language* was filtered to only show "en". This also filtered out all songs that have no lyrics. Additionally all songs with a language score lower then 0.85 were excluded. This subset is referred to as *BB-L-EN*. The 1,291 songs got reduced to 1,252 after the operation.

8 https://github.com/FelixSiegfriedRiedel/chart-lyrics-analysis

Likewise the same 39 songs were removed from the BB-T100 dataset resulting in a reduction from 10,500 to 10,061 rows. The Billboard dataset for English songs is called *BB-T100-EN*.

5.2. Modeling

As previously stated, the spaCy framework solely was used to process the lyrics. It is very well documented and updated regularly. In spaCy, text is processed using so-called processing pipelines (see figure 5.1). For this research project, an already trained pipeline `english_core_web_lg`[9] is utilized.

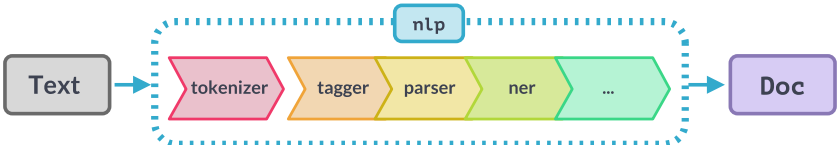


Figure 5.1.: Text Processing in spaCy (ExplosionAI 2021)

After the text is segmented into its underlying words, punctuation marks and symbols in a process that is called *tokenization*, the processing pipeline with its components is initialized. This results in a *doc* object that can be used to access information retrieved by the individual pipeline components. The tagger component for example assigns part-of-speech tags to tokens, such as 'NOUN' for nouns. (ExplosionAI 2021)

It is also possible to create custom pipelines and components. The LanguageDetector used in section 5.1.3 is such a component.

5.2.1. Modeling Techniques

Coronavirus' impact on the lyrics was measured in three different approaches. In section 5.2.2 common words in songs were identified similar to the top word analysis by Barman, Awekar, and Kothari. Additionally COVID-19-specific keywords were specified. The measure of a sentiment change in the lyrics in section 5.2.3 was inspired by the analyzes done by Barman, Awekar, and Kothari and Liu et al. that both were featured in chapter 3. With topic modeling the various topics addressed in the songs were identified.

9 `https://spacy.io/models/en#en_core_web_lg`

5.2.2. Keyword Analysis

For obtaining the keywords used by each song I made use of spaCy's part-of-speech tagging feature. This gave me the ability to only extract words with a certain role and position in a sentence. I decided to only consider nouns, proper nouns, verbs and adjectives as these are the most relevant words to deliver a meaning. Additionally the integrated stop words list was used to filter out common words in the English language as they would be present in most songs and therefore would not act as useful keywords to distinguish songs from each other.

```
import re
def get_keywords(text, nlp):
    text = re.sub(u"""[^\\x00-\\x7F\\x80-\\xFF\\u0100-\\u017F\\u0180-\\u024F
    \\u1E00-\\u1EFF]""", u'', text)
    doc = nlp(text)
    keywords = {}
    pos_tag = ['NOUN', 'ADJ', 'PROPN', 'VERB']
    for token in doc:
        if token.is_stop or token.lemma_ in nlp.Defaults.stop_words or
        ↪ token.is_punct:
            continue

        if token.pos_ in pos_tag:
            keyword = token.lemma_.lower()

            if keyword in keywords.keys():
                keywords[keyword]+=1
            else:
                keywords[keyword] = 1

    return dict(sorted(keywords.items(), key=lambda item: item[1],
    ↪ reverse=True))
```

Listing 5.3.: Keyword Extraction

Inspired by Foong's guide, I created the `get_keywords` function that can be seen in listing 5.3 (Foong 2020). After using regular expressions to filter out non-latin characters, the text was tokenized and the pipeline was initialized. Every token of the text was added to the keywords dictionary if the following conditions are given: Neither the token nor its lemma is a stop word, it is not a punctuation symbol and its part-of-speech tag is either noun, proper noun, verb or adjective. Before a token was added to the dictionary, it was converted into its original form. This process is called lemmatizing, for example *walk* would be the lemma of *walked*. To further avoid redundant entries, all the keywords are turned into lower case. Each keyword in the dictionary has a counter that was raised if the same

keyword is found again. The resulting dictionary was sorted by the keyword count in descending order.

(a) Top 100 Words in 2019

(b) Top 100 Words in 2020

Figure 5.2.: Comparison of top words used in the Billboard Weekly Hot 100 Charts

Analogical to Barman, Awekar, and Kothari, I created word clouds for the most used words in 2019 and 2020 based on the extracted keywords (see figure 5.2). The size of each word is relative to the amount of times it was found in the data. As the figure shows, both years share numerous of words that are favoured by the songwriters. Coronavirus-related keywords were not used frequently enough to be present in the top 100 words. In the appendix (A.5) the frequencies of the first 50 words for 2019 and 2020 can be found.

To examine if songs about the ongoing pandemic are present in the data and which words are used to describe the situation, I used the newly generated keywords. I started with a small list of words consisting of the most obvious words related to the virus and looked up songs that featured such words (*corona*, *coronavirus*, *pandemic*, *quarantine*).

After the first iteration the list was expanded with words that were mentioned in the results. This process was repeated multiple times until the results stayed the same. To gather more realistic results, songs with a release date prior to March 2020 were filtered out. The final list can be seen in listing 5.4 below.

```
covid_keywords = ['antidote', 'contracting', 'corona', 'coronavirus', 'cough',
    'covid', 'distancing', 'flu', 'hazmat', 'hospitable', 'mask', 'pandemic',
    'quarantine', 'sars', 'symptom', 'virus']
```

Listing 5.4.: Array with Keywords Related to Coronavirus

However, some of the 33 songs the keyword search returned did use the words in a different context thus needing additional validation. To further evaluate the songs, coronavirus-related keyword mentions in the songs descriptions and annotations were considered. In contrast to the lyrics the annotations and descriptions were addressing the pandemic more directly and therefore more precise keywords could be used to obtain usable results. Even though the number of keywords was reduced, this process returned more songs, 46 in total. Some songs that were not covered in the data before, due to the artists addressing the virus with unspecific words, were discovered through their descriptions and annotations like the song "Six Feet Apart" written by Luke Combs which lyrics can be seen below.

"[...] Giving hugs and shaking hands
It's a mystery, I suppose
Just how long this thing goes
But there'll be crowds and there'll be shows
And there will be light after dark
Some day when we aren't six feet apart [...]"

Combs 2020

5.2.3. Sentiment Analysis

To haven an impression whether the overall mood changed during the outbreak, the sentiment of each lyrics was retrieved using SpacyTextBlob (Edwardes 2021). The tool created by Edwardes returns polarity and subjectivity of a given text. The polarity can range between -1.0 and 1.0 and determines if the text is rather negative or positive. The subjectivity ranges between 0.0 (very objective) and 1.0 (very subjective).

After adding the component to the spaCy pipeline, it is automatically executed when running the pipeline and its values can be accessed via the `doc` object.

```python
import spacy
from spacytextblob.spacytextblob import SpacyTextBlob

nlp = spacy.load("en_core_web_lg")
nlp.add_pipe("spacytextblob")

line_1 = "Panic on the brain, world has gone insane"
line_2 = "And there will be light after dark"

doc = nlp(line_1)
print(line_1, doc._.polarity)
#Panic on the brain, world has gone insane -1.0

doc = nlp(line_2)
print(line_2, doc._.polarity)
#And there will be light after dark 0.125
```

Listing 5.5.: Sentiment Analysis Example

Two examples for the polarity can be seen in Listing 5.5. I used single lines of two specific songs from the *covid_lyrics* dataset generated by the keyword analysis. The first line by Joseph in "Level Of Concern" has a strong negative polarity (-1.0). The other line written by Combs however has a more positive sentiment (0.125).

Contrary to my assumptions, no obvious decrease in the overall sentiment of the chart songs was observable after the outbreak (figure 5.3). The polarity reached its lowest value of 0.016 in the week of 28 September 2019 and peaked in 12 December 2020. After the outbreak in March 2020 it stayed between 0.050 and 0.075 until it started rising consistently by the end of summer. The average sentiment in the fall months of 2020 was slightly higher compared with 2019.

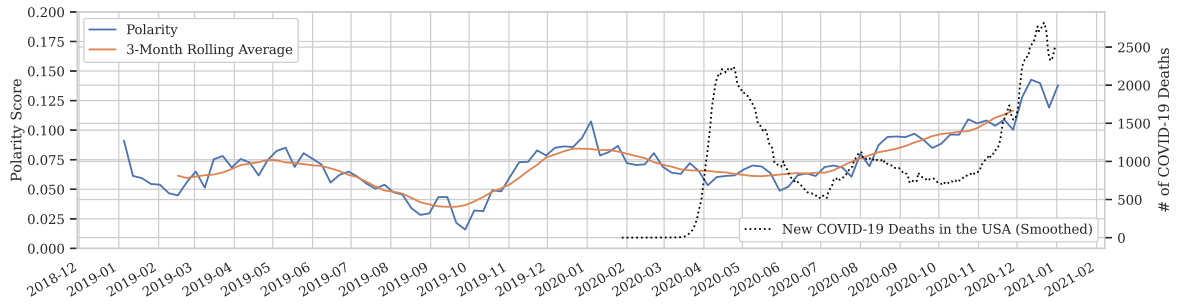


Figure 5.3.: Polarity Trend in Billboard Top 100 Charts in Comparison to Reported COVID-19 Deaths in the U.S.A

After comparing the sentiment to the trend of deaths caused by the virus, it became clear that there was no direct relation between the both.

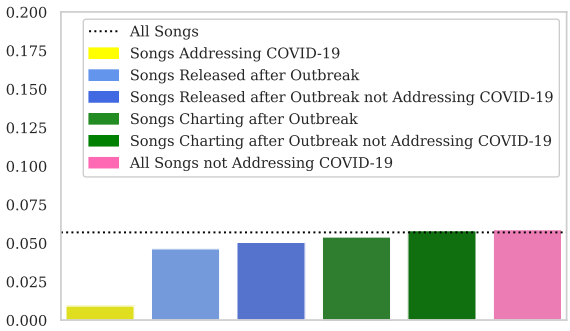


Figure 5.4.: Average Polarity in Billboard Top 100 Charts

Out of 4,220 chart placements after the outbreak in March 2020 only 5.4% are related to the virus. This explains the low influence on the sentiment. As seen in figure 5.4: Coronavirus-specific songs have an average polarity of 0.009 which is 84% lower than the average polarity of all songs charting between 2019 and 2020. These songs reduced the average polarity of songs released after the outbreak by 8% and the average polarity of songs that appeared on the charts after the outbreak by 7%. The data table for the figure can found in the appendix (A.6).

5.2.4. Topic Modeling

To obtain an understanding of the topics present in the music charts, the lyrics were classified using *Latent Dirichlet Allocation* (LDA).

The algorithm that was introduced by Blei, Ng, and Jordan creates word clusters that represent individual topics of a text corpus. For each document in the corpus the affiliation probabilities to the different topics are computed based on term frequencies within the text. (Blei, Ng, and Jordan 2003)

Before training an LDA model, the textual data needs to be transformed. After the text is tokenized, it needs to be converted into the *Bag-of-Words* format. A Bag-of-Words tuple consists of a word-ID (a unique number given to each individual word in the whole corpus) and the frequency of the word with that specific ID in the document. Different parameters of the model can be set depending on the goal of the modeling process, such as number of topics or amount of passes through the

corpus. The resulting topics need to be labeled manually based on the words that are present in the clusters.

While Y. Chen and Lerch retrieved five individual topics using the same algorithm, the thematic clusters for the chart lyrics of 2019 and 2020 were very different. After unsuccessfully experimenting with different parameters of the algorithm, I decided to train the model with a bigger dataset instead. For this new approach a dataset with 147,872 lyrics scraped from AZLyrics.com was used (Suarez 2020). I filtered out non English songs from Suarez' dataset with the language detector and tokenized and lemmatized the lyrics using a similar function as in the keyword analysis. The Gensim library (Řehůřek and Sojka 2010) supplied the framework to train the LDA model.

After training with various parameters, seven topic clusters were returned by the model. The following list shows my interpretations for the topics (bold) and the first 10 most frequent words within the clusters.

1. **Men**: man, big, old, ride, town, work, boy, roll, new
2. **Love**: love, time, feel, way, tell, think, need, thing, life
3. **Explicit Language**: nigga, bitch, fuck, shit, money, hit, ass, hoe, real
4. **Life / Spirituality**: god, die, lord, life, dead, soul, live, kill, blood
5. **Environment**: light, dream, night, eye, run, world, fall, hand, burn
6. **Interjections**: bang, boom, nae, dey, nan, pon, dat, deo, wonderful
7. **Women / Party**: baby, girl, little, tonight, dance, night, like, body, rock

In figure 5.5 the different clusters and their distance to each other is visualised by pyLDAvis. The size of the circles represent the word count in the respected topic and closer topics have more words in common. Judging by the overlapping circles the topic count could have been reduced to 4 but the topics yielded by that adjustment were to broad to be labeled in a logical manner.

The finished model was applied on the Billboard lyrics dataset (BB-L-EN) and the topics and their probability for each song were calculated. A code snippet feauturing the concrete settings of the model is featured in the appendix (B.6). On average five topics were assigned to a song. The topic with the highest probability was treated as the main topic of the piece.

As seen in figure 5.6, the size of the individual topic clusters greatly varies. 543 songs have *Explicit Language* as their top topic rendering it as the most frequent topic in the data which is also noticeable when looking at the keywords (see 5.2.2). On the other hand only 7 songs got *Interjections* assigned to them and more then half of these songs are Christmas songs like jingle bells.

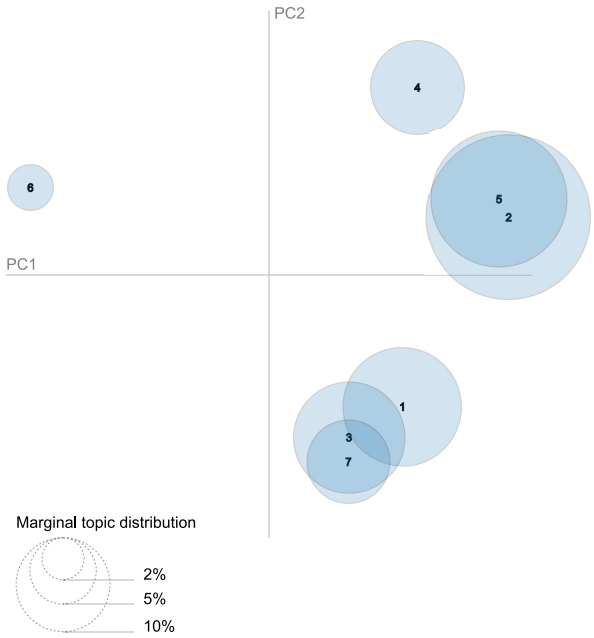


Figure 5.5.: Intertopic Distance Map for the AZLyrics LDA Model (Mabey and Susol 2015)

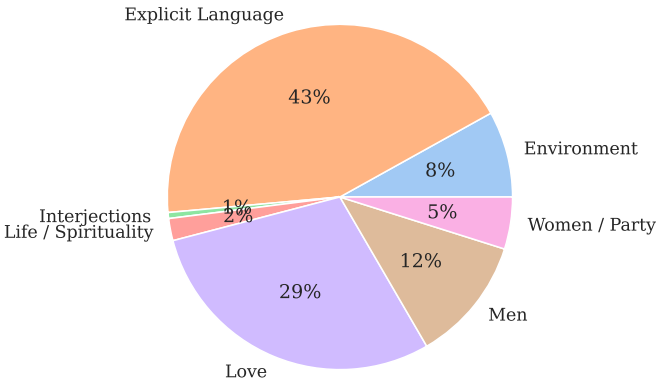
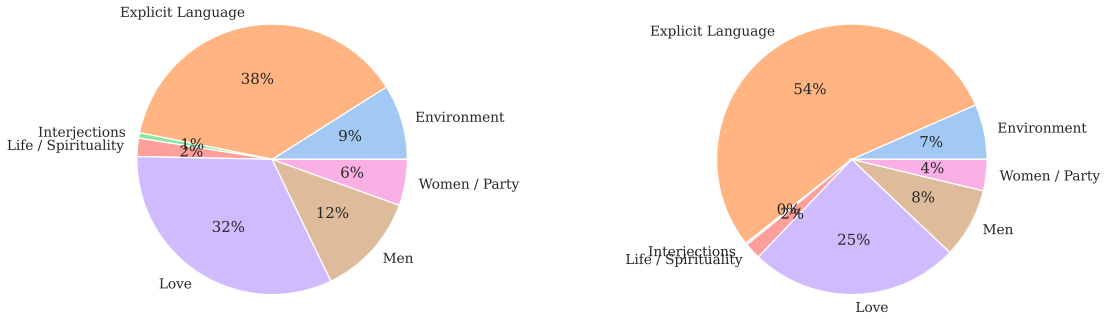


Figure 5.6.: Topic Distribution of Chart Songs (n=1,252)

When comparing the topic distributions of songs released before and after the outbreak as seen in figure 5.7 some notable differences can be seen that could be tied to the outbreak. Songs that were released after

29 February 2020 are 16% more likely to feature explicit language in their lyrics. They are 7% less likely to address love and the topic *Women / Party* also lost 2% of its share.



(a) Before March 2020 (n=779) **(b)** From March 2020 (n=455)

Figure 5.7.: Topic Distribution of Chart Songs Before and After the Outbreak of COVID-19

It needs to be noted that the analysis above only regards songs with a release date. Therefore, eighteen songs were left out.

How the frequency of the topic *Love* changed over the course of the weekly charts can be observed in figure 5.8. Although a decline in love-centered songs was visible in the pie charts before, the change within the US-American music charts is more subtle.

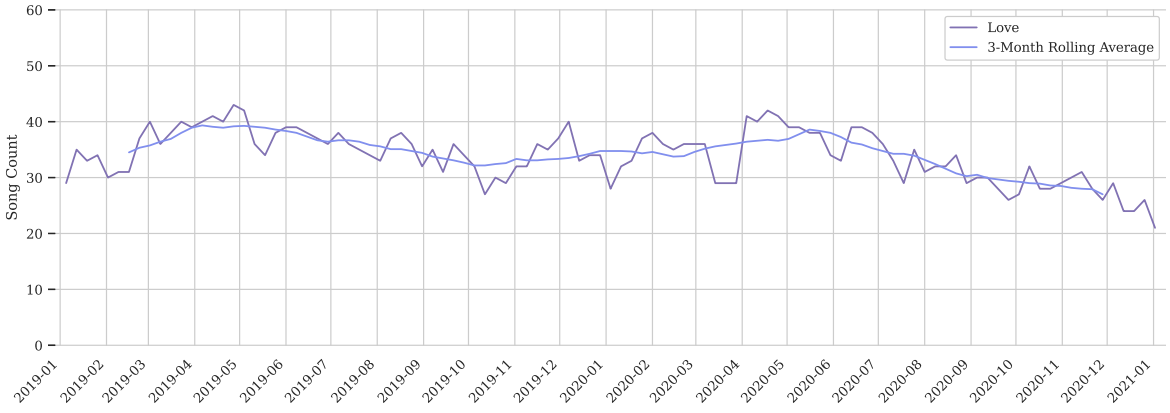


Figure 5.8.: Frequency Trend of Songs with Topic *Love* in Billboard Top 100 Charts

Until September 2020 the song count was averaging between 30 and 40. The most love-songs were released in the week of 27 April 2019 with a song count of 43. After September 2020 it started climbing down below 30 and reached its lowest point of 19 in the first week of 2021. According to the graph a light seasonality can be assumed with the amount of love songs rising until mid of summer and then declining towards the end of the year. See appendix A.7 for the trend of all topics.

6. Discussion

How can songs that address the coronavirus be classified through text mining?

How did the virus influence the US-american music charts?

In this section of the paper I am answering the two earlier proposed research questions and evaluate the results and weaknesses of the project and possible different approaches.

6.1. Interpretation of the Results

The identification of COVID-19-influenced songs was done in two steps. First the keywords of the lyrics, annotations and descriptions on Genius.com had to be extracted. This was achieved by combining different NLP techniques: Part-Of-Speech Tagging, Lemmatization and Filtering. Nouns, proper nouns, verbs and adjectives were converted into their root forms and the *stopwords*, the most common words, of the English language were filtered out. Then a custom list of keywords that are centered around the current pandemic was defined and the keyword results were scanned for these words. The result assumes that 46 of the 1,252 songs address coronavirus. The sentiment and topic analysis did not directly assist in discovering coronavirus songs but provided additional data to determine an influence on the charts.

With 10% of the songs that were released after the outbreak allegedly showing reference to the pandemic, the perceptible influence on the American music charts is insignificant. This also becomes clear when looking at the unexpected small reduction of 8% in the overall sentiment that can be seen in figure 5.4 on page 38. COVID-19-songs have an average polarity of 0.009 that can be interpreted as neither positive nor negative as the possible polarity ranges between -1 (very negative) and 1 (very positive). This close to neutral sentiment score can have multiple explanations: the songs could either have equally positive and negative polarity scores or they are all of neutral nature. With the score ranging between -0.285 and 0.410 and a standard deviation of 0.156 (see table 6.1) the first

explanation is more likely the case. This shows that songs about the pandemic can be of positive and negative nature. When I compared the polarity trend within the charts during 2019 and 2020 to the reported deaths to COVID-19 in the USA it became clear that a direct correlation is unlikely even though one could argue that more uplifting songs were played by the population to cope with the tragic numbers. This sentiment trend is contrary to the results of the study from Liu et al. that was presented in section 3.3 which stated an increase towards negative artist sentiment for rising coronavirus cases.

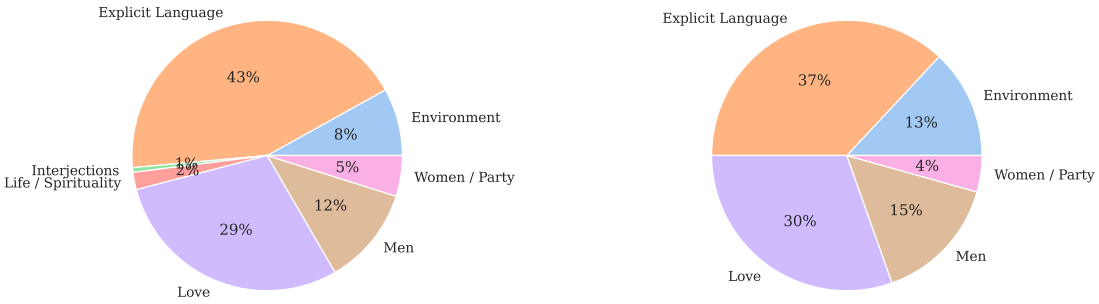
	sentiment_polarity
count	46.000
mean	0.009
std	0.156
min	-0.285
25%	-0.075
50%	-0.011
75%	0.093
max	0.410

Table 6.1.: Descriptive Statistics on the COVID-19-Songs Polarity Data

When solely looking at the topics of the songs released before and after the outbreak in March 2020 without taking into account the amount of times they appeared on the charts, an increase in songs with explicit language by 16% and a decrease in songs about love by 7% is noticeable. This is an indication that the songwriting style changed but it is unclear when exactly it changed as the time frames in which the lyrics were created are not in the data. Time between writing a song and releasing a song is very different for each artist. As the low amount of COVID-19 songs reported earlier suggests, this thematic change could be influenced by many other factors apart from the virus.

The least common topic clusters *Interjections* and *Life / Spirituality* are not present in the coronavirus songs. In general the topic distribution of these songs is very similar to the overall topic distribution (see figure 6.1) with a maximum difference of 6% for each topic.

I expected the artists would need a few months until a song of this type is released and indeed the first coronavirus song "Level Of Concern" was released in 9 April 2020 (Joseph 2020) roughly one month after the outbreak. However, it first appeared on the charts in 24 April 2020. Their average rank of 43 is lower then I assumed meaning these songs do not necessarily rank higher on the charts although 9 of them reached positions on the charts that are higher then 10.



(a) All Songs (n=1,252) **(b)** COVID-19 Songs (n=46)

Figure 6.1.: Topic Distribution of All Chart Songs and COVID-19 Songs

6.2. Errors & Weaknesses

Timing is a big weakness of this research. As hinted earlier, the time between writing and releasing a song is highly variable and many songs which might have been written during 2020 that potentially address the pandemic are yet to be released. I assume famous artists that land their songs on the charts regularly put a considerable amount of effort into the release of their songs which means it may take up to a couple of years until a piece is released and often songs are bundled into albums which take even longer to release. The time frame between March 2020 and January 2021 might have been too small for artists to release their coronavirus songs. As the charts change on a weekly basis, seasonality can be an influence on the themes and sentiment of the charts, therefore analyzing charts from multiple years before and after the outbreak would have been beneficial. This would better illustrate if a change in the polarity score is related to the virus or to the month it was released in. This weakness directly correlates with the size of the dataset. As training the topic model showed, a collection of 1,252 songs is not enough to receive comprehensible theme clusters and instead the model was trained with lyrics from AZLyrics.com which are neither particularly tied to chart songs nor up to date. With more years covered in the data the greater amount of English lyrics could lead to better modeling results.

Selecting and cleaning the data lead to another weakness. I solely relied on the results given by the language detector (Balaji 2019) for filtering out non-English songs. The threshold for the algorithms confidence was set to 0.85 which only excluded three songs. Even though some songs featured large

sections with non-English lyrics they are still included in the data like "Stay" by BTS (Jungkook et al. 2020).

"[...] 가만히 난 주문을 걸어
그 어느 때보다 크게 뛰는 heart
이 순간 우리 언제라도 어디 있대도
Together, wherever, yeah
Wherever, yeah [...]"

Jungkook et al. 2020

The refrain of said song that can be seen above only has a few English words however the language detection interprets it as English with a confidence score of 0.86. A higher threshold would have excluded songs similar to "Stay" and would have improved the overall quality of the data.

During topic analysis only a few of the available parameters were configured. Experimenting with more parameters and different settings could improve the models accuracy. The clusters greatly varied with the size of the dataset therefore working with a training set was not an option thus training the model needed a plethora of computation power and time. A faster PC or a cloud computing system would have enabled a better and more effective training process.

The keyword and topic analysis show that various interjections and profanities are present in the data. Although interjections were particularly filtered out with the Part-Of-Speech tagging functionality of spaCy, some words were falsely identified and were left in. I originally decided to leave in explicit language without expecting it to have such an impact on the topics and keywords. Even though explicit words can have a certain meaning to song, leaving them out might result in a better understanding of the overall themes present in the charts.

Even though the algorithms are very advanced and yielded many promising results in different NLP applications, classifying the songs was more complicated than first expected. A common way of expressing themes in poetry, literature and songwriting is to speak *between the lines* meaning some topics are addressed indirectly by using fitting words. For example Combs used the words *six feet apart* to address the contact restrictions recommended by the American Centers of Disease Control and Prevention (NCIRD 2021). Humans that lived during the pandemic are most likely to understand the reference. The keywords from said song on the other hand are not directly tied to the virus which was a big challenge during the identification part of the research. Extending the target keyword list was not an option as indirect references to the virus lead to more false-positive identifications of

songs. This problem was partly tackled by also extracting keywords from the descriptions and the annotations on Genius.com which often mention the virus when it was relevant to the song. However this approach lead to another problem: the sole dependency of the research on mostly user-generated content on the site. Therefore, if a Genius.com user falsely identified a song, that false information is also present in the data. To give an example: some of the songs were identified as COVID-19-specific because they had the required keywords in their description even though the description just stated that the song was written during the lockdown which does not directly imply the song is relating to the pandemic.

Another issue I want to address is the reproducibility of the project. In later stages of the analysis I decided to also mine the comments, descriptions and annotations from genius.com. As at this point the lyrics were already filtered, I only added the new columns to the English dataset BB-L-EN but I updated the genius.com scraper to always scrape this information for future uses. The Jupyter Notebooks that feature the validation, preparation and selection part of the research were updated as well to work with the new structure of the data and the procedures stayed the same. As the LyricsGenius framework throws timeout errors in different situations on every run, the new data has false lyrics at different locations compared to the original data therefore I would have to do the manual validation again which I decided not to do because I already had a filtered and validated dataset with English lyrics. All the files that are influenced by this have either "_new" or "N-" in the file name and are only used exemplary to show the data preparation, validation and selection process but still include flaws as the manual part was skipped.

Similarly the training process of the LDA model is not reproducible either. No random state was declared and therefore an identical model can not be achieved through training even with the same settings. I exported the model to the project directory and the Jupyter Notebook, that uses the model, retrieves the data from the existing one.

6.3. Outlook & Different Approaches

Apart from improving and fixing the used techniques, the research can be continued in many different ways.

Instead of just looking at the release date or the date a song appeared on the charts, the actual time the lyrics were written could be considered. However obtaining this metric is very complicated because the information usually does only appear in a very unstructured form like in interviews or in song descriptions.

During the data gathering process the comments on Genius.com were also scraped but they were not used for the analysis. In future projects one could analyze these comments and examine how people reacted to certain songs.

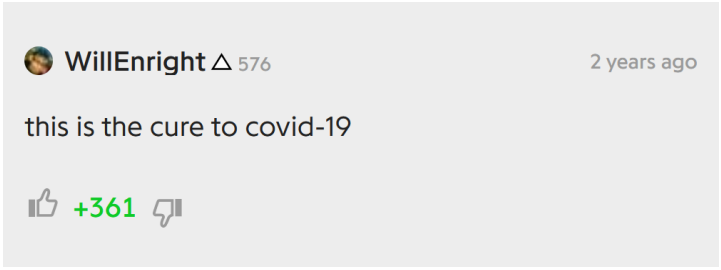


Figure 6.2.: Comment by Genius.com user WillEnright on The Weeknd's song "Too Late" (Tesfaye 2020b)

Some comments like the one seen in figure 6.2 indicate a positive sentiment toward songs released during the pandemic and it could be further analyzed in which context listeners mention the virus and songs that helped the people to cope with the situation might be identified.

Coronavirus influenced lyrics could be detected with the help of ML. Lines extracted from lyrics, newspaper articles and blogs that feature the virus could be collected. These lines could then be labeled and mixed with non-coronavirus lines to create a dataset that can train the classification algorithm which is later applied on unseen data.

The scope of the research could be extended to a global level. English songs are popular in many countries of the world and usually rank high on the local charts, therefore the algorithms could be applied to other countries as well. Although spaCy supports some other languages, lyrics could be translated into English with the help of MT to be suitable for the analysis. A global context gives the opportunity to compare the chart trends and the individual developments of the virus in the countries.

It could be analyzed whether the virus is less present in charts if a country has fewer COVID-19-cases and -deaths. During their analysis, that was featured earlier, Liu et al. used the Difference in Differences approach to reduce falsification. Because only the US-American charts were examined it was not possible to use this technique. By analyzing the charts of more countries said method becomes applicable and other factors untied to the virus that can be subtracted from the sentiment scores as the different countries had the outbreak at different times.

Instead of exclusively focusing on the Billboard charts other charts can be considered for example charts from streaming platforms like Spotify[1] or Apple Music[2]. That way a greater differentiation by city or genre is possible. Genius.com also features their own charts[3]. By using these charts, problems with differing song and artist names that occurred can be avoided because the lyrics are scraped from the same site.

1 https://charts.spotify.com/charts/overview/global

2 https://music.apple.com/browse/top-charts

3 https://genius.com/#top-songs

7. Conclusion

With its origin in Cryptography and Linguistics in the 40s and 50s of the 20th century NLP heavily changed in the past 70 years. As a combination of Mathematics, Information Technology and Linguistics it has a great variety of underlying techniques that are used to interpret natural language and especially AI and ML had a big influence on recent NLP advances.

Apart from applications like voice assistants, translators and chat bots it is used to extract and interpret information from texts.

This paper presented different practical examples of NLP within the context of music. With GPT-2 the website builder Zyro was able to create a tool that can generate lyrics similar to songs by Kendrick Lamar, Taylor Swift, or The Beatles (Kulp 2020). Y. Chen and Lerch developed a system that can create texts which syllables match a given melody. Other researchers used algorithms to examine the songwriting-style (Barman, Awekar, and Kothari 2019) and Liu et al. specifically analyzed if coronavirus caused a change in the listening habits of Last.fm users.

To showcase the functionalities of Text Mining – which is often used as a synonym for NLP – the lyrics of songs that appeared on Billboard Weekly Hot 100 charts between January 2019 and January 2021 were mined. The research was structured using the CRISP-DM model and apart from the actual modeling process the preceding steps were shown similarly to traditional Data Science projects. The goal was to extract themes and moods from songs and especially to identify whether the ongoing Coronavirus had an influence on songwriting and listening habits.

The language of the lyrics that were scraped from Genius.com using the LyricsGenius framework (W. Miller 2021) was automatically determined and all non-English songs or songs without lyrics were filtered out. Two complementary datasets were created during the process: BB-L-EN includes Genius.com data for the 1,252 English songs that appeared on the charts and BB-T100-EN includes the chart information gathered from Billboard for each of the songs.

For the analysis multiple techniques were utilized that are dependent on each other. During the keyword analysis I made use of spaCy's Part-Of-Speech tagging feature that allowed me to collect

the most frequent nouns, proper nouns, adjectives and verbs that are present in the lyrics, song descriptions and annotations. The comparison of the keywords of 2019 and 2020 in figure 7.1 was rather underwhelming and gave no hint towards an outbreak of a pandemic and in general both years had numerous words in common that were favoured by the songwriters. However the analysis gave the template to identify 46 songs that are related to COVID-19.

(a) Top 100 Words in 2019

(b) Top 100 Words in 2020

Figure 7.1.: Comparison of top words used in the Billboard Weekly Hot 100 Charts

The analysis of the sentiment was carried by a spaCy pipeline component that calculates two scores which indicate the polarity and the objectivity of a text whenever the pipeline is called. The score was plotted for every week of the charts and I assumed the polarity score would go down after the outbreak in March but instead it was rather consistent and even overtook the highest score of 2019 in the fall and winter months of 2020. When compared to the reported deaths to the virus in the U.S.A, no correlation was observable as seen in figure 7.2.

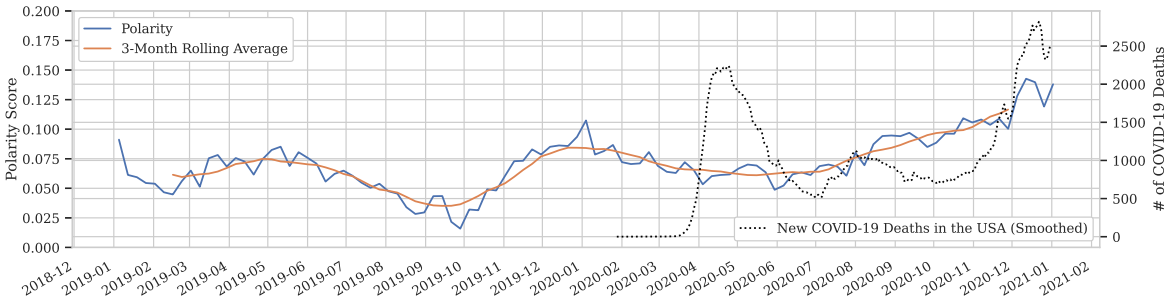
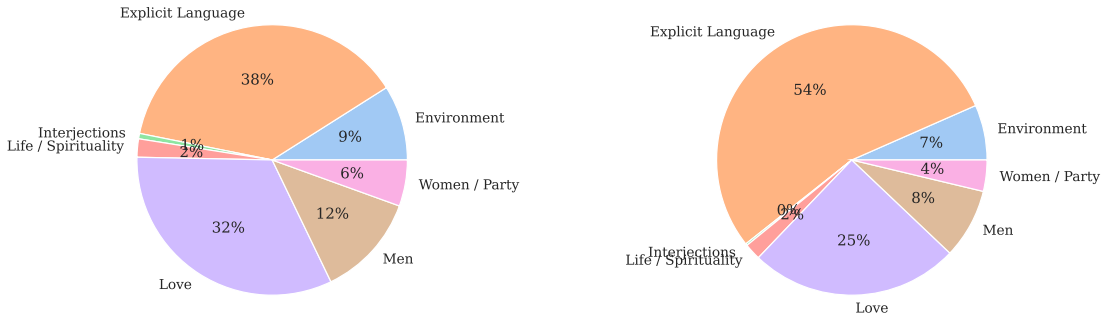


Figure 7.2.: Polarity Trend in Billboard Top 100 Charts in Comparison to Reported COVID-19 Deaths in the USA

The average polarity score of 0.009 of COVID-19-songs is 84% lower than the average of all songs. The score is so close to neutral because these songs have a high standard deviation. With only 10% of the songs that released after the outbreak being Coronavirus-songs the influence on the sentiment in the charts was smaller than expected.

For the topic modeling process the LDA algorithm was used. The model that was trained with more than 140,000 lyrics from AZLyrics.com yielded seven different topic clusters that were applied on the chart songs lyrics. The process resulted in an unequal topic distribution with 43% of the songs having Explicit Language as their top topic and 29% Love but only 2% of the songs are about life and spirituality. Before the outbreak of the virus the artists were 7% more likely to feature love and and 16% less likely to use explicit language in their songs compared to after the outbreak (see Figure 7.3).



(a) Before March 2020 (n=779) **(b)** From March 2020 (n=455)

Figure 7.3.: Topic Distribution of Chart Songs Before and After the Outbreak of COVID-19

In addition to the direct comparison the topic Love was plotted over the course of weekly charts of 2019 and 2020. This showed that the frequency of songs with that topic started declining with the end of summer in 2020.

Although great advances in the field were made, NLP still has its weaknesses that were partly brought to light with this thesis. Subjective tasks like determining a texts sentiment or topic can be done with algorithms but not yet to a level a human would interpret these texts. Especially in creative writing like it is present in song lyrics many phrases have different meanings than the actual words and lyrical means like irony stay undetected.

For projects, that are tied to certain events, it is very important to consider the timing for gathering and analyzing data. It was hard to interpret when exactly a song was written and even if it was released months after the outbreak it could have been written years ago. Considerable attention needs to be put towards cleaning and selecting the data and it is beneficial to determine high thresholds for certain values like the confidence score of an algorithm.

With Coronavirus being part of the daily lives of people in 2020 and later years many fields are influenced by it and address and adapt to the situation. As only the most successful songs manage to be part of the charts, it takes a while until a global event like the outbreak of a virus is present. Already a slight influence is noticeable that will be more concise in the coming years. Even though COVID-19-specific songs reduced the overall sentiment and they can also carry positive messages like love.

Bibliography

[1] P. Hancox: *A brief history of Natural Language Processing*. https://www.cs.bham.ac.uk/~pjh/sem1a5/pt1/pt1_history.html. (Accessed: 17 April 2021). 1996.

[2] P.-H. Chen: "Essential Elements of Natural Language Processing: What the Radiologist Should Know". In: *Academic Radiology* 27.1 (2020). Special Issue: Artificial Intelligence, pp. 6–12. ISSN: 1076-6332. DOI: https://doi.org/10.1016/j.acra.2019.08.010. URL: https://www.sciencedirect.com/science/article/pii/S1076633219304179.

[3] P. Kulp: "This AI Songwriting Platform Can Imitate Kendrick Lamar or Taylor Swift". In: *ADWEEK* (2020). URL: https://www.adweek.com/creativity/ai-songwriting-platform-imitate-music-stars-zyro/.

[4] M. Liu et al.: "PANDEMICS, MUSIC, AND COLLECTIVE SENTIMENT: EVIDENCE FROM THE OUTBREAK OF COVID-19". In: *International Society for Music Information Retrieval Conference 2020*. Oct. 2020.

[5] E. D. Liddy: "Natural Language Processing". In: *Encyclopedia of Library and Information Science*. 2nd ed. New York, NY: Marcel Decker, Inc., 2001. ISBN: 978-0824720759.

[6] K. Cohen: "Biomedical Natural Language Processing and Text Mining". In: Dec. 2014, pp. 141–177. ISBN: 9780124016781. DOI: https://doi.org/10.1016/B978-0-12-401678-1.00006-3.

[7] Y. R. Chao and G. Zipf: "Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology". In: *Language* 26 (1949), p. 394.

[8] W. N. Locke and A. D. Booth: *Machine Translation of Languages - Fourteen Essays*. Cambridge, Massachusetts and New York, NY: Technology Press of the Massachusetts Institute of Technology and John Wiley & Sons, Inc., 1955. ISBN: 978-0837184340.

[9] N. Chomsky: *Syntactic Structures*. 2nd ed. Berlin: Walter de Gruyter, 2002. ISBN: 978-3-110-17279-9.

[10] J. Weizenbaum: "ELIZA–A computer program for the study of natural language communication between man and machine". In: *Communications of the ACM* 9.1 (1966), pp. 36–45. DOI: 10.1145/365153.365168. URL: https://doi.org/10.1145/365153.365168.

[11] *Language and Machines: Computers in Translation and Linguistics - A Report by the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council*. Tech. rep. Washington, D. C.: National Academy of Sciences and National Research Council, 1966.

[12] T. Winograd: "Understanding natural language". In: *Cognitive Psychology* 3.1 (1972), pp. 1–191. ISSN: 0010-0285. DOI: https://doi.org/10.1016/0010-0285(72)90002-3. URL: https://www.sciencedirect.com/science/article/pii/0010028572900023.

[13] K. R. McKeown: "Discourse strategies for generating natural-language text". In: *Artificial Intelligence* 27.1 (1985), pp. 1–41. ISSN: 0004-3702. DOI: https://doi.org/10.1016/0004-3702(85)90082-7. URL: https://www.sciencedirect.com/science/article/pii/0004370285900827.

[14] S. A. Thompson and W. C. Mann: "Rhetorical structure theory: A framework for the analysis of texts". In: *IPrA Papers in Pragmatics* 1.1 (1987), pp. 79–105. ISSN: 2406-419x. DOI: https://doi.org/10.1075/iprapip.1.1.03tho. URL: https://www.jbe-platform.com/content/journals/10.1075/iprapip.1.1.03tho.

[15] P. Johri et al.: "Natural Language Processing: History, Evolution, Application, and Future Work". In: *Proceedings of 3rd International Conference on Computing Informatics and Networks*. Jan. 2021, pp. 365–375. ISBN: 978-981-15-9712-1. DOI: https://doi.org/10.1007/978-981-15-9712-1_31.

[16] A. Louis: *A Brief History of Natural Language Processing — Part 2*. https://medium.com/@antoine.louis/a-brief-history-of-natural-language-processing-part-2-f5e575e8e37. (Accessed: 17 April 2021). July 2020.

[17] Y. Bengio et al.: "A Neural Probabilistic Language Model". In: *J. Mach. Learn. Res.* 3.null (Mar. 2003), pp. 1137–1155. ISSN: 1532-4435.

[18] D. Ferrucci: "Introduction to 'This is Watson'". In: *IBM Journal of Research and Development* 56 (May 2012), 1:1–1:15. DOI: https://doi.org/10.1147/JRD.2012.2184356.

[19] R. Collobert and J. Weston: "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning". In: *Proceedings of the 25th International Conference on Machine Learning*. ICML '08. Helsinki, Finland: Association for Computing Machinery, 2008, pp. 160–167. ISBN: 9781605582054. DOI: 10.1145/1390156.1390177. URL: https://doi.org/10.1145/1390156.1390177.

[20] T. Mikolov, I. Sutskever, et al.: "Distributed Representations of Words and Phrases and their Compositionality". In: *Neural and Information Processing System (NIPS)*. 2013. URL: https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

[21] T. Mikolov, K. Chen, et al.: "Efficient Estimation of Word Representations in Vector Space". In: *ICLR*. 2013.

[22] I. Sutskever et al.: "Sequence to Sequence Learning with Neural Networks". In: NIPS'14. Montreal, Canada: MIT Press, 2014, pp. 3104–3112.

[23] Y. Wu et al.: "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation". In: *CoRR* abs/1609.08144 (2016). URL: http://arxiv.org/abs/1609.08144.

[24] A. M. Dai and Q. V. Le: "Semi-supervised sequence learning". In: *Advances in neural information processing systems* 28 (2015), pp. 3079–3087.

[25] J. Howard and S. Ruder: "Universal language model fine-tuning for text classification". In: *arXiv preprint arXiv:1801.06146* (2018).

[26] M. Honnibal: *Introducing spaCy*. https://explosion.ai/blog/introducing-spacy. (Accessed: 16 July 2021). Feb. 2015.

[27] C. Sammut and G. I. Webb: *Encyclopedia of Machine Learning*. 1st ed. New York, NY: Springer Publishing Company, Incorporated, 2011. ISBN: 978-0-387-30768-8.

[28] M. Bahja: "Natural language processing applications in business". In: *E-Business-Higher Education and Intelligence Applications*. IntechOpen, 2020.

[29] T. Brown et al.: "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[30] Y. Chen and A. Lerch: "Melody-Conditioned Lyrics Generation with SeqGANs". In: Dec. 2020, pp. 189–196. DOI: 10.1109/ISM.2020.00040.

[31] L. Yu et al.: "SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient". In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. AAAI'17. San Francisco, California, USA: AAAI Press, 2017, pp. 2852–2858.

[32] D. Jurafsky and J. H. Martin: "Speech and Language Processing - An Introduction to Natural Language Processing,Computational Linguistics, and Speech Recognition". Third Edition draft. 2020.

[33] D. M. Blei et al.: "Latent Dirichlet Allocation". In: *J. Mach. Learn. Res.* 3.null (Mar. 2003), pp. 993–1022. ISSN: 1532-4435.

[34] M. Barman et al.: *Decoding the Style and Bias of Song Lyrics*. July 2019.

[35] A. Caliskan et al.: "Semantics derived automatically from language corpora contain human-like biases". In: *Science* 356.6334 (2017), pp. 183–186. DOI: 10.1126/science.aal4230.

[36] A. G. Greenwald et al.: "Measuring individual differences in implicit cognition: the implicit association test." In: *Journal of personality and social psychology* 74.6 (1998), p. 1464.

[37] E. Zangerle et al.: "Leveraging Affective Hashtags for Ranking Music Recommendations". In: *IEEE Transactions on Affective Computing* 12.1 (2021), pp. 78–91. DOI: 10.1109/TAFFC.2018.2846596.

[38] M. Taboada et al.: "Lexicon-Based Methods for Sentiment Analysis". In: *Computational Linguistics* 37.2 (June 2011), pp. 267–307. ISSN: 0891-2017. DOI: 10.1162/COLI_a_00049. eprint: https://direct.mit.edu/coli/article-pdf/37/2/267/1798865/coli_a_00049.pdf. URL: https://doi.org/10.1162/COLI%5C_a%5C_00049.

[39] J. Angrist and J.-S. Pischke: "Mostly Harmless Econometrics: An Empiricist's Companion". In: Jan. 2009. ISBN: 9780691120348 (hardcover : alk. paper).

[40] C. Shearer: "The CRISP-DM model: the new blueprint for data mining". In: *Journal of data warehousing* 5.4 (2000), pp. 13–22.

[41] J. Ma: "Coronavirus: China's first confirmed Covid-19 case traced back to November 17". In: *South China Morning Post* (Mar. 13, 2020). URL: https://www.scmp.com/news/china/society/article/3074991/coronavirus-chinas-first-confirmed-covid-19-case-traced-back?module=perpetual_scroll&pgtype=article&campaign=3074991 (visited on 08/06/2021).

[42] N. Salari et al.: "Prevalence of stress, anxiety, depression among the general population during the COVID-19 pandemic: A systematic review and meta-analysis". In: *Globalization and Health* 16 (July 2020), pp. 1–11. DOI: 10.1186/s12992-020-00589-w.

[43] Billboard: *The Hot 100 – The Week of July 25, 2020*. https://www.billboard.com/charts/hot-100/2020-07-25. (Accessed: 18 August 2021). 2020.

[44] Billboard: *Billboard Charts Legend*. https://www.billboard.com/p/billboard-charts-legend. (Accessed: 19 August 2021). 2021.

[45] ExplosionAI: *Language Processing Pipelines*. https://spacy.io/usage/processing-pipelines. (Accessed: 25 September 2021). 2021.

[46] N. W. Foong: *Extract Keywords Using spaCy in Python*. https://betterprogramming.pub/extract-keywords-using-spacy-in-python-4a8415478fbf. (Accessed: 25 September 2021). Jan. 2020.

[47] NCIRD: *How COVID-19 Spreads*. https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/how-covid-spreads.html. (Accessed: 24 November 2021). July 2021.

Datasets

[48] D. Guzmán: *hot 100 Songs Weekly 2010-2020*. https://www.kaggle.com/diegoguzmn/hot-100-songs-weekly-20192020. (Last Update: 30 November 2020, Accessed: 18 March 2021). 2020.

[49] F. S. Riedel: *Billboard Hot 100 Weekly Charts 2020*. https://www.kaggle.com/felixsiegfriedriedel/billboard-hot-100-weekly-charts-2020. (Last Update: 18 March 2021, Accessed: 18 March 2021). 2021.

[50] T. Bertin-Mahieux et al.: "The Million Song Dataset". In: *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*. 2011.

[51] M. Schedl: "The LFM-1b Dataset for Music Retrieval and Recommendation". In: *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*. ICMR '16. New York, New York, USA: Association for Computing Machinery, 2016, pp. 103–110. ISBN: 9781450343596. DOI: 10.1145/2911996.2912004. URL: https://doi.org/10.1145/2911996.2912004.

[52] H. Ritchie et al.: "Coronavirus Pandemic (COVID-19)". In: *Our World in Data* (2020). URL: https://ourworldindata.org/coronavirus.

[53] A. Suarez: *AZLyrics song lyrics*. https://www.kaggle.com/albertsuarez/azlyrics. (Last Update: 14 May 2020, Accessed: 11 November 2021). 2020.

Tools

[54] *Genius API*. https://docs.genius.com/. (Accessed: 29 July 2021). 2021.

[55] J. W. Miller: *LyricsGenius*. https://lyricsgenius.readthedocs.io/en/master/. (Last Update: 17 April 2021, Accessed: 29 July 2021). 2021.

[56] A. Balaji: *LanguageDetector*. https://spacy.io/universe/project/spacy-langdetect. (Last Update: 2 May 2019, Accessed: 01 July 2021). 2019.

[57] S. Edwardes: *SpacyTextBlob*. https://spacy.io/universe/project/spacy-textblob. (Last Update: 5 May 2021, Accessed: 20 October 2021). 2021.

[58] R. Řehůřek and P. Sojka: "Software Framework for Topic Modelling with Large Corpora". English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. http://is.muni.cz/publication/884893/en. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[59] B. Mabey and M. Susol: *pyLDAvis*. https://pypi.org/project/pyLDAvis/. (Last Update: 24 March 2021, Accessed: 3 November 2021). 2015.

Lyrics

[60] A. M. Tesfaye: *Blinding Lights*. https://genius.com/The-weeknd-blinding-lights-lyrics. (Accessed: 19 August 2021). 2020.

[61] L. Combs: *Six Feet Apart*. https://genius.com/Luke-combs-six-feet-apart-lyrics. (Accessed: 14 October 2021). 2020.

[62] T. Joseph: *Level of Concern*. https://genius.com/Twenty-one-pilots-level-of-concern-lyrics. (Accessed: 21 October 2021). 2020.

[63] J. Jungkook et al.: *Stay*. https://genius.com/Bts-stay-lyrics. (Accessed: 30 November 2021). 2020.

[64] A. M. Tesfaye: *Too Late*. https://genius.com/The-weeknd-too-late-lyrics. (Accessed: 03 December 2021). 2020.

Figures

Tables

Listings

Appendix

A. Tables, Images, Examples

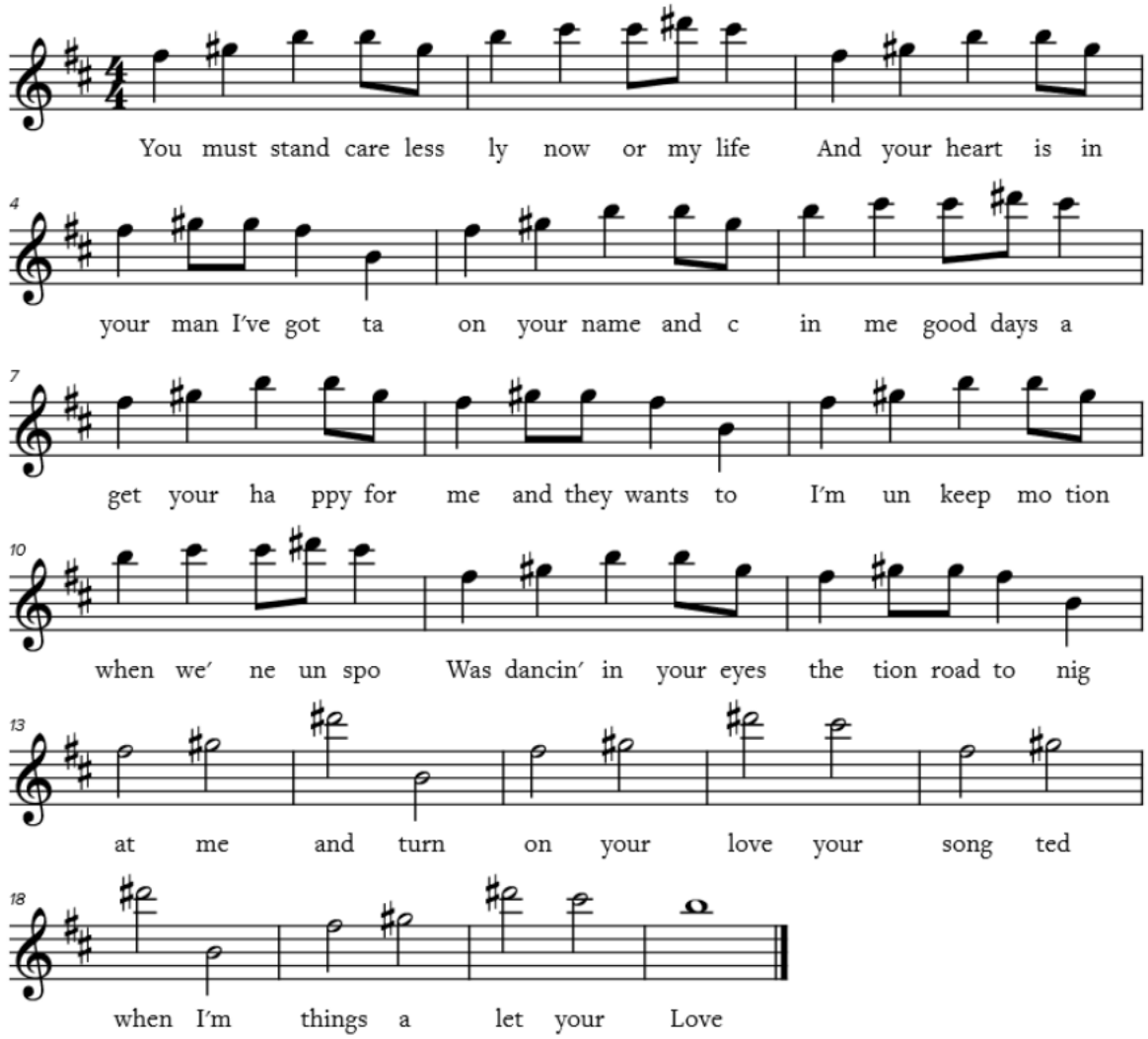
A.1. AI-Generated Lyrics Inspired by The Beatles

I like the way it doesn't matter where you turn
So if you choose to be free
Every step you take is a gift and not a command
And if you choose to be lonely
that's okay because it's not a sin
Love is all we need at this time
So keep the faith, boy
Love is all we need, all we need
All we need is love, all we need
I love you, all we need, all we need
All we need is love, all we need

The lyrics above were generated by Zyro's AI Music Festival Tool[1]

1 https://zyro.com/ai-festival

A.2. AI-Generated Lyrics Based on Melody Changes



A.3. Head of BB-T100-EN

To increase readability, the table was transposed.

	0	**1**	**2**
billboard_id	0	1	2
lyrics_id	0	1	2
song	Thank U, Next	Without Me	All I Want For Christmas Is You
artist	Ariana Grande	Halsey	Mariah Carey
peak_rank	1	2	3
rank_last_week	1	2	7
weeks_on_chart	8	12	30
rank	1	2	3
date	2019-01-05 00:00:00	2019-01-05 00:00:00	2019-01-05 00:00:00
first_appearance	2019-01-05	2019-01-05	2019-01-05
language	en	en	en
language_score	0.999997	0.999995	0.999996

A.4. Head of BB-L-EN

To increase readability, the table was transposed and entries were shortened.

	0	1	2
billboard_id	0	1	2
lyrics_id	0	1	2
artist	Ariana Grande	Halsey	Mariah Carey
first_artist	Ariana Grande	Halsey	Mariah Carey
song	Thank U, Next	Without Me	All I Want For Christmas Is You
weeks_on_chart	28.0	52.0	43.0
peak_rank	1.0	1.0	1.0
genius_id	4063065	3977187	204233
lyrics	Thought I'd end up with Sean\r\nBut he wasn't ...	Found you when your heart was broke\r\nI fille...	I don't want a lot for Christmas\r\nThere is j...
url	https://genius.com/Ariana-grande-thank-u-next-...	https://genius.com/Halsey-without-me-lyrics	https://genius.com/Mariah-carey-all-i-want-for...
length	2409	2095	1918
word_count	460	435	388
language	en	en	en
language_score	0.999997	0.999995	0.999996
first_appearance	2019-01-05 00:00:00	2019-01-05 00:00:00	2019-01-05 00:00:00
genius_primary_artist	Ariana Grande	Halsey	Mariah Carey
genius_description	On the lead single and titular track to her fi...	"Without Me" is the first new song released by...	"All I Want For Christmas Is You" is an uptemp...
genius_annotations	[(One taught me love\n One taught me patience\...	[(Gave love 'bout a hundred tries (Hundred tri...	[(I don't need to hang my stocking\n There upo...
genius_comments	[The Mac shoutout has me fully in tears, this ...	[The queen is ready to snatch our wigs once ag...	[i really like this song, it's about that time...
release_date	2018-11-03 00:00:00	2018-10-04 00:00:00	1994-11-01 00:00:00
annotation_ids	['15720075', '15720076', '15720054', '15720247...	['15517989', '15520369', '15518283', '15518820...	['8393500', '8393500', '21611023', '8393500', ...

A.5. Top 50 Keywords of the Billboard Weekly Hot 100 Charts

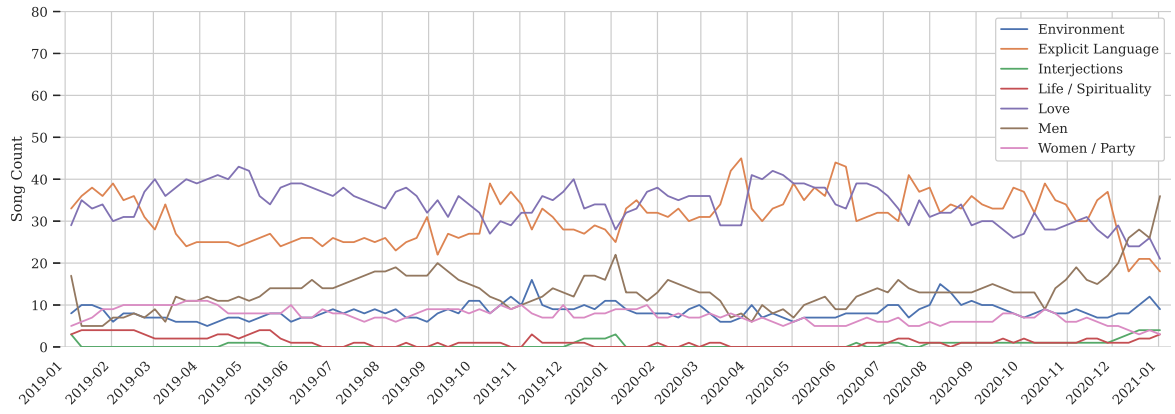
2019	
keyword	**count**
nigga	30
love	27
know	23
want	20
bitch	17
let	11
wanna	10
need	9
time	9
girl	9
baby	8
shit	8
bad	7
la	7
fuck	7
god	6
good	6
ayy	6
come	6
feel	5
tell	5
think	5
christmas	4
run	4
talk	4
dance	4
da	4
light	4
mind	4
look	4
man	3
body	3
woo	3
heart	3
find	3
thing	3
night	3
straight	3
real	3
check	3
summer	2
arm	2
jingle	2
rich	2
beautiful	2
pray	2
jesus	2
face	2
head	2
hate	2

2020	
keyword	**count**
know	59
nigga	51
bitch	30
love	27
want	18
ayy	13
time	12
girl	10
shit	10
la	10
let	9
wanna	9
come	9
baby	9
woo	6
run	6
night	5
way	5
need	5
look	5
tell	5
man	5
pussy	5
christmas	5
good	5
bad	4
think	4
shoot	4
da	4
hope	4
feel	4
fuck	4
life	4
big	4
hit	4
thing	3
country	3
mm	3
happy	3
god	3
leave	3
stay	3
money	3
mad	3
hold	3
like	3
lil	3
gang	3
hot	3
young	3

A.6. Average Sentiment Data Table

	Polarity Score
Songs Addressing COVID-19	0.009
Songs Released after Outbreak	0.046
Songs Released after Outbreak not Addressing COVID-19	0.05
Songs Charting after Outbreak	0.054
Songs Charting after Outbreak not Addressing COVID-19	0.058
All Songs not Addressing COVID-19	0.059
All Songs	0.057

A.7. Frequency Trend of All Topics



B. Source Code & Environment

B.1. PC Specs

Processor	AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx 2.10 GHz
Installed memory (RAM)	8.00 GB (5.91 GB useable)
System type	64-Bit Operating System, x64-based Processor
Windows edition	Windows 10 Pro
Version	20H2

B.2. Execution Order

All data and results are included in the repository which means running the procedures is unnecessary. See the execution order below to recreate the research process. The steps 1, 4 and 6 request data from the web and are very time consuming.

1. **top100_spider.py**
    - crawls Billboard Weekly Hot 100 Charts data
2. **billboard_weekly_100.ipynb**
    - consolidates G19-BB and R20-BB into BB-T100
3. **artist_song.ipynb**
    - groups unique songs and artists and removes censored words
    - generates BB-AS
4. **get_genius_lyrics.py get_genius_resources_v2.py**
    - utilizes LyricsGenius

- gets lyrics, comments, annotations, descriptions, urls, ids, release dates from genius.com
- generates BB-L_raw

5. **lyrics_validation.ipynb**
   - validates lyrics
   - exports invalid lyrics with manually added GeniusIDs
6. **get_genius_resources_v2.py**
   - updates invalid genius.com data
7. **lyrics_preparation.ipynb**
   - adds updated data to BB-L_raw
   - determines language with LanguageDetector
   - changes shape and sorting
   - generates BB-L
8. **data_selection.ipynb**
   - filters English songs
   - generates BB-T100-EN and BB-L-EN
9. **keywords.ipynb**
   - gets keywords
   - determines COVID-19 songs
   - generates wordclouds for most used words
10. **sentiment.ipynb**
    - calculates and plots sentiment scores for lyrics
11. **topics.ipynb**
    - trains LDA model with AZLyrics data
    - uses model to determine topics for lyrics
    - plots topic distributions and frequency trends

B.3. Installed Python Packages

name	version	name	version	name	version	name	version
anyio	3.3.0	ipykernel	5.3.4	pandas	1.3.2	requests-unixsocket	0.2.0
argon2-cffi	20.1.0	ipython	7.19.0	pandocfilters	1.4.3	scipy	1.5.4
async-generator	1.10	ipython-genutils	0.2.0	parsel	1.6.0	Scrapy	2.5.0
atomicwrites	1.4.0	ipywidgets	7.5.1	parso	0.8.2	seaborn	0.11.0
attrs	21.2.0	itemadapter	0.3.0	pathy	0.6.0	Send2Trash	1.8.0
Automat	20.2.0	itemloaders	1.0.4	pickleshare	0.7.5	service-identity	21.1.0
backcall	0.2.0	jedi	0.18.0	Pillow	8.3.1	setuptools	57.4.0
beautifulsoup4	4.9.3	Jinja2	3.0.1	pip	21.2.4	six	1.16.0
bleach	4.0.0	jmespath	0.10.0	pluggy	0.13.1	smart-open	5.2.0
blis	0.7.4	jsonschema	3.2.0	preshed	3.0.5	smmap	4.0.0
catalogue	2.0.5	jupyter	1.0.0	priority	1.3.0	sniffio	1.2.0
certifi	2021.5.30	jupyter-client	6.2.0	prometheus-client	0.11.0	soupsieve	2.2.1
cffi	1.14.6	jupyter-console	6.2.0	prompt-toolkit	3.0.19	spacy	3.1.1
chardet	4.0.0	jupyter-core	4.7.1	Protego	0.1.16	spacy-langdetect	0.1.2
charset-normalizer	2.0.4	jupyter-server	1.10.2	py	1.10.0	spacy-legacy	3.0.8
click	7.1.2	jupyter-server-mathjax	0.2.3	pyasn1	0.4.8	srsly	2.4.1
colorama	0.4.4	jupyterlab-pygments	0.1.2	pyasn1-modules	0.2.8	terminado	0.11.1
constantly	15.1.0	kiwisolver	1.3.1	pycparser	2.20	testpath	0.5.0
cryptography	3.4.7	langdetect	1.0.7	pydantic	1.8.2	thinc	8.0.8
cssselect	1.1.0	lxml	4.6.3	PyDispatcher	2.0.5	toml	0.10.2
cycler	0.10.0	lyricsgenius	3.0.1	Pygments	2.10.0	tornado	6.1
cymem	2.0.5	markdown-it-py	1.1.0	PyHamcrest	2.0.2	tqdm	4.62.1
decorator	5.0.9	MarkupSafe	2.0.1	pyOpenSSL	20.0.1	traitlets	5.0.5
defusedxml	0.7.1	matplotlib	3.4.3	pyparsing	2.4.7	Twisted	20.3.0
en-core-web-sm	3.1.0	mdit-py-plugins	0.2.8	pyrsistent	0.18.0	typer	0.3.2
entrypoints	0.3	mistune	0.8.4	pytest	6.2.4	typing-extensions	3.10.0.0
et-xmlfile	1.1.0	murmurhash	1.0.5	python-dateutil	2.8.2	urllib3	1.26.6
gitdb	4.0.7	nbclient	0.5.1	pytz	2021.1	w3lib	1.22.0
GitPython	3.1.18	nbconvert	6.0.7	pywin32	301	wasabi	0.8.2
h2	3.2.0	nbdime	3.1.0	pywinpty	1.1.3	wcwidth	0.2.5
hpack	3.0.0	nbformat	5.0.8	PyYAML	5.4.1	webencodings	0.5.1
hyperframe	5.2.0	nest-asyncio	1.5.1	pyzmq	22.2.1	websocket-client	1.2.1
hyperlink	21.0.0	notebook	6.1.5	qtconsole	5.0.1	widgetsnbextension	3.5.1
idna	3.2	numpy	1.21.2	QtPy	1.10.0	zope.interface	5.4.0
incremental	21.3.0	openpyxl	3.0.7	queuelib	1.6.1		
iniconfig	1.1.1	packaging	21.0	requests	2.26.0		

B.4. Billboard Weekly Hot 100 Scraper

```
import scrapy

class Top100ListSpider(scrapy.Spider):
    name = 'top100'
    start_urls = ['https://www.billboard.com/charts/hot-100/2020-01-04']
    week_dates = ['2020-01-04', '2020-01-11', '2020-01-18', '2020-01-25',
                  '2020-02-01', '2020-02-08', '2020-02-15', '2020-02-22',
                  '2020-02-29', '2020-03-07', '2020-03-14', '2020-03-21',
                  '2020-03-28', '2020-04-04', '2020-04-11', '2020-04-18',
                  '2020-04-25', '2020-05-02', '2020-05-09', '2020-05-16',
                  '2020-05-23', '2020-05-30', '2020-06-06', '2020-06-13',
                  '2020-06-20', '2020-06-27', '2020-07-04', '2020-07-11',
                  '2020-07-18', '2020-07-25', '2020-08-01', '2020-08-08',
                  '2020-08-15', '2020-08-22', '2020-08-29', '2020-09-05',
                  '2020-09-12', '2020-09-19', '2020-09-26', '2020-10-03',
                  '2020-10-10', '2020-10-17', '2020-10-24', '2020-10-31',
                  '2020-11-07', '2020-11-14', '2020-11-21', '2020-11-28',
                  '2020-12-05', '2020-12-12', '2020-12-19', '2020-12-26',
                  '2021-01-02']
    url = 'https://www.billboard.com/charts/hot-100/'
    ID = 0
    week_id = 0
    week_date = week_dates[week_id]

    def parse(self, response):
        for list_element in response.css('li.chart-list__element'):
            self.ID += 1
            yield {
            'id':
                self.ID,
            'rank':
                list_element.css('span.chart-element__rank__number::text').get(),
            'artist':
                list_element.css('span.chart-element__information__artist::text').get(),
            'song':
                list_element.css('span.chart-element__information__song::text').get(),
            'rank_last_week':
                list_element.css('span.chart-element__information__delta__text.text--last::text').get(),
            'peak_rank':
                list_element.css('span.chart-element__information__delta__text.text--peak::text').get(),
            'weeks_on_chart':
                list_element.css('span.chart-element__information__delta__text.text--week::text').get(),
            'date':
                self.week_date
            }
        self.week_id += 1
        if self.week_id < len(self.week_dates):
            self.week_date = self.week_dates[self.week_id]
            next_page_url = self.url+self.week_date
            yield scrapy.Request(next_page_url, self.parse)
        pass
```

B.5. Genius.com Scraper

```python
import lyricsgenius
import numpy as np
import pandas as pd
from requests.exceptions import HTTPError, Timeout


token_file = open("genius_token.txt", "r")
token = token_file.read()
token_file.close()

genius = lyricsgenius.Genius(token, response_format='plain', timeout=3, sleep_time=0.1)
genius.verbose = False
genius.remove_section_headers = True
genius.skip_non_songs = True
genius.retries = 1


def get_genius_resources(title, artist):
    result_dict = {}
    try:
        song = genius.search_song(title, artist)
        if song:
            result_dict['lyrics'] = song.lyrics
            result_dict['url'] = 'https://genius.com' + song.path
            id = song.id
            result_dict['id'] = id
            result_dict['primary_artist'] = song.primary_artist
            song = genius.song(song.id)['song']
            result_dict['description'] = song['description']['plain']
            result_dict['annotations'] = genius.song_annotations(id)
            result_dict['comments'] = []
            for comment in genius.song_comments(id, per_page=20)['comments']:
                result_dict['comments'].append(comment['body']['plain'])
            result_dict['release_date'] = song['release_date']
        else:
            result_dict['lyrics'] = '!NoSong!'
            result_dict['url'] = ''
            result_dict['id'] = ''
            result_dict['primary_artist'] = ''
            result_dict['description'] = ''
            result_dict['annotations'] = []
            result_dict['comments'] = []
            result_dict['release_date'] = ''
    except:
        result_dict['lyrics'] = '!Error!'
        result_dict['url'] = ''
        result_dict['id'] = ''
        result_dict['primary_artist'] = ''
        result_dict['description'] = ''
        result_dict['annotations'] = []
        result_dict['comments'] = []
        result_dict['release_date'] = ''
    return result_dict
```

B.6. LDA Model Configuration

```
azlyrics_lda_model = gensim.models.ldamodel.LdaModel(corpus=azlyrics_corpus,
                                                     id2word=id2word,
                                                     num_topics=7,
                                                     chunksize=4000,
                                                     passes=20,
                                                     alpha='auto',
                                                     iterations=200)
```