Sentiment Analysis of Songs over 62 years (1958-2019)

Abstract

The aim of this project is to perform sentiment analysis on representative samples of the most popular music from the past few decades to see if there was any significant change to the sentiment (positive or negative) in the language used in popular music lyrics over time. We analyzed the Billboard Hot 100 songs through each week in every year of our dataset using VADER (Valence Aware Dictionary for Sentiment Reasoning), and observed a trend of increasing negative sentiment and decreasing positive sentiment in song lyrics over the years.

Introduction

This study is an attempt to analyze how popular music has changed over the years. We all have heard from our elders or peers who are fond of older music from the 80s or 90s about how music lyrics have degraded over the years and the positivity that the songs from that timeline radiated simply isn't prevalent in modern music. We hope to learn through our project exactly how much has sentiment in popular music changed (or not), and in what ways.

The results of this project will help us in linking societal factors or changes/evolution of music over time to the changes in sentiments that we obtain. We also feel that since the most popular songs of the year are generally a good representation of what the common public prefers to hear at that time, getting the most prevalent sentiments of popular songs in each year and their changes over the years could be useful in determining if there is a change in public sentiment over the years, i.e : if the public is getting more aggressive/negative and increasing the usage of such sentiment words or inspecting other behavioral or preferential changes that the society might have been going through.

In this study we have attempted to answer the following research questions:

1. How have sentiments in songs have changed over the last 62 years (1958-2019)?
2. Which model (from Logistic Regression, Support Vector Machine, VADER, TextBob, and Gaussian Naive Bayes) performs better in song classification (Classes: Positive & Negative)?

Related Work

1. Review on Sentiment Analysis on Music by Stuti Shukla, Pooja Khanna, and Krishna Kant Agarwal (Shukla, Khanna and Agrawal). In this study the researchers perform sentiment analysis on songs' lyrics with logistic regression using character n-gram features for mood classification.
2. Applying data mining for sentiment analysis in music by Lucia Martin Gomez and Maria Navarro Caceres (Martín-Gómez and Caceres, 2018). In this research the researchers have used the WEKA framework, with its extension MEKA, to perform multilabel classification on songs to extract relationship patterns between emotions and song's music features. The dataset used is Emotions from Mullan consisting of 593 songs with 72 music features. Data-mining algorithms used are Random k-Labelsets and Multi-Label k-Nearest Neighbors.
3. Sentiment Analysis on the text of Harry Potter by Greg Rafferty (Rafferty, 2018). In this project the author performed a sentiment analysis on the Texts of Harry Potter to predict/anticipate the reader's emotions/mood while reading the texts. The author uses various models to perform the sentiment analysis including TextBlob, Vader, Naive Bayes, and Pattern Analyser.

Data

1. Billboard Hot weekly charts dataset
    1.1. Source: data.world (Miller *Billboard Hot weekly charts - dataset by kcmillersean*)
    1.2. Information: Billboard's Top 100 songs of the week list for all weeks from 1958 to 2019. Then sorted the dataframe by year
    1.3. Size (number of documents/song): 320395
    1.4. Primary Language of the documents: English
2. Lyrics DataFrame
    2.1. Source of lyrics: LyricsGenius (Miller *LyricsGenius: a Python client for the Genius.com API*)
    2.2. Information: Extracted the lyrics of unique songs in 'Billboard Hot weekly charts' for each year from 1958 to 2019 and merged it with Billboard Hot weekly charts dataset
    2.3. Size (number of documents/song): 28978
    2.4. Primary Language of the documents: English
3. Moody Lyrics (Test Dataset)
    3.1. Source: MoodLyrics (Çano *MoodyLyrics: A Sentiment Annotated Lyrics Dataset*)
    3.2. Information: Annotated Dataset of 2000 songs with Labels 'pos' and 'neg'. We extracted lyrics for these songs using the LyricsGenius API and merged with this dataset.
    3.3. Size (number of documents/song): 2000:
    3.4. Average number of sentences per document: 61.2385516507
    3.5. Primary Language of the documents: English

Method

Model Used

To cater to the purpose of this research we used VADER (Valence Aware Dictionary for Sentiment Reasoning). VADER is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. VADER analyses a text's sentiment with respect to its structure and words. VADER returns a dictionary of scores: positive, negative, neutral, and compound. Positive and Negative scores are in the range [0.0, 1.0] indicating extreme sentiment at 1.0 and none at 0.0. Compound score is a metric that calculates the sum of all the lexicon ratings which have been normalized between -1(extreme negative) and +1 (extreme positive). VADER performs better on sentences than on longer strings, so we calculated the three scores for songs by tokenizing its lyrics into sentences (each line = one sentence).

Analysis of the Models Used

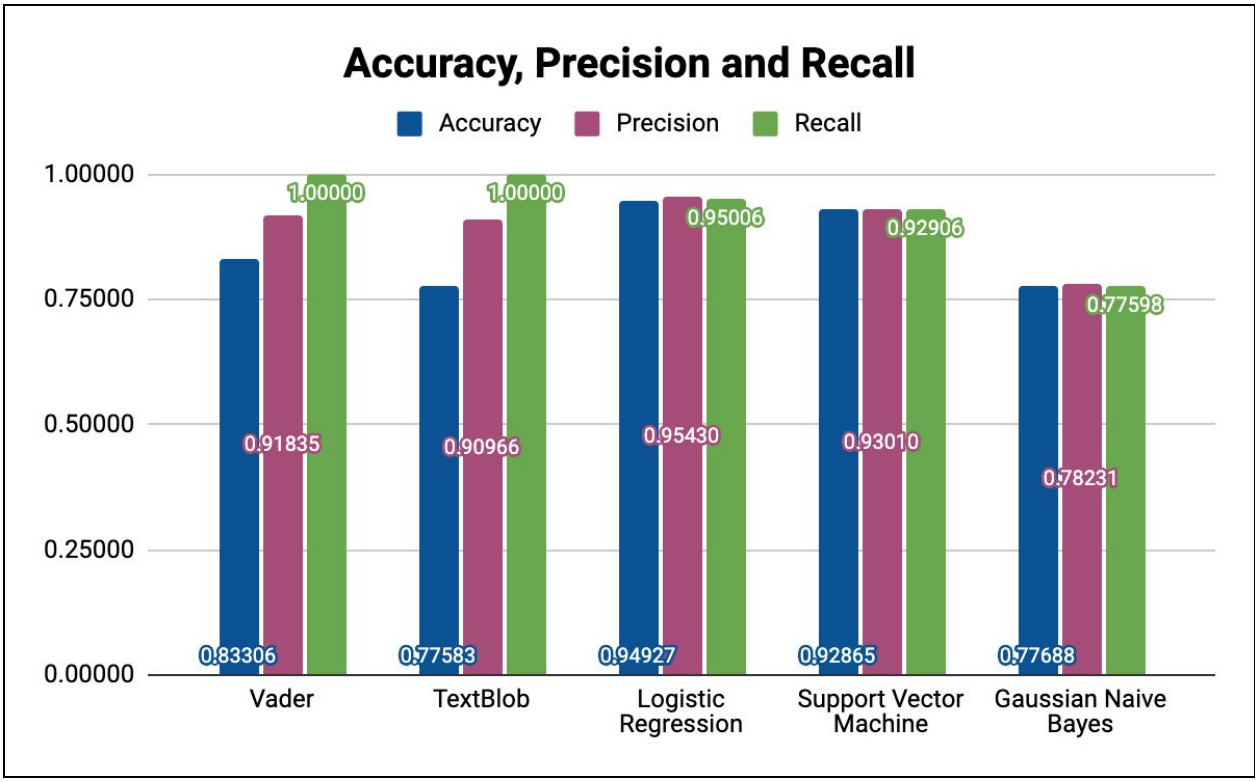
VADER and TextBlob are similar pre-trained Models, however, VADER works better with informal text documents containing slangs or emojis, whereas TextBlob almost always works better with formal text documents. Song lyrics can be classified as informal document type which may contain slangs at times, and therefore VADER performed better than TextBlob on our test dataset (shown in the graph below).

Gaussian Naive Bayes assumes that all features are independent of each other, however, in songs the structure is very important which is not taken into consideration by Naive Bayes. This is why Naive Bayes didn't perform well on our test dataset.

Logistic Regression and Linear Support Vector Machine generally perform comparably in practice. Both Logistic Regression and SVM are discriminative classifiers and so they learn explicit boundaries between classes. This is why both these models perform (and have performed) excellently on binary sentiment classification on our dataset.

VADER performed well with 83.33% Accuracy and 91.83% Precision on the test data. Logistic Regression and Support Vector Machine performed better than VADER. However, since our dataset is very big, creating vectors (using BERT or TF-IDF vectors) for Logistic Regression timed out. Since SVM algorithms are not incremental, they performed very slow on our dataset. Therefore, considering performance results, speed of the models, and limited computing power we decided to proceed with VADER.

Graph 1: VADER's Performance on Test Data vs Other Models' Performance



Assumptions Made

There were some assumptions made: general assumptions and assumptions specific to the way scores were normalized. The general assumption is that the sentiment of a song can solely be identified from the lyrics of the song. The specific assumptions lead to different interpretations and normalization of the scores reflected in Graphs 2, 3, and 4 respectively. The assumptions were:

1. Graph 2:
    a. Each sentence in a song holds equal importance to reflect the song's sentiment
    b. Each song holds equal importance to reflect that year's overall popular culture sentiment
2. Graph 3:
    a. Each song may hold different importance to reflect that year's overall popular culture sentiment
3. Graph 4:

a. Each sentence in a song may hold different importance to reflect the song's sentiment
b. Each song may hold different importance to reflect that year's overall popular culture sentiment

Results

<u>Baseline Algorithm</u>

Note: For VADER no training is required, so all 2000 songs from the MoodLyrics dataset will be used for testing.

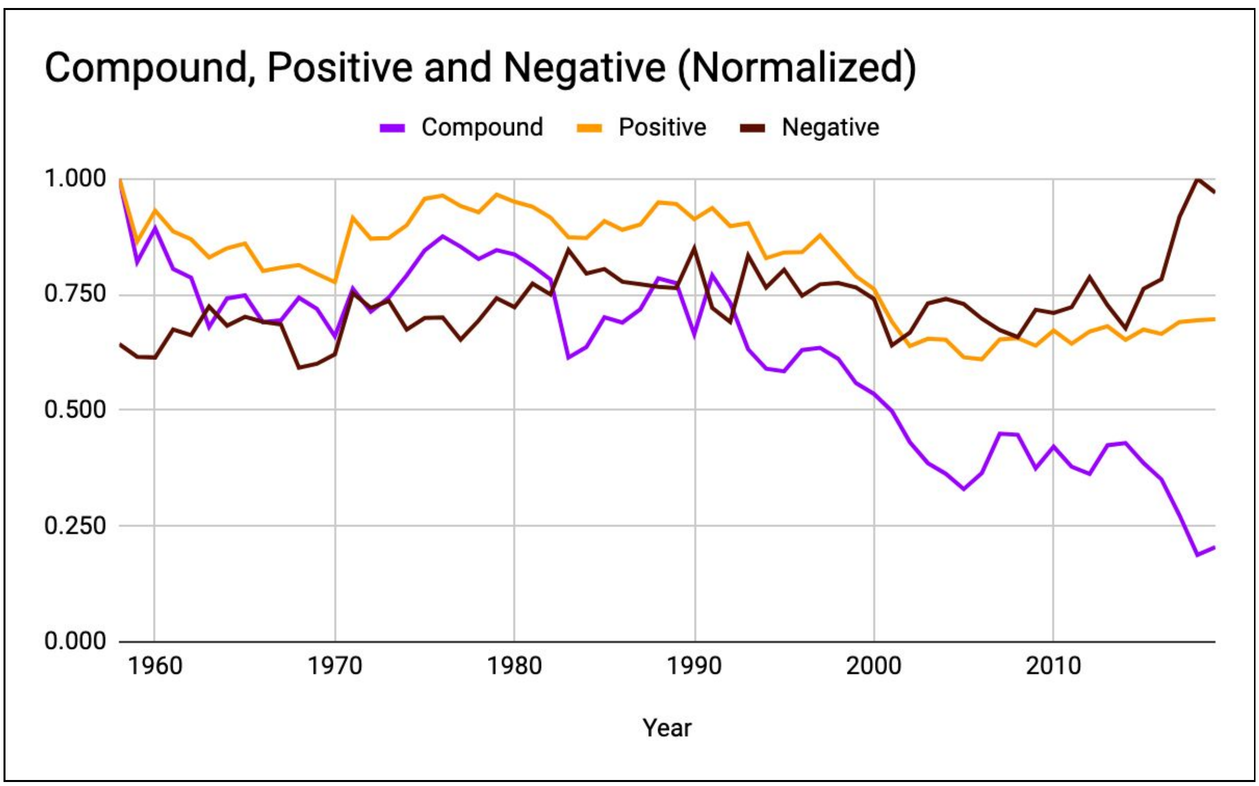
1. Tokenize the lyrics for each song into sentences (by line breaks: '\n')
2. Get the VADER dictionary for scores for each sentence
3. Sum up the scores for each sentences for a song
4. Compare the scores and label the song 'Positive' or 'Negative' depending on the higher score
5. Find Accuracy, Precision, and Recall by using the Label column in the dataset to compare with the Predicted Labels (found using VADER)

<u>Advanced Procedure</u>

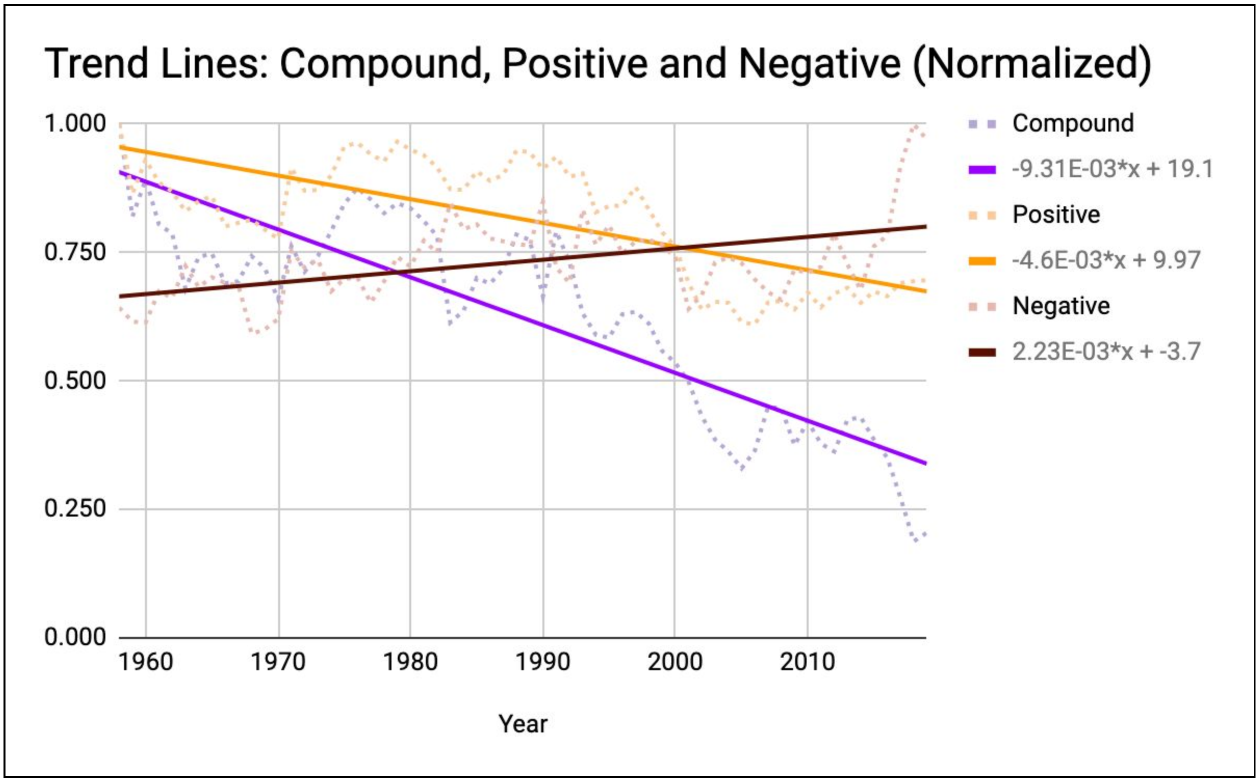
1. Use data.world to extract a list of all songs and their ids needed to scrape lyrics from genius.com using LyricGenius
2. Split the list of song ids by year
3. Use genius.com API for python to extract lyrics for each song
4. Procedure using VADER (Valence, Aware, Dictionary, and sEntiment Reasoning)
    a. Tokenize each document by line breaks (into lines)
    b. Get the VADER dictionary for scores for each sentence
    c. Sum up the scores for each sentences for a song
    d. Procedure 1/Graph 2:
        i. Divide each score by the number of sentences in that song (so that each song is equally contributes to the overall sentiment score for that year)
        ii. Add each score for each song (group by year)
        iii. Divide the aggregate scores by the number of songs (documents) in that year
        iv. Normalize scores for each sentiment by dividing the aggregate scores by the maximum aggregate score for each sentiment
        v. Plot graphs to compare and anticipate a trend (if one exists) for each sentiment score (Compound, Positive, and Negative)

e. Procedure 2/Graph 3:
    i. Add each score for each song (group by year)
    ii. Divide the aggregate scores by the number of songs (documents) in that year
    iii. Normalize scores for each sentiment by dividing the aggregate scores by the maximum aggregate score for each sentiment
    iv. Plot graphs to compare and anticipate a trend (if one exists) for each sentiment score (Compound, Positive, and Negative)
f. Procedure 3/Graph 4:
    i. Add each score for each song (group by year)
    ii. Store the number of total sentences in each year's songs' lyrics
    iii. Divide the aggregate scores by the number of total sentences in that year
    iv. Normalize scores for each sentiment by dividing the aggregate scores by the maximum aggregate score for each sentiment
    v. Plot graphs to compare and anticipate a trend (if one exists) for each sentiment score (Compound, Positive, and Negative)

Graph 2: Normalized Sentiment Scores

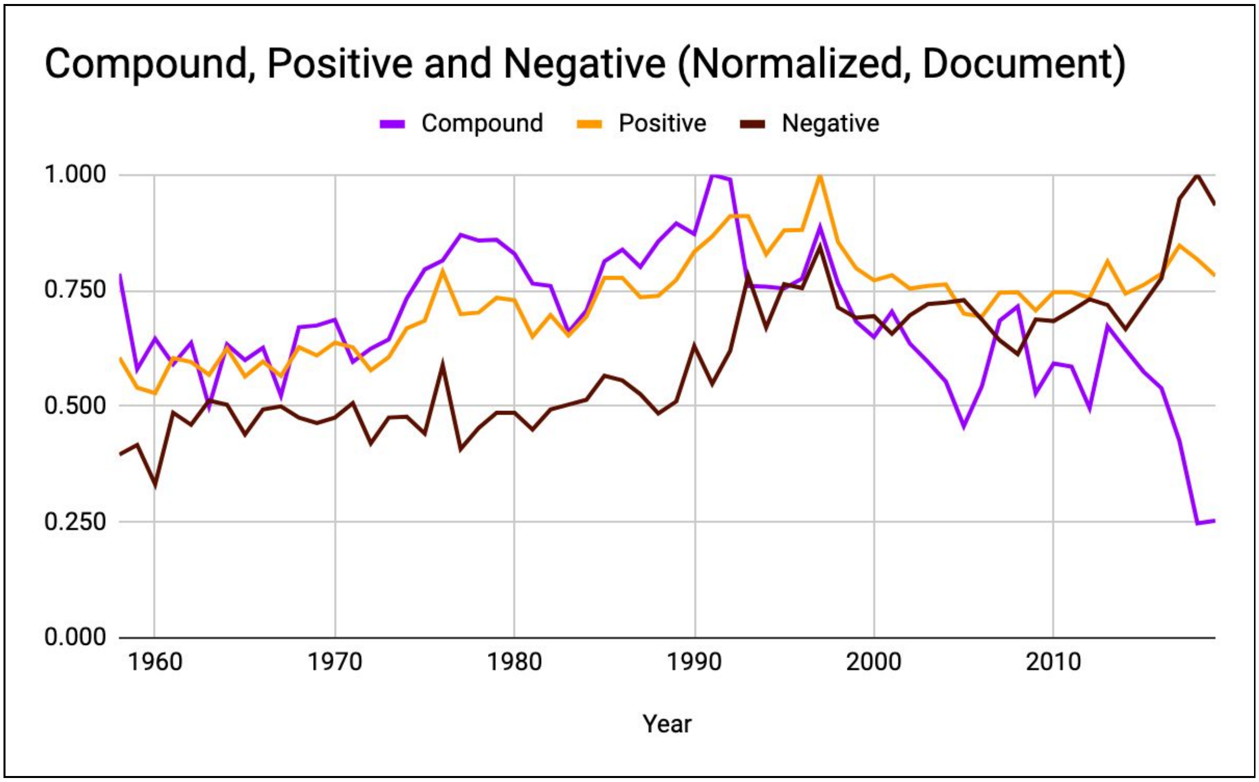


Graph 2A: Trend lines for Normalized Sentiment Scores

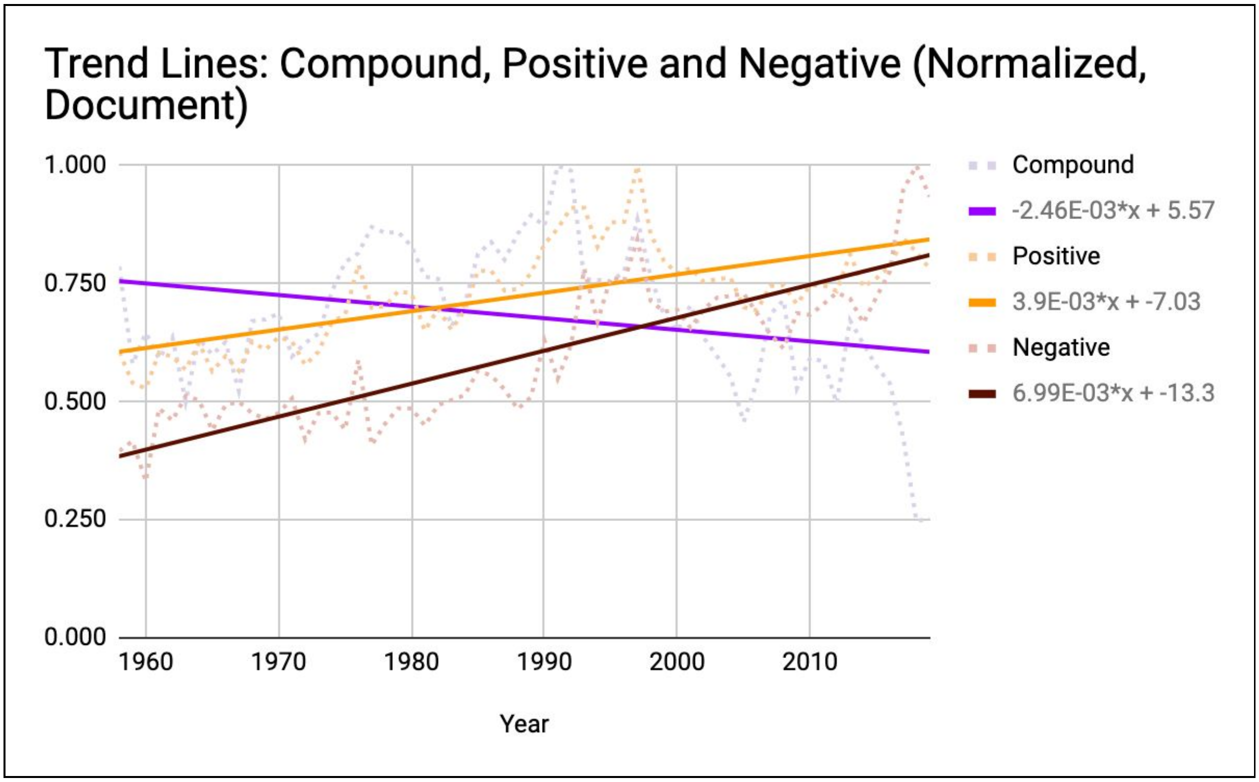


Inference: From Graph 2A we can see that that is a general trend of negativity increasing (slope: 2.23) and positivity and compound decreasing (slope: -4.6 and -9.31 respectively) over the years. The overall mean sentiment polarity is positive, however, there is a trend suggesting increase in negative sentiment.

Graph 3: Normalized Sentiment Scores (per Document)

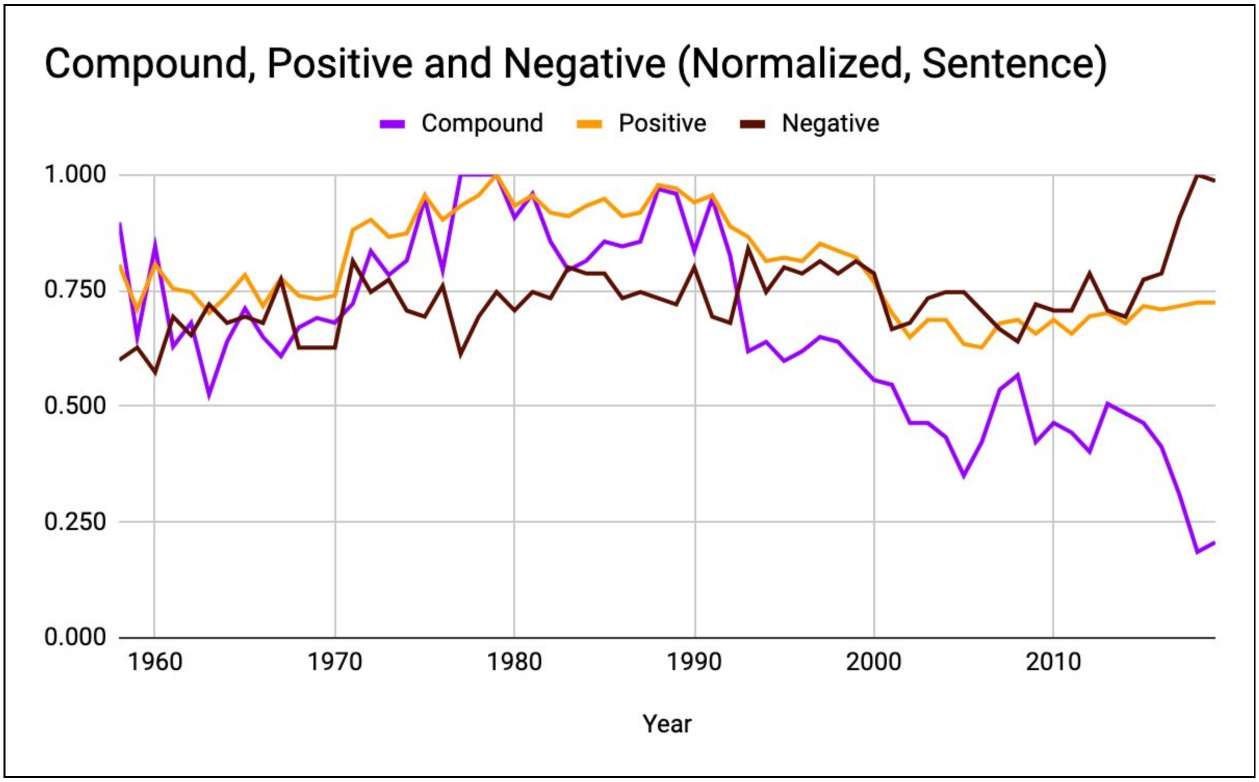


Graph 3A: Trend lines for Normalized Sentiment Scores (per Document)

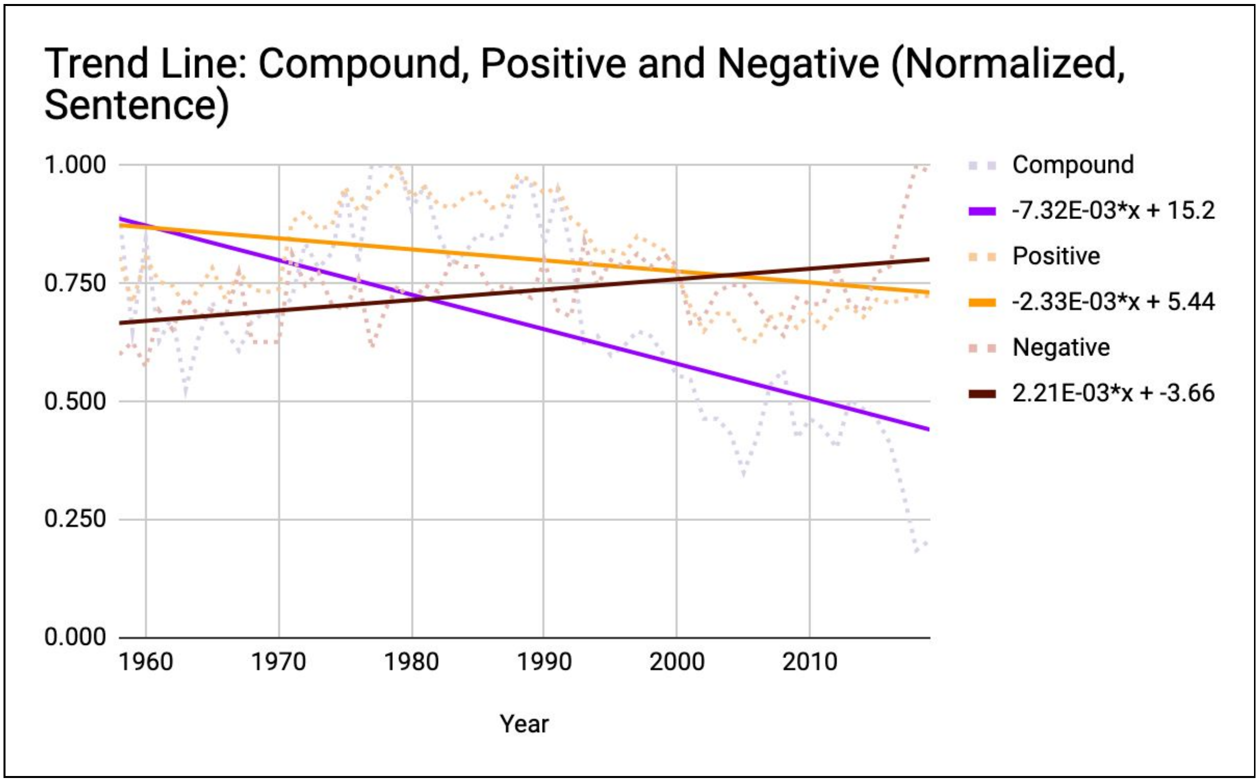


Inference: From Graph 3A we can see that that is a general trend of negativity and positivity increasing (slope: 6.99 and 3.9 respectively) and compound decreasing (slope: -2.46) over the years. Both sentiments (positive and negative) seem to increase over the years, however, negative sentiment seems to increase more rapidly compared to positive and so the overall mean compound sentiment polarity seems to decrease with time, heading towards a negative polarity.

<u>Graph 4: Normalized Sentiment Scores (per Sentence)</u>



Graph 4A: Trend lines for Normalized Sentiment Scores (per Sentence)



Inference: From Graph 4A we can see that that is a general trend of negativity increasing (slope: 2.21) and positivity and compound decreasing (slope: -2.33 and -7.32 respectively) over the years. The overall mean sentiment polarity is positive, however, there is a trend suggesting increase in negative sentiment.

Table 1: Top Keywords in Song Lyrics (by Frequency)

1958	2019
like	like
come	yeah
littleness	niggas
manning	bitches
knows	lil bitch
jump like	love
good	need
time	fuck

Interpreting the results

1. 2018 and 2019 had the most negative lyrics recorded in the last 62 years
2. 1979 (as per Procedure 1 and 3) and 1997 (as per Procedure 2) had the most positive lyrics recorded in the last 62 years
3. All three graphs (with different assumptions and normalizations) indicate an overall net positive polarity and suggest a general trend of the net sentiment polarity (decreasing) heading towards a more negative sentiment.
4. As seen in table 1, we can clearly see how top keywords from 1958 are usually used in a positive context to express joy, whereas top keywords from 2019 are mostly slangs, generally used in negative/offensive contexts to express anger and sorrow.

Discussion and Future Work

Through the results of this project, we observed that there has indeed been a gradual increase in negative sentiment polarity in popular song lyrics over the years. Depression among adults has increased significantly over the last few decades (Mental health issues increased significantly in young adults over the last decade: Shift may be due in part to rise of digital media, study suggests, 2020). Reports also suggest that violence and social unrest is increasing significantly and is at its highest. Music is a form of expression for many and so this might be why an increase in aggression/negativity in the society is reflected in music lyrics as well. As a result people are using more explicit and negative words. On the other hand, artists, to make the Billboard Hot 100 list, (with all the recent and new analytical tools available) might want to blend in with the popular culture by using explicit, negative and derogatory words in their songs, which explains a recent rapid increase in negative sentiment polarity.

Future applications of this project could be extending the scope of this project to other fields of popular culture and media to examine changing trends in sentiments in them over the years. For example, we could use a similar approach as this project on datasets of movie scripts, newspaper articles, and online articles to see if there has been a shift in sentiment in these as well over the years.

Bibliography

1. Çano, Erion. "MoodyLyrics: A Sentiment Annotated Lyrics Dataset." *MoodyLyrics*. ResearchGate, 1 Mar. 2017. Web. 01 Nov. 2020. <https://www.researchgate.net/publication/317031495_MoodyLyrics_A_Sentiment_Annotated_Lyrics_Dataset>.
2. Martín-Gómez, L. and Caceres, M., 2020. Applying Data Mining For Sentiment Analysis In Music. [online] Researchgate. Available at: <https://www.researchgate.net/publication/318510880_Applying_Data_Mining_for_Sentiment_Analysis_in_Music> [Accessed 4 December 2020].
3. Miller, John W. LyricsGenius. GitHub, 23 Sept. 2019. Web. 01 Nov. 2020. <https://github.com/johnwmillr/LyricsGenius>.
4. Miller, Sean. "Billboard Hot Weekly Charts - Dataset by Kcmillersean." Data.world. Data.world, 06 Jan. 2020. Web. 01 Nov. 2020. <https://data.world/kcmillersean/billboard-hot-100-1958-2017>
5. Rafferty, G., 2018. Basic NLP On The Texts Of Harry Potter: Sentiment Analysis. [online] Medium. Available at: <https://towardsdatascience.com/basic-nlp-on-the-texts-of-harry-potter-sentiment-analysis-1b474b13651d> [Accessed 4 December 2020].
6. Shukla, S., Khanna, P. and Agrawal, K., n.d. Review On Sentiment Analysis On Music. [online] Ieeexplore.ieee.org. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8286111>.