**Team 180: Sentiment Analysis on Music Lyrics**

Carlos Ordonez, Jingxiao Wang, David S. Stephen, Yevgen Polishchuk, Thomas W. Davis, Tomas Ordonez

{cordonez6, jwang960, dstephen, ypolishchuk3, tdavis330, tordonez3}@gatech.edu

## 1. INTRODUCTION

Music streaming platforms continue to improve curation of music to provide users with a personalized experience. We are analyzing different natural language processing libraries to perform sentiment analysis on a large data set of song lyrics. Our approach analyzes songs based on lyric sentiment and provides an interactive visualization tool. We believe that this approach enhances the music curation and differentiation efforts that these platforms aim to have.

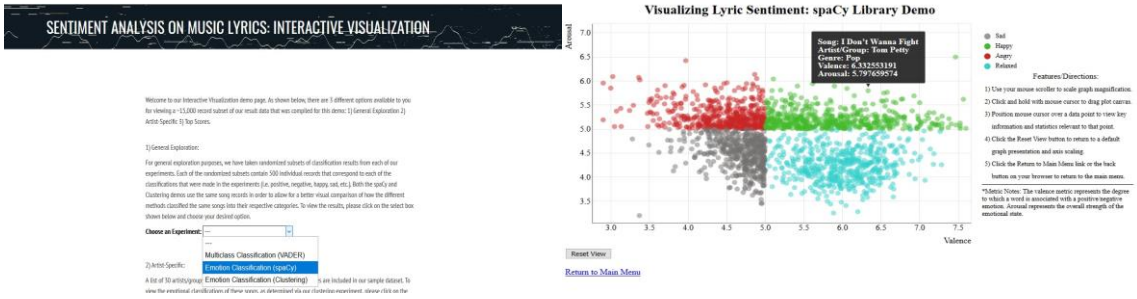


**Figure 1: User Interface to visualize sentiment analysis on music lyrics.** Left image shows user interface main menu with three sections to review visualizations based on general exploration, artist specific, and top scores. Right image shows visualization for classification of song lyrics based on valence, representing the degree to which a word is associated with a positive/negative emotion, and arousal, showing the strength of the emotion.

## 2. PROBLEM DEFINITION

The online music industry looks for new ways to differentiate themselves by creating a personalized user experience. We seek to create a novel sentiment analysis algorithm based on song lyrics, by combining natural language processing libraries and an interactive visualization.

## 3. SURVEY

Some music streaming services have recommendation algorithms using lyric and audio analysis, that combine social features to give songs a binary score [2][3]. This system has been used in research to understand movie review sentiment by analyzing the semantic relationship among words [16]. Most automatic music classification solutions are based entirely on audio feature analysis [9], aimed at improving mood classification in music by combining audio with lyric sentiment. While most studies focus on audio features, some show that lyric data can serve as a proxy for the melodic, structural, and rhythmic properties of an audio signal [15][4].

In the past, sentiment analysis has been approached using naïve Bayes classification, decision trees, Support Vector Machine (SVM), Collaborative filtering [10], and other alternative algorithms [5]. Research

conducted by Gupta [11] and Russell [12], found that proper weightage on words in different song parts such as the chorus combined with lyrics could yield better accuracy and mood classification.

Research on Twitter sentiment analysis showed their model outperformed a human rater [18]. Some researchers insist that analyzing lyric data with audio features yields better results than using just lyric text. By utilizing other algorithm types, we hope to find the model that best fits our data and improve the accuracy of predictions [5][6][7]. These predictions can be used for a score system or to improve the user experience of other music streaming features.

## 4. PROPOSED METHOD

Established sentiment analysis approaches are based on machine learning or on lexical features [7]. Our proposed analysis looks to expand the range of polarity from a multiclass classification that has positive, neutral, and negative to an emotion classification such as sad, happy, angry, or relaxed.

We analyzed different natural language processing libraries using Python in order to perform sentiment analysis on a Kaggle dataset containing 250,000 lyrics from over two thousand singers from different times, genres and countries [26]. Our approach analyzes songs based on lyric sentiment presented with a user interface.

The interface is a webpage that provides visualizations of several options and features from a subset of general results from each of our experiments, as well as records pertaining to specific artists/groups and top metric scores. The three sections of our interface show visualizations for 1) General Exploration to choose an experiment (multiclass classification with Vader, emotion classification with spaCy, and emotion classification using Clustering). 2) Artist-Specific to explore Clustering classification by choosing a specific artist, and 3) Top Scores to explore top 100 scores by choosing Vader or spaCy. Creating an interactive representation of our results can help listeners better understand the relationship between the music they like and the mood of the song [1].

We divided our analysis into two iterations. For the first iteration, we performed an initial analysis with multiclass classification to learn more about natural language processing. We used the NLP library “VADER” (Valance Aware Dictionary and Sentiment Reasoner), a lexicon and rule-based sentiment analysis tool, to determine if lyrics are positive, neutral, or negative [21]. We initially proposed to analyze the results using state-of-practice benchmarks [17]. A list of 11 benchmarks used to evaluate the results of the VADER library [17]. However, due to time constraints, we opted to analyze the VADER results as described in our experiments below.

For the second iteration, we performed an analysis to expand the range of polarity from a multiclass classification that has positive, neutral, and negative to an emotion classification such as sad, happy, angry, or relaxed. We used the NLP library “spaCy”, an industry adopted tool for information extraction and for understanding large volumes of text [22]. The results were also examined in our experiments.

Evaluating our spaCy visualization using the thresholds assigned brought up the question of whether there was another way to assign sentiment boundaries within our dataset. We evaluated a KMeans clustering algorithm to find subgroups within our dataset using valence and arousal.

Furthermore, most music recommendation engines and/or related analytical ventures do not produce interactive visualizations. We provide an interactive user interface to visualize and explore our results.

### 4.a. Intuition: Improving state of the art

Our project proposes the following innovations:

- Music streaming services provide a personalized experience but do not offer a way to understand the sentiment of a song. We offer a quantitative way to score emotional classification on music lyrics. For example, for the song "Best of You" (Artist: Foo Fighters), our model shows a valence of 6.88 and arousal of 3.91, showing a sentiment of "relaxed".
- Our novel visualization represents song valence and arousal scores calculated with our algorithm and displayed with an interactive scatterplot. Current sentiment analysis does not provide such an intuitive and interactive interface to represent song metrics.
- Our method categorizes songs based on emotional classification and provides filters for song meta-data characteristics. As we improve our interface, we hope to provide better user control and visualize clustered characteristics.

**4.b. Approach description: Algorithms, user interfaces, etc.**

VADER is a lexicon and rule-based sentiment analysis tool, used to determine the positivity or negativity of text [21]. We used it for our initial analysis for multiclass classification of lyrics. It provides the following polarity values: positive, negative, neutral, and a compound score, which calculates the sum of valences of each word in the lexicon, normalized from –1 to 1 [25]. Compound scores greater than or equal to .05 are considered “positive”, while scores less than or equal to -.05 are considered “negative” [25]. Those that fall between -.05 and .05 are deemed “neutral” [25]. The success benchmark of this approach is described in our experiments section.

To go beyond the multiclass sentiment classification provided by VADER, we used a lexicon-based approach to take both word valence and arousal values into consideration. In the context of sentiment analysis, valence represents a degree to which a word is associated with a positive or a negative emotion, while arousal shows the strength of the emotion. Using spaCy [22], a library for information extraction, we removed "stop" words (e.g. “is”, “at”, “for”) as they are not found in most lexicon dictionaries, thus making it difficult to assign values. Next, we used a lexicon dictionary from the work by Warriner [23] to assign valence and arousal values to the top 10 frequent words for each song. We used the most frequent words in a song for our analysis to mitigate the collective effect of neutral words with single occurrences. These values were used to assign valence and arousal scores for each song.

The output of the spaCy analysis assigns each song a valence and arousal score, ranging from 0-9. Looking to assign sentiment labels using these scores, we chose to use Thayer’s two-dimensional emotional model [27]. This model illustrates the relationship between valence and arousal score values and more granular sentiment labels of happy, calm, angry and sad. Each song is assigned to a Thayer’s model quadrant using a clustering method, described in more detail below.

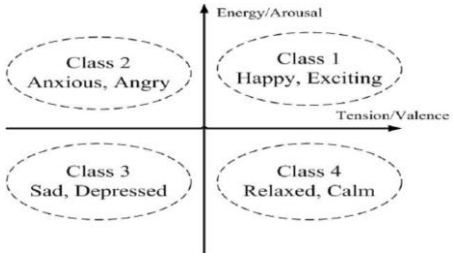


**Figure 2: Thayer’s 2-D emotional model**

Looking to improve upon these thresholds, we decided to use a kmeans clustering algorithm. We used the valence and arousal scores to attempt to find better sentiment assignments.

To display our results, we developed an interactive visualization that allows users to view a subset of general results from each of our experiments, as well as records pertaining to specific artists/groups and top metric scores. The three sections of our interface show visualizations for 1) General Exploration to choose an experiment (multiclass classification with Vader, emotion classification with spaCy, and emotion classification using Clustering). 2) Artist-Specific to explore Clustering classification by choosing a specific artist, and 3) Top Scores to explore top 100 scores by choosing Vader or spaCy.

The visualizations are interactive scatterplots with mouse-scroller operated magnification, draggable plot canvases, and mouseover tooltips that display key information relevant to each data point. The data points are also color-coded according to their classification (such as positive/negative or happy/sad, etc.) and have a corresponding plot legend. For demo purposes, our interactive visualization runs utilizing a subset of our total experimental results. This subset contains ~15,000 records. 500 records corresponding to each classification from our VADER and spaCy library-based experiments (positive, negative, neutral, happy, sad, angry, and relaxed) were randomly selected and compiled. Records corresponding to 30 artists/groups spanning a variety of genres and time periods were also included and are visualized using their clustering-based classifications. We also compiled the top 100 scores pertaining to positive/negative compound values (VADER) and valence/arousal (spaCy).

## 5. EXPERIMENTS

As previously described, we divided our analysis into two iterations. A multiclass classification using the NLP library “Vader” to determine if lyrics are positive, neutral, or negative, and an emotion classification using the NLP library “spaCy” to determine if song lyrics are sad, happy, angry, or relaxed. We describe the experiments performed and the design of upcoming experiments.

### 5.a. Description of our testbed

<u>These are completed questions that our experiments are designed to answer:</u>

- Does the effect of header keywords and uppercase vs lowercase provide any insights besides how the NLP Vader library builds the features?
- Would our analysis for multiclass classification using the NLP Vader library outperform a human rater, as seen in research [17] with social media sentiment analysis?
- How would the scores compare using each library, Vader for multiclass classification (positive, neutral, or negative) and spaCy for emotional classification (sad, happy, angry, or relaxed)?
- Would a clustering algorithm provide improved sentiment classification based on valence and arousal values?

### 5b. Experiments - Details and Observations

#### *Experiment 1, analyzing results of original vs cleaned dataset for classification with NLP Vader*

VADER’s library is based on social media text, which makes use of capitalization and emoticons [17]. We wanted to evaluate if the effect of header keywords and uppercase vs lowercase provided any insights besides how the library builds the features.

The cleaning process was performed by removing the header keywords 'verse', 'hook', 'chorus', and 'intro'. Then we took the averages of the positive and negative values of the compound feature that VADER

provides and comparing the clean and un-clean data. The difference between clean/unclean data for the positive values was 0.32% and 0.42% for the negative ones. We consider that these changes are irrelevant, which implies that there is no need to clean the data when analyzing lyrics with VADER. We learned that the library's lexicon only affects adverbs and verbs, by summing the valences of each word [24]. Surveying the data, we found that the lyrics do not emphasize emotion through capitalization.

***Experiment 2, Success benchmark for classification with NLP Vader***

We wanted to evaluate if the NLP Vader library would outperform a human rater, as seen in research [17] with social media sentiment analysis, even though the library was built with a different type of dataset than song lyrics. Following a human rater approach [18], we ran three samples of 5 songs with a fixed random number to replicate the experiment (random_state=1, 2, and 3). We had two participants, each manually rate the samples (total of 15 songs) and score them based on these values: negative (-1), neutral (0), and positive (1). Also, to avoid confirmation bias, first we rated the lyrics before looking at the calculated scores. The results showed that human vs VADER matched 40% of the labels and human vs spaCy matched 30% of the labels. We found it difficult to check agreement between human-rater label and the Vader features, due to the small sample and the number of human-raters (2 participants, each rating 15 songs). The outcome would significantly improve to account for subjective ratings in a future development by crowd sourcing human rating with a larger dataset.

***Experiment 3, analyzing results of classification (Vader) vs emotion classification (spaCy)***

We analyzed the sentiment relationship between the two NLP packages VADER and spaCy. We wanted to know how many songs that were identified by VADER as positive were identified as Happy or Relaxed by spaCy. Similarly, find the number of songs identified as negative by VADER that were identified as Sad or Angry by spaCy. We found that VADER and SPACY agreed on 66.03% of the 4,583 songs, which were True Positives, and True Negatives as correctly identified. On the other hand, they disagreed on 33.97% of the total songs.

| Sentiment_Analysis | Confussion_Type | Confussion_Value | Percentage_of_Total_Songs |
|---|---|---|---|
| Positive, Happy or Relaxed | TP | 2759 | 60.20% |
| Positive, Sad or Angry | FP | 57 | 1.24% |
| Negative, Sad or Angry | TN | 267 | 5.83% |
| Negative,Happy or Relaxed | FN | 1500 | 32.73% |

**Figure 3:** Confusion matrix comparing multiclass classification (positive, neutral, or negative) and spaCy for emotional classification (sad, happy, angry, or relaxed)

We believe that this is related to the fact that most songs had their neutral score higher than positive or negative scores. Possibly creating some ambiguity when the song was tagged negative by VADER and tagged Happy or Relaxed by SpaCy. A possible reason for this would be that VADER was built for use in social media posts and not on lyrics.

***Experiment 4, analyzing if a clustering algorithm improves emotion classification based on valence and arousal values***

Using the sum-of-squared distances for each data point to its center, we found that there doesn't really seem be a clear best number of centers. We could choose anywhere from 4-6 centers. Since our initial analysis was 4 sentiments, we proceeded with 4 centers. When compared to the original plot of our spaCy

output using assigned thresholds, it's clear the KMeans finds dramatically different sentiment boundaries within our dataset.

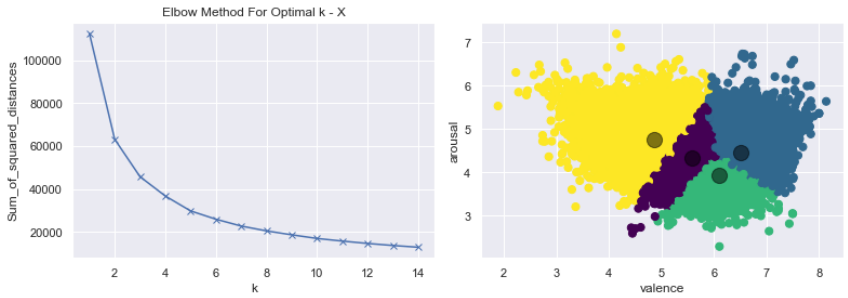


**Figure 4:** Left: sum-of-square distance measures for each number of clusters, k. Right: Scatterplot of a KMeans clustering of valence and arousal scores using k=4 clusters.

## 6. CONCLUSIONS

In this paper, we analyzed different natural language processing libraries in order to perform sentiment analysis on a Kaggle dataset containing 250,000 lyrics [26]. Our approach analyzes songs based on lyric sentiment presented with a visualization tool. Similar work showed that music classification solutions are based entirely on audio feature analysis [9]. We offer a quantitative way to score emotional classification on music lyrics. Our novel visualization represents song valence and arousal scores calculated with our algorithm and displayed with an interactive scatterplot. Current sentiment analysis does not provide such an intuitive and interactive interface to represent song metrics.

We evaluated experiments to understand the effect of header keywords on lyrics (intro, chorus) and uppercase vs lowercase. Using VADER we learned that they don't provide any insight since the library's lexicon only affects adverbs and verbs. Surveying the data, we found that the lyrics do not emphasize emotion through capitalization. We evaluated VADER's performance on song lyrics, even though we knew that the library was built using a different dataset (social media posts). Using a human-rater approach with two participants rating a small sample, we found it difficult to check agreement between human-rater label and VADER's features. A future evaluation to improve accounting for subjective ratings is to crowd source human rating with larger datasets. Also, we compared the results of using classification with VADER vs emotion classification with spaCy. Our experiments showed that VADER creates some ambiguity when when songs are tagged as neutral, possibly for the same reason that the library was built with a different type of dataset. Furthermore, we evaluated if a clustering algorithm would provide improved sentiment classification based on valence and arousal values.

Since VADER was built using social media posts and not lyrics, further evaluations could be performed to build a model that uses valence scores for words as input features and human-rater labels for song lyrics as classification targets. Our interactive interface representing our results can help listeners understand the relationship between the music they like and the mood of the song. We believe that this approach contributes to the enhancement of music curation and differentiation that music platforms aim to have.

### Distribution of team member effort

All team members have contributed similar amounts of effort.

**REFERENCES**


1. M. Cartwright. “Seeing Sound: Investigating the Effects of Visualizations and Complexity on Crowdsourced Audio Annotations”. In Proc. ACM HCI. 2017

2. A. Jandar, J. Abraham, K. Khanna, R. Dubey. “Emotion Analysis of Songs Based on Lyrical and Audio Features”. IJAIA. 2015

3. X. Hu, S. Downie. “When Lyrics Outperform Audio for Music Mood Classification: A Feature Analysis”. ISMIR. 2010

4. Cano, E. “Text-based Sentiment Analysis and Music Emotion Recognition.” Polytechnic University of Turin, 2018

5. Fang, X., and J. Zhan. “Sentiment analysis using product review data.” Journal of Big Data. 2015

6. Shanmugapriya, K. and B. Srinivasan. “An Efficient Method for Determining Sentiment from Song Lyrics Based on WordNet Representation Using HMM.” International Journal of Innovative Research in Computer and Communication Engineering 3, no. 2, 2015

7. P. Gonçalves, M. Araújo, F. Benevenuto, M. Cha. “Comparing and combining sentiment analysis methods”. In Proc. ACM COSN. 2013

8. A. Oudenne, S. Chasins. “Identifying the Emotional Polarity of Song Lyrics through Natural Language Processing”. 2010

9. X. Hu, S. Downie. “Improving mood classification in music digital libraries by combining lyrics and audio”. In Proc. JCDL. 2010

10. A. Paudel, M. Ghimire, B. Bajracharya, N. Bhattarai. “Personality Based Music Recommendation System”. 2017

11. S. Gupta. “Music Data Analysis: A State-of-the-art Survey”. 2014

12. A. Jamdar, J. Abraham, K. Khanna, R. Dubey. “Emotion Analysis of Songs Based on Lyrical and Audio Features”. International Journal of Artificial Intelligence & Applications (IJAIA) Vol. 6, No. 3, May 2015.

13. Rachman, Fika Hastarita, Riyanarto Sarno, and Chastine Fatichah. "Music Emotion Classification based on Lyrics-Audio using Corpus based Emotion." International Journal of Electrical & Computer Engineering (2088-8708) 8, no. 3 (2018).

14. He, Hui, Jianming Jin, Yuhong Xiong, Bo Chen, Wu Sun, and Ling Zhao. "Language feature mining for music emotion classification via supervised learning from lyrics." In International Symposium on Intelligence Computation and Applications, pp. 426-435. Springer, Berlin, Heidelberg, 2008.

15. Fell, Michael, and Caroline Sporleder. "Lyrics-based analysis and classification of music." In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 620-631. 2014.

16. A. Maas, R. Daly, P. Pham, D. Huang, A. Ng, C. Potts. “Learning Word Vectors for Sentiment Analysis”. In Proc. HLT. Pages 142–150. 2011

17. CJ. Hutto, E. Gilbert. “VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text”. AAAI. 2014

18. A. Agarwal, B. Xie, I. Vovsha, O. Rambow, R. Passonneau. “Sentiment Analysis of Twitter Data”. In Proc. LSM. Pages 30-38. 2011

19. L. Pollacci, R.Guidotti, G. Rossetti, F. Giannotti, D. Pedreschi. "The Fractal Dimension of Music: Geography, Popularity and Sentiment Analysis". ISTI-CNR. 2018

20. Y. Chang, W. Yeh, Y. Hsing, C. Wang. "Refined distributed emotion vector representation for social media sentiment analysis". PLOS One. 2019

21. https://pypi.org/project/vaderSentiment/

22. https://spaCy.io/

23. Warriner, A.B., Kuperman, V. & Brysbaert, M. Norms of valence, arousal, and dominance for 13,915 English lemmas. Behav Res 45, 1191–1207 (2013)

24. Wikipedia, The Free Encyclopedia, s.v. "Valency (linguistics)," (accessed April 1, 2020), https://en.wikipedia.org/wiki/Valency_(linguistics)

25. Pandey, P. “Simplifying Sentiment Analysis using VADER in Python (on Social Media Text).” medium.com. https://medium.com/analytics-vidhya/simplifying-social-media-sentiment-analysis-using-vader-in-python-f9e6ec6fc52f (accessed April 15, 2020).

26. Detkov, N. "250,000+ lyrics over 2k singers." Kaggle Dataset. https://www.kaggle.com/detkov/lyrics-dataset

27. Yeh, Chia-Hung & Lin, Hung-Hsuan & Chang, Hsuan. “An Efficient Emotion Detection Scheme for Popular Music”. 2009

**APPENDIX**

- Linguistic Inquiry Word Count (LIWC) - (Polarity-based) Lexicon with almost 4,500 words organized into 1 of 76 categories, including 905 words that are split into binary categories used for sentiment analysis. No support of acronyms, initialisms, slang, or emoticons.
- General Inquirer (GI) - (Polarity-based) Oldest Lexicon (1966) with about 11,000 words organized into 1 of 183 categories, including 1,915 of which that are labeled positive and 2,291 that are labeled negative. No support of acronyms, initialisms, slang, or emoticons.

- Affective Norms for English Words (ANEW) - (Valence-based) Lexicon of 1,043 words that are ranked in terms of their pleasure, arousal, and dominance with a range of 1-9 for each.
- HuLiu04 - (Polarity-based) Lexicon of nearly 6,800 words; 2,006 labeled as positive and 4,783 labeled as negative.
- Word-Sense Disambiguation (WSD) - Process by which the “sense” of a word that has multiple meanings is derived from the context of which it is used.
- SentiWordNet (SWN) - (Valence-based) Lexicon of 147,306 synsets ranked in terms of positivity, negativity, and neutrality. Each has a range of 0-1, and their collective sum equals 1.
- SenticNet (SCN) - Uses both AI and Semantic Web techniques to process natural language opinions and sentiment.
- Naive Bayes (NB) - Classifier that relies on Bayesian probability.
- Maximum Entropy (ME) - Machine learning technique that uses multinomial logistic regression.
- Support Vector Machine - Classifier that separates data points using a hyperplane.