Fatima Lois E. Suarez
TIGP SNHCC

NLPIR Project Writeup

Sentiment Analysis and Topic Extraction of BTS (방탄소년단) English Song Lyrics

I. Introduction

K-Pop (Korean pop) boy group BTS (방탄소년단) have managed to win the hearts of a global audience despite releasing songs primarily in Korean. They have been commended particularly for discussing issues relevant to today's youth, touching upon topics not as often discussed in the K-Pop genre such as mental health, self-perception, and personal growth.

I wanted to use NLP techniques such as sentiment analysis to extract the emotions conveyed in their songs from the English translations of the lyrics. I would like to also try performing topic extraction to look at the central themes expressed in each song. Lastly, I would like to see how the sentiment and topics in their songs have changed over time.

This project was inspired by the following projects:

- *'The Data Science of "Someone Like You" or Sentiment Analysis of Adele's Songs'* by Preetish Panda, Prompt Cloud. Retrieved from: *https://www.kdnuggets.com/2018/09/sentiment-analysis-adele-songs.html*

- *'Using Machine Learning to Analyze Taylor Swift's Lyrics'* by Ian Freed. Retrieved from: https://www.codecademy.com/resources/blog/taylor-swift-lyrics-machine-learning/
- '*Data Analysis of K-POP: Playing with Spotify API'* by Nancy Yan. Retrieved from: https://nancyyanyu.github.io/posts/63adf3bb/
- *'The Data Science of K-Pop: Understanding BTS through data and A.I.'* by Haebichan Jung. Retrieved from: https://towardsdatascience.com/the-data-science-of-k-pop-understanding-bts-through-data-and-a-i-part-1-50783b198ac2

Why this study?

- I am a big fan of K-Pop, and have been a fan of BTS' music since late 2016. I find a lot of comfort not just in the musicality of their songs (production, overall vibes - mostly since I'm not fluent in Korean), but also in the messages they convey and themes they discuss.
- I have limited knowledge in data mining, so I was inclined to use an existing dataset.

Limitations of the study:

- *the use of the English translation of lyrics:* I am not fluent in Korean, and so far all the lessons in the NLPIR course have used English as a basis. In this case however, some contextual meaning might be lost especially those significant in a Korean context (ex. '뱁새' or 'crow tit' is not easily understood in English, but in Korean it refers to a person that is trying too hard)
- *why not sort by genre?* By both Western and Korean music standards, K-Pop songs by idol groups (of which BTS is a part of) are typically 'lumped' into one singular category (compared to non-idol groups, like solo artists and indie bands)
- my own limited experience in coding in both Python and R
- my own limited experience on data mining

Lyrics data sourced from: https://www.kaggle.com/kailic/bts-lyrics
Timeframe considered: 2013 - 2020 (19 Studio Albums including Repackages)

II. Challenges Encountered & Modifications Made

Issues with BTS dataset:

- 19 albums but only 131 unique songs with lyrics
    - ***repackaged albums*** - common in K-Pop to release new albums with the same songs and include a few new songs
- overlapping songs in dataset ⇢ *cannot be analyzed by album in a straightforward manner*
- data is unlabeled (for sentiment analysis)

As such, I modified my original objectives, particularly for the sentiment analysis step. Instead, I tried to ask how well non-K-Pop lyrics will be able to evaluate the sentiment expressed in BTS' lyrics.

Addition of a (labeled) non-BTS lyrics dataset

An additional dataset of non-K-Pop lyrics with enough data for training a machine learning was added. This data was sourced from: https://data.mendeley.com/datasets/3t9vbwxgr5/3.

It includes around 20,000 song lyrics of songs released between 1950 and 2019, and it also includes metadata of all the songs. Included in song metadata is **valence,** which was used for labeling.

valence: describes musical positiveness conveyed by a song

- *high valence*: more positive (e.g. happy, cheerful)
- *low valence*: more negative (e.g. sad, angry)

As the BTS lyrics dataset did not come with metadata, I used the Spotify API and the spotipy library to extract the metadata and valence values for the 131 unique BTS songs and added them manually to the dataset.

Finally, a numerical label was assigned to each song for binary classification:

- positive: valence < 0.5 == 1
- negative: valence ≥ 0.5 == 0

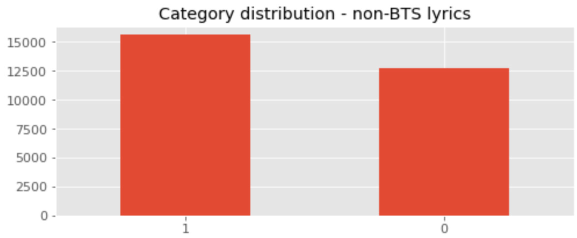
Category distribution of datasets according to valence scores

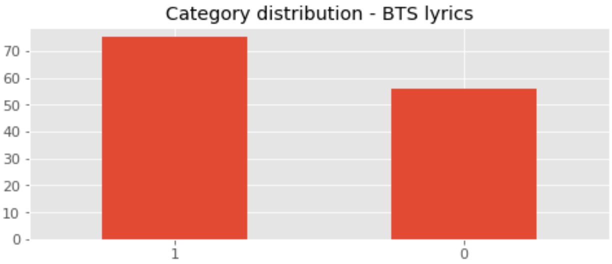
Both datasets seem to have similar distribution of positive vs. negative songs, but with more positively tagged songs.

III. Methodology

0. Preprocessing:

- remove instrumentals, skits, duplicate songs
- tokenization, remove punctuation, lemmatization





1. Sentiment Analysis

1. Extracting features: *CountVectorizer* and *TfIdfVectorizer* were both used to compute the term-document matrix and compared afterwards
2. Train binary classifier: *Multinomial Naive Bayes*
3. Evaluation: classification report (accuracy and F1 scores for each category) + confusion matrix

2. Topic Modeling

Method used: *Latent Dirichlet Allocation (LDA)*

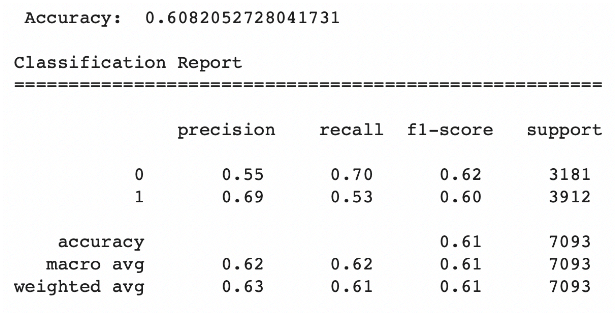
1. Analysis: create term-document matrix from processed data using CountVectorizer
2. Choosing number of topics: coherence score was computed vs. number of topics
3. Train LDA model - visualization: pyLDAvis intertopic distance map (2D)
4. Classify song lyrics into topics according to highest probability for each song - visualization: plot song topics over time

IV. Results & Discussion

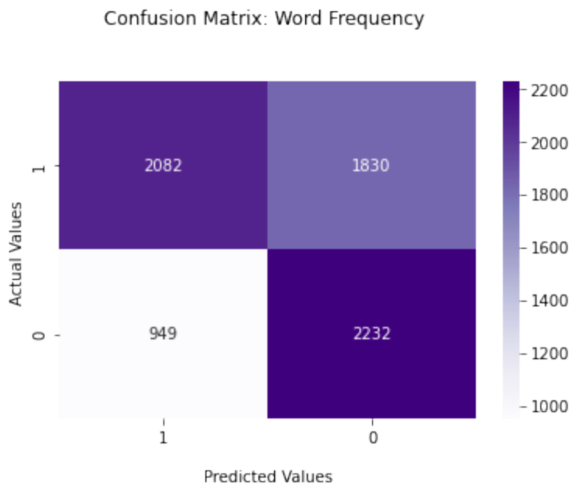
A. Sentiment Analysis

Training & testing a binary classifier on non-K-pop lyrics:

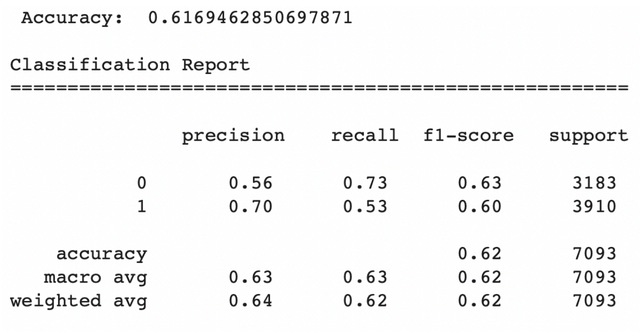
- using word frequency:



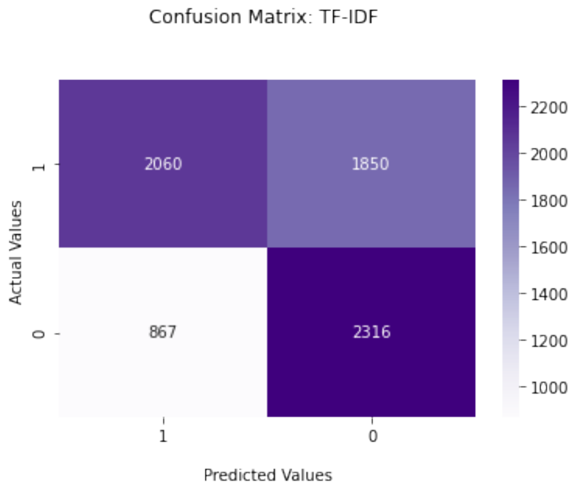
```
 Accuracy:  0.6082052728041731

Classification Report
======================================================

              precision    recall  f1-score   support

           0       0.55      0.70      0.62      3181
           1       0.69      0.53      0.60      3912

    accuracy                           0.61      7093
   macro avg       0.62      0.62      0.61      7093
weighted avg       0.63      0.61      0.61      7093
```



- using TF-IDF:

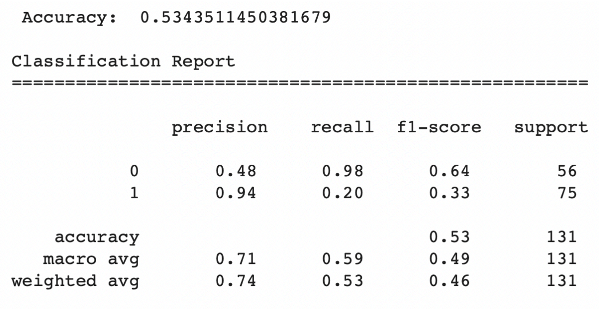


```
 Accuracy:  0.6169462850697871

Classification Report
======================================================

              precision    recall  f1-score   support

           0       0.56      0.73      0.63      3183
           1       0.70      0.53      0.60      3910

    accuracy                           0.62      7093
   macro avg       0.63      0.63      0.62      7093
weighted avg       0.64      0.62      0.62      7093
```

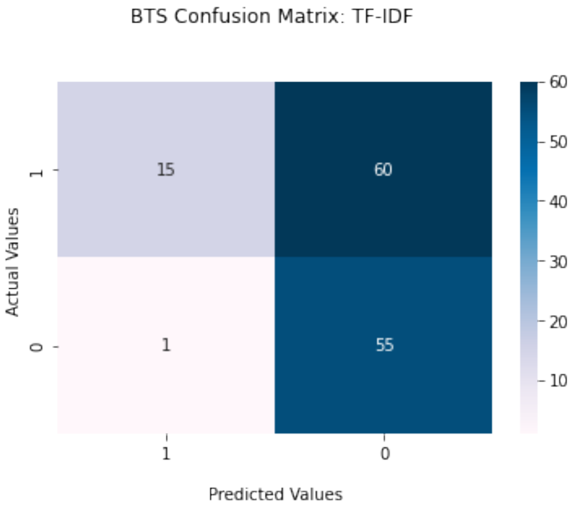


Using classifier on BTS' lyrics:

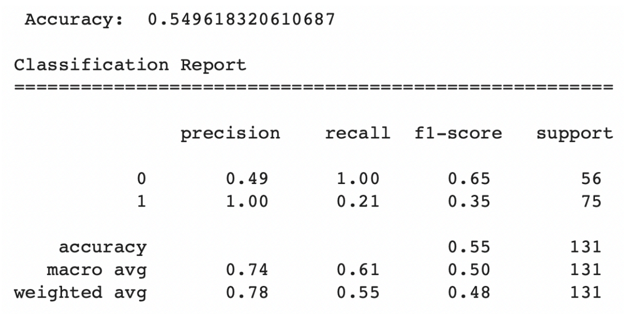
- using word frequency:



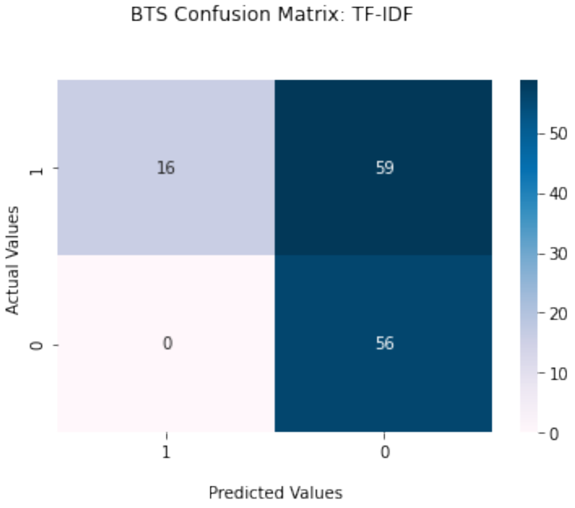
```
 Accuracy:  0.5343511450381679

Classification Report
=====================================================

              precision    recall  f1-score   support

           0       0.48      0.98      0.64        56
           1       0.94      0.20      0.33        75

    accuracy                           0.53       131
   macro avg       0.71      0.59      0.49       131
weighted avg       0.74      0.53      0.46       131
```



- using TF-IDF:



```
 Accuracy:  0.549618320610687

Classification Report
=====================================================

              precision    recall  f1-score   support

           0       0.49      1.00      0.65        56
           1       1.00      0.21      0.35        75

    accuracy                           0.55       131
   macro avg       0.74      0.61      0.50       131
weighted avg       0.78      0.55      0.48       131
```

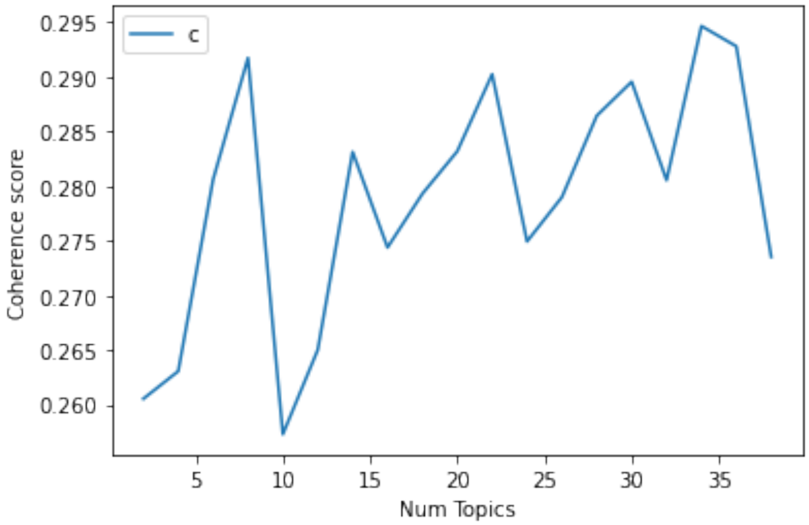


Remarks:

- the use of valence for labeling: Spotify have not revealed how valence is computed, but it uses other musical features as opposed to just lyrics
- computing the feature matrix: BTS lyrics have to be included so that the classifier can be used. This increases the number of features and suggests that the training data's feature vectors will be too sparse (especially if words from the BTS lyrics don't frequently appear in other song lyrics)
- pre-processing:
  - lemmatization: ex. lemmatize("was") —> "wa"
  - Korean words without direct English translations were written phonetically instead

B. Topic modeling

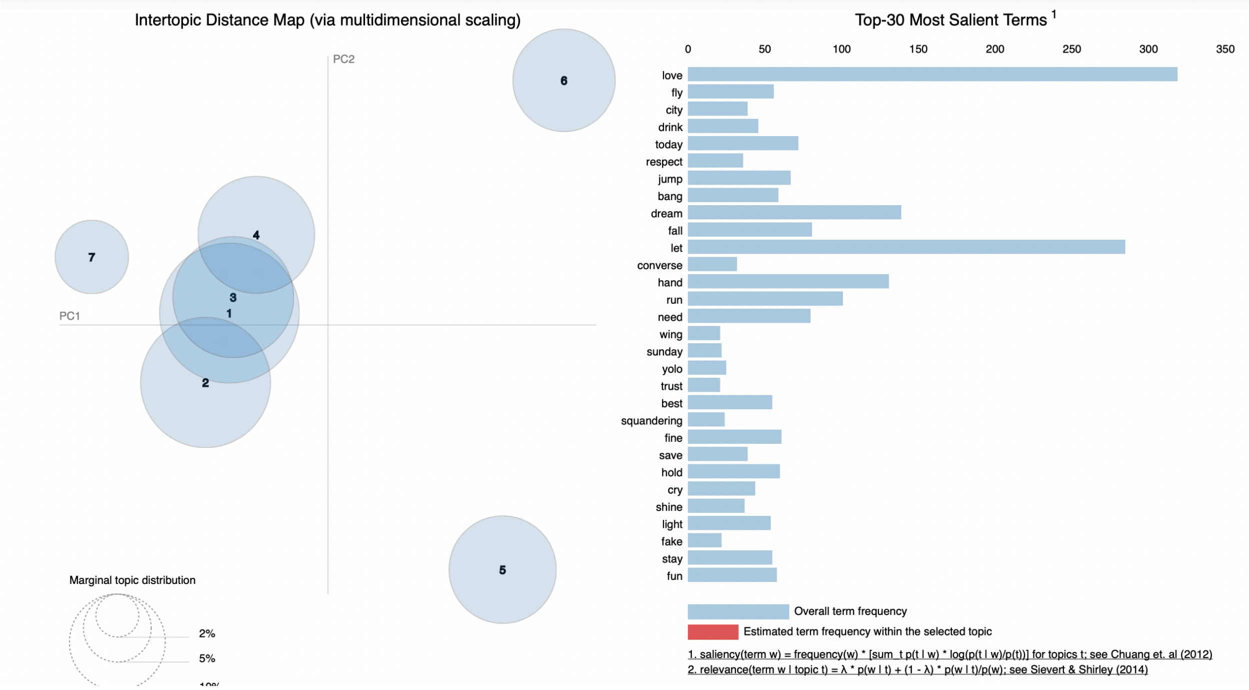
Choosing number of topics: this graph generates different plots every time I run it, so I chose num_topics = 7.



Generated topics + top words for each topic:

topic	top words
1	know, let, day, bang, fine, want, stay, way, fun, moment
2	know, converse, want, save, high, get, stay, yolo, time, squandering
3	dream, say, make, want, world, love, know, life, got, crazy
4	love, know, drink, fall, respect, say, live, look, fake, shot
5	let, jump, get, time, feel, hand, want, look, say, day
6	city, today, fly, hand, wing, night, want, trust, come, shine
7	love, let, run, need, say, keep, day, stop, sunday, turn

Intertopic distance map:

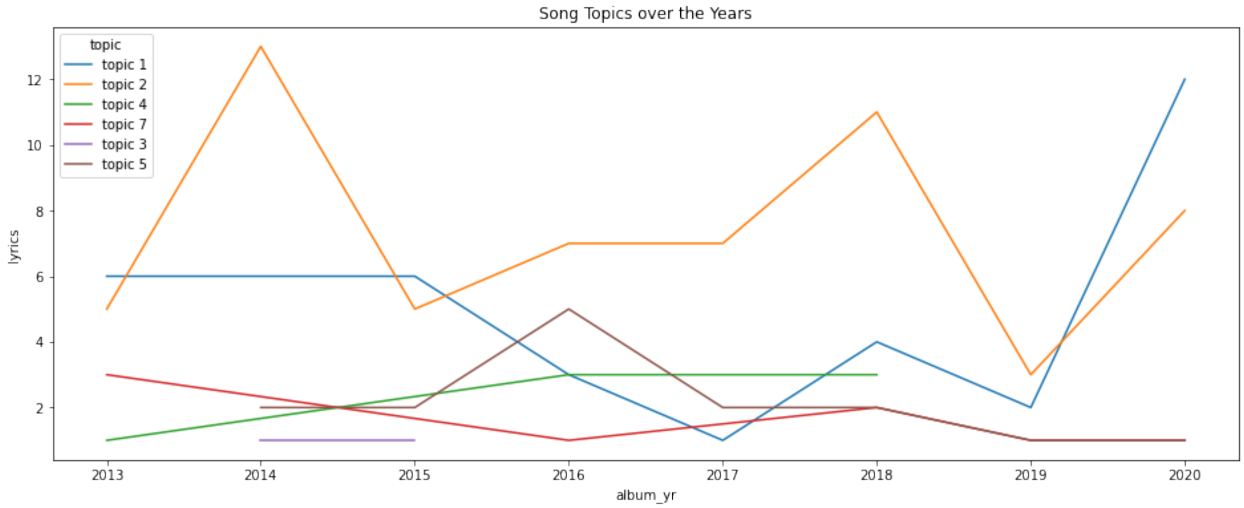


Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Salient Terms [1]

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Plotting song topics over time (by song release date):



Remarks:

- too many songs assigned to 2 topics; one topic had no assigned to it
    - compute topic similarity to improve clustering and topic assignment
- assigning topic names based on top words is not straightforward

References:

- *'The Data Science of "Someone Like You" or Sentiment Analysis of Adele's Songs'* by Preetish Panda, Prompt Cloud. Retrieved from: *https://www.kdnuggets.com/2018/09/sentiment-analysis-adele-songs.html*
- *TIGP SNHCC* - Introduction to Social Networks - Text Mining Lab Homework
- *'Using Machine Learning to Analyze Taylor Swift's Lyrics'* by Ian Freed. Retrieved from: *https://www.codecademy.com/resources/blog/taylor-swift-lyrics-machine-learning/*
- *'Taylor Swift Song Recommendation with NLP and Topic Modeling'* by Annie Shieh. Retrieved from: *https://medium.com/@annieshieh12/taylor-swift-song-recommendation-with-nlp-and-topic-modeling-8594636608fa*
    - Github (source codes): *https://github.com/anshieh12/Taylor_swift_song_recommender*
- *'What Makes A Song Likeable?'* by Ashrit Shetty. Retrieved from: *https://towardsdatascience.com/what-makes-a-song-likeable-dbfdb7abe404*
- *Text Mining with R: A Tidy Approach* by Julia Silge and David Robinson (2017)
- *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions* by Bing Liu (2015)
- *'Data Analysis of K-POP: Playing with Spotify API'* by Nancy Yan. Retrieved from: https://nancyyanyu.github.io/posts/63adf3bb/
- *'The Data Science of K-Pop: Understanding BTS through data and A.I.'* by Haebichan Jung. Retrieved from: https://towardsdatascience.com/the-data-science-of-k-pop-understanding-bts-through-data-and-a-i-part-1-50783b198ac2
- *'Is K-pop still K-pop without the 'K'?'* by Haley Yang. Retrieved from: https://koreajoongangdaily.joins.com/2021/10/12/entertainment/kpop/twice-the-feels-english-kpop-songs-boa-eat-you-up/20211012155647679.html