Kelly Cosgrove
12/09/2021

Pop Punk Song Search
Self-Evaluation

1. Have you completed what you have planned?

    a. I have completed what I planned for the most part. I built a Vector Space Model using a corpus of Pop-Punk song lyrics from the most popular Pop-Punk artists of the past 20 years. I was also able to build an HTML-based GUI that allows users to interact with the VSM.

    b. One limitation I had that I did not originally plan on was only using the top 20 most popular songs by each artist that I included in the corpus. When planning this project out, I had initially thought I would include each artist's **entire** discography. However, due to an issue with the Genius API erroring out pretty regularly, I determined that this would not be possible and had to limit my corpus scope.

    c. I also diverged from a few of the Python libraries I had originally set out to use. To pull song lyrics, I originally planned on using **lyrics-corpora**; however, as stated in my progress report, that library turned out to be very buggy and not usable, so I leveraged **lyricsgenius** instead. Additionally, for the front-end search, I initially wanted to leverage **Dash**, as this was the only front-end Python library I had heard of. However, I quickly learned that Dash is more for displaying data-based dashboards, and my use case did not seem applicable for it. So, I used the **Flask** Python library instead.

2. Have you got the expected outcome? If not, discuss why.

    a. The vector space model does perform reasonably well. In all ten test cases, the VSM did return the expected song in the top 4 best matches, although it was more uncommon to see the expected song as the #1 match. There are definitely ways that I could have improved its performance, like building the VSM corpus with n-gram phrases rather than just Bag-of-Words. However, that did not look possible to implement through Gensim, so I decided to keep the simpler BoW implementation. Overall, I am happy with the performance.