Exploring The Smiths' Albums

The Smiths are one of the most influential bands of the 80s, with a catalog that explores interesting themes such as love, sorrow, and death. I wanted to explore their albums and possibly conduct a sentiment and theme analysis to provide some insight into their catalog.

My original research question was, “What are the main recurring themes, emotional patterns, and linguistic characteristics present in The Smiths' lyrics, and how do these evolve and perform over their albums?” This question turned out to be a little too broad, specifically when asking about linguistic characteristics; in hindsight, it was unrealistic. The question I was actually able to answer was, “What thematic patterns are present within The Smiths' lyrics and albums, and how do they evolve over time?” I found the easiest way to store each album was a combination of dataframes and classes to allow me to store the album data and create functions to explore the album contents.

I wanted to keep their catalog organized, so I decided to do so with two classes, Album and Song, which would store all the information gathered. The Album class is the main one, which holds the songs in the album with these attributes:

- **Name**: The title of the album.
- **Artist**: The artist's name, which in this case is always The Smiths.
- **Release Date**: The date the album was released.
- **Track list**: A list of the tracks on the album, each of which is its own class containing attributes like name, theme, lyrics, and sentiment.
- **Overall Theme**: The overarching theme of each album.

The Album class has two main methods to describe the album and show the project results in an easily accessible way:

1. **description**: A summary of the album that includes the album's name, artist, release date, and the number of tracks.
2. **full_description**: This method creates a detailed description of the album, including the full tracklist, where each song's name, duration, and theme are displayed. This method also displays the

album cover using IPython display. I decided to include this method because, after gathering all the information, the project still felt unfinished; I felt this at least gave a way to display the findings.

The data was initially obtained using Spotify to get The Smiths' catalog. This caused issues due to many albums that didn't contain new songs. I worked around this by creating a list of the albums I knew contained original songs and filtering out only those. I would have liked to find a better solution, but at the time, it was causing too many issues, and I felt it was best to move on. Once I got the album names, I continued using the Spotify API to get the song names. At this point, I had all the information about the albums and songs except for lyrics, so I initialized my classes for Album and Song. I then used Genius to get the lyrics for each song and add them to their respective objects.

Once I had the lyrics, I had to clean them by doing things like removing stop words and tokenizing them. Some lyrics also contained a lot of markdown and descriptions, which I filtered out based on their presence. The general lyrics all needed stop words removed, but I ensured that I kept the raw lyrics for future reference. Tokenizing the lyrics allowed me to perform theme analysis and sentiment analysis with only the words that provided content or value. I found that this removed words like "you" and "me," which I thought were important since, when referring to “you,” songs were often love songs. I did, however, avoid using the raw lyrics in theme analysis since it found most songs were love songs solely due to the word "you," which, while often correct for The Smiths, removed some thematic depth.

After obtaining clean lyrics, I moved on to sentiment analysis. The first song I analyzed was “Heaven Knows I’m Miserable Now” because I wanted to know how the sentiment analysis modules would perform. It gave the song a 0.97 compound score, indicating a positive sentiment, which I disagreed with. I still conducted sentiment analysis for all the songs since I felt it added a good layer to the project, though I wouldn’t necessarily rely on it. In the future, I’d like to explore how an LLM would perform and how easy it would be to implement one, as ChatGPT seemed to understand the song’s sentiment well.

Since I disagreed with some results from the sentiment analysis, even after trying multiple modules, I created a function to identify themes in the lyrics. I did this using four main themes: sorrow, death, love, and joy—these were the themes I felt best represented the discography while remaining manageable. I created a dictionary with these themes and associated keywords. I was then able to identify themes for each song by seeing which keywords were most frequent. I felt these themes better represented the songs

than the sentiment scores. In the future, I would like to use more categories, allowing for more specific themes.

The most common theme across all albums was Love, with 30 out of 59 songs having that theme—this was after removing the keyword “you,” which initially made almost every song a love song. The second most common theme was sorrow/sadness, with 15 songs. Overall, I agree with the themes assigned to each song, as much of their discography focuses on love with a generally sad undertone. This sentiment was something that couldn’t be fully captured by any of the tools I used; sentiment analyzers often rated the songs as positive when they weren’t. This was expected since much of the sadness is conveyed through metaphor. Creating a tool that could understand this would be an interesting future project.

After gathering themes for each song and deciding on each album’s overall theme based on its songs, I looked at the most common lyrics. I then used the WordCloud module to visualize this, creating word clouds for specific songs, albums, and the discography as a whole. Surprisingly, the most common word was “oh,” which is technically a filler word but greatly shapes the band’s style. Considering the prevalence of filler words across songs could make for an interesting project, especially in rock genres.

Along with the word cloud I also created a bar chart of themes amongst songs to visualize which themes were the most prevalent. I also created a line chart of themes over time which showed some interesting trends. At the beginning of their catalog the most common theme was love, which in itself is not a surprise as it's the most common theme in general. The interesting thing is that during the middle of their catalog sorrow/sadness was actually the most prevalent theme. This led me to look at what happened during this time and I found their band was facing some tension with one of them getting arrested due to their heroin addiction. The end of their catalog actually has majority love songs which was interesting after finding out they were beefing. This was explained by the fact this album wasn't actually new and was just a compilation of songs they hadn't officially released. With all this information you can see the story of how they came together making love songs but as time passed and they started growing apart this showed in their music through the sad undertones.

This project explored The Smiths' discography, revealing recurring themes of love, sorrow, and death that characterize their music. Using a structured approach with classes to organize albums and songs, and applying both sentiment and theme analysis, I discovered how the thematic focus evolved across albums. While sentiment analysis highlighted emotional tones, it missed the nuanced irony in The Smiths' lyrics.

However, thematic analysis and WordCloud visualizations were effective, showcasing central themes and unique elements like filler words that shape the band's style. Future expansions could include creating a more accurate sentiment model to capture complex emotions and themes more effectively.