Roger White
DATA 340

Natural Language Processing in the Context of Music Recommendation Systems

---

Abstract

This project aims to analyze wikipedia pages of musical artists and identify other artists mentioned in the page in the hopes of aiding in the efficacy of content based filtering music recommendation algorithms. The methodology of this project is as follows: 1) downloading wikipedia pages for musical artists, 2) processing and cleaning the text, 3) collecting all other wikipedia links in the page, 4) building a model to predict whether the links belong to other musical artists. Using a dataset of roughly 300 observations, I was able to construct a model with an accuracy score above 80%. This means that given any wikipedia page, the model was able to accurately predict whether it belonged to a musical artist 80% of the time. This could have potential implications for content based filtering music algorithms as this could allow for additional metadata to be incorporated into such models.

Introduction

The medium of music consumption has changed markedly over the past decade. For most of modern history, music has been consumed via a physical medium (vinyl, cd, cassette tape)[1]. As a result, the process in which individuals discovered new music also took place via a physical medium. Whether it was through word of mouth or magazine, individuals were largely in charge of their own music discovery process. Since 2010 however, the medium of music consumption known as *streaming* has grown in popularity. In 2022, 32% of music was consumed via a streaming service such as Spotify or Apple Music, and of this category, 74% were internet users[2]. What this indicates is that access to the internet is likely to affect one's choice in medium, and as global internet usage continues to grow, so will the popularity of streaming services. Streaming services have undoubtedly profited from such trends, with companies like Spotify being valued at over 30 billion dollars on the public market, and Apple Music reportedly bringing in nearly 7 billion dollars in annual revenue for 2021[3].

As the industry begins to consolidate, the primary differentiating factor between companies is how well they can retain users. Accessibility, ease of use, and content all affect a

[1](Richter)
[2] (Cooke)
[3] (Shewale)

service's user retention, but a large component is also their ability to recommend music that their users will enjoy. As a result, recommendation algorithms have become extremely valuable to streaming services. With this being said, reproducibility is a concern of these platforms, thus in depth discussion regarding how these algorithms work can only be left to speculation. What is known however, is the general approaches that platforms take in designing recommendation algorithms. The first is known as **collaborative based filtering**. This is known as the 'Netflix approach' due to the company pioneering this design in order to effectively recommend TV-shows and movies to their viewers. The design is as follows: a model predicts how likely a user is to enjoy a song based on similarities between user listening profiles. To better understand this, picture two listeners: listener A and listener B. If listener A listens to songs X,Y, and Z, while listener B listens to songs X and Z, the goal of a collaborative based filtering recommendation algorithm would be to identify the similarities between these listener profiles and recommend listener B song Y. The second approach is known as **content based filtering**. This approach attempts to identify content similarities between songs, such as similarities in tempo, instrumentation, and arrangement[4]. These are all variables that can be derived from the raw audio signals of the track itself, but metadata such as artist name and release date can also be helpful in identifying track similarities. This project is aiming to capture additional metadata surrounding artists, that is: an artist's influences and collaborators. The idea is that if a listener likes a particular artist, it might help improve a recommendation algorithm's efficacy if it can also take into account which artists the artist in question has been influenced by or collaborated with. The logic here is that artists that influence or collaborate with a particular artist are likely to produce similar sounding music, thus being of interest to listeners who prefer that particular artist.

Methodology/Dataset

To identify influences or collaborators for a particular artist, I first start by collecting public wikipedia data. To do so in python, I have utilized the Beautiful Soup and Requests library.

```
from bs4 import BeautifulSoup
import requests

url = 'https://en.wikipedia.org/wiki/Tame_Impala'

page = requests.get(url)
soup = BeautifulSoup(page.content, "html.parser")
```

Downloading the data

[4] (Pastukhov)

This snippet of code downloads the wikipedia page for the artist ‘Tame Impala’. The page is then processed from HTML to text using Beautiful Soup.

```
content_container = soup.find('div', {'class': 'mw-parser-output'})

for element in content_container.find_all(['p','h2']):
   if element.name == 'h2' and element.find('span', {'id': 'References'}):
     break
   if element.name == 'p':
       content.append(element.get_text())

 content_str = ''

 for i in content:
   content_str += i
```

Processing the data

Now that we have the wikipedia page for the artist, we need to then process the page such that the text data can be used for further analysis. To start, we don’t want to use the entire page as there is a lot of text that won’t be used. In this code snippet, I’ve decided to create a string of all the text from the title section up until but not including the references section. Because all wikipedia pages have a standardized format, this code can be used for any wikipedia page.

```
import re

Content = content_str.replace('\n','')
Clean_content = Content.replace('\\','')
Cleaned_text = re.sub(r'\[\d+\]', '', Clean_content)
```

Cleaning the data

Now that we have our wikipedia page content as a string, there is further processing that can be done. Here, I have decided to remove three items that frequently occur within the content of wikipedia pages: 1) newline characters, 2) backslashes, and 3) footnote references. None of these items will be necessary for analyzing the content of artist wikipedia pages. For removing the footnote references (they take the format of; [n]), I have utilized the regular expressions library.

At this point, we have a full body of wikipedia page content in the format of a string that is ready to be processed for analysis. One thing I noticed whilst processing several wikipedia pages is that the first sentence of every page is a brief description of what the page is about. This standardized format is incredibly helpful, as the first sentence of any wikipedia page should be enough for a model to identify whether the page belongs to an artist or not.

```
sentences = re.split(r'(?<=[.!?]) +', clean_text)
sentences[0]
```

Retrieving the first sentence of the page

After going through several music related wikipedia pages and extracting the first sentence of each, I then classify the pages based on whether they belong to an artist or not. Take the two examples from the dataset below.

	Page Title	Description	Class
12	Nick Allbrook	Nicholas Allbrook (born 23 November 1987) is an Australian psychedelic rock musician, singer, and songwriter.	1

Nick Allbrook is classified as an artist and assigned a class variable of 1

30	Triple J	Triple J is a government-funded, national Australian radio station that began broadcasting in 1975 as a division of the Australian Broadcasting Corporation (ABC).	0

Triple J is classified as a non-artist and assigned a class variable of 0

One particular strength of this methodology is that almost all of the observations are music related. In the example above, Triple J is a music related proper noun, but not an artist. If the data were strictly artist pages and non-music related pages, the model may confuse music related pages with artist pages as a result of having similar descriptions.

After repeating this process with over 300 wikipedia pages, the resulting dataset consists of 312 observations, 101 being artist pages, and the remaining 211 being non-artist pages. This data is then used to train a classification model.

```
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.svm import SVC
```

```
text_vectorizer = CountVectorizer()
X_text = text_vectorizer.fit_transform(cleaned_data['Description'])
y = cleaned_data['Class']

X_train, X_test, y_train, y_test = train_test_split(X_text, y, test_size=0.8)

clf = SVC()
clf.fit(X_train, y_train)
```

Training the model

Here, I have decided to utilize Scikit-learn's C-Support Vector Classification algorithm. In order to do so, I vectorize the description text and fit the model onto the dataset. I then utilize a train-test split with 20% of the data used to train the model and the remaining 80% to test it. This is to ensure that the model isn't overfit.

Results

The main highlight of this section is the accuracy of the model. Depending on the random state, the model was able to predict whether a wikipedia page belonged to an artist roughly 80% of the time, which is quite impressive given the lack of input features and relatively small dataset. It is not likely that this accuracy score is bolstered by the model being overfit, as the train test split of 20-80 respectively should prevent this from happening. Instead, the model's accuracy can largely be attributed to wikipedia's standardized format and similarities between musical artist descriptions.

8	Pond (Australian band)	Pond are an Australian psychedelic rock band from Perth, Western Australia, formed in 2008.	1
9	Kevin Parker (musician)	Kevin Richard Parker (born 20 January 1986) is an Australian singer, songwriter, musician, record producer, and DJ, best known for his musical project Tame Impala, for which he writes, performs, records, and produces the music.	1
10	Jay Watson	Jay Watson is an Australian multi-instrumentalist, producer, singer and songwriter.	1

In the sub-section of the dataset above, one can see how the presence of certain words like 'singer','musician', or 'band' can be dead giveaways for a page belonging to a musical artist.

Discussion

This project has proven that wikipedia pages for artists can be a helpful resource for identifying an artist's collaborators or influences thanks to the standardized format that wikipedia utilizes. Given the model, one is able to take a wikipedia page for an artist and identify other

artists mentioned within that artists wikipedia page. As previously mentioned, the hope is that the artists identified in a page produce similar sounding music to the artist in question, but this may not always be the case.

Using the example of Tame Impala, while many artists mentioned on this page are influences and collaborators as seen above (Pond, Jay Watson), some are not. Take Aphex Twin for example. The context of this artist being mentioned in Tame Impala's wikipedia page is as a prospective collaborator, not a past collaborator. Utilizing such a model then without taking into account the context in which an artist is mentioned might produce unwarranted results. There might be two ways to deal with this problem.

The first might be to predict whether the artist mentioned is an influence or collaborator using the sentence in which the artist is mentioned as an input feature for a separate model. One problem with this is that such categories are much more difficult to classify than the categories our current model predicts, since there is no standardized format for artists that are mentioned in a wikipedia page (there are very few similarities between the context in which artists and influences are mentioned). Another problem with this approach is that influences and collaborators may not lead an artist to producing similar sounding music, thus to a music recommendation algorithm, such information would be irrelevant.

The second way to approach this problem is to track the term frequency values for artists mentioned in a wikipedia page. By doing so, it wouldn't necessarily matter if an artist is a collaborator or influence. If they are mentioned at a disproportionately high frequency, then it is likely that that artist is similar to the artist in question. A problem with this is that more popular artists may be mentioned more frequently, so in order to combat this problem one must take into account the popularity of an artist and measure the term frequency relative to that artist's popularity.

While there are some challenges to this concept, one must remember the broader context in which a model like this might be used. As previously discussed, music recommendation algorithms are quite sophisticated and take into account many different variables. This project isn't looking to replace the existing systems in place, but rather to aid them. In essence, this work could help add important metadata to a content-based recommendation algorithm. Despite having minimal impact to recommendation systems as a whole, thanks to the importance of such algorithms, this additional resource for content-based filtering could be the difference between market dominance and insolvency.

Conclusion

This project has proven that by utilizing natural language processing techniques, Wikipedia can be a helpful resource in identifying additional metadata to be used in content based filtering music recommendation algorithms. Given the existing trends in the music industry, it's likely that streaming services will only continue to grow as it becomes more and

more convenient for internet users to consume music via such a medium. As a result, this project's relevancy will only continue to grow in the near future.

References

Cooke, Chris. "Music consumption at all time high powered by streaming and video apps." *Complete Music Update*, 17 November 2022, https://archive.completemusicupdate.com/article/music-consumption-at-all-time-high-powered-by-streaming-and-video-apps/. Accessed 23 December 2023.

Pastukhov, Dmitry. "How Spotify's Algorithm Works? A Complete Guide to Spotify Recommendation System [2022]." *Music Tomorrow*, 9 February 2022, https://www.music-tomorrow.com/blog/how-spotify-recommendation-system-works-a-complete-guide-2022. Accessed 23 December 2023.

Richter, Felix. "Infographic: From Tape to Tidal: 4 Decades of U.S. Music Sales." *Statista*, 24 June 2022, https://www.statista.com/chart/17244/us-music-revenue-by-format/. Accessed 23 December 2023.

Shewale, Rohit. "Apple Music Statistics For 2024 (Usage & Financials)." *DemandSage*, 8 December 2023, https://www.demandsage.com/apple-music-statistics/. Accessed 23 December 2023.