Sentiment Analysis by Gender on 2020's Top Charting Albums

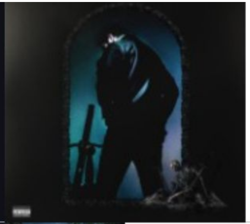
By Niki Vasan and Rose Feng

What can music streaming behavior tell us about gender-based societal biases and preferences?

Related Research

1. Anglada-Tort, Manuel & Krause, Amanda & North, A.(2019). *Popular music lyrics and musicians' gender over time: A computational approach. Psychology of Music*. 49. 426-444. 10.1177/0305735619871602.
2. Barman, M. P., Awekar, A., & Kothari, S. (2019, July). Decoding the style and bias of song lyrics. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 1165-1168).
3. Lytle, B. (2019, December 31). *Using natural language processing to analyze Spotify 2019 top global artists*. Medium. Retrieved November 11, 2021, from https://briannalytle7.medium.com/using-natural-language-processing-to-analyze-spotify-2019-top-global-artists-e5449d8b5133.

Stage 1: Corpus Creation

1		**Hollywood's Bleeding** Post Malone
2		**My Turn** Lil Baby
3		**Please Excuse Me For Being Antisocial** Roddy Ricch
4		**Fine Line** Harry Styles
5		**Folklore** Taylor Swift

Stage 1: Corpus Creation

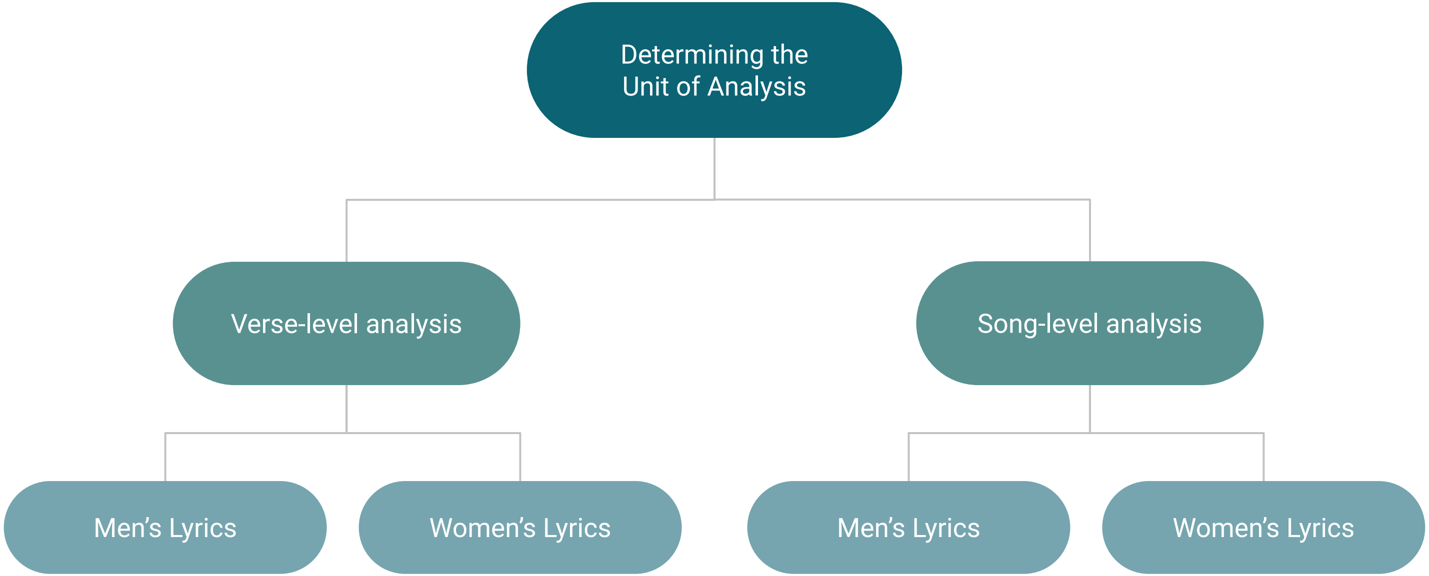
```
# ****************************** get json file of track info in the album******************************************
albumtitle = genius.search_album(album_title, artist_name) # use lyricsgenius wrapper to genius api search query
album_title = re.sub("[^a-zA-Z0-9]", "", album_title) # remove spaces in input album title
albumtitle.save_lyrics() # write json output to a file

# ****************************get dictionary of song titles and album artist**************************************
album_dict = {}
with open(f'Lyrics_{album_title}.json') as a:
  albumtitle_json = json.load(a) # load album object
  song_artist = albumtitle_json['artist']['name'] # use artist from the album not from song (remove features)
  for track in albumtitle_json['tracks']:
    song_title = track['song']['title']
    song_title = re.sub('\u200b', '', song_title) # remove unnecessary characters before song title
    album_dict[song_title] = song_artist # key: song title, value: artist
#print(album_dict)

 # *************************get a dictionary of songs and lyrics urls**********************************************
album_url_dict = {} # dictionary to store urls
for item in album_dict: # iterate through artist and song title
    song_title = item
    song_artist = album_dict[item]
    # URL for a search via the Genius API:
    genius_search_url = f'http://api.genius.com/search?q={song_title}&access_token={client_access_token}'
    # API call
    resp = requests.get(genius_search_url)
    data = resp.json() # save as json
    # now search for match w/ artist
    for song in data['response']['hits']:
        if song['result']['primary_artist']['name'] == song_artist:
            # if there's a match, get the url
            lyrics_url = song['result']['url']
            album_url_dict[song_title] = lyrics_url
            # status update so we can make sure we're on the right track...
            print("Matched! Artist: " + song_artist + " and title: " + song_title)
```

Stage 1: Determining the Unit of Sentiment Analysis

Comparing Absolute Differences → Human Labeling Approach

	Song-Names	Song-Artists	HL Class	Song-Level-Scores	Verse-Level-Scores	Absolute-Difference	Accurate-Y-N
0	Hollywood's Bleeding	Post Malone	Strongly Negative	0.7503	-0.244050	0.506250	N
1	Can't Die	Juice WRLD	Moderately Negative	0.9898	0.417250	0.572550	N
2	For the Night	Pop Smoke	Neutral	-0.9979	-0.914933	0.082967	N
3	High Fashion	Roddy Ricch	Moderately Positive	0.9482	0.109150	0.839050	Y
4	Watermelon Sugar	Harry Styles	Strongly Positive	0.9502	0.203763	0.746438	Y
5	When the Party's Over	Billie Eilish	Strongly Negative	0.9964	0.452560	0.543840	N
6	Playing Games	Summer Walker	Moderately Negative	0.9180	0.436200	0.481800	N
7	Juicy	Doja Cat	Neutral	0.9977	0.887420	0.110280	N
8	Happiness Over Everything (H.O.E)	Jhene Aiko	Moderately Positive	-0.9879	-0.801300	0.186600	N
9	Soulmate	Lizzo	Extremely Positive	0.9999	0.793844	0.206056	Y

Stage 2: Sentiment Analysis



Result: Album Dataframes

Men

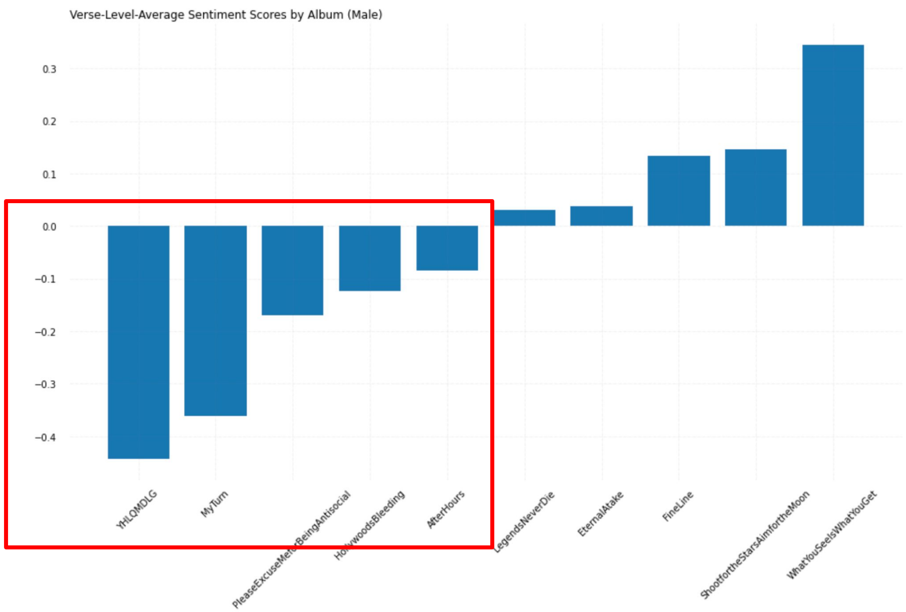
Album	Verse-Level-Average	Song-Level-Average
HollywoodsBleeding	-0.123975	-0.089721
MyTurn	-0.361158	-0.499138
PleaseExcuseMeforBeingAntisocial	-0.169493	-0.177093
FineLine	0.134158	0.328936
EternalAtake	0.036813	-0.391412
ShootfortheStarsAimfortheMoon	0.146611	0.237026
AfterHours	-0.084088	-0.218371
LegendsNeverDie	0.030895	-0.123156
WhatYouSeeIsWhatYouGet	0.345420	0.596062
YHLQMDLG	-0.443686	-0.946820

Women

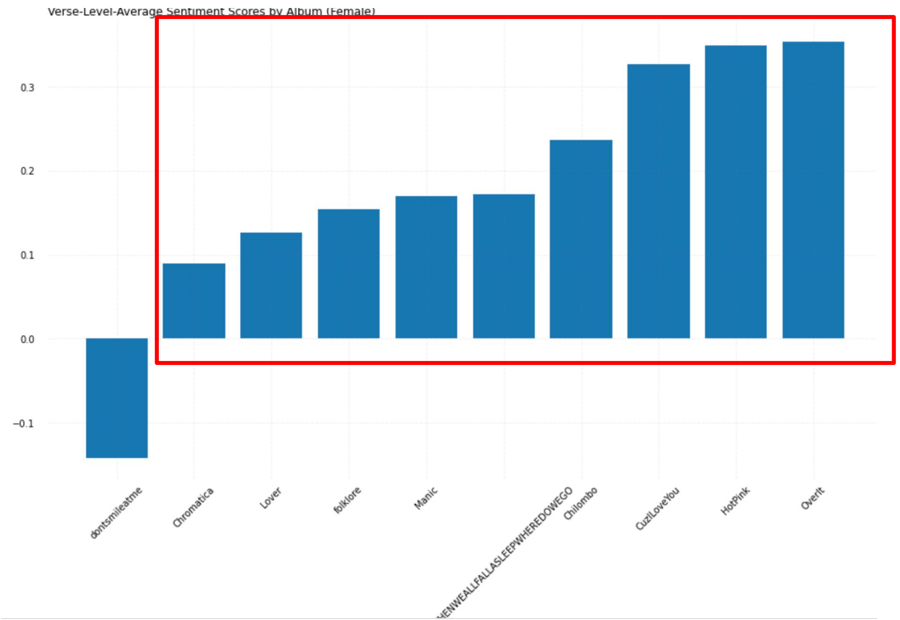
Album	Verse-Level-Average	Song-Level-Average
folklore	0.154577	0.224569
WHENWEALLFALLASLEEPWHEREDOWEGO	0.172262	0.215015
Lover	0.126749	0.201129
OverIt	0.353774	0.341845
Manic	0.169664	0.099992
Chilombo	0.236165	0.187667
HotPink	0.349611	0.620627
CuzILoveYou	0.326391	0.367350
dontsmileatme	-0.141817	-0.435900
Chromatica	0.089364	0.167908

Result: Verse-Level Score Distributions

Men



Women



Next Steps

1. Evaluate efficacy of sentiment analysis
2. Analyze results across genres
3. If time, try a different sentiment analyzer or a run a topic model

Thank you!

Any questions?