

```
In [1]: 1 import json
        2 import re
        3 from ast import literal_eval
```

Open json file with songs + albums + lyrics

```
In [2]: 1 with open('lyrics_taylor_swift.json') as f:
        2     lyrics_list = json.load(f)
```

Add correct albums for songs with mislabeled albums

```
In [3]: 1 for song in lyrics_list:
2     if song['song_title'] == 'you need to calm down ':
3         song['album'] = 'lover'
4     if song['song_title'] == 'lover ':
5         song['album'] = 'lover'
6     if song['song_title'] == 'blank space ':
7         song['album'] = '1989 deluxe'
8     if song['song_title'] == '...ready for it? ':
9         song['album'] = 'reputation'
10    if song['song_title'] == 'willow ':
11        song['album'] = 'evermore'
12    if song['song_title'] == 'cruel summer ':
13        song['album'] = 'lover'
14    if song['song_title'] == 'the archer ':
15        song['album'] = 'lover'
16    if song['song_title'] == 'untouchable ':
17        song['album'] = 'fearless'
18    if song['song_title'] == 'the man ':
19        song['album'] = 'lover'
20    if song['song_title'] == 'bad blood ':
21        song['album'] = 'reputation'
22    if song['song_title'] == 'miss americana & the heartbreak prince ':
23        song['album'] = 'lover'
24    if song['song_title'] == 'london boy ':
25        song['album'] = 'lover'
26    if song['song_title'] == 'cornelia street ':
27        song['album'] = 'lover'
28    if song['song_title'] == 'wildest dreams ':
29        song['album'] = '1989 deluxe'
30    if song['song_title'] == 'daylight ':
31        song['album'] = 'lover'
32    if song['song_title'] == 'death by a thousand cuts ':
33        song['album'] = 'lover'
34    if song['song_title'] == 'i forgot that you existed ':
35        song['album'] = 'lover'
36    if song['song_title'] == 'paper rings ':
37        song['album'] = 'lover'
38    if song['song_title'] == 'false god ':
39        song['album'] = 'lover'
40    if song['song_title'] == 'i think he knows ':
41        song['album'] = 'lover'
42    if song['song_title'] == 'afterglow ':
```

```
43     song['album'] = 'lover'
44     if song['song_title'] == "soon you'll get better ":
45         song['album'] = 'lover'
46     if song['song_title'] == 'shake it off ':
47         song['album'] = '1989 deluxe'
48     if song['song_title'] == 'its nice to have a friend ':
49         song['album'] = 'lover'
50     if song['song_title'] == 'everything has changed ':
51         song['album'] = 'red deluxe edition'
52     if song['song_title'] == 'mean ':
53         song['album'] = 'speak now'
54     if song['song_title'] == 'fifteen ':
55         song['album'] = 'fearless'
56     if song['song_title'] == 'sparks fly ':
57         song['album'] = 'speak now'
58     if song['song_title'] == 'picture to burn ':
59         song['album'] = 'taylor swift'
60     if song['song_title'] == 'the story of us ':
61         song['album'] = 'speak now'
62     if song['song_title'] == 'ours ':
63         song['album'] = 'speak now'
64     if song['song_title'] == 'tim mcgraw ':
65         song['album'] = 'taylor swift'
66     if song['song_title'] == 'cold as you ':
67         song['album'] = 'taylor swift'
68     if song['song_title'] == 'tied together with a smile ':
69         song['album'] = 'taylor swift'
70     if song['song_title'] == 'the other side of the door ':
71         song['album'] = 'fearless'
72     if song['song_title'] == 'jump then fall ':
73         song['album'] = 'fearless'
74     if song['song_title'] == 'if this was a movie ':
75         song['album'] = 'speak now'
76     if song['song_title'] == 'im only me when im with you ':
77         song['album'] = 'taylor swift'
78     if song['song_title'] == 'superstar ':
79         song['album'] = 'fearless'
80     if song['song_title'] == 'come in with the rain ':
81         song['album'] = 'fearless'
82     if song['song_title'] == 'superman ':
83         song['album'] = 'speak now'
84     if song['song_title'] == 'today was a fairytale ':
85         song['album'] = 'fearless'
```

Delete unnecessary elements

```
In [4]: 1 albums = ["taylor swift", "fearless", "speak now", "red deluxe edition", "1989 deluxe", "reputation"]
        2
        3 for song in lyrics_list:
        4     if song['album'] not in albums:
        5         lyrics_list.remove(song)
```

```
In [5]: 1 for song in lyrics_list:
        2     if song['album'] == 'unknown':
        3         lyrics_list.remove(song)
```

Create dictionary with songs and albums

```
In [6]: 1 albums = ["taylor swift", "fearless", "speak now", "red deluxe edition", "1989 deluxe", "reputation"]
        2 song_album_dic = {}
        3 for song in lyrics_list:
        4     if song['album'] in albums:
        5         song_album_dic[song['song_title'].strip()] = song['album']
        6     else:
        7         pass
```

Create dictionary with songs and lyrics

```
In [7]: 1 lyrics_dict = {}
        2
        3 for song in lyrics_list:
        4     lyrics_dict[song['song_title'].strip()] = song['lyrics']
```

Define function to split camelCase lyric string

```
In [8]: 1 def split_camelcase(lyric):
        2     return re.findall(r'[A-Z]?[a-z]+|[A-Z]+(?:[A-Z]|$)', lyric)
```

Run function to split camelCase lyric strings

```
In [9]: 1 for song, lyrics in lyrics_dict.items():
        2     new_lyrics = []
        3     lyric = [lyric.replace(lyric, str(split_camelcase(lyric))) for lyric in lyrics]
        4     new_lyrics.extend(lyric)
        5     lyrics_dict[song] = new_lyrics
```

Remove double quotation marks

```
In [10]: 1 for song, lyrics in lyrics_dict.items():
        2     new_lyrics = []
        3     for lyric in lyrics:
        4         lyric = literal_eval(lyric)
        5         new_lyrics.append(lyric)
        6         lyrics_dict[song] = new_lyrics
```

Merge individual word lists to create one list for each song

```
In [11]: 1 for song, lyrics in lyrics_dict.items():
        2     new_lyrics = []
        3     for lyric in lyrics:
        4         for word in lyric:
        5             new_lyrics.append(word)
        6             lyrics_dict[song] = new_lyrics
```

Save new json file

```
In [12]: 1 with open("song_lyrics_dict.json", 'w') as f:
        2     json.dump(lyrics_dict, f, indent=4)
```

Import tools for sentiment analysis

```
In [13]: 1 import pandas as pd
          2 from nltk.tokenize.treebank import TreebankWordDetokenizer
          3 from helpers import *
          4 from sklearn.feature_extraction.text import CountVectorizer
          5 import numpy as np
          6 import matplotlib.pyplot as plt
          7 from nltk.sentiment.vader import SentimentIntensityAnalyzer
```

Detokenize lyrics

```
In [14]: 1 corpus = {}
          2 for song, lyrics in lyrics_dict.items():
          3     detokenized_lyrics = TreebankWordDetokenizer().detokenize(lyrics)
          4     # print(detokenized_lyrics)
          5     corpus[song] = [detokenized_lyrics.lower()]
          6     # print(corpus)
```

Create dataframe

```
In [15]: 1 df = pd.DataFrame(data=corpus).T.rename(columns={0: "lyrics"})
```

```
In [16]: 1 df
```

```
Out[16]:
```

	lyrics
cardigan	vintage tee brand new phone high heels on cobb...
exile	i can see you standing honey with his arms aro...
lover	we could leave the christmas lights up til jan...
the 1	m doing good m on some new shit been saying ye...
look what you made me do	i don t like your little games don t like your...
...	...
perfect have i loved	if you love me then i love you swear by the fr...
matches	i shined your shoes i poured the wine i gave y...
check out this view	i dont need a lot but there s one thing love h...
sunshine	i walk in the room there s a beautiful sunshin...
you're not sorry (taylor's version)	all this time i was wasting hoping you would c...

212 rows × 1 columns

Find unique words in lyrics

```
In [17]: 1 def find_unique(list_of_words):  
2     unique_list = []  
3     for word in list_of_words:  
4         if word not in unique_list:  
5             unique_list.append(word)  
6     return unique_list
```

```
In [18]: 1 words = []  
2 for word in df['lyrics'].tolist():  
3     words.append(find_unique(lyrics_to_words(word).split()))
```

Add column of unique words to dataframe

```
In [19]: 1 df['words'] = words
          2 df
```

Out[19]:

		lyrics	words
	cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...
	exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...
	lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...
	the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...
	look what you made me do	i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...

	perfect have i loved	if you love me then i love you swear by the fr...	[love, swear, freckle, moon, maybe, enough, li...
	matches	i shined your shoes i poured the wine i gave y...	[shined, shoe, poured, wine, gave, time, patch...
	check out this view	i dont need a lot but there s one thing love h...	[dont, need, lot, one, thing, love, taught, ne...
	sunshine	i walk in the room there s a beautiful sunshin...	[walk, room, beautiful, sunshine, ray, across,...
	you're not sorry (taylor's version)	all this time i was wasting hoping you would c...	[time, wasting, hoping, would, come, around, g...

212 rows × 2 columns

Add album column to dataframe


```
In [20]: 1 df['album'] = [song_album_dic[song_name] if song_name in song_album_dic else "Other" for song_name in df.index]
2 df
```

```
Out[20]:
```

		lyrics	words	album
	cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...	folklore
	exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...	folklore
	lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...	lover
	the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...	folklore
	look what you made me do	i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...	reputation

	perfect have i loved	if you love me then i love you swear by the fr...	[love, swear, freckle, moon, maybe, enough, li...	Other
	matches	i shined your shoes i poured the wine i gave y...	[shined, shoe, poured, wine, gave, time, patch...	Other
	check out this view	i dont need a lot but there s one thing love h...	[dont, need, lot, one, thing, love, taught, ne...	Other
	sunshine	i walk in the room there s a beautiful sunshin...	[walk, room, beautiful, sunshine, ray, across,...	Other
	you're not sorry (taylor's version)	all this time i was wasting hoping you would c...	[time, wasting, hoping, would, come, around, g...	Other

212 rows × 3 columns

Delete unnecessary rows from dataframe and reset index

```
In [21]: 1 indices_to_delete = df[df['album'] == 'Other'].index
2 df.drop(indices_to_delete, inplace = True)
```

```
In [22]: 1 df.to_csv('new_lyrics_words_albums.csv')
```

```
In [23]: 1 indexed_df = df.reset_index()  
2 indexed_df
```

Out [23]:

		index	lyrics	words	album
0		cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...	folklore
1		exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...	folklore
2		lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...	lover
3		the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...	folklore
4	look what you made me do		i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...	reputation
...	
162	taylor swift's reputation (reputation tour intro)		taylor swift taylor swift taylor swift taylor ...	[taylor, swift, declined, address, rumor, glob...	taylor swift
163	long live/new year's day		i said remember this moment in the back of my ...	[said, remember, moment, back, mind, time, sto...	taylor swift
164	i did something bad (reputation tour remix)		i never trust a narcissist oh but they love me...	[never, trust, narcissist, oh, love, play, em,...	taylor swift
165	state of grace (acoustic version)		m walking fast through the traffic lights busy...	[walking, fast, traffic, light, busy, street, ...	red deluxe edition
166	treacherous (original demo recording)		put your lips close to mine as long as they do...	[put, lip, close, mine, long, dont, touch, foc...	red deluxe edition

167 rows × 4 columns

```
In [24]: 1 indexed_df.drop([119, 131, 132, 142, 148, 153, 154, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165])
```

```
In [25]: 1 indexed_df
```

```
Out[25]:
```

	index	lyrics	words	album
0	cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...	folklore
1	exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...	folklore
2	lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...	lover
3	the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...	folklore
4	look what you made me do	i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...	reputation
...
149	invisible	she can t see the way your eyes light up when ...	[see, way, eye, light, smile, never, notice, s...	taylor swift
150	a place in this world	i don t know what i want so don t ask me cause...	[know, want, ask, cause, still, trying, figure...	taylor swift
151	superman	tall dark and superman he puts papers in his b...	[tall, dark, superman, put, paper, briefcase, ...	speak now
152	a perfectly good heart	why would you wanna break a perfectly good hea...	[would, wanna, break, perfectly, good, heart, ...	taylor swift
155	today was a fairytale	today was a fairytale you were the prince i us...	[today, fairytale, prince, used, damsel, distr...	fearless

149 rows × 4 columns

```
In [26]: 1 indexed_df.reset_index(drop = True, inplace = True)
```

```
In [27]: 1 indexed_df
```

```
Out[27]:
```

	index	lyrics	words	album
0	cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...	folklore
1	exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...	folklore
2	lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...	lover
3	the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...	folklore
4	look what you made me do	i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...	reputation
...
144	invisible	she can t see the way your eyes light up when ...	[see, way, eye, light, smile, never, notice, s...	taylor swift
145	a place in this world	i don t know what i want so don t ask me cause...	[know, want, ask, cause, still, trying, figure...	taylor swift
146	superman	tall dark and superman he puts papers in his b...	[tall, dark, superman, put, paper, briefcase, ...	speak now
147	a perfectly good heart	why would you wanna break a perfectly good hea...	[would, wanna, break, perfectly, good, heart, ...	taylor swift
148	today was a fairytale	today was a fairytale you were the prince i us...	[today, fairytale, prince, used, damsel, distr...	fearless

149 rows × 4 columns

Create dataframe of all words in lyrics with counts by album

```
In [28]: 1 set_words = []
2 set_albums = []
3
4 for i in indexed_df.index:
5     for word in indexed_df['words'].iloc[i]:
6         set_words.append(word)
7         set_albums.append(indexed_df['album'].iloc[i])
```

```
In [29]: 1 words_albums_df = pd.DataFrame({'words': set_words, 'album': set_albums})
```

```
In [30]: 1 stopwords = ["di", "n't", "oh", "ooh", "ai", "oooh", "wo", "mmm", "mmmmmm", "lyrics", "chorus", "v
```

```
In [31]: 1 cv = CountVectorizer(stop_words=stopwords)
```

```
In [32]: 1 text_cv = cv.fit_transform(words_albums_df['words'].iloc[:])
2 data_cv = pd.DataFrame(text_cv.toarray(), columns=cv.get_feature_names())
3 data_cv['album'] = words_albums_df['album']
```

```
In [33]: 1 vect_words = data_cv.groupby('album').sum().T
2 vect_words = vect_words.reset_index(level=0).rename(columns={'index': 'words'})
3 vect_words = vect_words.rename_axis(columns='')
```

```
In [34]: 1 vect_words = vect_words[['words', "taylor swift", "fearless", "speak now", "red deluxe edition", "19
```

```
In [35]: 1 vect_words
```

```
Out[35]:
```

	words	taylor swift	fearless	speak now	red deluxe edition	1989 deluxe	reputation	lover	folklore	evermore
0	abigail	0	1	0	0	0	0	0	0	0
1	absentmindedly	0	1	0	0	0	0	0	0	0
2	absurd	0	0	1	0	0	0	0	0	0
3	accent	0	0	0	0	0	0	1	0	0
4	accident	0	0	0	0	0	0	1	0	1
...
2818	yououttherestanding	0	0	0	0	0	0	0	0	1
2819	youre	0	0	0	3	0	1	1	0	0
2820	yourhands	0	0	0	0	0	0	1	0	0
2821	yousure	0	0	0	0	0	0	0	1	0
2822	youusedtowork	0	0	0	0	0	0	0	1	0

2823 rows × 10 columns

```
In [36]: 1 vect_words.to_csv('new_words_albums.csv',index=False)
```

Calculate ratios for words/lyrics/songs/albums, and create dataframe by album

```
In [37]: 1 def calc_words_stats(df,main_df):
2     unique_words = []
3     total_words = []
4     total_songs = []
5     albums = []
6
7     for value in df.columns[1:]:
8         unique_words.append(np.count_nonzero(df[value]))
9         total_words.append(sum(df[value]))
10        albums.append(str(value))
11        total_songs.append(main_df['album'][main_df['album']==value].count())
12
13    data = pd.DataFrame({'album':albums,'unique words':unique_words,'total words':total_words,'total
14    data['words per songs'] = round(data['total words'] / data['total songs'],0)
15    data['words per songs'] = data['words per songs'].astype('int')
16    return data
```

```
In [38]: 1 words_stats = calc_words_stats(vect_words,indexed_df)
```

```
In [39]: 1 words_stats
```

```
Out[39]:
```

	album	unique words	total words	total songs	words per songs
0	taylor swift	483	873	14	62
1	fearless	678	1411	19	74
2	speak now	809	1532	17	90
3	red deluxe edition	732	1371	19	72
4	1989 deluxe	625	1068	15	71
5	reputation	767	1280	16	80
6	lover	835	1393	18	77
7	folklore	895	1299	16	81
8	evermore	845	1288	15	86

```
In [40]: 1 words_stats.to_csv('new_words_stats.csv')
```

Calculate sentiment metrics for lyrics, and create dataframe of scores by song

```
In [41]: 1 negative = []  
2 neutral = []  
3 positive = []  
4 compound = []  
5  
6 sid = SentimentIntensityAnalyzer()
```

```
In [42]: 1 for i in indexed_df.index:  
2     scores = sid.polarity_scores(indexed_df['lyrics'].iloc[i])  
3     negative.append(scores['neg'])  
4     neutral.append(scores['neu'])  
5     positive.append(scores['pos'])  
6     compound.append(scores['compound'])
```

```
In [43]: 1 indexed_df['negative'] = negative
2 indexed_df['neutral'] = neutral
3 indexed_df['positive'] = positive
4 indexed_df['compound'] = compound
```

```
In [44]: 1 indexed_df
```

```
Out[44]:
```

	index	lyrics	words	album	negative	neutral	positive	compound
0	cardigan	vintage tee brand new phone high heels on cobb...	[vintage, tee, brand, new, phone, high, heel, ...	folklore	0.061	0.746	0.194	0.9942
1	exile	i can see you standing honey with his arms aro...	[see, standing, honey, arm, around, body, laug...	folklore	0.080	0.798	0.122	0.9321
2	lover	we could leave the christmas lights up til jan...	[could, leave, christmas, light, til, january,...	lover	0.036	0.737	0.227	0.9961
3	the 1	m doing good m on some new shit been saying ye...	[good, new, shit, saying, yes, instead, though...	folklore	0.072	0.755	0.173	0.9889
4	look what you made me do	i don t like your little games don t like your...	[like, little, game, tilted, stage, role, made...	reputation	0.083	0.778	0.139	0.9820
...
144	invisible	she can t see the way your eyes light up when ...	[see, way, eye, light, smile, never, notice, s...	taylor swift	0.071	0.699	0.230	0.9926
145	a place in this world	i don t know what i want so don t ask me cause...	[know, want, ask, cause, still, trying, figure...	taylor swift	0.084	0.781	0.134	0.9091
146	superman	tall dark and superman he puts papers in his b...	[tall, dark, superman, put, paper, briefcase, ...	speak now	0.057	0.752	0.191	0.9962
147	a perfectly good heart	why would you wanna break a perfectly good hea...	[would, wanna, break, perfectly, good, heart, ...	taylor swift	0.006	0.712	0.282	0.9978
148	today was a fairytale	today was a fairytale you were the prince i us...	[today, fairytale, prince, used, damsel, distr...	fearless	0.023	0.853	0.123	0.9866

149 rows x 8 columns

```
In [45]: 1 indexed_df.to_csv('new_indexed_df.csv')
```

Create dataframe of mean sentiment scores by album


```
In [46]: 1 means_df = indexed_df.groupby([ 'album' ]).mean()  
        2 means_df
```

Out [46]:

	negative	neutral	positive	compound
album				
1989 deluxe	0.109467	0.701600	0.188867	0.531393
evermore	0.132533	0.715333	0.152267	0.203673
fearless	0.065895	0.766842	0.167211	0.677200
folklore	0.108438	0.759313	0.132375	0.283987
lover	0.104944	0.695667	0.199500	0.633728
red deluxe edition	0.083211	0.740737	0.176105	0.701105
reputation	0.086313	0.749188	0.164563	0.725894
speak now	0.086529	0.766000	0.147471	0.418335
taylor swift	0.079429	0.765214	0.155357	0.464893

Create plot comparing mean sentiment scores by album

```
In [47]: 1 for name, group in means_df.groupby('album'):
2         plt.scatter(group['positive'], group['negative'], label=name)
3         plt.legend()
4
5         plt.xlim([-0.05, 1.0])
6         plt.ylim([-0.05, 1.0])
7         plt.title("Average Sentiment of Each Album")
8         plt.xlabel('Positive Sentiment')
9         plt.ylabel('Negative Sentiment')
10
11        plt.savefig("album_sentiments.png", format="png", bbox_inches='tight')
12        plt.show()
```

