In [4]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [5]:
```
comments = pd.read_csv(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube D
```

```
C:\Users\aabid\AppData\Local\Temp\ipykernel_3660\3918574829.py:1: DtypeWarning: C
olumns (2,3) have mixed types. Specify dtype option on import or set low_memory=F
alse.
  comments = pd.read_csv(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube
Data Analysis/UScomments.csv', on_bad_lines='skip')
```

In [6]:
```
comments.head(5)
```

Out[6]:

	video_id	comment_text	likes	replies
0	XpVt6Z1Gjjo	Logan Paul it's yo big day ‼️‼️‼️	4	0
1	XpVt6Z1Gjjo	I've been following you from the start of your...	3	0
2	XpVt6Z1Gjjo	Say hi to Kong and maverick for me	3	0
3	XpVt6Z1Gjjo	MY FAN . attendance	3	0
4	XpVt6Z1Gjjo	trending 😉	3	0

In [7]:
```
comments.isnull().sum()
```

Out[7]:
```
video_id         0
comment_text    26
likes            0
replies          0
dtype: int64
```

In [8]:
```
comments.dropna(inplace=True)
```

In [73]:
```
comments.isnull().sum()
```

Out[73]:
```
video_id        0
comment_text    0
likes           0
replies         0
Polarity        0
dtype: int64
```

Performing Sentiment Analysis

In [10]:
```
from textblob import TextBlob
```

In [11]:
```
comments.head(6)
```

Out[11]:

	video_id	comment_text	likes	replies
0	XpVt6Z1Gjjo	Logan Paul it's yo big day ‼️‼️‼️	4	0
1	XpVt6Z1Gjjo	I've been following you from the start of your...	3	0
2	XpVt6Z1Gjjo	Say hi to Kong and maverick for me	3	0
3	XpVt6Z1Gjjo	MY FAN . attendance	3	0
4	XpVt6Z1Gjjo	trending 😉	3	0
5	XpVt6Z1Gjjo	#1 on trending AYYEEEEE	3	0

In [12]:
```
TextBlob("Logan Paul it's yo big day ‼️‼️‼️").sentiment.polarity
```

Out[12]:
```
0.0
```

In [13]:
```
polarity = []
for comment in comments['comment_text']:
    try:
        polarity.append(TextBlob(comment).sentiment.polarity)
    except:
        polarity.append(0)
```

In [14]:
```
len(polarity)
```

Out[14]:
```
691374
```

In [15]:
```
comments['Polarity'] = polarity
```

In [16]:
```
Filter1 = comments['Polarity']==1
```

In [17]:
```
comments_positive = comments[Filter1]
```

In [18]:
```
filter2 = comments['Polarity']==-1
```

In [19]:
```
comments_negative = comments[filter2]
```

In [20]:
```
from wordcloud import WordCloud, STOPWORDS
```

In [21]:
```
set(STOPWORDS)
```

Out[21]:
```
{'a',
 'about',
 'above',
 'after',
 'again',
 'against',
 'all',
 'also',
 'am',
 'an',
 'and',
 'any',
 'are',
 "aren't",
 'as',
 'at',
 'be',
 'because',
 'been',
 'before',
 'being',
 'below',
 'between',
 'both',
 'but',
 'by',
 'can',
 "can't",
 'cannot',
 'com',
 'could',
 "couldn't",
 'did',
 "didn't",
 'do',
 'does',
 "doesn't",
 'doing',
 "don't",
 'down',
 'during',
 'each',
 'else',
 'ever',
 'few',
 'for',
 'from',
 'further',
 'get',
 'had',
 "hadn't",
 'has',
 "hasn't",
 'have',
 "haven't",
 'having',
 'he',
 "he'd",
 "he'll",
 "he's",
```

```
'hence',
'her',
'here',
"here's",
'hers',
'herself',
'him',
'himself',
'his',
'how',
"how's",
'however',
'http',
'i',
"i'd",
"i'll",
"i'm",
"i've",
'if',
'in',
'into',
'is',
"isn't",
'it',
"it's",
'its',
'itself',
'just',
'k',
"let's",
'like',
'me',
'more',
'most',
"mustn't",
'my',
'myself',
'no',
'nor',
'not',
'of',
'off',
'on',
'once',
'only',
'or',
'other',
'otherwise',
'ought',
'our',
'ours',
'ourselves',
'out',
'over',
'own',
'r',
'same',
'shall',
"shan't",
'she',
```

```
"she'd",
"she'll",
"she's",
'should',
"shouldn't",
'since',
'so',
'some',
'such',
'than',
'that',
"that's",
'the',
'their',
'theirs',
'them',
'themselves',
'then',
'there',
"there's",
'therefore',
'these',
'they',
"they'd",
"they'll",
"they're",
"they've",
'this',
'those',
'through',
'to',
'too',
'under',
'until',
'up',
'very',
'was',
"wasn't",
'we',
"we'd",
"we'll",
"we're",
"we've",
'were',
"weren't",
'what',
"what's",
'when',
"when's",
'where',
"where's",
'which',
'while',
'who',
"who's",
'whom',
'why',
"why's",
'with',
"won't",
```

```
 'would',
 "wouldn't",
 'www',
 'you',
 "you'd",
 "you'll",
 "you're",
 "you've",
 'your',
 'yours',
 'yourself',
 'yourselves'}
```

In [22]:
```
comments['comment_text']
```

Out[22]:
```
0                          Logan Paul it's yo big day !!!!!
1         I've been following you from the start of your...
2                       Say hi to Kong and maverick for me
3                                    MY FAN . attendance
4                                              trending 😉
                              ...
691395                                             Лучшая
691396    qu'est ce que j'aimerais que tu viennes à Roan...
691397                          Ven a mexico! 😍 te amo LP
691398                                   Islığı yeter...
691399    Kocham tą piosenkę😍❤❤❤byłam zakochana po uszy ...
Name: comment_text, Length: 691374, dtype: object
```

In [23]:
```
type(comments['comment_text'])
```

Out[23]:
```
pandas.core.series.Series
```

In [24]:
```
total_positive_comments = ' '.join(comments_positive['comment_text'])
```

In [25]:
```
wordcloud = WordCloud(stopwords=set(STOPWORDS)).generate(total_positive_comments
```

In [26]:
```
plt.imshow(wordcloud)
plt.axis('off')
```

Out[26]:
```
(np.float64(-0.5), np.float64(399.5), np.float64(199.5), np.float64(-0.5))
```

In [27]:
```
total_negative_comments = ' '.join(comments_negative['comment_text'])
```

In [28]:
```
wordcloud = WordCloud(stopwords=set(STOPWORDS)).generate(total_negative_comments
```

In [29]:
```
plt.imshow(wordcloud)
plt.axis('off')
```

Out[29]:
```
(np.float64(-0.5), np.float64(399.5), np.float64(199.5), np.float64(-0.5))
```

Performing Emoji Analysis

In [30]:
```
import emoji
```

In [31]:
```
emoji.__version__
```

Out[31]:
```
'2.2.0'
```

In [32]:
```
comments['comment_text'].head(6)
```

Out[32]:
```
0                    Logan Paul it's yo big day ‼️‼️‼️
1    I've been following you from the start of your...
2                   Say hi to Kong and maverick for me
3                                MY FAN . attendance
4                                          trending 😉
5                             #1 on trending AYYEEEEE
Name: comment_text, dtype: object
```

In [33]:
```
comment = 'trending 😉'
```

In [34]:
```
[char for char in comment if char in emoji.EMOJI_DATA]
```

Out[34]:
```
['😉']
```

In [35]:
```
all_emoji_list = []
for comment in comments['comment_text'].dropna():
    for char in comment:
        if char in emoji.EMOJI_DATA:
            all_emoji_list.append(char)
```

In [36]:
```
all_emoji_list[:10]
```

Out[36]:
```
['‼', '‼', '‼', '😳', '😭', '👍', '', '', '😍', '💋']
```

In [37]:
```
from collections import Counter
```

In [38]:
```
Counter(all_emoji_list).most_common(10)
```

Out[38]:
```
[('😂', 36987),
 ('😍', 33453),
 ('❤', 31119),
 ('🔥', 8694),
 ('😭', 8398),
 ('👏', 5719),
 ('😘', 5545),
 ('👍', 5476),
 ('💔', 5359),
 ('💕', 5147)]
```

In [39]:
```
Counter(all_emoji_list).most_common(10)[0]
```

Out[39]:
```
('😂', 36987)
```

In [40]:
```
Counter(all_emoji_list).most_common(10)[0][0]
```

Out[40]:
```
'😂'
```

In [41]:
```
Counter(all_emoji_list).most_common(10)[0][1]
```

Out[41]:
```
36987
```

In [42]:
```
emojis = [Counter(all_emoji_list).most_common(10)[i][0] for i in range(10)]
```

In [43]:
```
freq = [Counter(all_emoji_list).most_common(10)[i][1] for i in range(10)]
```

In [44]:
```
import plotly.graph_objs as go
from plotly.offline import iplot
```

In [45]:
```
trace = go.Bar(x = emojis, y= freq)
```

In [46]:
```
iplot([trace])
```

Collect Entire data of Youtube

In [47]:
```
import os
```

In [49]:
```
files = os.listdir(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube Data
```

In [50]:
```
files
```

Out[50]:
```
['CAvideos.csv',
 'CA_category_id.json',
 'DEvideos.csv',
 'DE_category_id.json',
 'FRvideos.csv',
 'FR_category_id.json',
 'GBvideos.csv',
 'GB_category_id.json',
 'INvideos.csv',
 'IN_category_id.json',
 'JPvideos.csv',
 'JP_category_id.json',
 'KRvideos.csv',
 'KR_category_id.json',
 'MXvideos.csv',
 'MX_category_id.json',
 'RUvideos.csv',
 'RU_category_id.json',
 'USvideos.csv',
 'US_category_id.json']
```

In [51]:
```
files_csv = [file for file in files if '.csv' in file]
```

In [52]:
```
files_csv
```

Out[52]:
```
['CAvideos.csv',
 'DEvideos.csv',
 'FRvideos.csv',
 'GBvideos.csv',
 'INvideos.csv',
 'JPvideos.csv',
 'KRvideos.csv',
 'MXvideos.csv',
 'RUvideos.csv',
 'USvideos.csv']
```

In [53]:
```
import warnings
from warnings import filterwarnings
filterwarnings('ignore')
```

In [55]:
```
full_df = pd.DataFrame()
path = r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube Data Analysis\add
for file in files_csv:
    current_df =pd.read_csv(path+'/'+file, encoding='iso-8859-1', on_bad_lines='
    full_df = pd.concat([full_df, current_df], ignore_index=True)
```

In [56]:
```
full_df.shape
```

Out[56]:
```
(375942, 16)
```

How to export your data into(csv, json, db)

In [57]:
```
full_df[full_df.duplicated()].shape
```

Out[57]:
```
(36417, 16)
```

In [58]:
```
full_df = full_df.drop_duplicates()
```

In [59]:
```
full_df.shape
```

Out[59]:
```
(339525, 16)
```

In [ ]:
```
full_df.to_csv(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube Data Anal
```

In [ ]:
```
full_df.to_json(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube Data Ana
```

In [60]:
```
from sqlalchemy import create_engine
```

In [61]:
```
engine = create_engine('sqlite:///C:\Aabid Study\LEARNING DATA SCIENCE\Projects\
```

In [62]:
```
full_df.to_sql('Users', con = engine, if_exists='append')
```

Out[62]:
```
339525
```

Which category has the maxixmum likes?

In [63]:
```
full_df.head(5)
```

Out[63]:

	video_id	trending_date	title	channel_title	category_id	publish_tim
0	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. BeyoncÃ©	EminemVEVO	10	2017-11 10T17:00:03.000
1	0dBIkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11 13T17:00:00.000
2	5qpjK5DgCt4	17.14.11	Racist Superman | Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11 12T19:05:24.000
3	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11 12T18:01:41.000
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11 09T11:04:14.000

In [64]:
```
full_df['category_id'].unique()
```

Out[64]:
```
array([10, 23, 24, 25, 22, 26,  1, 28, 20, 17, 29, 15, 19,  2, 27, 43, 30,
       44])
```

In [65]:
```
json_df = pd.read_json(r'C:\Aabid Study\LEARNING DATA SCIENCE\Projects\Youtube D
```

In [66]:
```
json_df
```

Out[66]:

	kind	etag	
0	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
1	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
2	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
3	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
4	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
5	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
6	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
7	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
8	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
9	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
10	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
11	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
12	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
13	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
14	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi

	kind	etag	
15	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
16	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
17	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
18	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
19	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
20	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
21	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
22	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
23	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
24	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
25	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
26	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
27	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
28	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
29	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi

	kind	etag	
30	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi
31	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/S730Ilt-Fi-emsQJv...	'youtube#vi

In [67]:
```
json_df['items']
```

Out[67]:
```
0     {'kind': 'youtube#videoCategory', 'etag': '"m2...
1     {'kind': 'youtube#videoCategory', 'etag': '"m2...
2     {'kind': 'youtube#videoCategory', 'etag': '"m2...
3     {'kind': 'youtube#videoCategory', 'etag': '"m2...
4     {'kind': 'youtube#videoCategory', 'etag': '"m2...
5     {'kind': 'youtube#videoCategory', 'etag': '"m2...
6     {'kind': 'youtube#videoCategory', 'etag': '"m2...
7     {'kind': 'youtube#videoCategory', 'etag': '"m2...
8     {'kind': 'youtube#videoCategory', 'etag': '"m2...
9     {'kind': 'youtube#videoCategory', 'etag': '"m2...
10    {'kind': 'youtube#videoCategory', 'etag': '"m2...
11    {'kind': 'youtube#videoCategory', 'etag': '"m2...
12    {'kind': 'youtube#videoCategory', 'etag': '"m2...
13    {'kind': 'youtube#videoCategory', 'etag': '"m2...
14    {'kind': 'youtube#videoCategory', 'etag': '"m2...
15    {'kind': 'youtube#videoCategory', 'etag': '"m2...
16    {'kind': 'youtube#videoCategory', 'etag': '"m2...
17    {'kind': 'youtube#videoCategory', 'etag': '"m2...
18    {'kind': 'youtube#videoCategory', 'etag': '"m2...
19    {'kind': 'youtube#videoCategory', 'etag': '"m2...
20    {'kind': 'youtube#videoCategory', 'etag': '"m2...
21    {'kind': 'youtube#videoCategory', 'etag': '"m2...
22    {'kind': 'youtube#videoCategory', 'etag': '"m2...
23    {'kind': 'youtube#videoCategory', 'etag': '"m2...
24    {'kind': 'youtube#videoCategory', 'etag': '"m2...
25    {'kind': 'youtube#videoCategory', 'etag': '"m2...
26    {'kind': 'youtube#videoCategory', 'etag': '"m2...
27    {'kind': 'youtube#videoCategory', 'etag': '"m2...
28    {'kind': 'youtube#videoCategory', 'etag': '"m2...
29    {'kind': 'youtube#videoCategory', 'etag': '"m2...
30    {'kind': 'youtube#videoCategory', 'etag': '"m2...
31    {'kind': 'youtube#videoCategory', 'etag': '"m2...
Name: items, dtype: object
```

In [68]:
```
json_df['items'][0]
```

Out[68]:
```
{'kind': 'youtube#videoCategory',
 'etag': '"m2yskBQFythfE4irbTIeOgYYfBU/Xy1mB4_yLrHy_BmKmPBggty2mZQ"',
 'id': '1',
 'snippet': {'channelId': 'UCBR8-60-B28hp2BmDPdntcQ',
  'title': 'Film & Animation',
  'assignable': True}}
```

In [69]:
```
cat_dict = {}
for item in json_df['items'].values:
    cat_dict[int(item['id'])] = item['snippet']['title']
```

In [70]:
```
cat_dict
```

Out[70]:
```
{1: 'Film & Animation',
 2: 'Autos & Vehicles',
 10: 'Music',
 15: 'Pets & Animals',
 17: 'Sports',
 18: 'Short Movies',
 19: 'Travel & Events',
 20: 'Gaming',
 21: 'Videoblogging',
 22: 'People & Blogs',
 23: 'Comedy',
 24: 'Entertainment',
 25: 'News & Politics',
 26: 'Howto & Style',
 27: 'Education',
 28: 'Science & Technology',
 29: 'Nonprofits & Activism',
 30: 'Movies',
 31: 'Anime/Animation',
 32: 'Action/Adventure',
 33: 'Classics',
 34: 'Comedy',
 35: 'Documentary',
 36: 'Drama',
 37: 'Family',
 38: 'Foreign',
 39: 'Horror',
 40: 'Sci-Fi/Fantasy',
 41: 'Thriller',
 42: 'Shorts',
 43: 'Shows',
 44: 'Trailers'}
```

In [75]:
```
full_df['category_name'] = full_df['category_id'].map(cat_dict)
```

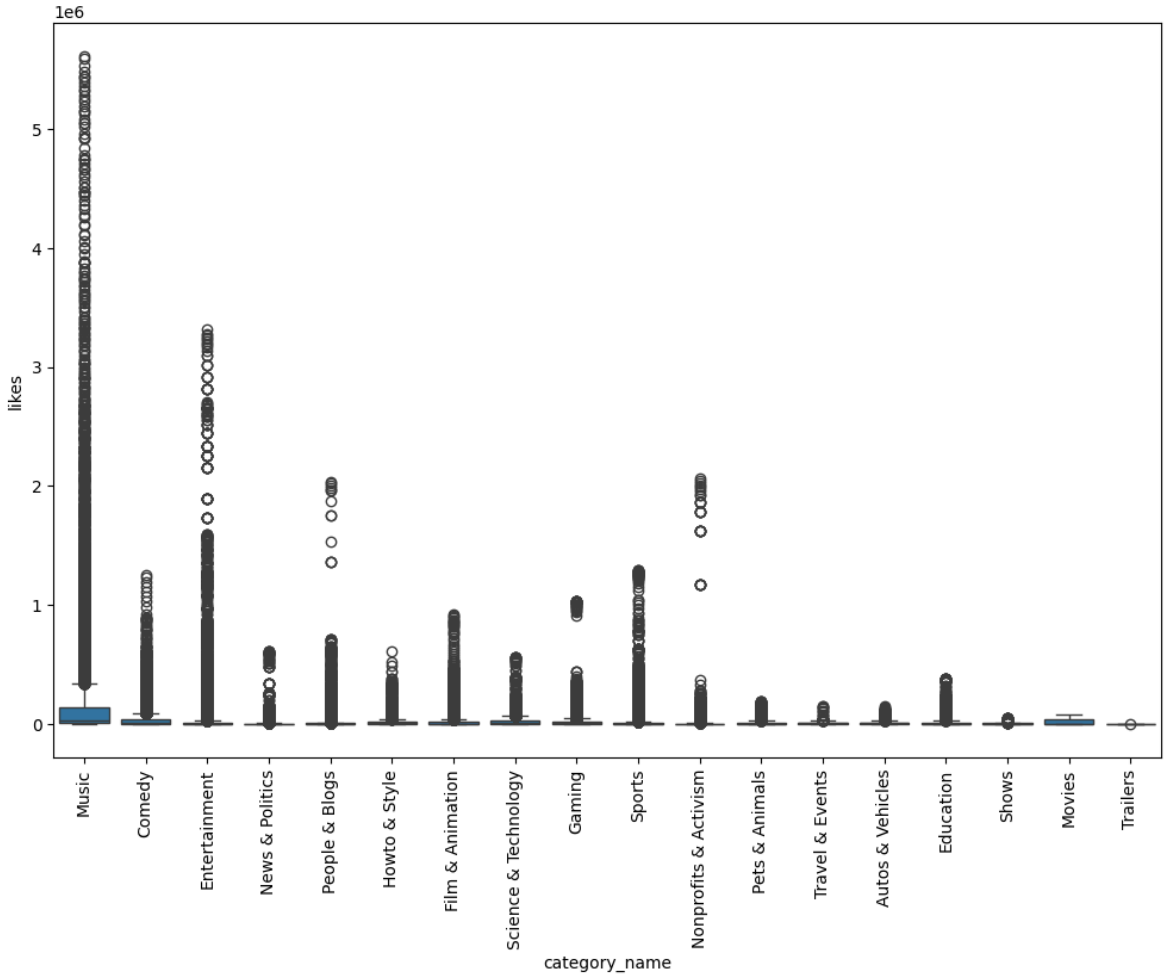
In [76]:
```
full_df.head(5)
```

Out[76]:

	video_id	trending_date	title	channel_title	category_id	publish_tim
0	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. BeyoncÃ©	EminemVEVO	10	2017-11 10T17:00:03.000
1	0dBIkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11 13T17:00:00.000
2	5qpjK5DgCt4	17.14.11	Racist Superman | Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11 12T19:05:24.000
3	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11 12T18:01:41.000
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11 09T11:04:14.000

In [79]:
```
plt.figure(figsize=(12,8))
sns.boxplot(x='category_name',y='likes', data=full_df)
plt.xticks(rotation= 'vertical')
```

Out[79]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17],
 [Text(0, 0, 'Music'),
  Text(1, 0, 'Comedy'),
  Text(2, 0, 'Entertainment'),
  Text(3, 0, 'News & Politics'),
  Text(4, 0, 'People & Blogs'),
  Text(5, 0, 'Howto & Style'),
  Text(6, 0, 'Film & Animation'),
  Text(7, 0, 'Science & Technology'),
  Text(8, 0, 'Gaming'),
  Text(9, 0, 'Sports'),
  Text(10, 0, 'Nonprofits & Activism'),
  Text(11, 0, 'Pets & Animals'),
  Text(12, 0, 'Travel & Events'),
  Text(13, 0, 'Autos & Vehicles'),
  Text(14, 0, 'Education'),
  Text(15, 0, 'Shows'),
  Text(16, 0, 'Movies'),
  Text(17, 0, 'Trailers')])
```



Finding out whether audience is engaged or not?

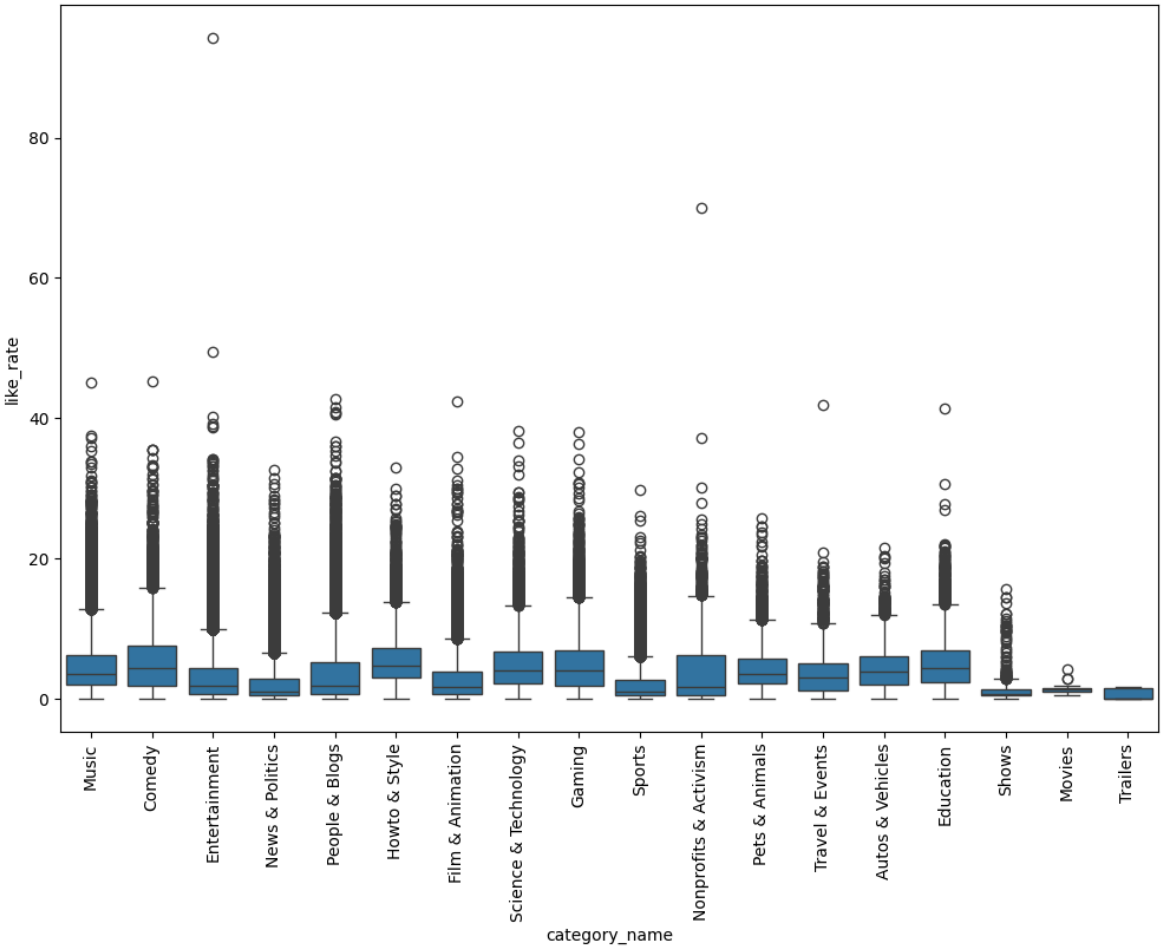
In [80]:
```
full_df['like_rate'] = (full_df['likes']/full_df['views'])*100
full_df['dislike_rate'] = (full_df['dislikes']/full_df['views'])*100
full_df['comment_count_rate'] = (full_df['comment_count']/full_df['views'])*100
```

In [81]:
```
full_df.columns
```

Out[81]:
```
Index(['video_id', 'trending_date', 'title', 'channel_title', 'category_id',
       'publish_time', 'tags', 'views', 'likes', 'dislikes', 'comment_count',
       'thumbnail_link', 'comments_disabled', 'ratings_disabled',
       'video_error_or_removed', 'description', 'category_name', 'like_rate',
       'dislike_rate', 'comment_count_rate'],
      dtype='object')
```

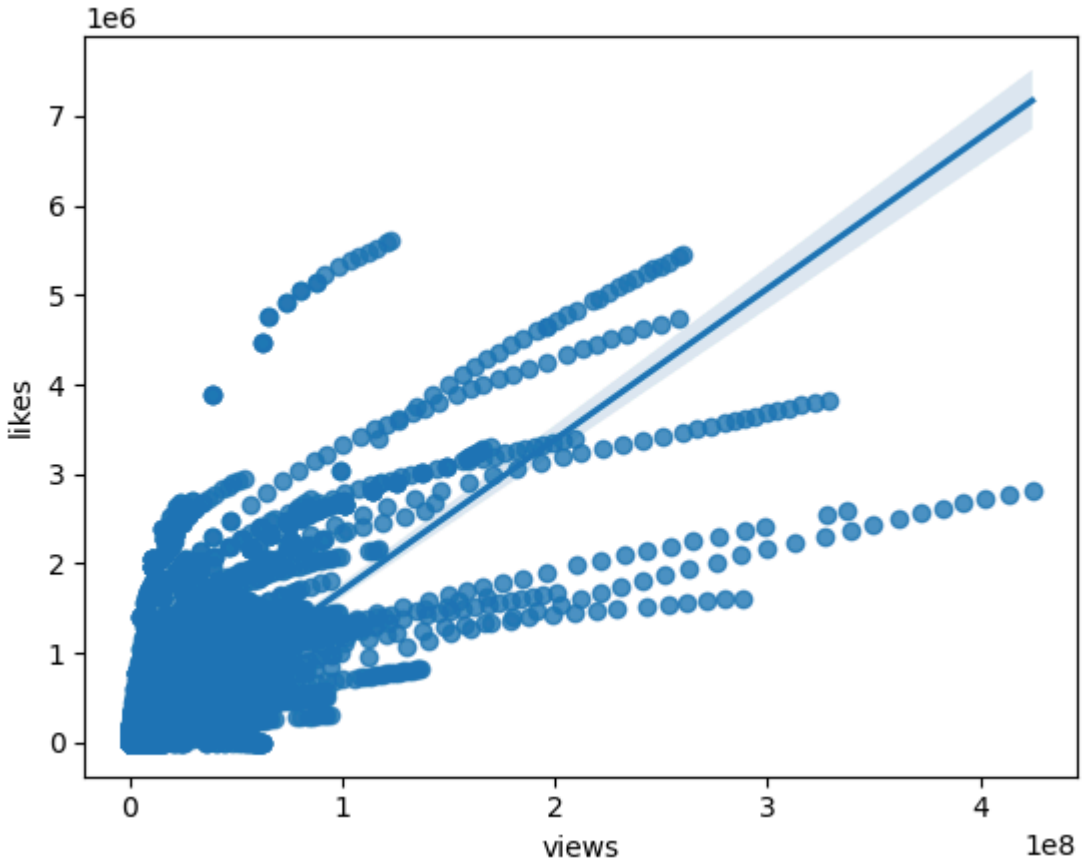
In [82]:
```
plt.figure(figsize=(12,8))
sns.boxplot(x='category_name',y='like_rate', data=full_df)
plt.xticks(rotation= 'vertical')
```

Out[82]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17],
 [Text(0, 0, 'Music'),
  Text(1, 0, 'Comedy'),
  Text(2, 0, 'Entertainment'),
  Text(3, 0, 'News & Politics'),
  Text(4, 0, 'People & Blogs'),
  Text(5, 0, 'Howto & Style'),
  Text(6, 0, 'Film & Animation'),
  Text(7, 0, 'Science & Technology'),
  Text(8, 0, 'Gaming'),
  Text(9, 0, 'Sports'),
  Text(10, 0, 'Nonprofits & Activism'),
  Text(11, 0, 'Pets & Animals'),
  Text(12, 0, 'Travel & Events'),
  Text(13, 0, 'Autos & Vehicles'),
  Text(14, 0, 'Education'),
  Text(15, 0, 'Shows'),
  Text(16, 0, 'Movies'),
  Text(17, 0, 'Trailers')])
```



In [83]:
```
sns.regplot(x='views', y='likes', data=full_df )
```

Out[83]:
```
<Axes: xlabel='views', ylabel='likes'>
```



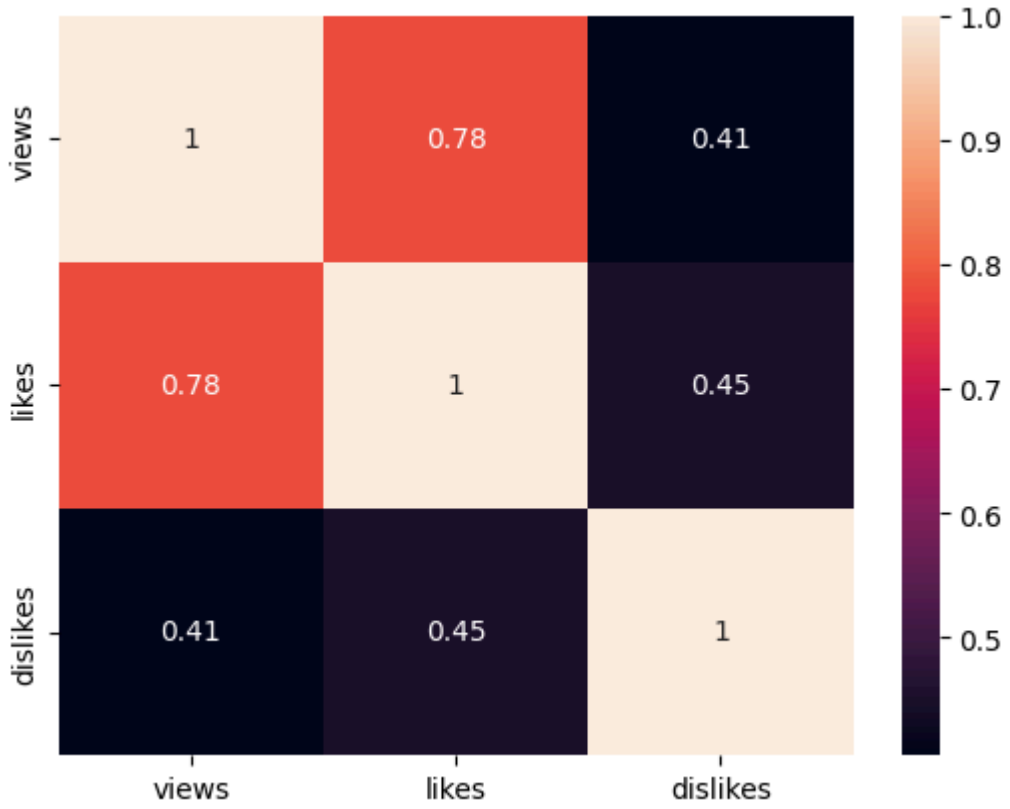
In [84]:
```
full_df[['views', 'likes', 'dislikes']].corr()
```

Out[84]:

	views	likes	dislikes
views	1.000000	0.779531	0.405428
likes	0.779531	1.000000	0.451809
dislikes	0.405428	0.451809	1.000000

In [86]:
```
sns.heatmap(full_df[['views', 'likes', 'dislikes']].corr(), annot=True)
```

Out[86]: <Axes: >



Which Channel has the largest number of trending videos?

In [87]:
```
full_df.head(6)
```

Out[87]:

	video_id	trending_date	title	channel_title	category_id	publish_tim
0	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. BeyoncÃ©	EminemVEVO	10	2017-11 10T17:00:03.000
1	0dBIkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11 13T17:00:00.000
2	5qpjK5DgCt4	17.14.11	Racist Superman | Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11 12T19:05:24.000
3	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11 12T18:01:41.000
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11 09T11:04:14.000
5	0yIWz1XEeyc	17.14.11	Jake Paul Says Alissa Violet CHEATED with LOGA...	DramaAlert	25	2017-11 13T07:37:51.000

In [88]:
```
full_df['channel_title'].value_counts()
```

Out[88]:
```
channel_title
The Late Show with Stephen Colbert    710
WWE                                   643
Late Night with Seth Meyers           592
TheEllenShow                          555
Jimmy Kimmel Live                     528
                                     ...
The Secrets of Dairy                    1
babygranderecords                       1
24 Oras News Today                      1
turk2doubleoh8                          1
BOOMPANOT.COM                           1
Name: count, Length: 37824, dtype: int64
```

In [93]:
```
cdf = full_df.groupby(['channel_title']).size().sort_values(ascending=False).res
```

In [95]:
```
cdf = cdf.rename(columns={0:'Total Videos'})
```

In [96]:
```
cdf
```

Out[96]:

	channel_title	Total Videos
0	The Late Show with Stephen Colbert	710
1	WWE	643
2	Late Night with Seth Meyers	592
3	TheEllenShow	555
4	Jimmy Kimmel Live	528
...	...	...
37819	Tesoros Enterrados	1
37820	Anton Neverov	1
37821	Thalia 444	1
37822	Thaitv6 Official	1
37823	Thad Broman	1

37824 rows × 2 columns

In [97]:
```
import plotly.express as px
```

In [98]:
```
px.bar(data_frame=cdf[:20], x='channel_title',y='Total Videos')
```

Does Punctuations in title and tags have any relation with views, likes, dislikes, comments?

In [99]:
```
import string
```

In [100…
```
string.punctuation
```

Out[100…
```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

In [110…
```
def punc_count(text):
    return len([char for char in text if char in string.punctuation])
```

In [111…
```
sample = full_df[0:10000]
```

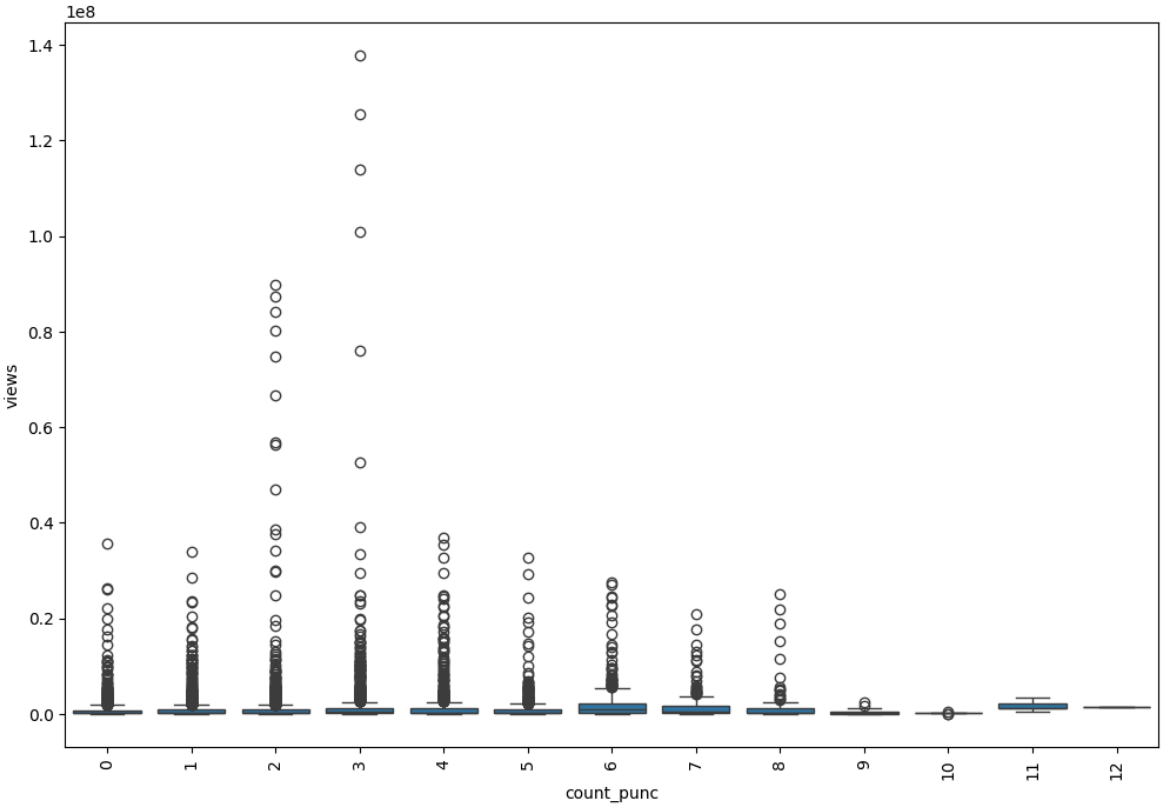
In [112…
```
sample['count_punc'] = sample['title'].apply(punc_count)
```

In [113…
```
sample['count_punc']
```

Out[113…
```
0       4
1       1
2       3
3       3
4       3
       ..
9995    6
9996    0
9997    1
9998    0
9999    6
Name: count_punc, Length: 10000, dtype: int64
```

In [115…
```
plt.figure(figsize=(12,8))
sns.boxplot(x='count_punc',y='views', data=sample)
plt.xticks(rotation= 'vertical')
```

Out[115…
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
 [Text(0, 0, '0'),
  Text(1, 0, '1'),
  Text(2, 0, '2'),
  Text(3, 0, '3'),
  Text(4, 0, '4'),
  Text(5, 0, '5'),
  Text(6, 0, '6'),
  Text(7, 0, '7'),
  Text(8, 0, '8'),
  Text(9, 0, '9'),
  Text(10, 0, '10'),
  Text(11, 0, '11'),
  Text(12, 0, '12')])
```



In [ ]: