In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
data = pd.read_csv(r'D:\Career\Udemy\DA\Youtube_project_shan_singh/USComments.csv',error_ba
data.head()
```

```
b'Skipping line 41589: expected 4 fields, saw 11\nSkipping line 51628: expec
ted 4 fields, saw 7\nSkipping line 114465: expected 4 fields, saw 5\n'
b'Skipping line 142496: expected 4 fields, saw 8\nSkipping line 189732: expe
cted 4 fields, saw 6\nSkipping line 245218: expected 4 fields, saw 7\n'
b'Skipping line 388430: expected 4 fields, saw 5\n'
C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py:
3165: DtypeWarning: Columns (2,3) have mixed types.Specify dtype option on i
mport or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

Out[2]:

	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 [3]:

```
data.shape
```

Out[3]:

```
(691400, 4)
```

In [4]:

```
data.dropna(inplace = True)
```

In [5]:

```
data.isnull().sum()
```

Out[5]:

```
video_id        0
comment_text    0
likes           0
replies         0
dtype: int64
```

Sentiment Analysis

In [6]:

```
from textblob import TextBlob
```

In [7]:

```
a = data[0:10000]
```

In [8]:

```
Polarity = []
for i in data['comment_text']:
    try:
        Polarity.append(TextBlob(i).sentiment.polarity)
    except:
        Polarity.append(0)
```

In [9]:

```
data['Polarity'] = Polarity
```

In [10]:

```
positive = data[data['Polarity'] == 1]
negative = data[data['Polarity'] == -1]
```

In [11]:

```
from wordcloud import WordCloud, STOPWORDS
```

In [12]:

```
positive['comment_text'][0:10]
```

Out[12]:

```
64                                       yu are the best
156    Power is the disease.  Care is the cure.  Keep...
227    YAS Can't wait to get it! I just need to sell ...
307                                    This is priceless
319                                  Summed up perfectly
325                     This is awesome. 1:20 XDDDDDDDDD
416                                   BEST MOVIE EVER!!!
433    Power is the disease.  Care is the cure.  Keep...
447         The greatest movie about the greatest movie.
469           It's Harry guys he's Spiderman best friend
Name: comment_text, dtype: object
```

In [13]:

```
negative['comment_text'][0:10]
```

Out[13]:

```
512     BEN CARSON IS THE MAN!!!!! THEY HATE HIM CAUSE...
562     Well… The brain surgeon Ben Carson just proved...
952            WHY DID YOU MAKE FURRY FORCE?! SO NASTY!!!
1371                                      WTF BRUH!!!!!!
1391                         cheeseus christ thats insane!!!
1932            this is the worst thing i've heard. ever.
2043    Economy is horrible in Cuba. It's going to be ...
2088                           Sub to me if this is terrible
2192                                            PATHETIC
2410    I don't like this sportscaster  sounds very an...
Name: comment_text, dtype: object
```

In [14]:

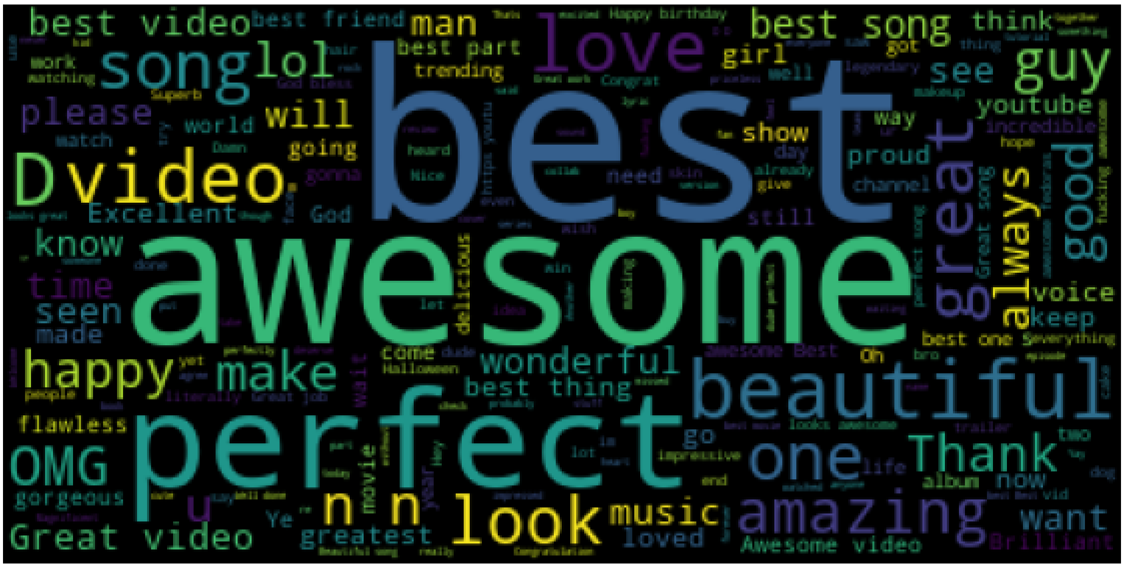
```
Total_posi = ' '.join(positive['comment_text'])
Total_nega = ' '.join(negative['comment_text'])
```

In [15]:

```
b = WordCloud(stopwords = set(STOPWORDS)).generate(Total_posi)
plt.figure(figsize = (10,10))
plt.imshow(b)
plt.axis('off')
```

Out[15]:

```
(-0.5, 399.5, 199.5, -0.5)
```

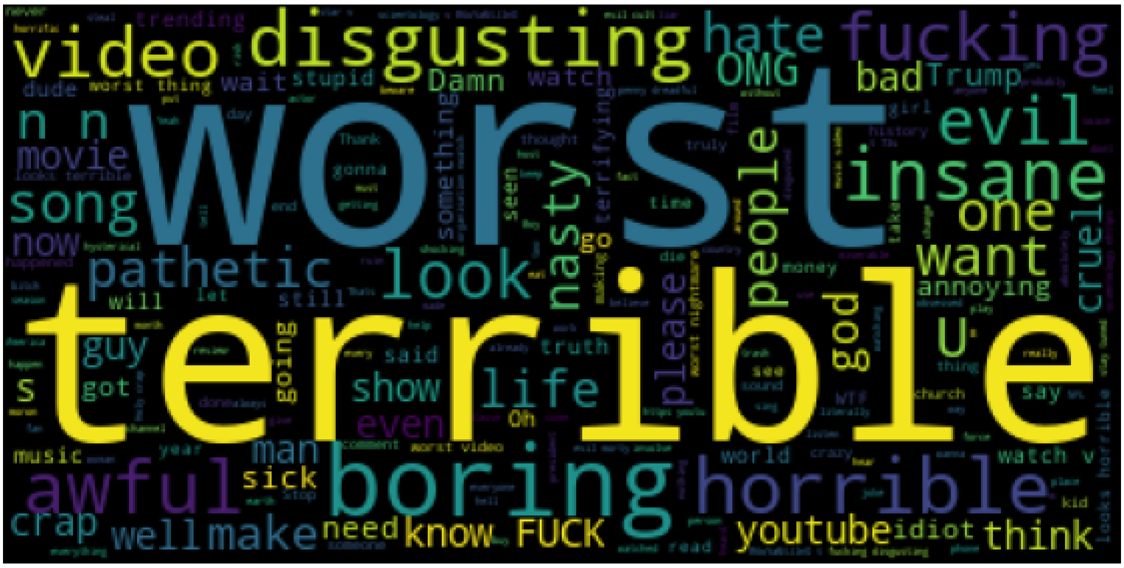


In [16]:

```
b = WordCloud(stopwords = set(STOPWORDS)).generate(Total_nega)
plt.figure(figsize = (10,10))
plt.imshow(b)
plt.axis('off')
```

Out[16]:

```
(-0.5, 399.5, 199.5, -0.5)
```



Emoji Analysis

In [17]:

```
import emoji
```

In [18]:

```
data.head(15)
```

Out[18]:

	video_id	comment_text	likes	replies	Polarity
0	XpVt6Z1Gjjo	Logan Paul it's yo big day ‼️‼️‼️	4	0	0.000000
1	XpVt6Z1Gjjo	I've been following you from the start of your...	3	0	0.000000
2	XpVt6Z1Gjjo	Say hi to Kong and maverick for me	3	0	0.000000
3	XpVt6Z1Gjjo	MY FAN . attendance	3	0	0.000000
4	XpVt6Z1Gjjo	trending 😉	3	0	0.000000
5	XpVt6Z1Gjjo	#1 on trending AYYEEEEE	3	0	0.000000
6	XpVt6Z1Gjjo	The end though 😭👍🏻❤️	4	0	0.000000
7	XpVt6Z1Gjjo	#1 trending!!!!!!!!!	3	0	0.000000
8	XpVt6Z1Gjjo	Happy one year vlogaversary	3	0	0.800000
9	XpVt6Z1Gjjo	You and your shit brother may have single hand...	0	0	-0.135714
10	XpVt6Z1Gjjo	There should be a mini Logan Paul too!	0	0	0.000000
11	XpVt6Z1Gjjo	Dear Logan, I really wanna get your Merch but ...	0	0	0.200000
12	XpVt6Z1Gjjo	Honestly Evan is so annoying. Like its not fun...	0	0	-0.023333
13	XpVt6Z1Gjjo	Casey is still better then logan	0	0	0.500000
14	XpVt6Z1Gjjo	aw geez rick this guy is the face of YouTube.	0	0	0.000000

In [19]:

```
emoji_list = []
for a in data['comment_text']:
    for b in a:
        if b in emoji.UNICODE_EMOJI_ENGLISH:
            emoji_list.append(b)
```

In [20]:

```
len(emoji_list)
```

Out[20]:

```
294549
```

In [21]:

```
emoji_list[0:10]
```

Out[21]:

```
['‼', '‼', '‼', '😉', '😭', '👍', ' ', '❤', '😍', '💋']
```

In [22]:

```
from collections import Counter
```

In [23]:

```
Emojis = [Counter(emoji_list).most_common(20)[i][0] for i in range(20)]
```

In [24]:

```
Emojis
```

Out[24]:

```
['😂',
 '😍',
 '🖤',
 '🔥',
 '😭',
 '👏',
 '😘',
 '👍',
 '💖',
 '💕',
 '♥',
 '😊',
 ' ',
 '💜',
 '😬',
 '👌',
 '💙',
 '😢',
 ' ',
 '🤣']
```

In [25]:

```
Freqs = [Counter(emoji_list).most_common(20)[i][1] for i in range(20)]
Freqs
```

Out[25]:

```
[36987,
 33453,
 31119,
 8694,
 8398,
 5719,
 5545,
 5476,
 5359,
 5147,
 4909,
 3596,
 3438,
 3429,
 3381,
 3112,
 2831,
 2672,
 2549,
 2279]
```

In [26]:

```
import plotly.graph_objs as go
```

In [27]:

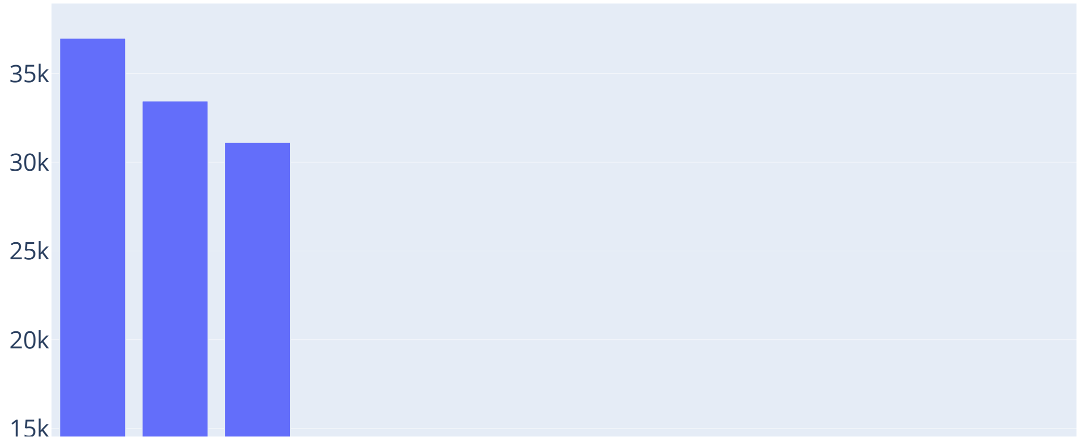
```
from plotly.offline import iplot
```

In [28]:

```
a = go.Bar(x = Emojis, y = Freqs)
```

In [29]:

```
iplot([a])
```



In [30]:

```
import os
```

In [31]:

```
path = r'D:\Career\Udemy\DA\Youtube_project_shan_singh\additional_data'
```

In [32]:

```
a = os.listdir(path)
a
```

Out[32]:

```
['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 [33]:

```
files_csv = [a[i] for i in range(0,len(a),2)]
files_csv
```

Out[33]:

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

In [34]:

```
full_df = pd.DataFrame()
for a in files_csv:
    current_df = pd.read_csv(path+'/'+a,encoding = 'iso-8859-1',error_bad_lines = False)
    current_df['Country'] = a.split('.')[0][0:2]
    full_df = pd.concat([full_df,current_df])
```

In [35]:

```
cate = pd.read_csv(r'D:\Career\Udemy\DA\Youtube_project_shan_singh/category_file.txt',sep =
cate
```

Out[35]:

	Category_id Category_name
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 [36]:

```
cate.reset_index(inplace = True)
```

In [37]:

```
cate.columns = ['category_id','category_name']
```

In [38]:

```
cate.set_index('category_id',inplace = True)
```

In [39]:

```
dct = cate.to_dict()
```

In [40]:

```
dct['category_name']
```

Out[40]:

```
{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 [41]:

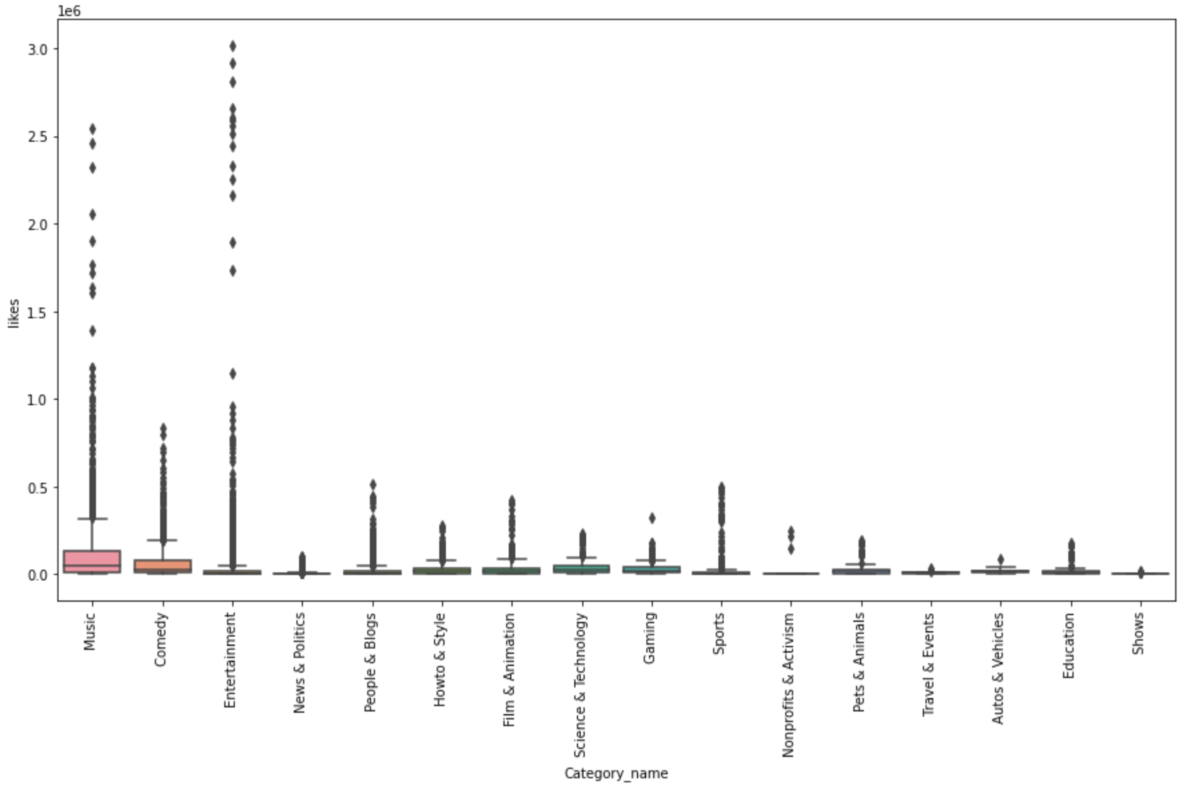
```
full_df['Category_name'] = full_df['category_id'].map(dct['category_name'])
```

In [42]:

```
plt.figure(figsize = (15,8))
sns.boxplot(x = 'Category_name',y = 'likes',data = full_df[0:10000])
plt.xticks(rotation = 'vertical')
```

Out[42]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15]),
 [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')])
```



Check whether the audience is engaged or not?

In [43]:

```
full_df.columns
```

Out[43]:

```
Index(['video_id', 'trending_date', 'title', 'channel_title', 'category_id',
       'publish_time', 'tags', 'views', 'likes', 'dislikes', 'comment_coun
t',
       'thumbnail_link', 'comments_disabled', 'ratings_disabled',
       'video_error_or_removed', 'description', 'Country', 'Category_name'],
      dtype='object')
```

In [44]:

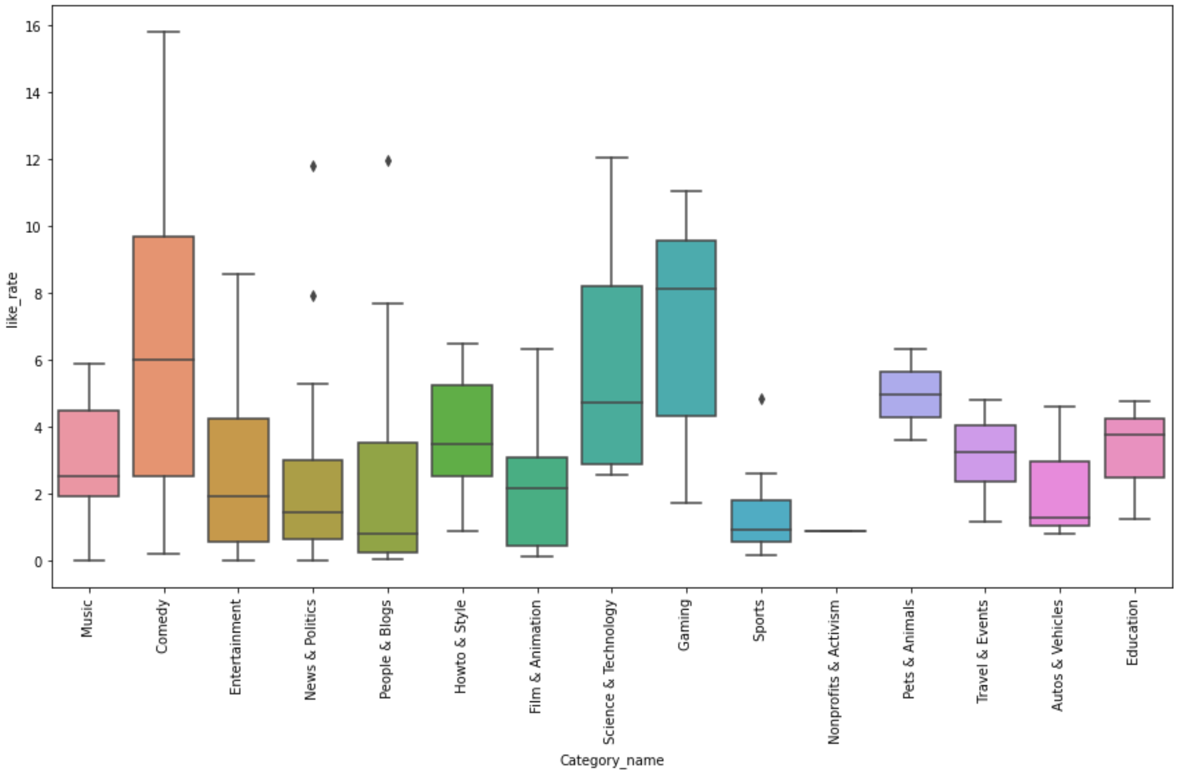
```
full_df['like_rate'] = (full_df['likes'] / full_df['views'] * 100)
full_df['dislike_rate'] = (full_df['dislikes'] / full_df['views'] * 100)
full_df['comment_rate'] = (full_df['comment_count'] / full_df['views'] * 100)
```

In [45]:

```
plt.figure(figsize = (15,8))
sns.boxplot(x = 'Category_name', y = 'like_rate',data = full_df[0:200])
plt.xticks(rotation = 'vertical')
```

Out[45]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14]),
 [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')])
```

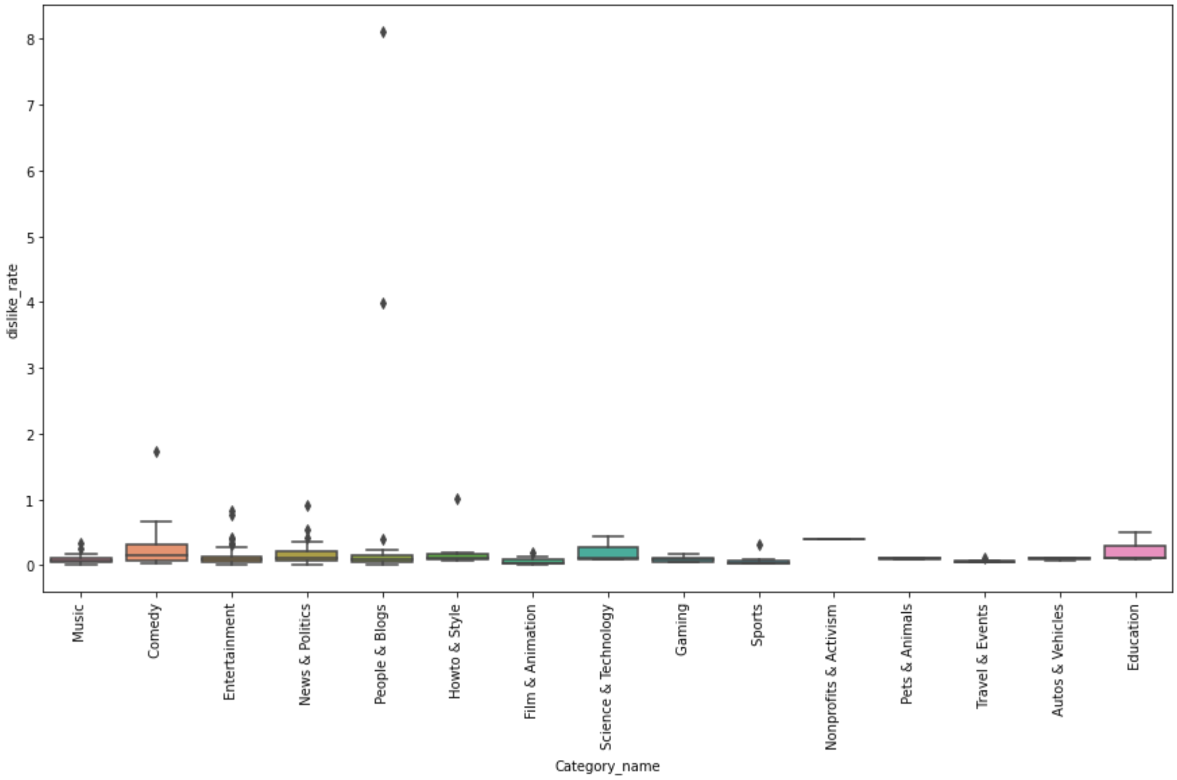


In [46]:

```
plt.figure(figsize = (15,8))
sns.boxplot(x = 'Category_name', y = 'dislike_rate',data = full_df[0:200])
plt.xticks(rotation = 'vertical')
```

Out[46]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14]),
 [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')])
```

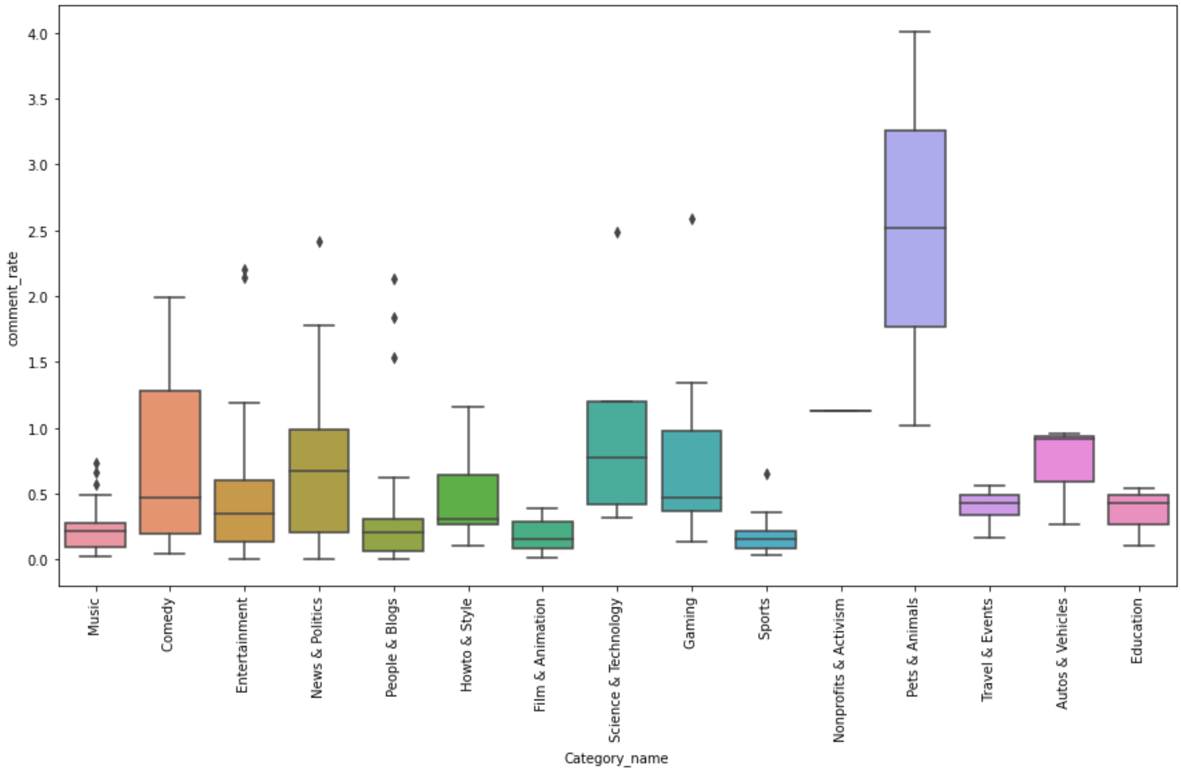


In [47]:

```
plt.figure(figsize = (15,8))
sns.boxplot(x = 'Category_name', y = 'comment_rate',data = full_df[0:200])
plt.xticks(rotation = 'vertical')
```

Out[47]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14]),
 [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')])
```

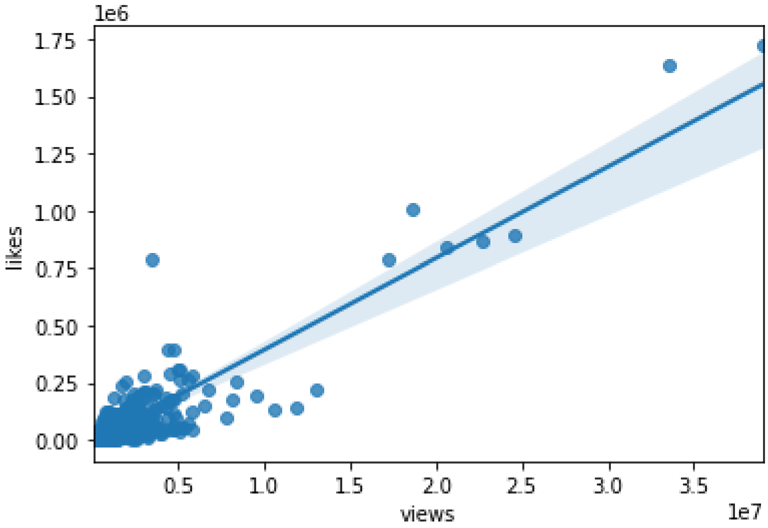


In [48]:

```
sns.regplot(x = 'views',y = 'likes',data = full_df[0:1000])
```

Out[48]:

```
<AxesSubplot:xlabel='views', ylabel='likes'>
```

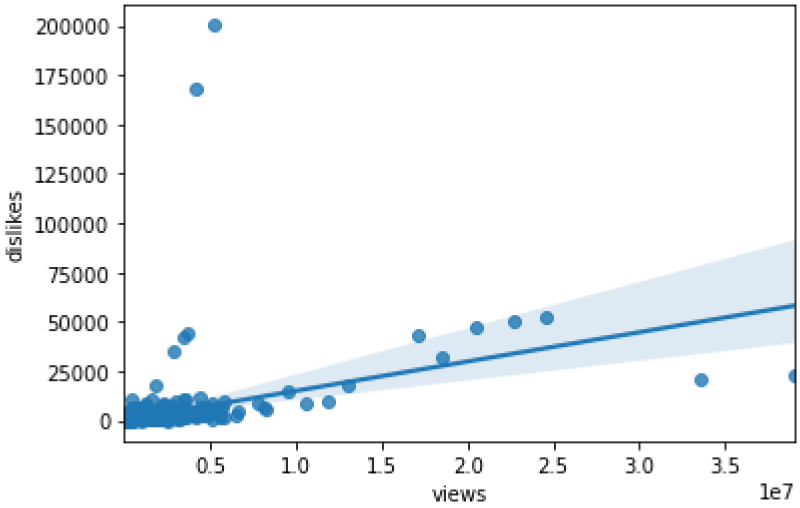


In [49]:

```
sns.regplot(x = 'views',y = 'dislikes',data = full_df[0:1000])
```

Out[49]:

```
<AxesSubplot:xlabel='views', ylabel='dislikes'>
```

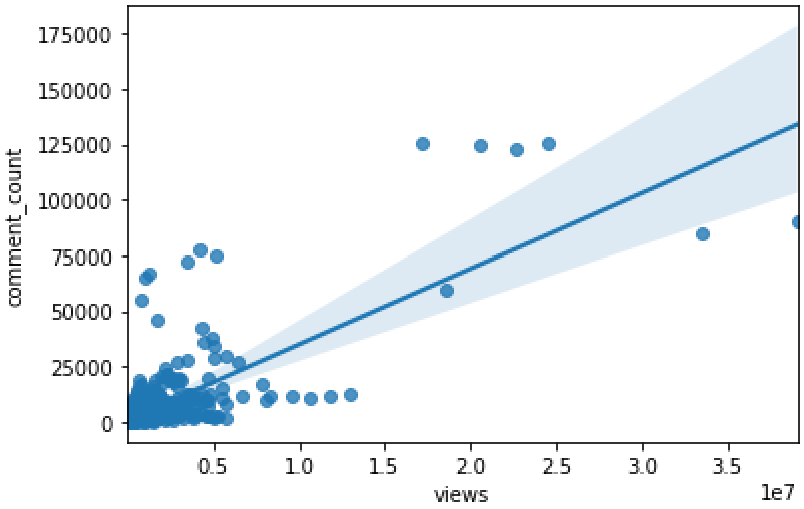


In [50]:

```
sns.regplot(x = 'views',y = 'comment_count',data = full_df[0:1000])
```

Out[50]:

```
<AxesSubplot:xlabel='views', ylabel='comment_count'>
```



In [51]:

```
corr = full_df[['views','likes','dislikes','comment_count']].corr()
corr
```

Out[51]:

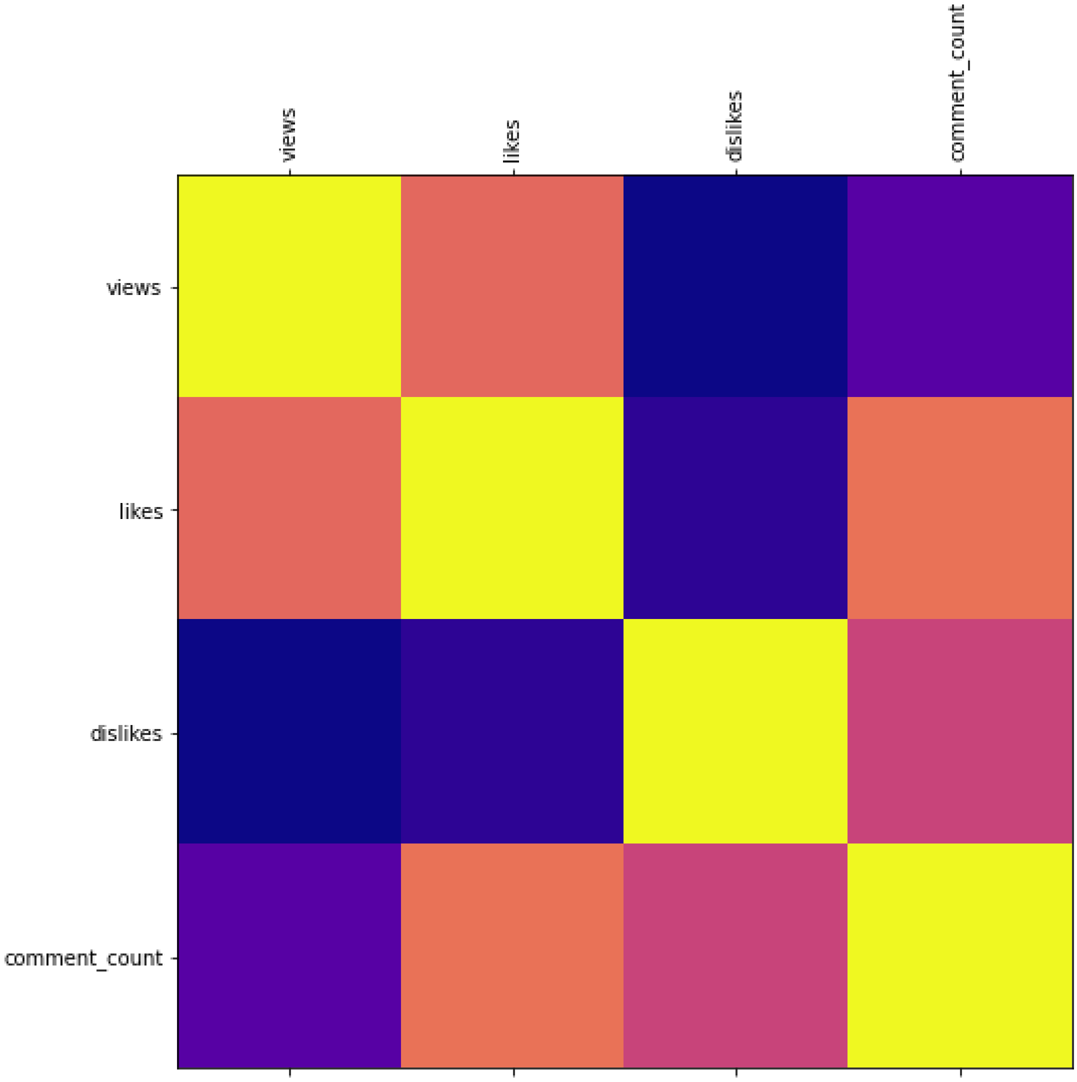
	views	likes	dislikes	comment_count
views	1.000000	0.777796	0.421653	0.510030
likes	0.777796	1.000000	0.453710	0.794490
dislikes	0.421653	0.453710	1.000000	0.705182
comment_count	0.510030	0.794490	0.705182	1.000000

In [52]:

```
fig = plt.figure(figsize = (15,8))
plt.matshow(corr,cmap = 'plasma',fignum = fig.number)
plt.xticks(range(len(corr.columns)),corr.columns,rotation = 'vertical')
plt.yticks(range(len(corr.columns)),corr.columns)
```

Out[52]:

```
([<matplotlib.axis.YTick at 0x24260fe73d0>,
  <matplotlib.axis.YTick at 0x24260fe1f70>,
  <matplotlib.axis.YTick at 0x24260fbae80>,
  <matplotlib.axis.YTick at 0x24260fea8b0>],
 [Text(0, 0, 'views'),
  Text(0, 1, 'likes'),
  Text(0, 2, 'dislikes'),
  Text(0, 3, 'comment_count')])
```

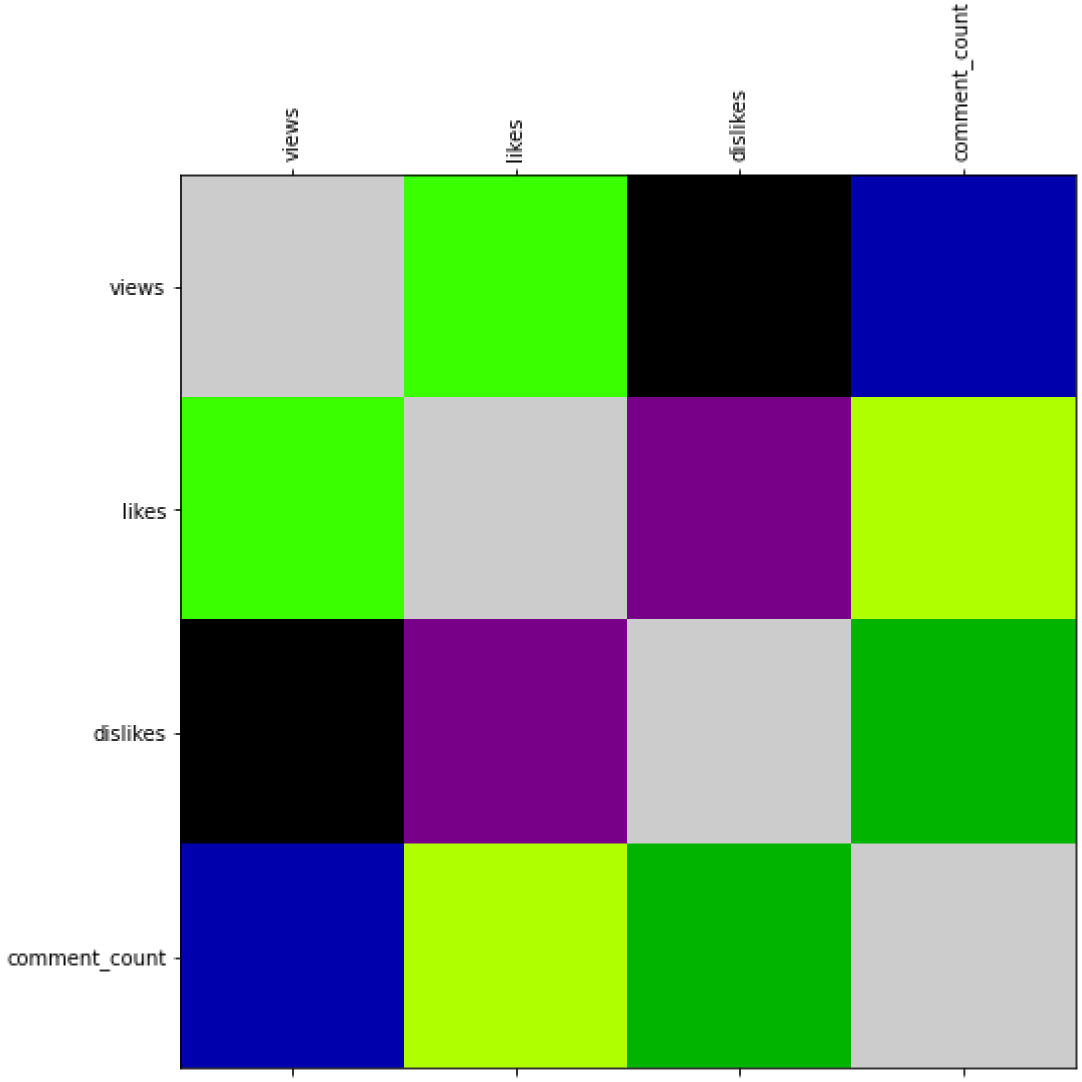


In [53]:

```
fig = plt.figure(figsize = (15,8))
plt.matshow(corr,cmap = 'nipy_spectral',fignum = fig.number)
plt.xticks(range(len(corr.columns)),corr.columns,rotation = 'vertical')
plt.yticks(range(len(corr.columns)),corr.columns)
```

Out[53]:

```
([<matplotlib.axis.YTick at 0x2425bbf2cd0>,
  <matplotlib.axis.YTick at 0x24261439c70>,
  <matplotlib.axis.YTick at 0x2424b7943d0>,
  <matplotlib.axis.YTick at 0x2425bcf0250>],
 [Text(0, 0, 'views'),
  Text(0, 1, 'likes'),
  Text(0, 2, 'dislikes'),
  Text(0, 3, 'comment_count')])
```

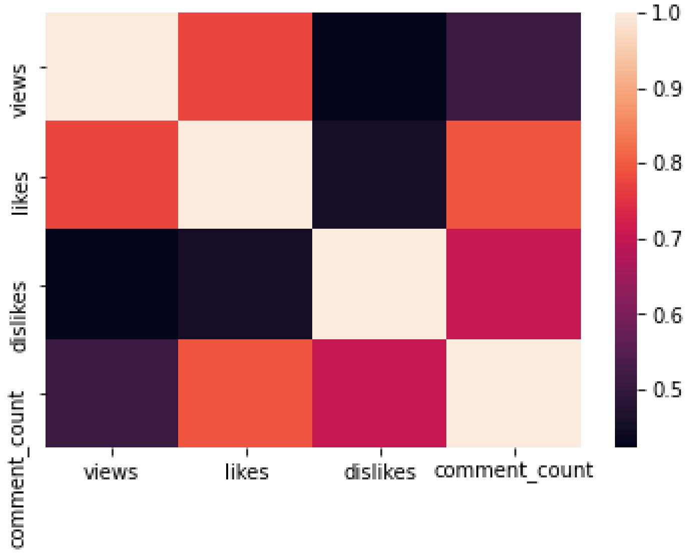


In [54]:

```
sns.heatmap(full_df[['views','likes','dislikes','comment_count']].corr())
```

Out[54]:

```
<AxesSubplot:>
```

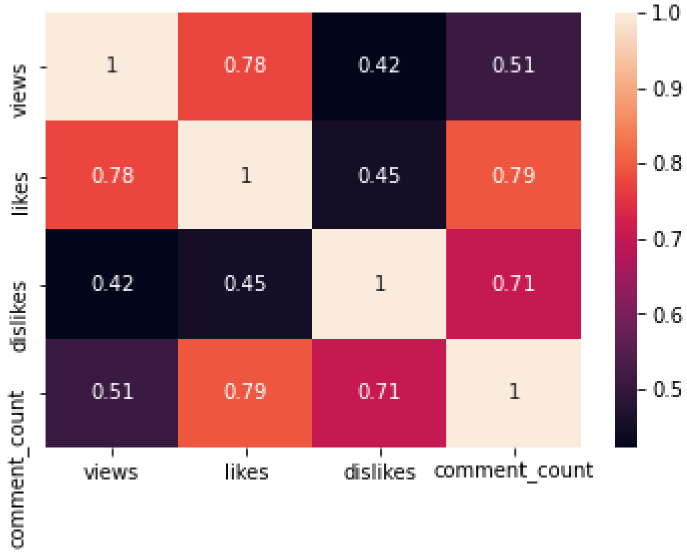


In [55]:

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

Out[55]:

```
<AxesSubplot:>
```



Channels with most no of trending videos

In [56]:

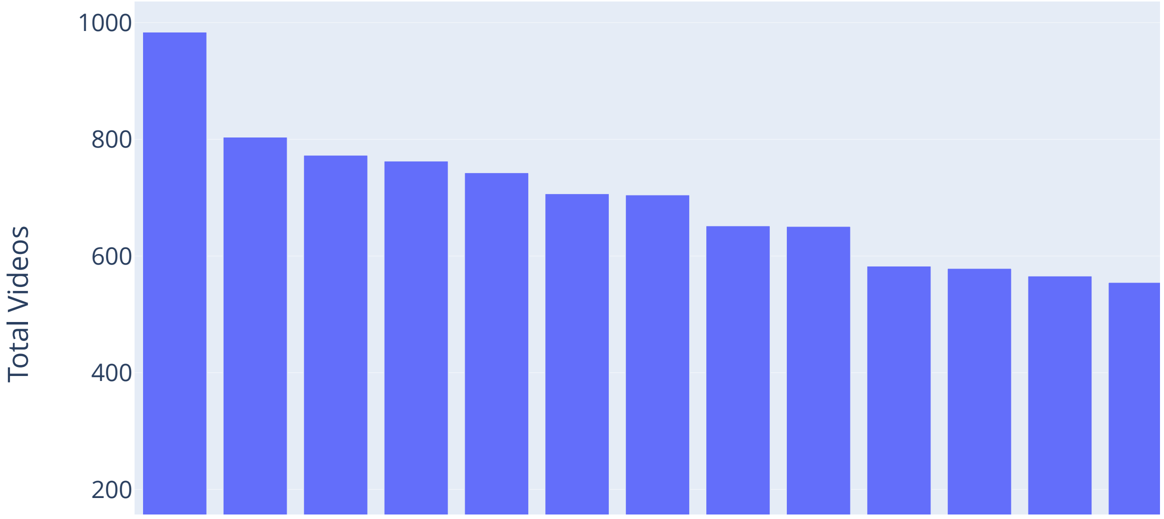
```
title')['video_id'].count().sort_values(ascending = False).to_frame().reset_index().rename(c
```

In [57]:

```
import plotly.express as px
```

In [58]:

```
px.bar(data_frame = cdf[0:20],x = 'channel_title',y = 'Total Videos')
```

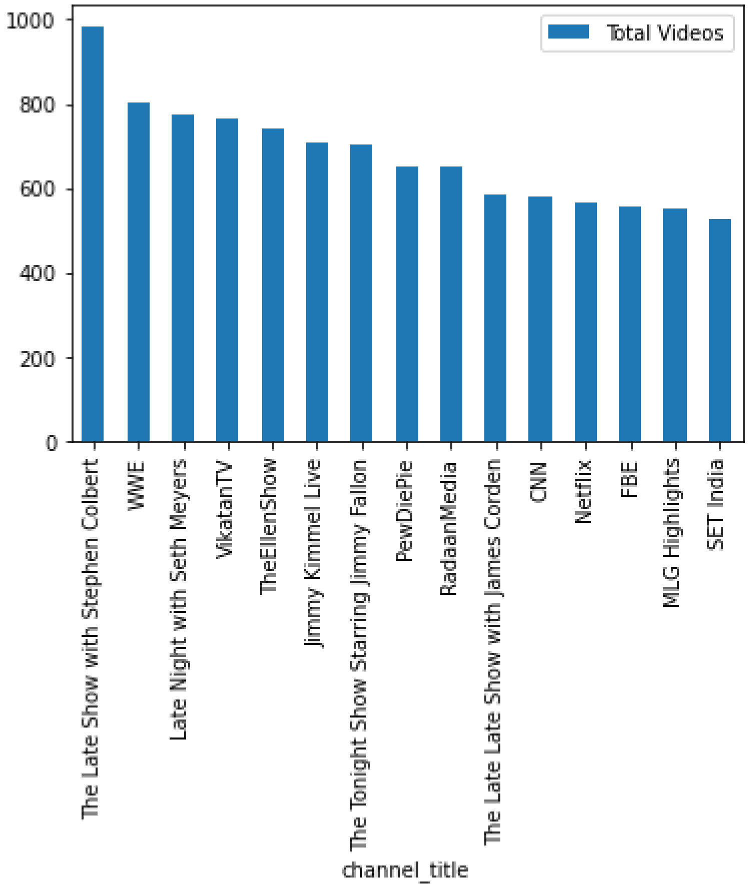


In [59]:

```
cdf[0:15].plot.bar(x = 'channel_title',y = 'Total Videos')
```

Out[59]:

```
<AxesSubplot:xlabel='channel_title'>
```



Punctuation

In [60]:

```
import string
```

In [61]:

```
string.punctuation
```

Out[61]:

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

In [62]:

```
full_df['title'].head(10)
```

Out[62]:

```
0          Eminem - Walk On Water (Audio) ft. BeyoncÃ©
1                        PLUSH - Bad Unboxing Fan Mail
2    Racist Superman | Rudy Mancuso, King Bach & Le...
3                             I Dare You: GOING BALD!?
4          Ed Sheeran - Perfect (Official Music Video)
5    Jake Paul Says Alissa Violet CHEATED with LOGA...
6               Vanoss Superhero School - New Students
7                   WE WANT TO TALK ABOUT OUR MARRIAGE
8                 THE LOGANG MADE HISTORY. LOL. AGAIN.
9    Finally Sheldon is winning an argument about t...
Name: title, dtype: object
```

In [63]:

```
def punc_count(x):
    return len([c for c in x if c in string.punctuation])
```

In [64]:

```
sample = full_df[0:500]
```

In [65]:

```
pd.options.mode.chained_assignment = None
```

In [66]:

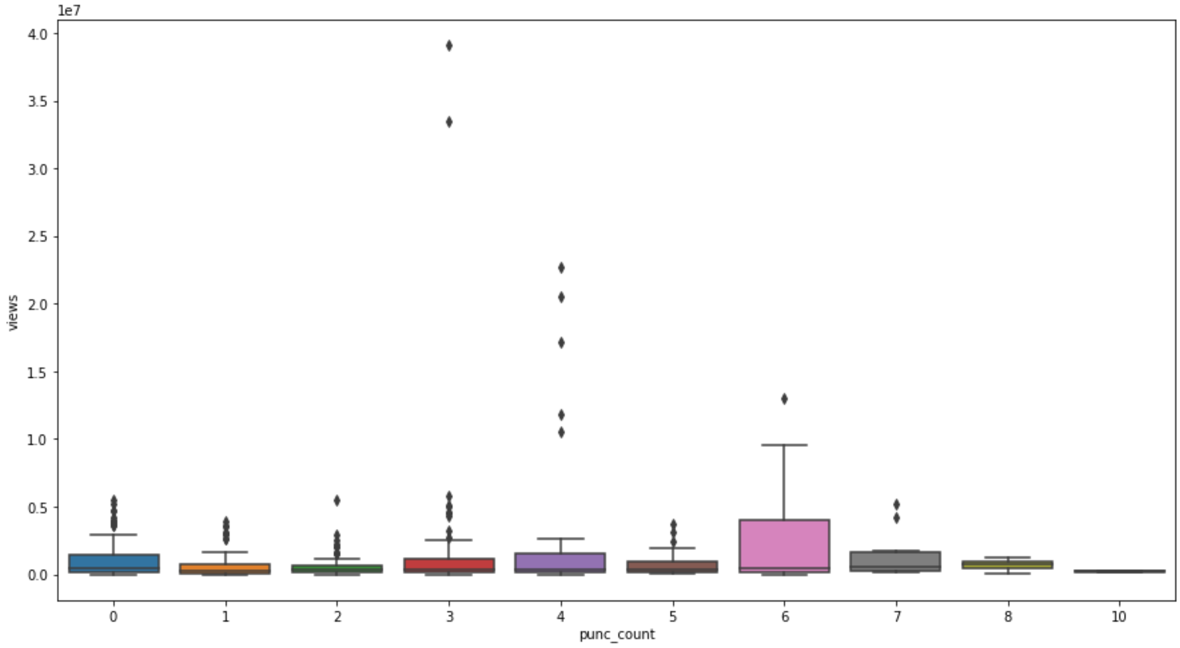
```
sample['punc_count'] = sample['title'].apply(punc_count)
```

In [67]:

```
plt.figure(figsize = (15,8))
sns.boxplot(data = sample,x = 'punc_count',y = 'views')
```

Out[67]:

```
<AxesSubplot:xlabel='punc_count', ylabel='views'>
```



In [68]:

```
sample['punc_count'].corr(sample['views'])
```

Out[68]:

```
0.10383139848353996
```