- YOUTUBE TRENDING KEYWORD PREDICTION

In [1]:
```
# importing the necessary libraries

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

sns.set()
```

In [2]:
```
# reading CSV files

Ng_data = pd.read_csv("NG_youtube_data.csv", encoding='utf-8')
US_data = pd.read_csv("us_Youtube_data.csv", encoding='utf-8')
```

In [3]:
```
Ng_data.head(3)
```

Out[3]:

	title	videoId	channelId	channelTitle	categoryId	V
0	FAMILY MATTERS | Brodashaggi | Mr Ibu | Kiriku...	Zv1i-Cdn4Ik	UCG6orNVuXIICv9_ifH6msIA	official BRODA SHAGGI	23	
1	Best Friends in the World: Senior Year | Episo...	BmCkK7W7j0Q	UCCl71WmqkG8BjVuySOSCh_Q	Neptune3 Studios	1	
2	ROMOLA ALAKARA - Latest 2022 Yoruba Movie Star...	seZb9P7H6MU	UCK2-495cMvdhaMqXcLlgmcg	YORUBAPLAY	24	

In [4]:
```
US_data.head(3)
```

Out[4]:

	channelId	channelTitle	videoId	publishedAt	videoTi
0	UCU1_l0ZJyTK_7HZZ3Ruw8Dg	MAPS	pTnk3ziVVRM	2014-01-10T01:24:57.000Z	Psyched Horizo Beyo Psychothera Wor
1	UCLuO2lUqHrPIlpx0hFenV2g	Tink Tink Club	cuJjSeHZIrg	2015-06-18T16:56:04.000Z	Episode 3 Dr. Jan Fadim
2	UCihqrkaOgVMfLNo2W1hSliA	Podcast Bunk	luyuZfWtGgg	2016-05-01T05:33:13.000Z	#3 Microdos from T Adam and Drew Sh

In [5]:
```
Ng_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 14 columns):
 #   Column                   Non-Null Count  Dtype 
---  ------                   --------------  ----- 
 0   title                    50 non-null     object
 1   videoId                  50 non-null     object
 2   channelId                50 non-null     object
 3   channelTitle             50 non-null     object
 4   categoryId               50 non-null     int64 
 5   VideoCategoryDescription  50 non-null     object
 6   Description              44 non-null     object
 7   viewCount                50 non-null     int64 
 8   likeCount                50 non-null     int64 
 9   dislikeCount             50 non-null     int64 
 10  commentCount             50 non-null     int64 
 11  favoriteCount            50 non-null     int64 
 12  publishedAt              50 non-null     object
 13  duration                 50 non-null     object
dtypes: int64(6), object(8)
memory usage: 5.6+ KB
```

In [6]:
```
US_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 115 entries, 0 to 114
Data columns (total 16 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   channelId           115 non-null    object 
 1   channelTitle        115 non-null    object 
 2   videoId             115 non-null    object 
 3   publishedAt         115 non-null    object 
 4   videoTitle          115 non-null    object 
 5   videoDescription    109 non-null    object 
 6   videoCategoryId     115 non-null    int64  
 7   videoCategoryLabel  115 non-null    object 
 8   duration            115 non-null    object 
 9   durationSec         115 non-null    int64  
 10  definition          115 non-null    object 
 11  caption             115 non-null    bool   
 12  viewCount           115 non-null    int64  
 13  likeCount           111 non-null    float64
 14  dislikeCount        111 non-null    float64
 15  commentCount        113 non-null    float64
dtypes: bool(1), float64(3), int64(3), object(9)
memory usage: 13.7+ KB
```

In [7]:
```
US_data = US_data.iloc[:80]
US_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 80 entries, 0 to 79
Data columns (total 16 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   channelId           80 non-null     object 
 1   channelTitle        80 non-null     object 
 2   videoId             80 non-null     object 
 3   publishedAt         80 non-null     object 
 4   videoTitle          80 non-null     object 
 5   videoDescription    76 non-null     object 
 6   videoCategoryId     80 non-null     int64  
 7   videoCategoryLabel  80 non-null     object 
 8   duration            80 non-null     object 
 9   durationSec         80 non-null     int64  
 10  definition          80 non-null     object 
 11  caption             80 non-null     bool   
 12  viewCount           80 non-null     int64  
 13  likeCount           76 non-null     float64
 14  dislikeCount        76 non-null     float64
 15  commentCount        78 non-null     float64
dtypes: bool(1), float64(3), int64(3), object(9)
memory usage: 9.6+ KB
```

In [8]:
```
Ng_data.sort_values(["viewCount", "likeCount", "commentCount"], ascending
```

In [9]:
```
Ng_data.head()
```

Out[9]:

	title	videoId	channelId	channelTitle	categoryId
35	SEE WAHALA O	qo-K3YlcfU0	UCjSzBGfo9gDXP0OerKJ9GZg	Twyse Ereme	24
26	Blaqbonez - Fake Nikes (Feat. Blxckie & Cheque...	oL_NxgHJ_M4	UC0iZ_gqCk22K0jWscf75lhg	Blaqbonez	10
40	PERFECT ASSISTANT- Watch Maurice Sam and Shine...	MyVG-UcdLU8	UCnZ28GESUBXyZSIvzcs1RXA	ChinneyLoveEze Tv	24
14	The Reign of Omo Igbo	ASnTvDizlWY	UCIwUNuYEcZiiJqjU04yN30A	AyoAjewole Woliagba-YPM	23
17	Aranda Ekun Part 3 - Latest Yoruba Movie 2022 ...	bC95DBzHuw8	UC3BYT_REpOi_DNhluVO2ndw	Yorubahood	24

In [10]:
```
US_data.sort_values(["viewCount", "likeCount", "commentCount"], ascending
US_data.head(4)
```

Out[10]:

	channelId	channelTitle	videoId	publishedAt	vid
8	UC2nzGL6I-SRc_gkBLWLDVWg	ellie pipkins	akZrftACY1I	2016-06-04T12:39:29.000Z	Microd SUPERNA
74	UCGcF4QHx-bUGNT7nIo-cxDQ	Motion Brand	DwDowxM5HLI	2016-05-20T10:16:37.000Z	Micro
3	UCgbWWPn3VYYzxjffZbfj9GQ	Alan Springwind	cng_ZhQf8iY	2016-01-25T04:48:22.000Z	Micro Away Th Ol
50	UCxw4-gbOtqRLdh9C3Ne-XAw	E-PAK Machinery, Inc.	b0RDMUgn_3E	2016-06-03T20:28:37.000Z	Fully Au Micro Pisto

In [11]:
```
# Renaming some columns in the US data

US_data["VideoCategoryDescription"] = US_data["videoCategoryLabel"]
US_data["Description"] = US_data["videoDescription"]
```

In [12]:
```
Ng_data.isna().sum()
```

Out[12]:
```
title                       0
videoId                     0
channelId                   0
channelTitle                0
categoryId                  0
VideoCategoryDescription    0
Description                 6
viewCount                   0
likeCount                   0
dislikeCount                0
commentCount                0
favoriteCount               0
publishedAt                 0
duration                    0
dtype: int64
```

In [13]:
```
US_data.isna().sum()
```

Out[13]:
```
channelId                   0
channelTitle                0
videoId                     0
publishedAt                 0
videoTitle                  0
videoDescription            4
videoCategoryId             0
videoCategoryLabel          0
duration                    0
durationSec                 0
definition                  0
caption                     0
viewCount                   0
likeCount                   4
dislikeCount                4
commentCount                2
VideoCategoryDescription    0
Description                 4
dtype: int64
```

In [14]:
```
Ng_data = Ng_data.dropna()
print(Ng_data.isna().sum())
```

```
title                       0
videoId                     0
channelId                   0
channelTitle                0
categoryId                  0
VideoCategoryDescription    0
Description                 0
viewCount                   0
likeCount                   0
dislikeCount                0
commentCount                0
favoriteCount               0
publishedAt                 0
duration                    0
dtype: int64
```

In [15]:
```
US_data = US_data.dropna()
print(US_data.isna().sum())
```

```
channelId                   0
channelTitle                0
videoId                     0
publishedAt                 0
videoTitle                  0
videoDescription            0
videoCategoryId             0
videoCategoryLabel          0
duration                    0
durationSec                 0
definition                  0
caption                     0
viewCount                   0
likeCount                   0
dislikeCount                0
commentCount                0
VideoCategoryDescription    0
Description                 0
dtype: int64
```

In [16]:
```
Ng_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 44 entries, 35 to 46
Data columns (total 14 columns):
 #   Column                    Non-Null Count  Dtype 
---  ------                    --------------  ----- 
 0   title                     44 non-null     object
 1   videoId                   44 non-null     object
 2   channelId                 44 non-null     object
 3   channelTitle              44 non-null     object
 4   categoryId                44 non-null     int64 
 5   VideoCategoryDescription  44 non-null     object
 6   Description               44 non-null     object
 7   viewCount                 44 non-null     int64 
 8   likeCount                 44 non-null     int64 
 9   dislikeCount              44 non-null     int64 
 10  commentCount              44 non-null     int64 
 11  favoriteCount             44 non-null     int64 
 12  publishedAt               44 non-null     object
 13  duration                  44 non-null     object
dtypes: int64(6), object(8)
memory usage: 5.2+ KB
```

In [17]:
```
# subsetting the columns to just the specific columns necessary.

Ng_data =Ng_data[["title","videoId", "VideoCategoryDescription", "Descrip

Ng_data.head()
```

Out[17]:

	title	videoId	VideoCategoryDescription	Description	vie
35	SEE WAHALA O	qo-K3YlcfU0	Entertainment	#IzzGone #TwyseAndFamily\n\nTwyse and Family T...	
26	Blaqbonez - Fake Nikes (Feat. Blxckie & Cheque...	oL_NxgHJ_M4	Music	Get "Young Preacher" by BLAQBONEZ here: https...	
40	PERFECT ASSISTANT- Watch Maurice Sam and Shine...	MyVG-UcdLU8	Entertainment	Subscribe to our YouTube channel\nhttps://www....	
17	Aranda Ekun Part 3 - Latest Yoruba Movie 2022 ...	bC95DBzHuw8	Entertainment	The youths of Ilu Iloro want Morolayo Adunni t...	
49	Police Internship	velB8gLx5GM	Comedy	Tegwolo's father has just joined the police. T...	

In [24]:
```
# subsetting the columns to just the specific columns necessary.

US_data =US_data[["channelTitle","videoId", "VideoCategoryDescription", "

US_data.head(5)
```

Out[24]:

	channelTitle	videoId	VideoCategoryDescription	De
8	ellie pipkins	akZrftACY1I	People & Blogs	a original. If you would like to su
3	Alan Springwind	cng_ZhQf8iY	Entertainment	https://www.spreaker.com/user/s
16	CABIvideo	xl7pRQGIMdg	Science & Technology	Farmer to farmer training fig
72	ICRISAT Co	dgWKr3Mymcw	Nonprofits & Activism	Microdosing: Up-scaling Dr Ta
60	ICRISAT Co	VjrWr7mpruI	Nonprofits & Activism	Future of microdosing Dr Ramac

In [19]:
```
Ng_data.describe()
```

Out[19]:

	viewCount	likeCount	commentCount
count	4.400000e+01	4.400000e+01	44.000000
mean	2.338803e+06	1.467370e+05	3672.886364
std	5.549736e+06	5.714969e+05	7529.054492
min	8.037500e+04	1.745000e+03	0.000000
25%	2.030915e+05	4.876750e+03	304.250000
50%	6.039595e+05	1.166200e+04	613.000000
75%	1.657581e+06	5.696525e+04	2732.500000
max	3.454361e+07	3.757194e+06	32438.000000

In [20]:
```
US_data.describe()
```

Out[20]:

	viewCount	likeCount	commentCount
count	72.000000	72.000000	72.000000
mean	14431.930556	264.500000	119.861111
std	65331.304830	1054.726772	495.047936
min	2.000000	0.000000	0.000000
25%	153.000000	0.000000	0.000000
50%	610.000000	5.000000	1.500000
75%	2364.750000	28.000000	25.250000
max	526243.000000	7046.000000	3672.000000

In [21]:
```
Ng_data.corr()
```

```
/var/folders/0c/ztv6vp6971zcyw9kflycnw100000gn/T/ipykernel_13980/3142781
792.py:1: FutureWarning: The default value of numeric_only in DataFrame.
corr is deprecated. In a future version, it will default to False. Selec
t only valid columns or specify the value of numeric_only to silence thi
s warning.
  Ng_data.corr()
```

Out[21]:

	viewCount	likeCount	commentCount
viewCount	1.000000	0.954152	0.714238
likeCount	0.954152	1.000000	0.641059
commentCount	0.714238	0.641059	1.000000

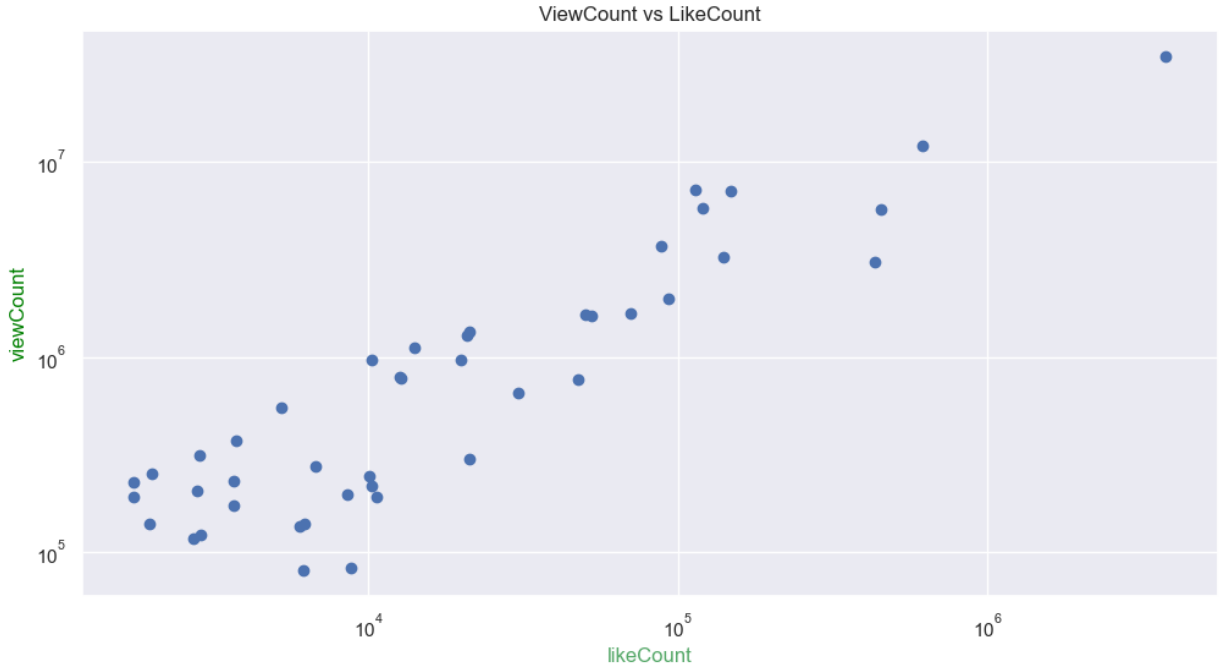
In [22]:
```
US_data.corr()
```

```
/var/folders/0c/ztv6vp6971zcyw9kflycnw100000gn/T/ipykernel_13980/2308284
19.py:1: FutureWarning: The default value of numeric_only in DataFrame.c
orr is deprecated. In a future version, it will default to False. Select
only valid columns or specify the value of numeric_only to silence this
warning.
  US_data.corr()
```

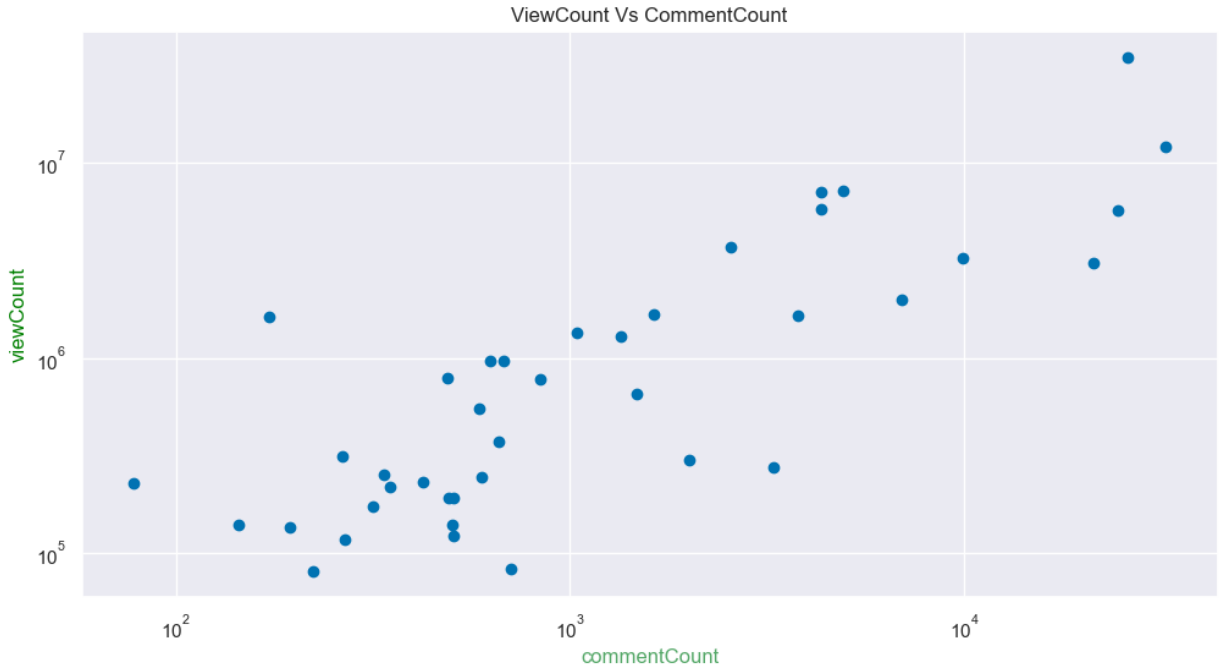
Out[22]:

	viewCount	likeCount	commentCount
viewCount	1.000000	0.932612	0.615707
likeCount	0.932612	1.000000	0.839458
commentCount	0.615707	0.839458	1.000000

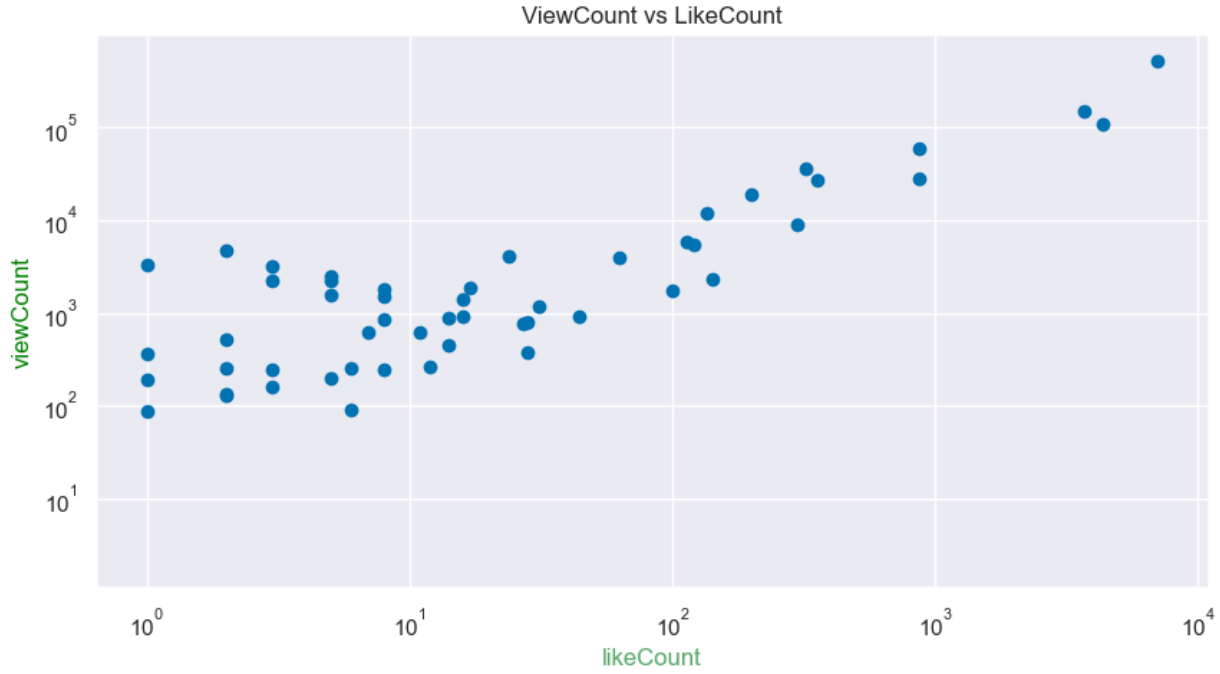
In [25]:
```
plt.figure(figsize=(12,6))
plt.scatter(Ng_data['likeCount'], Ng_data['viewCount'])
plt.xscale("log")
plt.yscale('log')
plt.style.use('seaborn-v0_8-colorblind')
plt.title("ViewCount vs LikeCount")
plt.ylabel("viewCount", color= "Green")
plt.xlabel("likeCount", color= "g")
plt.show()
```



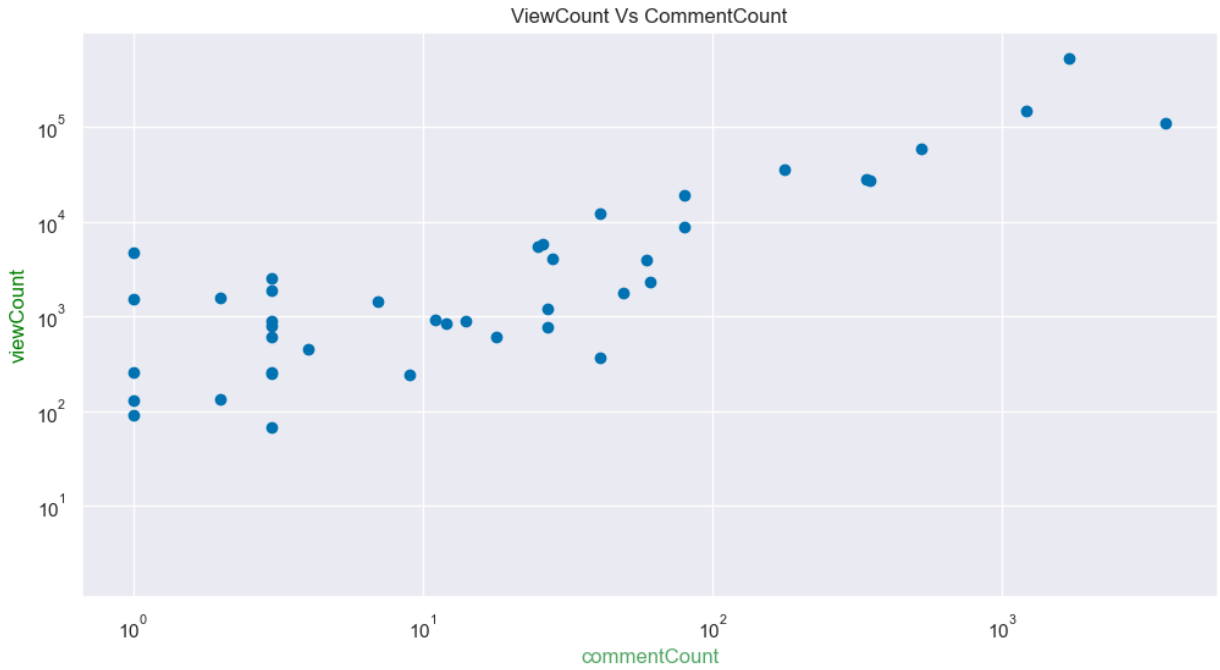
In [26]:
```
# plotting ViewCount against likeCount

plt.figure(figsize=(12,6))
plt.scatter(Ng_data['commentCount'], Ng_data['viewCount'])
plt.xscale("log")
plt.yscale('log')
plt.style.use('seaborn-v0_8-colorblind')
plt.title("ViewCount Vs CommentCount")
plt.ylabel("viewCount", color= "Green")
plt.xlabel("commentCount", color= "g")
plt.show()
```



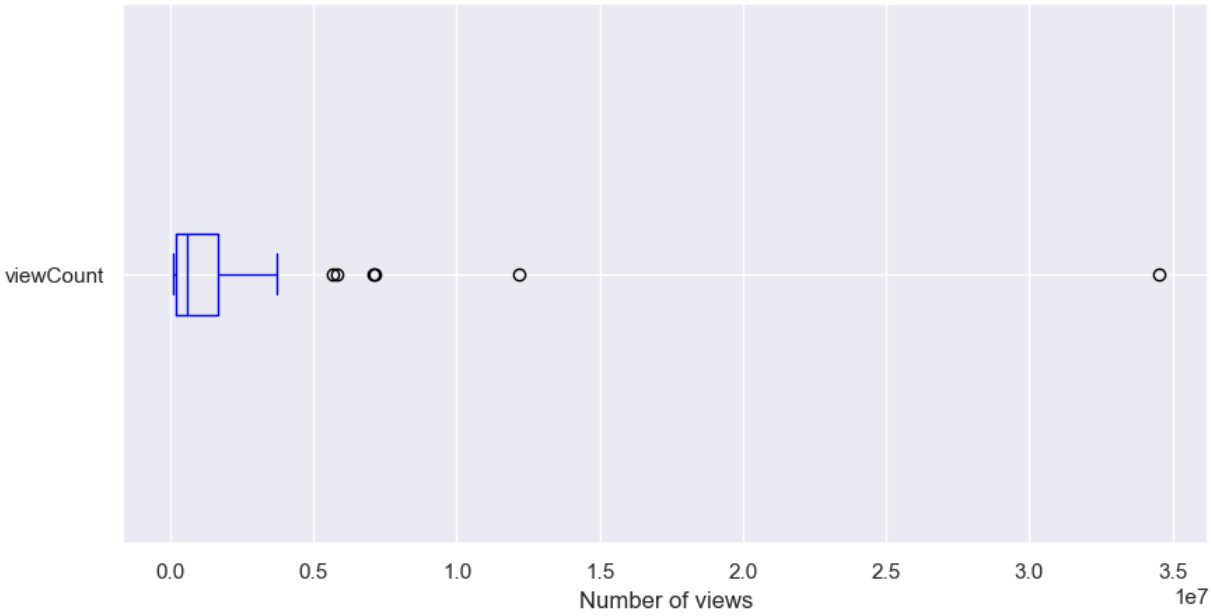
In [27]:
```
plt.figure(figsize=(10,5))
plt.scatter(US_data['likeCount'], US_data['viewCount'])
plt.xscale("log")
plt.yscale('log')
plt.style.use('seaborn-v0_8-colorblind')
plt.title("ViewCount vs LikeCount")
plt.ylabel("viewCount", color= "Green")
plt.xlabel("likeCount", color= "g")
plt.show()
```



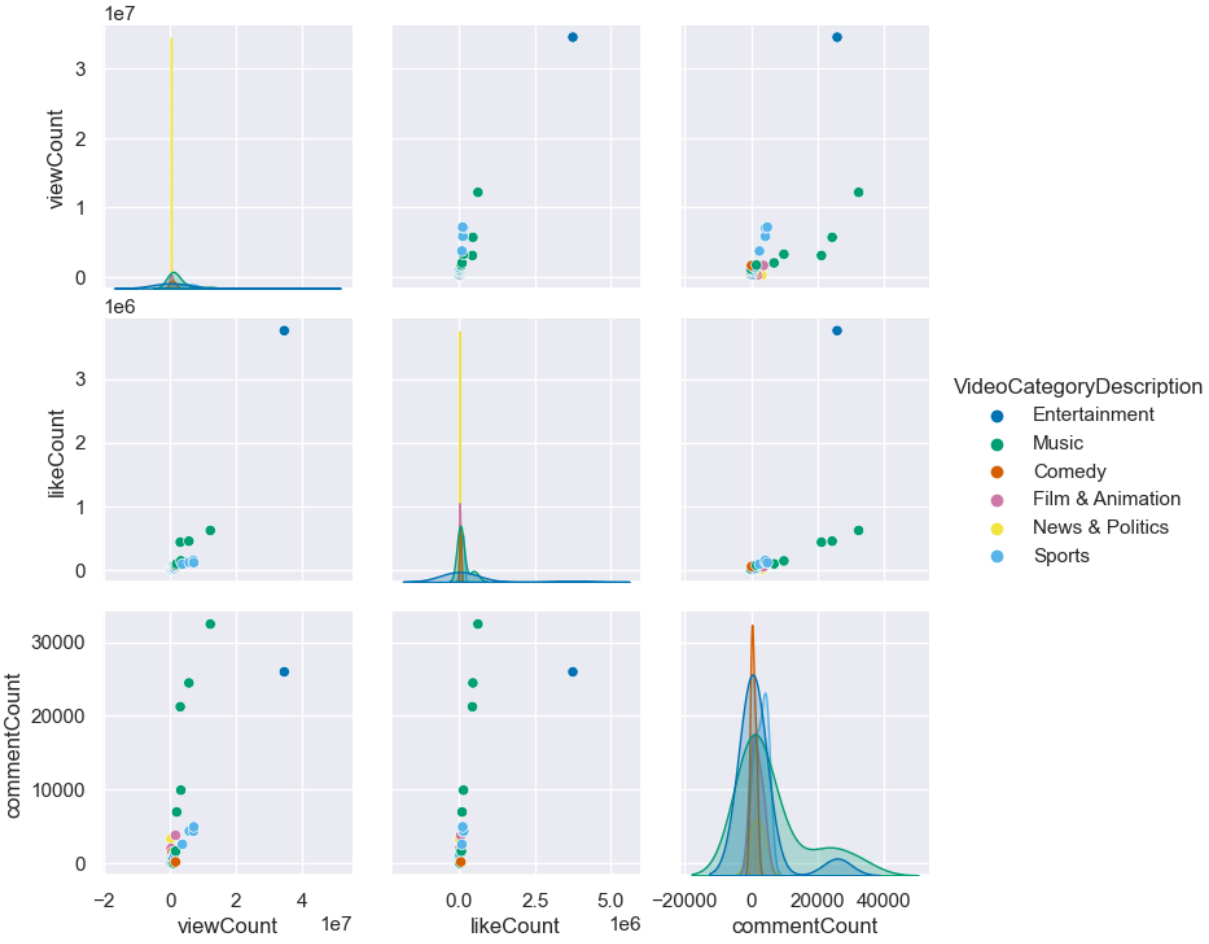
In [28]:
```
# plotting ViewCount against likeCount

plt.figure(figsize=(12,6))
plt.scatter(US_data['commentCount'], US_data['viewCount'])
plt.xscale("log")
plt.yscale('log')
plt.style.use('seaborn-v0_8-colorblind')
plt.title("ViewCount Vs CommentCount")
plt.ylabel("viewCount", color= "Green")
plt.xlabel("commentCount", color= "g")
plt.show()
```

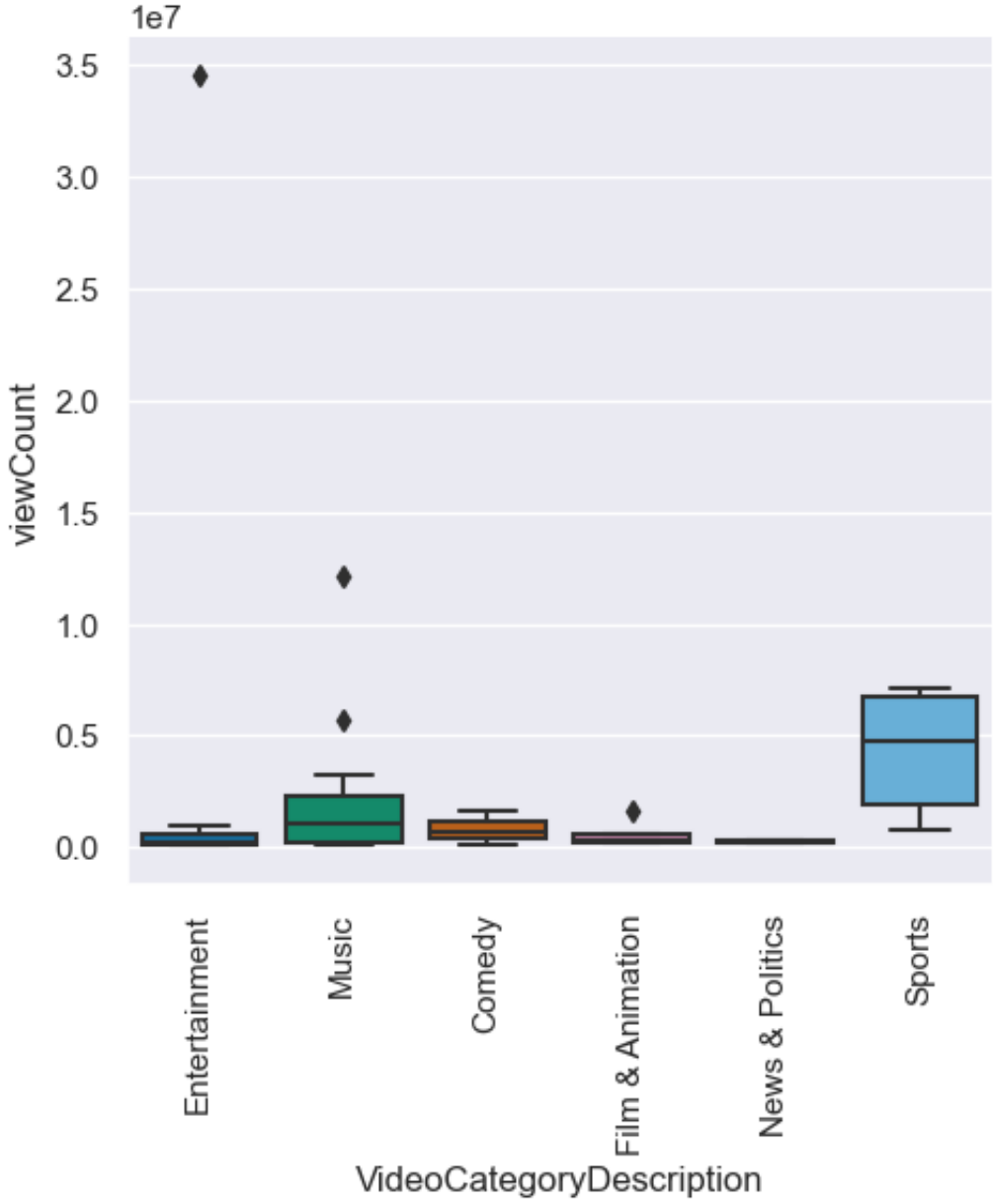


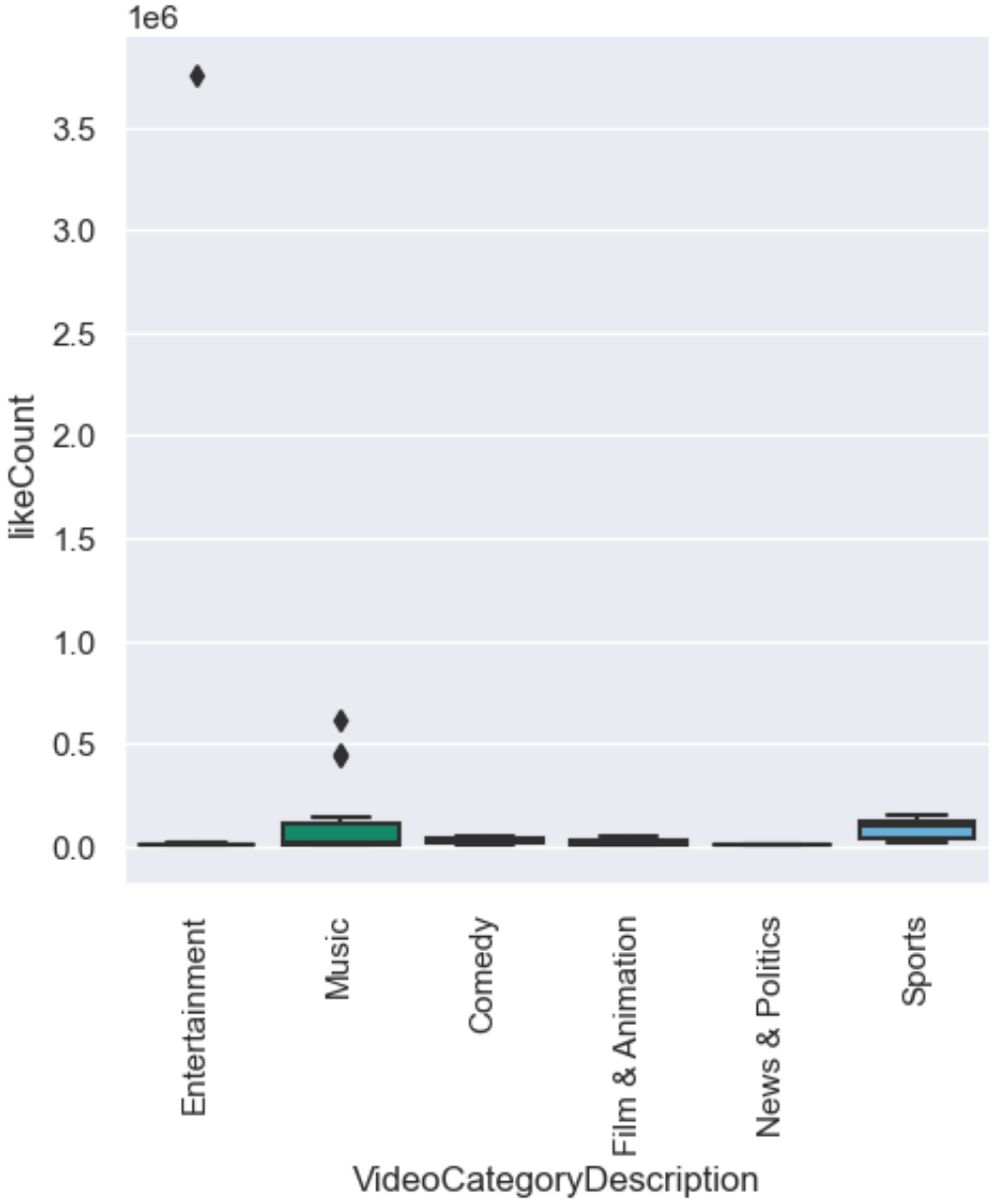
In [29]:
```
plt.figure(figsize= (12,6))
Ng_data['viewCount'].plot(kind='box', vert=False, color='blue',
                          figsize=(10,5))

plt.xlabel('Number of views')
plt.show()
# Boxplots with seaborn
sns.pairplot(data=Ng_data, hue="VideoCategoryDescription")
sns.catplot(x="VideoCategoryDescription", y="viewCount", kind="box", data
sns.catplot(x="VideoCategoryDescription", y="likeCount", kind="box", data
sns.catplot(x="VideoCategoryDescription", y="commentCount", kind="box", d
sns.displot(Ng_data, x="viewCount", hue="VideoCategoryDescription")
```

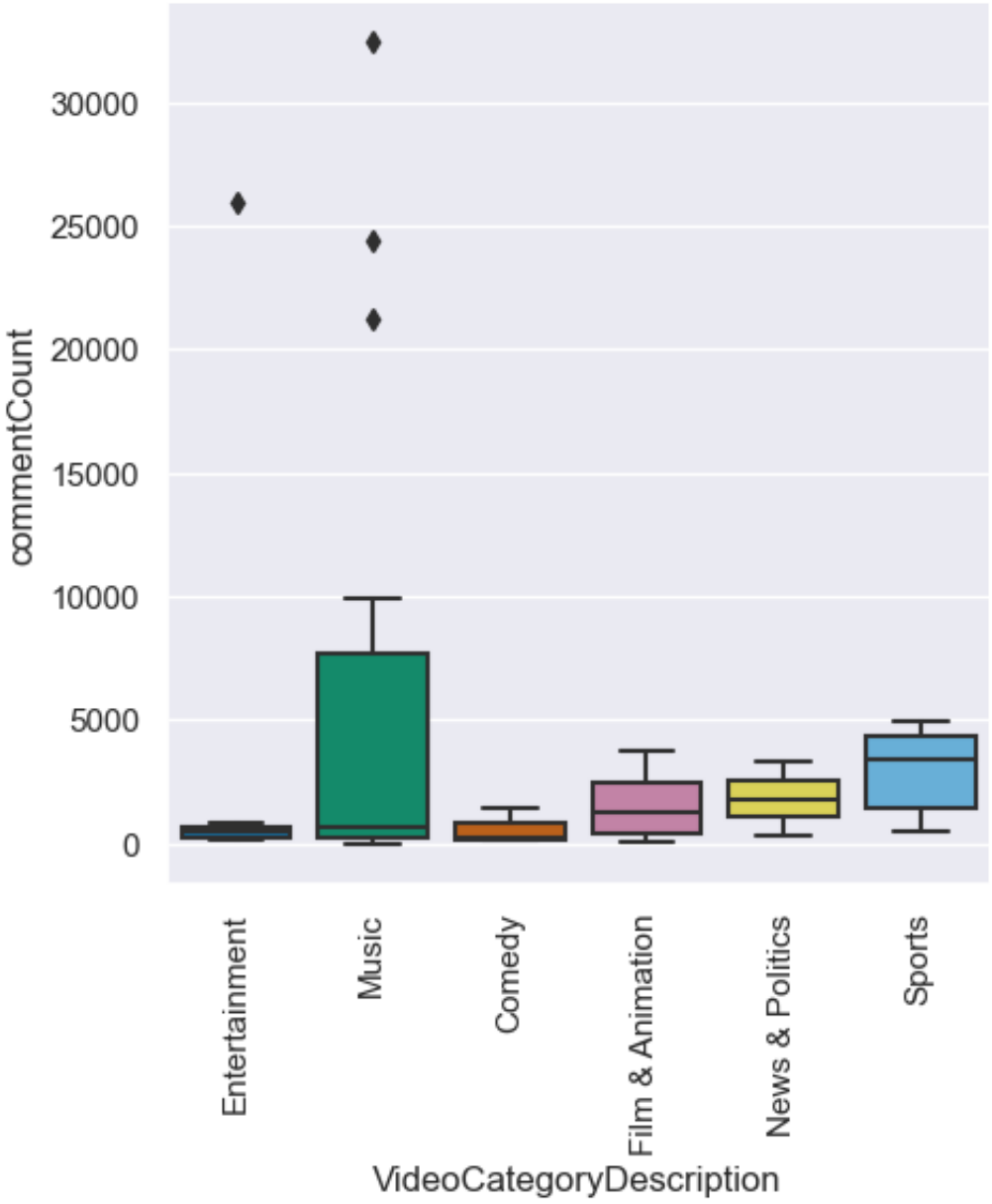


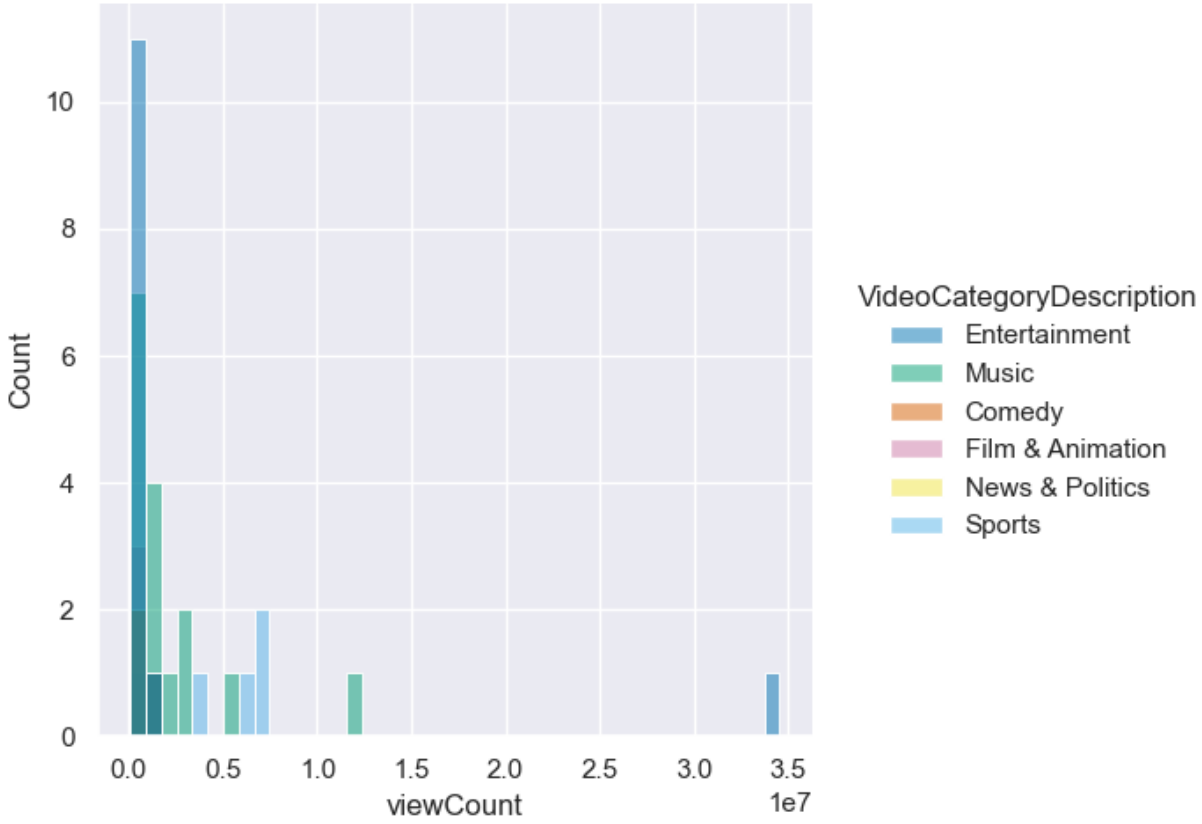
Out[29]: <seaborn.axisgrid.FacetGrid at 0x14e716fa0>



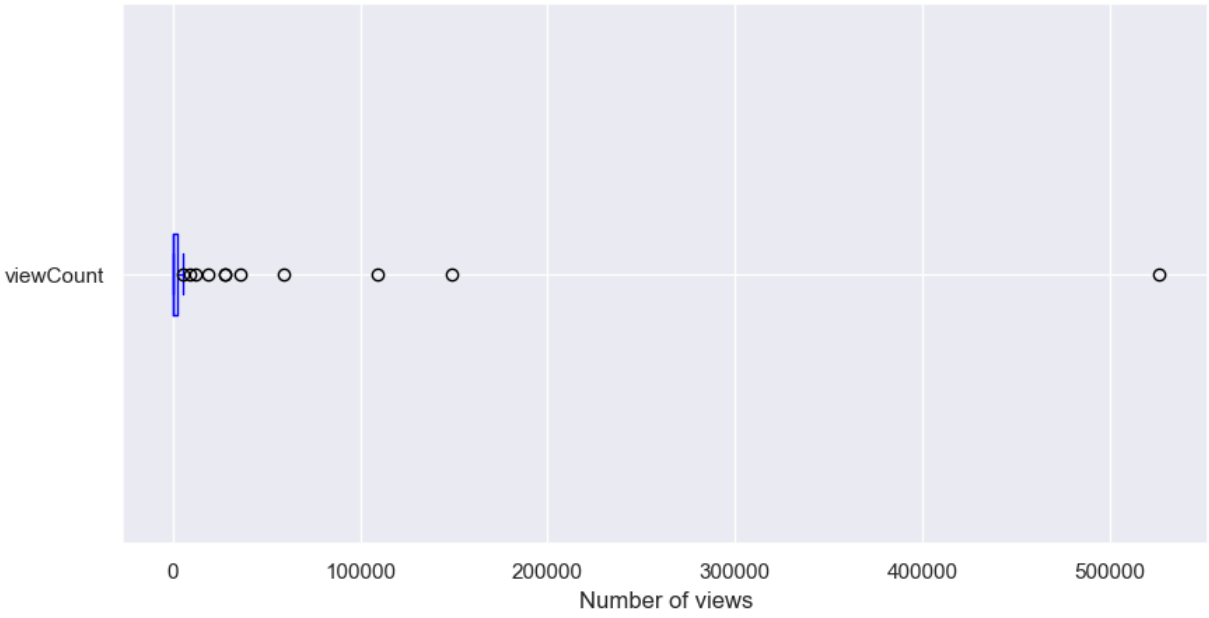

1e7
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
viewCount
Entertainment
Music
Comedy
Film & Animation
News & Politics
Sports
VideoCategoryDescription


1e6
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
likeCount
Entertainment
Music
Comedy
Film & Animation
News & Politics
Sports
VideoCategoryDescription

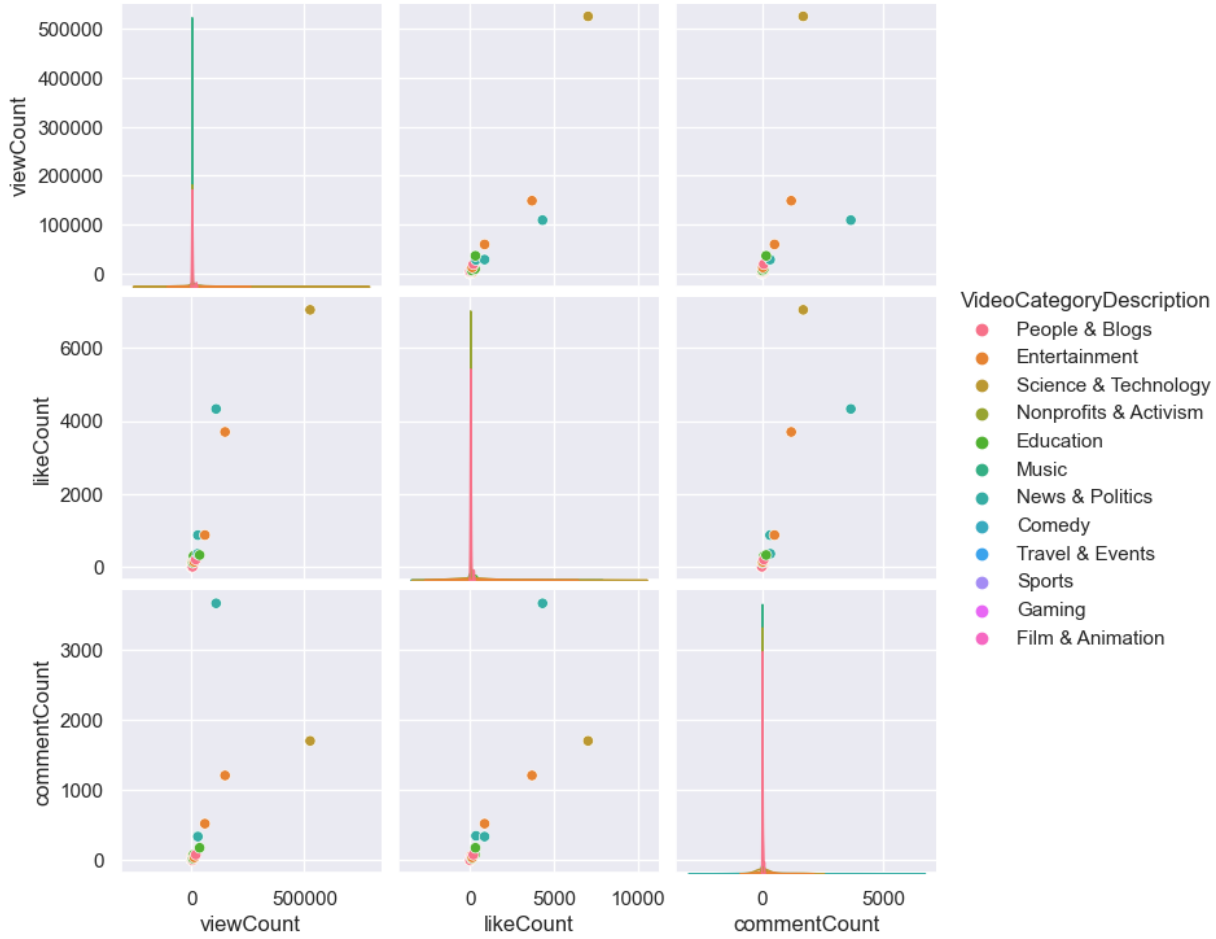

commentCount
30000
25000
20000
15000
10000
5000
0
Entertainment
Music
Comedy
Film & Animation
News & Politics
Sports
VideoCategoryDescription

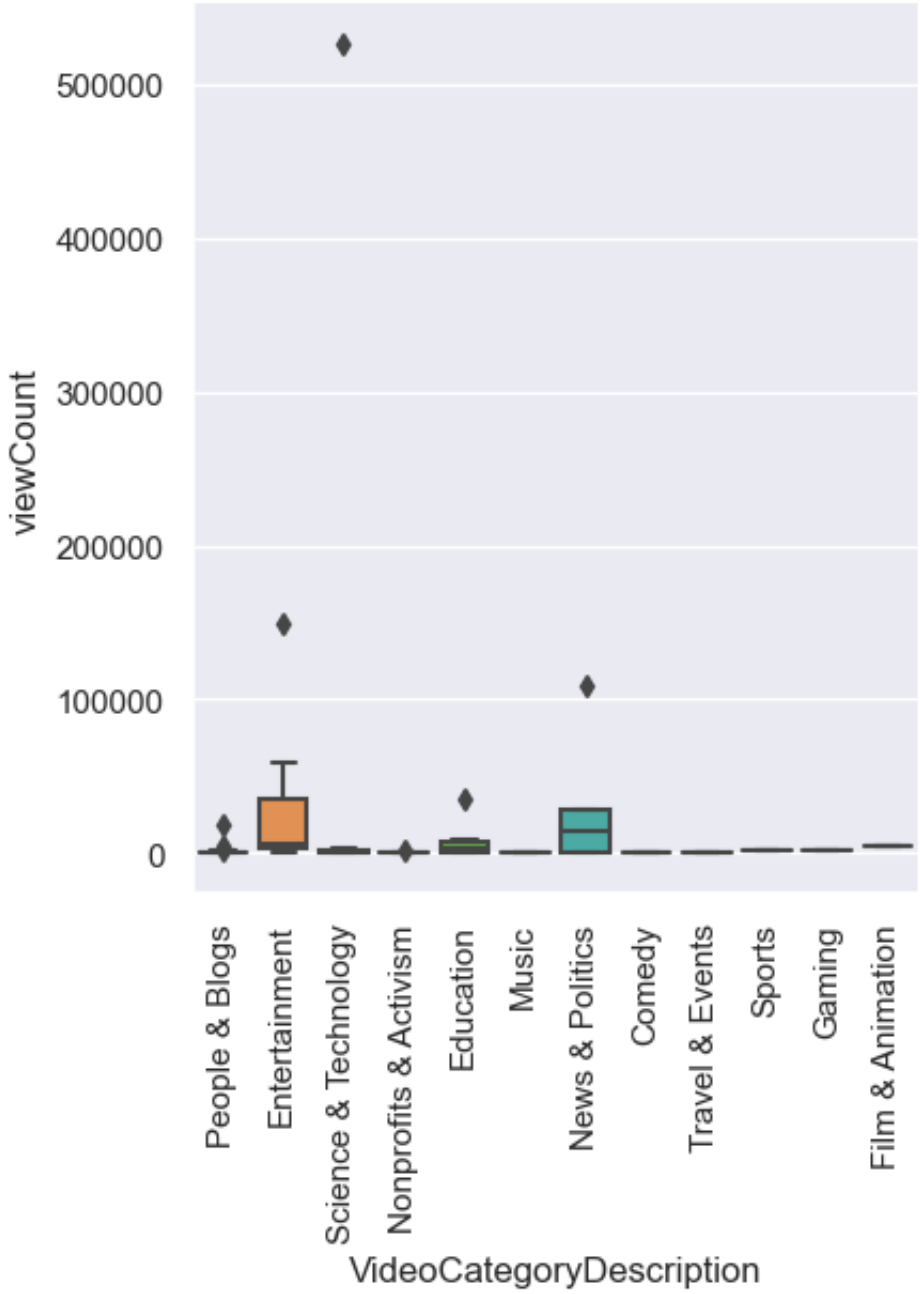


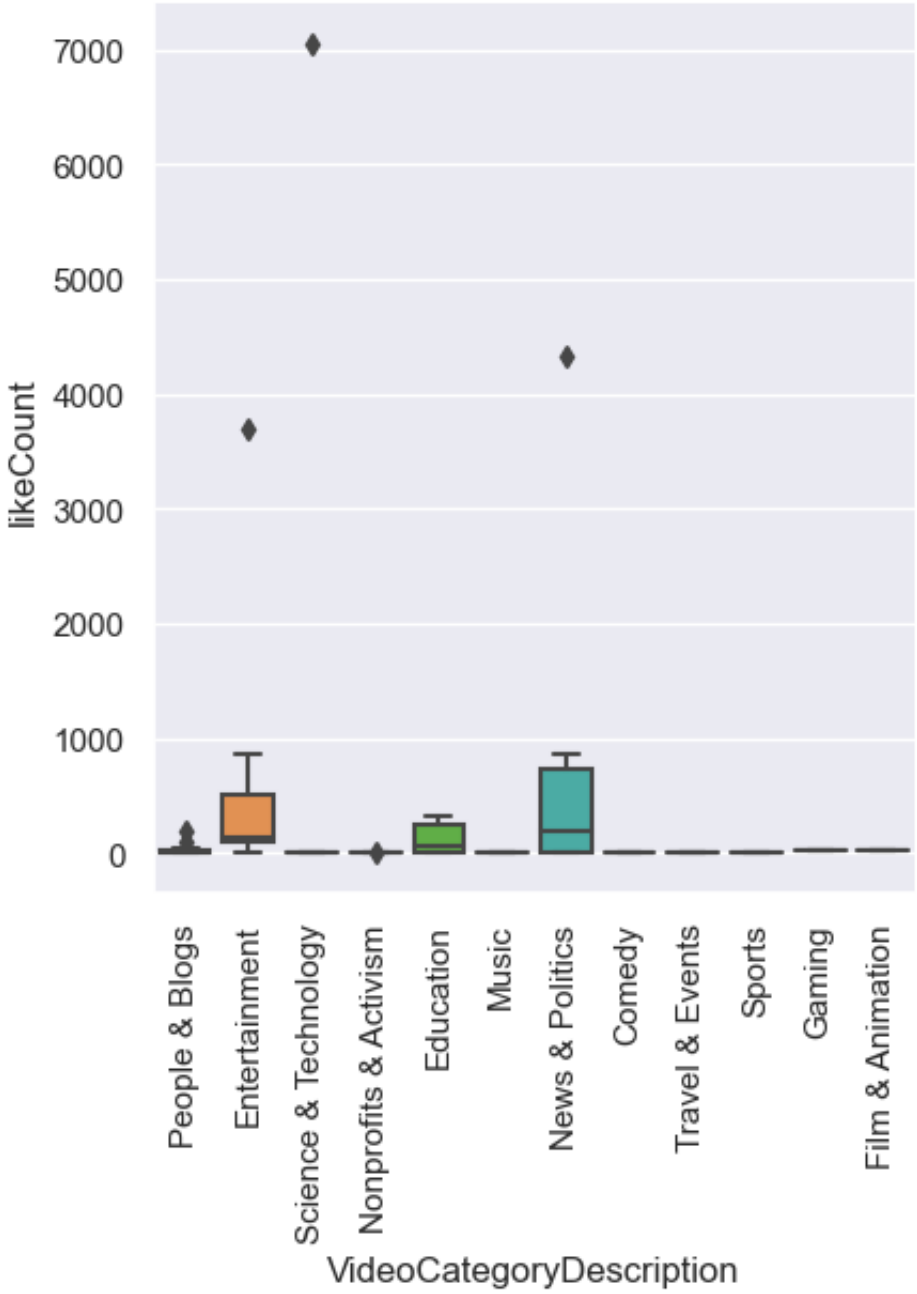
In [30]:
```
plt.figure(figsize= (12,6))
US_data['viewCount'].plot(kind='box', vert=False, color='blue',
                          figsize=(10,5))

plt.xlabel('Number of views')
plt.show()
# Boxplots with seaborn
sns.pairplot(data=US_data, hue="VideoCategoryDescription")
sns.catplot(x="VideoCategoryDescription", y="viewCount", kind="box", data
sns.catplot(x="VideoCategoryDescription", y="likeCount", kind="box", data
sns.catplot(x="VideoCategoryDescription", y="commentCount", kind="box", d
sns.displot(US_data, x="viewCount", hue="VideoCategoryDescription")
```

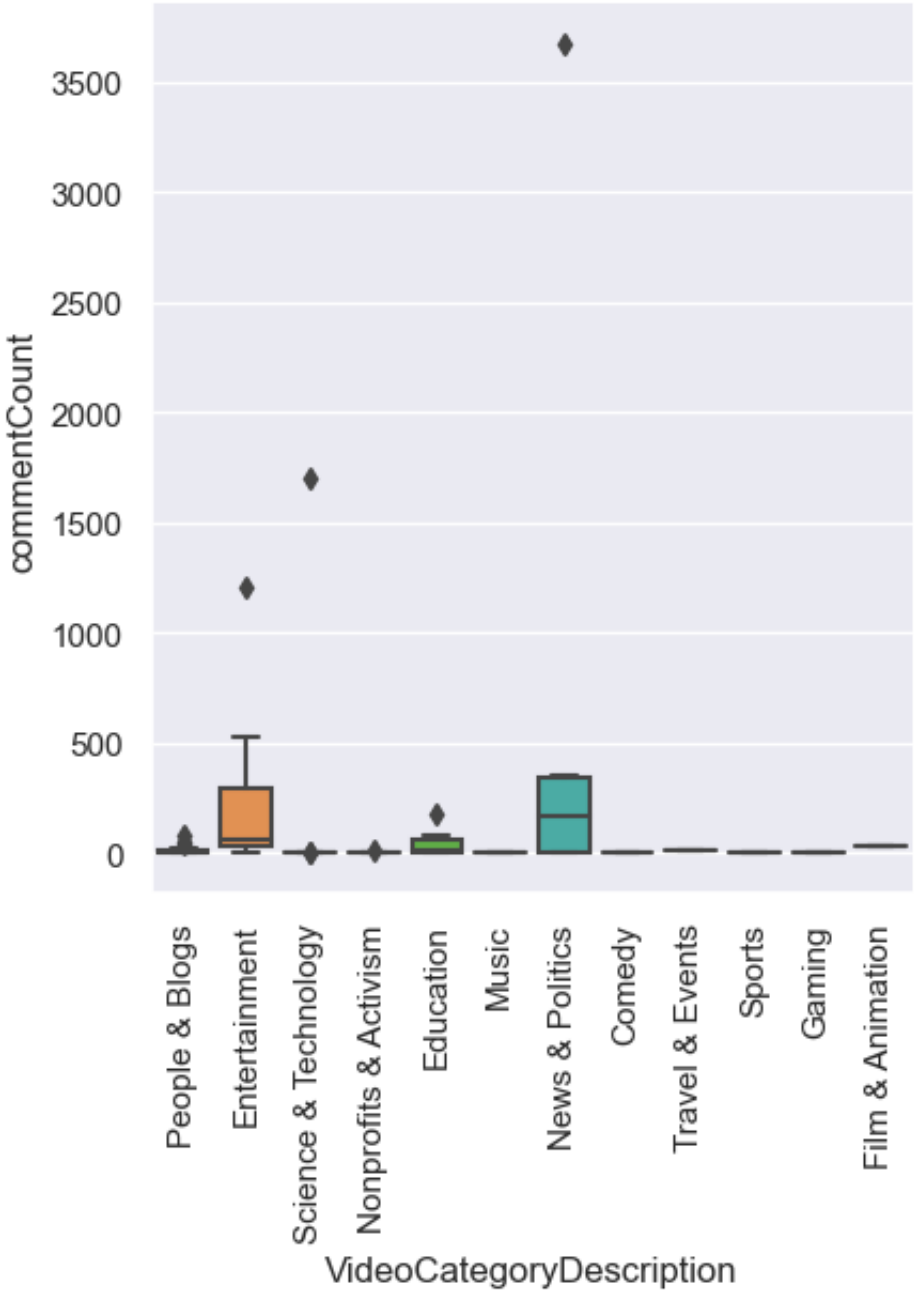


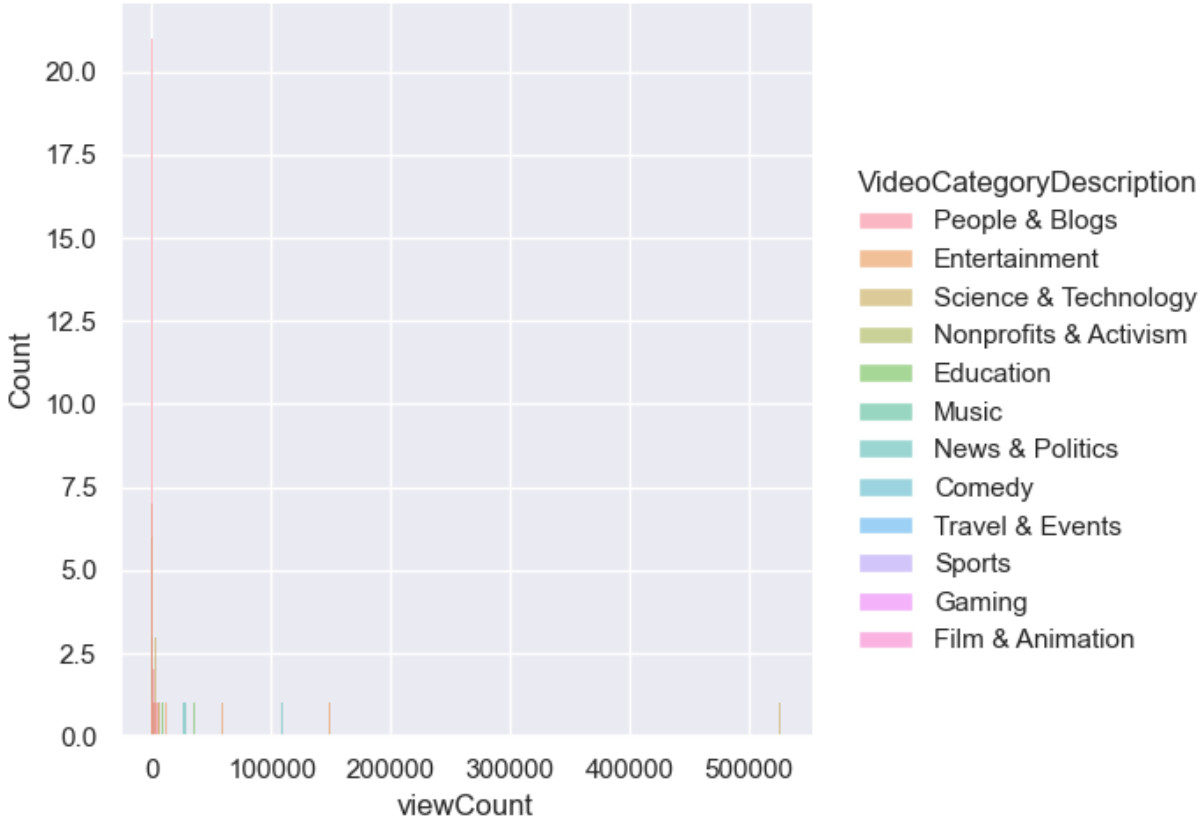
Out[30]: <seaborn.axisgrid.FacetGrid at 0x14f1fb9d0>




viewCount
500000
400000
300000
200000
100000
0
People & Blogs
Entertainment
Science & Technology
Nonprofits & Activism
Education
Music
News & Politics
Comedy
Travel & Events
Sports
Gaming
Film & Animation
VideoCategoryDescription


7000
6000
5000
4000
3000
2000
1000
0
likeCount
People & Blogs
Entertainment
Science & Technology
Nonprofits & Activism
Education
Music
News & Politics
Comedy
Travel & Events
Sports
Gaming
Film & Animation
VideoCategoryDescription


commentCount
3500
3000
2500
2000
1500
1000
500
0
People & Blogs
Entertainment
Science & Technology
Nonprofits & Activism
Education
Music
News & Politics
Comedy
Travel & Events
Sports
Gaming
Film & Animation
VideoCategoryDescription



In [31]:
```
import tensorflow as tf

import string
import nltk
from nltk.tokenize import RegexpTokenizer
from nltk.corpus import stopwords
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /Users/kel/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[31]: True

In [32]:
```
# subsetting the Nigeria title column
Ng_words = Ng_data[["title"]]
Ng_words.head()
```

Out[32]:

	title
35	SEE WAHALA O
26	Blaqbonez - Fake Nikes (Feat. Blxckie & Cheque...
40	PERFECT ASSISTANT- Watch Maurice Sam and Shine...
17	Aranda Ekun Part 3 - Latest Yoruba Movie 2022 ...
49	Police Internship

In [33]:
```
#Create an empty list

Ng_words_list = list()

indiv_lines = Ng_words['title'].values.tolist()

for line in indiv_lines:

    #create word tokens and remove punctuations in one go, r means regula
    rem_tok_punc = RegexpTokenizer(r'\w+')

    tokens = rem_tok_punc.tokenize(line)

    #convert all words to lower case
    words = [w.lower() for w in tokens]

    #Invoke all the english stopwords and removing duplicates using the s
    stop_word_list = set(stopwords.words("english"))

    #Remove stop words
    words = [w for w in words if not w in stop_word_list]

    #append words in the Ng_words_list
    Ng_words_list.append(words)

len(Ng_words_list)
```

Out[33]: 44

In [34]:
```
import gensim
```

In [35]:
```
Embedding_Dim = 100
#train word2vec model

model = gensim.models.Word2Vec(sentences = Ng_words_list, vector_size = E

#determining the vocabulary size

words = model.wv
print("The Vocabulary Size is ", len(words))
```

```
The Vocabulary Size is  220
```

In [36]:
```
model.save("word2vec.model")
```

In [37]:
```
vector = model.wv['wizkid']  # get numpy vector of a word
sims = model.wv.most_similar('wizkid', topn=10)  # get other similar word
sims
```

Out[37]:
```
[('oko', 0.2880608141422272),
 ('first', 0.23433852195739746),
 ('assistant', 0.2162284404039383),
 ('baby', 0.21406975388526917),
 ('rihanna', 0.20493315160274506),
 ('rock', 0.19968552887439728),
 ('lateef', 0.1950550377368927),
 ('sam', 0.1880389302968979),
 ('yoruba', 0.18208980560302734),
 ('tell', 0.1766613870859146)]
```

In [38]:
```
# subsetting the US title column

US_words = US_data[["channelTitle"]]
US_words
```

Out[38]:

	channelTitle
8	ellie pipkins
3	Alan Springwind
16	CABIvideo
72	ICRISAT Co
60	ICRISAT Co
...	...
33	Your Mate Tom
30	The Natural Way of Healing
7	Black Pigeon Speaks
29	PsychedSubstance
9	PsychedSubstance

72 rows × 1 columns

In [39]:
```
#Create an empty list

US_words_list = list()

indiv_lines = US_words['channelTitle'].values.tolist()

for line in indiv_lines:

    #create word tokens and remove punctuations in one go, r means regula
    rem_tok_punc = RegexpTokenizer(r'\w+')

    tokens = rem_tok_punc.tokenize(line)

    #convert all words to lower case
    words = [w.lower() for w in tokens]

    #Invoke all the english stopwords and removing duplicates using the s
    stop_word_list = set(stopwords.words("english"))

    #Remove stop words
    words = [w for w in words if not w in stop_word_list]

    #append words in the Ng_words_list
    US_words_list.append(words)

len(US_words_list)
```

Out[39]: 72

In [40]:
```
Embedding_Dim = 100
#train word2vec model

model1 = gensim.models.Word2Vec(sentences = US_words_list, vector_size =

#determining the vocabulary size

words_US = model1.wv
print("The Vocabulary Size is ", len(words_US))
```

```
The Vocabulary Size is  97
```

In [41]:
```
word_model = model1.save("word2vec.model1")
word_model
```

In [43]:
```
vector_US = model1.wv['ellie']  # get numpy vector of a word
sims_US = model1.wv.most_similar("ellie", topn=10)  # get other similar w
sims_US
```

Out[43]:
```
[('tom', 0.2506372332572937),
 ('co', 0.22956909239292145),
 ('improve', 0.22115959227085114),
 ('microlife', 0.2203265130519867),
 ('icrisat', 0.1781262308359146),
 ('way', 0.16201633214950562),
 ('tno', 0.1575070172548294),
 ('everything', 0.15164580941200256),
 ('area', 0.14308993518352509),
 ('nate', 0.14100120961666107)]
```

In [ ]:

In [ ]: