youtube-data

November 13, 2023

[129]:
```
# analyzing the youtube data
# download the dataset from kaggle https://www.kaggle.com/datasnaek/youtube-new
# performing data cleaning and EDA in the dataset
# importing the libraries
```

[130]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import json
import warnings
warnings.filterwarnings("ignore")
pd.set_option('display.max_columns', None)
```

[131]:
```
# reading the json file with pandas.read_json() method
```

[132]:
```
df=pd.read_json('archive (12)/CA_category_id.json')
df.head()
```

[132]:

	kind	etag	items
0	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	{'kind': 'youtube#videoCategory', 'etag': '"ld…
1	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	{'kind': 'youtube#videoCategory', 'etag': '"ld…
2	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	{'kind': 'youtube#videoCategory', 'etag': '"ld…
3	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	
4	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	

```
3  {'kind': 'youtube#videoCategory', 'etag': '"ld…
4  {'kind': 'youtube#videoCategory', 'etag': '"ld…
```

[133]:
```
# kind column has only one unique value
# it is not needed to extract
```

[134]:
```
df['kind'].unique()
```

[134]:
```
array(['youtube#videoCategoryListResponse'], dtype=object)
```

[135]:
```
# kind column has only one unique value
# it is not needed to extract
```

[136]:
```
df['etag'].unique()
```

[136]:
```
array(['"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2qTj13hkQZk"'],
      dtype=object)
```

[137]:
```
# column kind has the json data
# kind,etag,id,snippet,channerlid,title and assignable are the keys present in␣
 ↪the kind column
```

[138]:
```
row=df.iloc[:1,2]
for i in row:
    print(i)
```

```
{'kind': 'youtube#videoCategory', 'etag':
'"ld9biNPKjAjgjV7EZ4EKeEGrhao/Xy1mB4_yLrHy_BmKmPBggty2mZQ"', 'id': '1',
'snippet': {'channelId': 'UCBR8-60-B28hp2BmDPdntcQ', 'title': 'Film &
Animation', 'assignable': True}}
```

[139]:
```
# we also know that kind and etag have only one unique value
# so we are taking only id,channel id ,title,assignable
# we are saving all data of json in kind column in dictionary
```

[140]:
```
d={'id':[],'channelId':[],'title':[],'assignable':[]}
for item in df['items']:
    d['id'].append(item['id'])
    d['channelId'].append(item['snippet']['channelId'])
    d['title'].append(item['snippet']['title'])
    d['assignable'].append(item['snippet']['assignable'])
```

[141]:
```
# we are making a dataset with dictionay by pandas.DataFrame() method
```

[142]:
```
df2=pd.DataFrame(d)
df2.head()
```

[142]:

	id	channelId	title	assignable
0	1	UCBR8-60-B28hp2BmDPdntcQ	Film & Animation	True
1	2	UCBR8-60-B28hp2BmDPdntcQ	Autos & Vehicles	True
2	10	UCBR8-60-B28hp2BmDPdntcQ	Music	True
3	15	UCBR8-60-B28hp2BmDPdntcQ	Pets & Animals	True
4	17	UCBR8-60-B28hp2BmDPdntcQ	Sports	True

[143]:
```
# concatenating the columns of df1 and df2
# deleting the column items which has the json data
```

[144]:
```
df3=pd.concat([df,df2],axis=1)
CA_category=df3.drop(columns=['items'],axis=1)
CA_category.head()
```

[144]:

	kind	etag	id	channelId	title	assignable
0	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	1	UCBR8-60-B28hp2BmDPdntcQ	Film & Animation	True
1	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	2	UCBR8-60-B28hp2BmDPdntcQ	Autos & Vehicles	True
2	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	10	UCBR8-60-B28hp2BmDPdntcQ	Music	True
3	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	15	UCBR8-60-B28hp2BmDPdntcQ	Pets & Animals	True
4	youtube#videoCategoryListResponse	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	17	UCBR8-60-B28hp2BmDPdntcQ	Sports	True

[145]:
```
# inspecting the dataset with info() method
```

[146]:
```
CA_category.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   kind        31 non-null     object
 1   etag        31 non-null     object
 2   id          31 non-null     object
```

```
 3   channelId   31 non-null     object
 4   title       31 non-null     object
 5   assignable  31 non-null     bool
dtypes: bool(1), object(5)
memory usage: 1.4+ KB
```

[147]:
```
# column id has datatype "object" converting it into the "int32"
```

[148]:
```
CA_category['id']=CA_category['id'].astype('int32')
```

[149]:
```
# json file contains the video category data
# only 30 video categories are there
```

[150]:
```
CA_category.shape
```

[150]:
```
(31, 6)
```

[151]:
```
# dataset contains 10 json files and 10 csv files
# each json and csv for one country
# countries are USA, Great Britain, Germany, Canada, France, Russia, Mexico,␣
 ↪South Korea, India and Japan.
```

[152]:
```
import os

base_dir=os.getcwd()
base_dir+='\\archive (12)'
for file in os.listdir(base_dir):
    print(file)
```

```
.ipynb_checkpoints
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
```

[153]:
```
# performing the above operations in all the json files
# saving all the dataFrames in a list
```

[154]:
```
all_json_csv=[]
all_json_files=['archive (12)/CA_category_id.json',
                'archive (12)/DE_category_id.json',
                'archive (12)/FR_category_id.json',
                'archive (12)/CA_category_id.json',
                'archive (12)/GB_category_id.json',
                'archive (12)/IN_category_id.json',
                'archive (12)/JP_category_id.json',
                'archive (12)/KR_category_id.json',
                'archive (12)/MX_category_id.json',
                'archive (12)/RU_category_id.json',
                'archive (12)/US_category_id.json']
for file in all_json_files:
    #print(file.split('.'))


    df=pd.read_json(file)
    d={'id':[],'channelId':[],'title':[],'assignable':[]}
    for item in df['items']:
        d['id'].append(item['id'])
        d['channelId'].append(item['snippet']['channelId'])
        d['title'].append(item['snippet']['title'])
        d['assignable'].append(item['snippet']['assignable'])

    df2=pd.DataFrame(d)
    df3=pd.concat([df,df2],axis=1)
    category=df3.drop(columns=['items'],axis=1)

    category['id']=category['id'].astype('int32')
    all_json_csv.append(category)
```

[155]:
```
# its time to perform data cleaning on csv files
# json files contains the data of video category
# csv files contains the videos data
```

[156]:
```
CAvideos=pd.read_csv('archive (12)/CAvideos.csv')
CAvideos=CAvideos.rename(columns={'category_id':'id'})
CAvideos.head()
```

[156]:
```
       video_id trending_date  \
0  n1WpP7iowLc      17.14.11
```

```
1  0dBIkQ4Mz1M      17.14.11
2  5qpjK5DgCt4      17.14.11
3  d380meD0W0M      17.14.11
4  2Vv-BfVoq4g      17.14.11

                                                title channel_title  id  \
0        Eminem - Walk On Water (Audio) ft. Beyoncé    EminemVEVO  10
1                        PLUSH - Bad Unboxing Fan Mail     iDubbbzTV  23
2  Racist Superman | Rudy Mancuso, King Bach & Le…  Rudy Mancuso  23
3                           I Dare You: GOING BALD!?      nigahiga  24
4       Ed Sheeran - Perfect (Official Music Video)    Ed Sheeran  10

              publish_time  \
0  2017-11-10T17:00:03.000Z
1  2017-11-13T17:00:00.000Z
2  2017-11-12T19:05:24.000Z
3  2017-11-12T18:01:41.000Z
4  2017-11-09T11:04:14.000Z

                                                 tags     views    likes  \
0  Eminem|"Walk"|"On"|"Water"|"Aftermath/Shady/In…  17158579   787425
1  plush|"bad unboxing"|"unboxing"|"fan mail"|"id…   1014651   127794
2  racist superman|"rudy"|"mancuso"|"king"|"bach"…   3191434   146035
3  ryan|"higa"|"higatv"|"nigahiga"|"i dare you"|"…   2095828   132239
4  edsheeran|"ed sheeran"|"acoustic"|"live"|"cove…  33523622  1634130

   dislikes  comment_count                                thumbnail_link  \
0     43420         125882  https://i.ytimg.com/vi/n1WpP7iowLc/default.jpg
1      1688          13030  https://i.ytimg.com/vi/0dBIkQ4Mz1M/default.jpg
2      5339           8181  https://i.ytimg.com/vi/5qpjK5DgCt4/default.jpg
3      1989          17518  https://i.ytimg.com/vi/d380meD0W0M/default.jpg
4     21082          85067  https://i.ytimg.com/vi/2Vv-BfVoq4g/default.jpg

   comments_disabled  ratings_disabled  video_error_or_removed  \
0              False             False                   False
1              False             False                   False
2              False             False                   False
3              False             False                   False
4              False             False                   False

                                        description
0  Eminem's new track Walk on Water ft. Beyoncé i…
1  STill got a lot of packages. Probably will las…
2  WATCH MY PREVIOUS VIDEO  \n\nSUBSCRIBE  http…
3  I know it's been a while since we did this sho…
4   : https://ad.gt/yt-perfect\n : https://atlant…
```

[157]:
```
# inspecting the data
```

[158]:
```
CAvideos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40881 entries, 0 to 40880
Data columns (total 16 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   video_id                40881 non-null  object
 1   trending_date           40881 non-null  object
 2   title                   40881 non-null  object
 3   channel_title           40881 non-null  object
 4   id                      40881 non-null  int64 
 5   publish_time            40881 non-null  object
 6   tags                    40881 non-null  object
 7   views                   40881 non-null  int64 
 8   likes                   40881 non-null  int64 
 9   dislikes                40881 non-null  int64 
 10  comment_count           40881 non-null  int64 
 11  thumbnail_link          40881 non-null  object
 12  comments_disabled       40881 non-null  bool  
 13  ratings_disabled        40881 non-null  bool  
 14  video_error_or_removed  40881 non-null  bool  
 15  description             39585 non-null  object
dtypes: bool(3), int64(5), object(8)
memory usage: 4.2+ MB
```

[159]:
```
# trending date column has the object type data
```

[160]:
```
CAvideos['trending_date']
```

[160]:
```
0        17.14.11
1        17.14.11
2        17.14.11
3        17.14.11
4        17.14.11
           ...
40876    18.14.06
40877    18.14.06
40878    18.14.06
40879    18.14.06
40880    18.14.06
Name: trending_date, Length: 40881, dtype: object
```

[161]:
```
# converting the trending date column to "date" type
```

```
# the year is in the form of YY format,converting it into YYYY format by addign␣
 ↪2000 to it
```

[162]:
```
date=CAvideos['trending_date'].str.split('.')
d={'trending_date':[]}
for i in date:
    d['trending_date'].append(i[1]+'-'+i[2]+'-'+str(int(i[0])+2000))
date=pd.DataFrame(d)
CAvideos=CAvideos.drop(columns='trending_date',axis=1)
```

[163]:
```
# concatenating the trending date to the dataset
```

[164]:
```
CAvideos=pd.concat([CAvideos,date],axis=1)
```

[165]:
```
# converting the trending date column to the "date type" using pd.to_datetime()
```

[166]:
```
CAvideos['trending_date']=pd.
 ↪to_datetime(CAvideos['trending_date'],format='%d-%m-%Y')
```

[167]:
```
CAvideos['trending_date']
```

[167]:
```
0       2017-11-14
1       2017-11-14
2       2017-11-14
3       2017-11-14
4       2017-11-14
           …
40876   2018-06-14
40877   2018-06-14
40878   2018-06-14
40879   2018-06-14
40880   2018-06-14
Name: trending_date, Length: 40881, dtype: datetime64[ns]
```

[168]:
```
# inspecting the column publish time
# it contains the publish time and date ,seperating it into date and time
```

[169]:
```
CAvideos['publish_time']=pd.to_datetime(CAvideos['publish_time'])

CAvideos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40881 entries, 0 to 40880
Data columns (total 16 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   video_id                40881 non-null  object
```

```
 1   title                   40881 non-null  object
 2   channel_title           40881 non-null  object
 3   id                      40881 non-null  int64
 4   publish_time            40881 non-null  datetime64[ns, UTC]
 5   tags                    40881 non-null  object
 6   views                   40881 non-null  int64
 7   likes                   40881 non-null  int64
 8   dislikes                40881 non-null  int64
 9   comment_count           40881 non-null  int64
 10  thumbnail_link          40881 non-null  object
 11  comments_disabled       40881 non-null  bool
 12  ratings_disabled        40881 non-null  bool
 13  video_error_or_removed  40881 non-null  bool
 14  description             39585 non-null  object
 15  trending_date           40881 non-null  datetime64[ns]
dtypes: bool(3), datetime64[ns, UTC](1), datetime64[ns](1), int64(5), object(6)
memory usage: 4.2+ MB
```

[170]:
```
# appending the published date ,extracted from publish time column
```

[171]:
```
CAvideos['published_date']=pd.
 ↪to_datetime(CAvideos['publish_time'],format='%d-%m-%Y')
```

[172]:
```
# publish time has the date and time together converting it into time
```

[173]:
```
CAvideos['published_time']=CAvideos['publish_time'].dt.time
#CAvideos['published_time']=CAvideos['published_time'].dt.strftime('%H:%M:%S')
```

[174]:
```
# checking for duplicted row in dataset
```

[175]:
```
CAvideos.duplicated(subset=['video_id']).sum()
```

[175]:
```
16454
```

[176]:
```
# removing the duplicated rows
```

[177]:
```
CAvideos=CAvideos.drop_duplicates('video_id',keep='first')
CAvideos.shape
```

[177]:
```
(24427, 18)
```

[178]:
```
# renaming the column "category id" to "id"
```

[179]:
```
CAvideos=CAvideos.rename(columns={'category_id':'id'})
```

[180]:
```
# performing the all above operations in the all csv files
# printing the sample dataset
```

[181]:
```
all_csv=[]
all_csv_files=['archive (12)/CAvideos.csv',
               'archive (12)/DEvideos.csv',
               'archive (12)/FRvideos.csv',
               'archive (12)/GBvideos.csv',
               'archive (12)/JPvideos.csv',
               'archive (12)/INvideos.csv',
               'archive (12)/KRvideos.csv',
               'archive (12)/MXvideos.csv',
               'archive (12)/RUvideos.csv',
               'archive (12)/USvideos.csv',]
for csv in all_csv_files:
    try:
        videos=pd.read_csv(csv,encoding='utf-8')
        #print('a')
    except:
        videos=pd.read_csv(csv,encoding="ISO-8859-1")
        #print('b')
    print(videos.shape,end=' ')
    videos['country']=csv.split('/')[1][:2]
    date=videos['trending_date'].str.split('.')
    d={'trending_date':[]}
    for i in date:
        d['trending_date'].append(i[1]+'-'+i[2]+'-'+str(int(i[0])+2000))
    date=pd.DataFrame(d)
    videos=videos.drop(columns='trending_date',axis=1)

    videos=pd.concat([videos,date],axis=1)
    videos['trending_date']=pd.
 ↪to_datetime(date['trending_date'],format='%d-%m-%Y')

    videos=videos.drop_duplicates('video_id',keep='first')
    videos=videos.rename(columns={'category_id':'id'})
    videos['publish_time']=pd.to_datetime(videos['publish_time'])
    videos['published_date']=videos['publish_time'].dt.date
    videos['published_time']=videos['publish_time'].dt.time
    videos=videos.drop(columns='publish_time',axis=1)
    all_csv.append(videos)
videos.head()
```

```
(40881, 16) (40840, 16) (40724, 16) (38916, 16) (20523, 16) (37352, 16) (34567,
16) (40451, 16) (40739, 16) (40949, 16)
```

[181]:

	video_id	title
0	2kyS6SvSYSE	WE WANT TO TALK ABOUT OUR MARRIAGE
1	1ZAPwfrtAFY	The Trump Presidency: Last Week Tonight with J…
2	5qpjK5DgCt4	Racist Superman | Rudy Mancuso, King Bach & Le…

```
3  puqaWrEC7tY                    Nickelback Lyrics: Real or Fake?
4  d380meD0W0M                           I Dare You: GOING BALD!?

           channel_title  id  \
0            CaseyNeistat  22
1         LastWeekTonight  24
2            Rudy Mancuso  23
3  Good Mythical Morning  24
4                nigahiga  24

                                                tags    views   likes  \
0                                  SHANtell martin   748374   57527
1  last week tonight trump presidency|"last week …  2418783   97185
2  racist superman|"rudy"|"mancuso"|"king"|"bach"…  3191434  146033
3  rhett and link|"gmm"|"good mythical morning"|"…   343168   10172
4  ryan|"higa"|"higatv"|"nigahiga"|"i dare you"|"…  2095731  132235

   dislikes  comment_count                                thumbnail_link  \
0      2966          15954  https://i.ytimg.com/vi/2kyS6SvSYSE/default.jpg
1      6146          12703  https://i.ytimg.com/vi/1ZAPwfrtAFY/default.jpg
2      5339           8181  https://i.ytimg.com/vi/5qpjK5DgCt4/default.jpg
3       666           2146  https://i.ytimg.com/vi/puqaWrEC7tY/default.jpg
4      1989          17518  https://i.ytimg.com/vi/d380meD0W0M/default.jpg

   comments_disabled  ratings_disabled  video_error_or_removed  \
0              False             False                   False
1              False             False                   False
2              False             False                   False
3              False             False                   False
4              False             False                   False

                                        description country trending_date  \
0  SHANTELL'S CHANNEL - https://www.youtube.com/s…      US    2017-11-14
1  One year after the presidential election, John…      US    2017-11-14
2  WATCH MY PREVIOUS VIDEO   \n\nSUBSCRIBE   http…      US    2017-11-14
3  Today we find out if Link is a Nickelback amat…      US    2017-11-14
4  I know it's been a while since we did this sho…      US    2017-11-14

  published_date published_time
0     2017-11-13       17:13:01
1     2017-11-13       07:30:00
2     2017-11-12       19:05:24
3     2017-11-13       11:00:04
4     2017-11-12       18:01:41
```

[182]:
```
# cheking the country column
```

[183]:
```
videos['country'].unique()
```

[183]:
```
array(['US'], dtype=object)
```

[184]:
```
videos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 6351 entries, 0 to 40766
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   video_id                6351 non-null   object
 1   title                   6351 non-null   object
 2   channel_title           6351 non-null   object
 3   id                      6351 non-null   int64
 4   tags                    6351 non-null   object
 5   views                   6351 non-null   int64
 6   likes                   6351 non-null   int64
 7   dislikes                6351 non-null   int64
 8   comment_count           6351 non-null   int64
 9   thumbnail_link          6351 non-null   object
 10  comments_disabled       6351 non-null   bool
 11  ratings_disabled        6351 non-null   bool
 12  video_error_or_removed  6351 non-null   bool
 13  description             6249 non-null   object
 14  country                 6351 non-null   object
 15  trending_date           6351 non-null   datetime64[ns]
 16  published_date          6351 non-null   object
 17  published_time          6351 non-null   object
dtypes: bool(3), datetime64[ns](1), int64(5), object(9)
memory usage: 812.5+ KB
```

[185]:
```
# merging the json files dataset and csv files datasets by country wise
# merging the datasets based on the "column" and performing "inner" join
```

[186]:
```
merged=[]

for json,csv in zip(all_json_csv,all_csv):
    df=pd.merge(json,csv,on='id',how='inner')
    print(csv.isna().sum().sum(),end=' ')
    print(df.isna().sum().sum())
    merged.append(df)
```

```
973 971
1305 1296
2479 2475
58 55
1342 1342
```

```
294 294
1602 1601
3627 3534
2166 2069
102 101
```

[187]:
```
# checking the no of rows in each dataset
```

[188]:
```
for df in merged:
    print(df.shape)
```

```
(24377, 23)
(29425, 23)
(30477, 23)
(3266, 23)
(12899, 23)
(16238, 23)
(15757, 23)
(33302, 23)
(32968, 23)
(6337, 23)
```

[189]:
```
# concatenating all countries dataset column wise
# printing the shape of the dataset
```

[190]:
```
merged_df=pd.concat([df for df in merged],axis=0)
merged_df.shape
```

[190]:
```
(205046, 23)
```

[191]:
```
# columns of the dataset
```

[192]:
```
merged_df.columns
```

[192]:
```
Index(['kind', 'etag', 'id', 'channelId', 'title_x', 'assignable', 'video_id',
       'title_y', 'channel_title', 'tags', 'views', 'likes', 'dislikes',
       'comment_count', 'thumbnail_link', 'comments_disabled',
       'ratings_disabled', 'video_error_or_removed', 'description', 'country',
       'trending_date', 'published_date', 'published_time'],
      dtype='object')
```

[193]:
```
# inspecting the dataset
# the column "description" has 13738 null values
```

[194]:
```
205046-191308
```

[194]:
```
13738
```

[195]:
```
merged_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 205046 entries, 0 to 6336
Data columns (total 23 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   kind                    205046 non-null  object
 1   etag                    205046 non-null  object
 2   id                      205046 non-null  int32
 3   channelId               205046 non-null  object
 4   title_x                 205046 non-null  object
 5   assignable              205046 non-null  bool
 6   video_id                205046 non-null  object
 7   title_y                 205046 non-null  object
 8   channel_title           205046 non-null  object
 9   tags                    205046 non-null  object
 10  views                   205046 non-null  int64
 11  likes                   205046 non-null  int64
 12  dislikes                205046 non-null  int64
 13  comment_count           205046 non-null  int64
 14  thumbnail_link          205046 non-null  object
 15  comments_disabled       205046 non-null  bool
 16  ratings_disabled        205046 non-null  bool
 17  video_error_or_removed  205046 non-null  bool
 18  description             191308 non-null  object
 19  country                 205046 non-null  object
 20  trending_date           205046 non-null  datetime64[ns]
 21  published_date          205046 non-null  object
 22  published_time          205046 non-null  object
dtypes: bool(4), datetime64[ns](1), int32(1), int64(4), object(13)
memory usage: 31.3+ MB
```

[196]:
```
# checking the kind column
```

[197]:
```
merged_df['kind'].unique()
```

[197]:
```
array(['youtube#videoCategoryListResponse'], dtype=object)
```

[198]:
```
# column kind has only one unique value
# so removing it from the dataset
```

[199]:
```
merged_df=merged_df.drop(columns='kind',axis=1)
merged_df.head()
```

[199]:

	etag	id	\
0	"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…	1	

```
1  "ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…   1
2  "ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…   1
3  "ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…   1
4  "ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2…   1

                 channelId           title_x  assignable     video_id  \
0  UCBR8-60-B28hp2BmDPdntcQ  Film & Animation        True  FyZMnhUtLfE
1  UCBR8-60-B28hp2BmDPdntcQ  Film & Animation        True  fy-CuCzaPp8
2  UCBR8-60-B28hp2BmDPdntcQ  Film & Animation        True  kzwfHumJyYc
3  UCBR8-60-B28hp2BmDPdntcQ  Film & Animation        True  -Toujzk3ysk
4  UCBR8-60-B28hp2BmDPdntcQ  Film & Animation        True  kOnQocd799Y

                                          title_y  \
0          | Game Of Hunting 12 TV
1  Rooster Teeth Animated Adventures - Drunk Baby…
2  Sharry Mann: Cute Munda ( Song Teaser) | Parmi…
3  Gerrymandering, Conservative Comedians, Bobby …
4  Everything Wrong With Baby Driver In 14 Minute…

             channel_title  \
0
1             Rooster Teeth
2           Lokdhun Punjabi
3  Real Time with Bill Maher
4                CinemaSins

                                                 tags    views  likes  \
0   |"   "|" "|" "|" "|" "|" "|" "|" "|" "…   158815    218
1  Rooster Teeth|"RT"|"animation"|"television"|"f…   308568  19541
2  sharry mann|"sharry mann new song"|"sharry man…  1096327  33966
3  DNC Chairwoman Donna Brazile|"Hacks"|"The Insi…   458964   4903
4  baby driver|"edgar wright"|"everything wrong w…  2736733  58967

   dislikes  comment_count                                thumbnail_link  \
0        30            186  https://i.ytimg.com/vi/FyZMnhUtLfE/default.jpg
1        70            495  https://i.ytimg.com/vi/fy-CuCzaPp8/default.jpg
2       798            882  https://i.ytimg.com/vi/kzwfHumJyYc/default.jpg
3       392           1289  https://i.ytimg.com/vi/-Toujzk3ysk/default.jpg
4      3492           7915  https://i.ytimg.com/vi/kOnQocd799Y/default.jpg

   comments_disabled  ratings_disabled  video_error_or_removed  \
0              False             False                   False
1              False             False                   False
2              False             False                   False
3              False             False                   False
4              False             False                   False
```

```
                                                 description country trending_date  \
0  Thanks for watching the drama! Help more peopl…      CA    2017-11-14
1  Miles gets stuck at work one night watching ov…      CA    2017-11-14
2  Presenting Sharry Mann latest Punjabi Song  Cu…      CA    2017-11-14
3  Subscribe to the Real Time YouTube: http://its…      CA    2017-11-14
4  Here's a movie that most people, including us,…      CA    2017-11-14

  published_date published_time
0     2017-11-12       16:00:01
1     2017-11-13       14:00:03
2     2017-11-12       12:20:39
3     2017-11-11       05:30:08
4     2017-11-09       17:14:11
```

[200]:
```
# checking the etag column
```

[201]:
```
merged_df['etag'].unique()
```

[201]:
```
array(['"ld9biNPKjAjgjV7EZ4EKeEGrhao/1v2mrzYSYG6onNLt2qTj13hkQZk"',
       '"m2yskBQFythfE4irbTIeOgYYfBU/1v2mrzYSYG6onNLt2qTj13hkQZk"',
       '"XI7nbFXulYBIpL0ayR_gDh3eu1k/1v2mrzYSYG6onNLt2qTj13hkQZk"'],
      dtype=object)
```

[202]:
```
# column etag has only one unique value
# so removing it from the dataset
```

[203]:
```
merged_df=merged_df.drop(columns='etag',axis=1)
```

[204]:
```
merged_df.shape
```

[204]:
```
(205046, 21)
```

[205]:
```
# checking for duplicate rows
```

[206]:
```
merged_df.duplicated(subset=['video_id']).sum()
```

[206]:
```
22747
```

[207]:
```
# removing all duplicated  rows
```

[208]:
```
merged_df=merged_df.drop_duplicates('video_id',keep='first')
merged_df.shape
```

[208]:
```
(182299, 21)
```

[209]:
```
#checking for null values
# column description has 12967 null values
```

[210]:
```
merged_df.isna().sum()
```

[210]:
```
id                          0
channelId                   0
title_x                     0
assignable                  0
video_id                    0
title_y                     0
channel_title               0
tags                        0
views                       0
likes                       0
dislikes                    0
comment_count               0
thumbnail_link              0
comments_disabled           0
ratings_disabled            0
video_error_or_removed      0
description                 12967
country                     0
trending_date               0
published_date              0
published_time              0
dtype: int64
```

[211]:
```
# removing the column description
```

[212]:
```
merged_df=merged_df.drop(columns='description',axis=1)
```

[213]:
```
# renaming the column title_x to category
```

[214]:
```
merged_df=merged_df.rename(columns={'title_x':'category'})
merged_df.columns
```

[214]:
```
Index(['id', 'channelId', 'category', 'assignable', 'video_id', 'title_y',
       'channel_title', 'tags', 'views', 'likes', 'dislikes', 'comment_count',
       'thumbnail_link', 'comments_disabled', 'ratings_disabled',
       'video_error_or_removed', 'country', 'trending_date', 'published_date',
       'published_time'],
      dtype='object')
```

[215]:
```
# Ratio of likes-dislikes in different categories
# to calculate that divide sum of likes with sum of dislikes by category wise
```

[216]:
```
likes=merged_df.groupby('category')['likes'].sum()
dislikes=merged_df.groupby('category')['dislikes'].agg('sum')
likes_dislikes_ratio=likes/dislikes
```

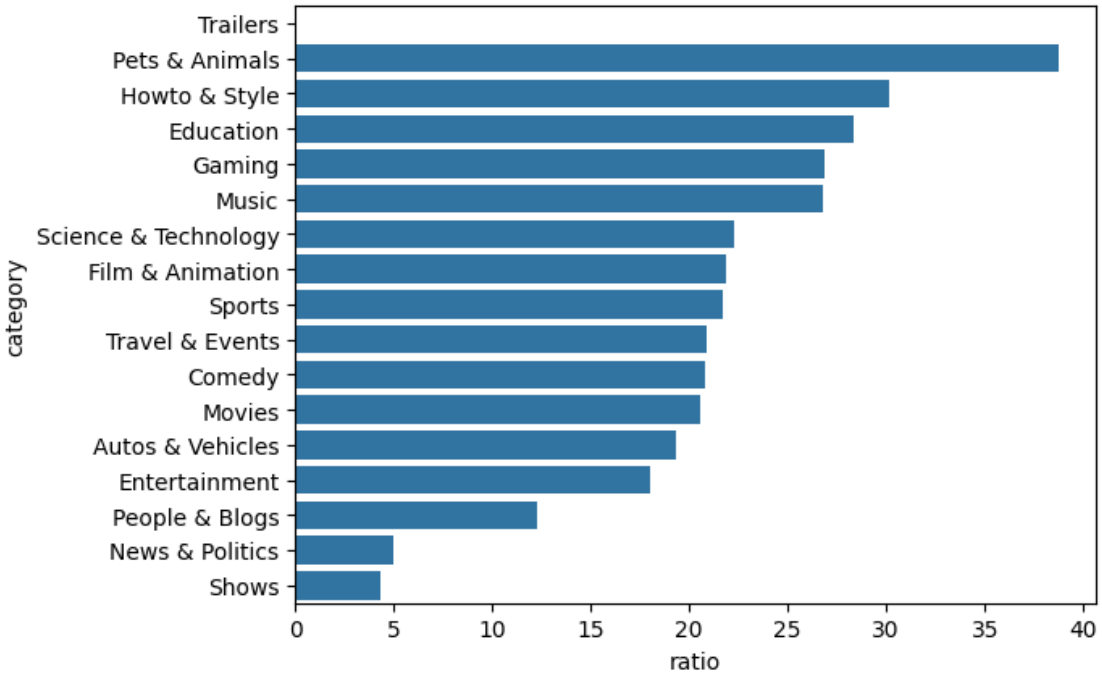
```
likes_dislikes_ratio=likes_dislikes_ratio.sort_values(ascending=False)
likes_dislikes_ratio=pd.DataFrame(likes_dislikes_ratio,columns=['ratio'])
likes_dislikes_ratio
```

[216]:

category	ratio
Trailers	inf
Pets & Animals	38.716438
Howto & Style	30.117611
Education	28.369119
Gaming	26.891060
Music	26.785348
Science & Technology	22.320670
Film & Animation	21.850140
Sports	21.739690
Travel & Events	20.921760
Comedy	20.827985
Movies	20.570413
Autos & Vehicles	19.335142
Entertainment	18.056741
People & Blogs	12.298057
News & Politics	4.993195
Shows	4.312365

[217]:
```
sns.barplot(x='ratio',y='category',data=likes_dislikes_ratio)
```

[217]: <Axes: xlabel='ratio', ylabel='category'>



[218]:
```
'''observation :
pets and Animals, howto and style, education,gaming, music
are the top 5 categories which have highest like-dislikes ratio'''
```

[218]:
```
'observation :\npets and Animals, howto and style, education,gaming, music \nare
the top 5 categories which have highest like-dislikes ratio'
```

[219]:
```
#Users like videos from which category more?
# to see that group all rows by categorywise and sum all the likes
```

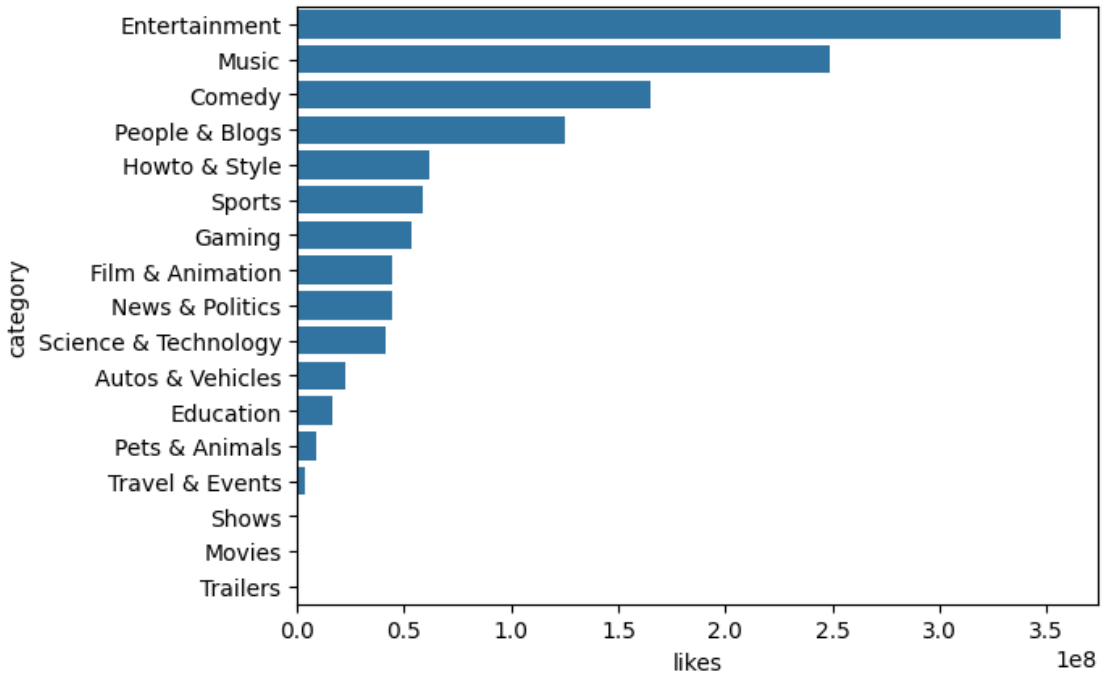
[220]:
```
likes=merged_df.groupby('category')['likes'].agg('sum').
 ↪sort_values(ascending=False)
likes=pd.DataFrame(likes)
likes
```

[220]:

category	likes
Entertainment	356340961
Music	248765865
Comedy	165274771
People & Blogs	124930558
Howto & Style	61492873
Sports	58786164
Gaming	53135955
Film & Animation	44686683
News & Politics	44092067
Science & Technology	41552718
Autos & Vehicles	22394889
Education	16749752
Pets & Animals	9113114
Travel & Events	3421733
Shows	986018
Movies	14461
Trailers	57

[221]:
```
sns.barplot(x='likes',y='category',data=likes)
```

[221]:
```
<Axes: xlabel='likes', ylabel='category'>
```



[222]:
```
'''
observation:
Entertainment ,music,comedy , blogs,sports are the top 5 categories which has␣
 ↪more no of likes
'''
```

[222]:
```
'\nobservation:\nEntertainment ,music,comedy , blogs,sports are the top 5
categories which has more no of likes\n'
```

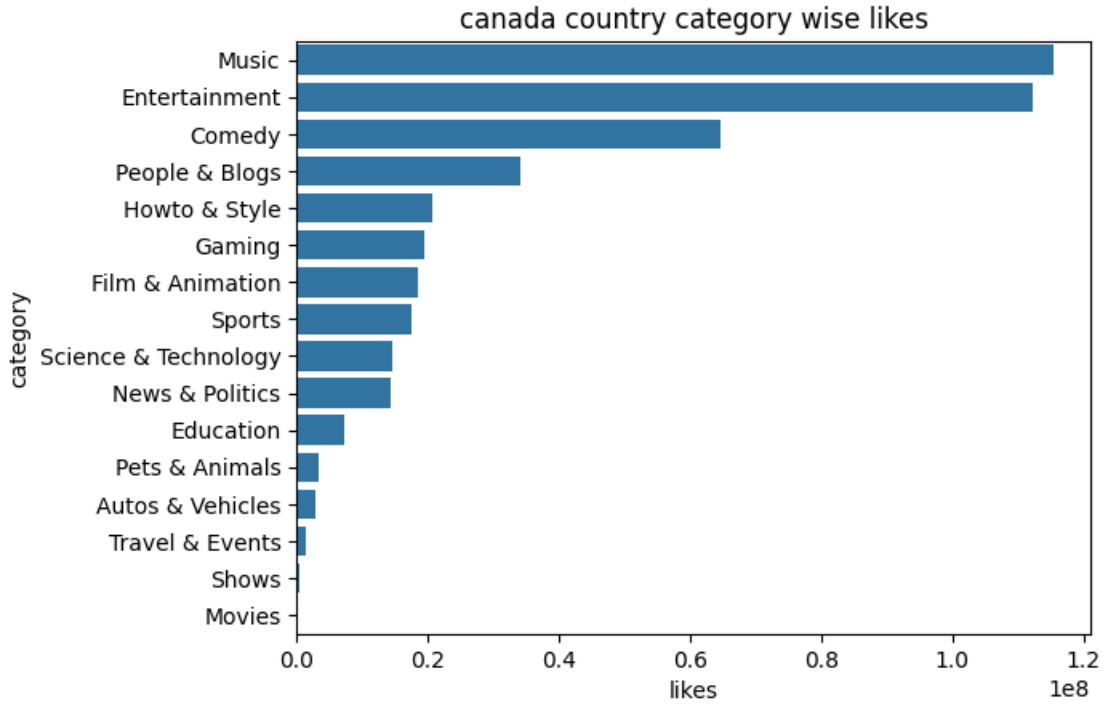
[223]:
```
# replacing the country column according to the country map dictionary
```

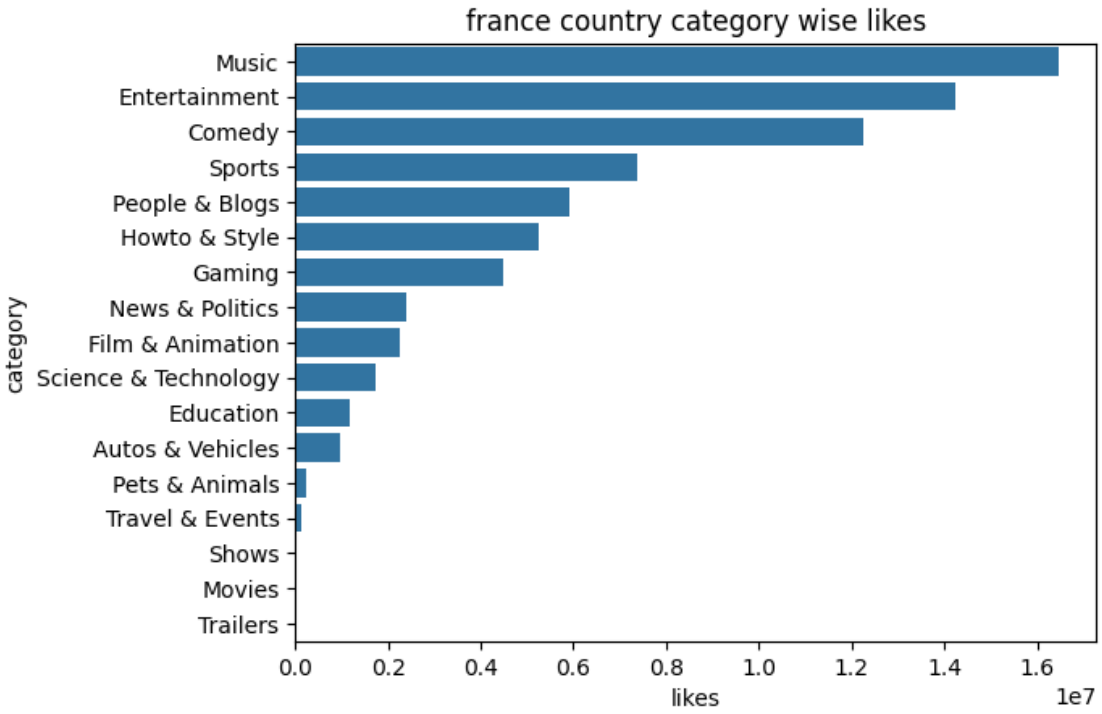
[224]:
```
country_map={'CA':'canada','FR':'france','GB':'britan','JP':'japan','IN':
 ↪'india','KR':'korea','MX':'mexico','RU':'Russia','US':'america','DE':
 ↪'germany'}
merged_df['country']=merged_df['country'].replace(country_map)
countries=merged_df['country'].unique()
countries
```

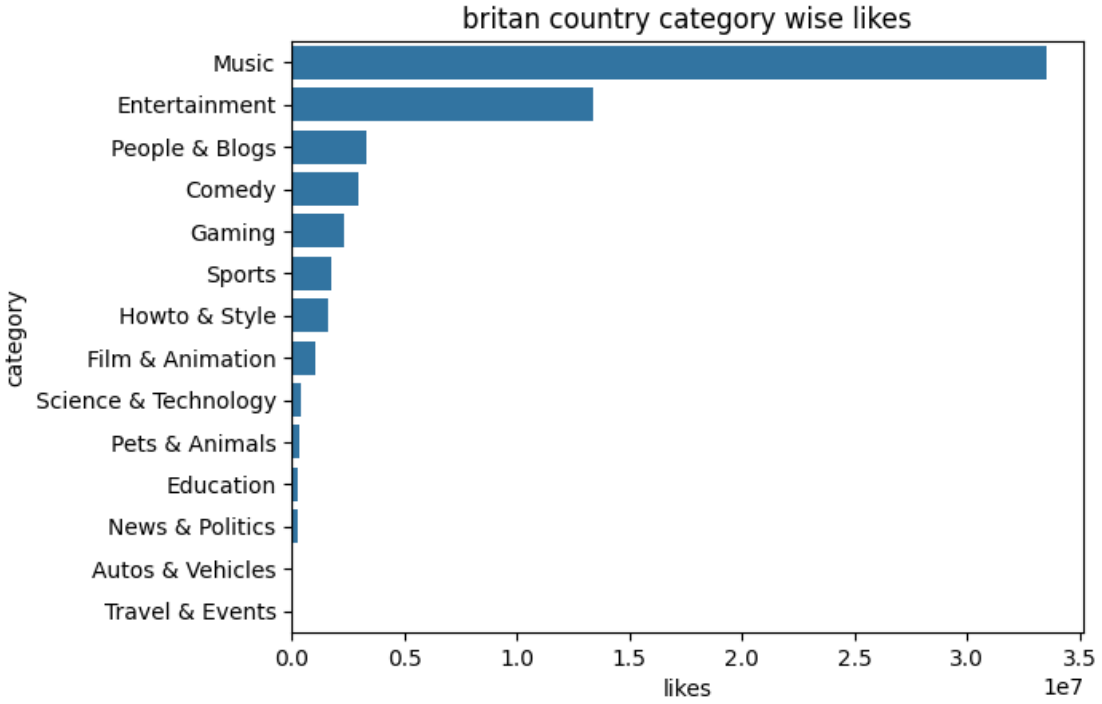
[224]:
```
array(['canada', 'germany', 'france', 'britan', 'japan', 'india', 'korea',
       'mexico', 'Russia', 'america'], dtype=object)
```

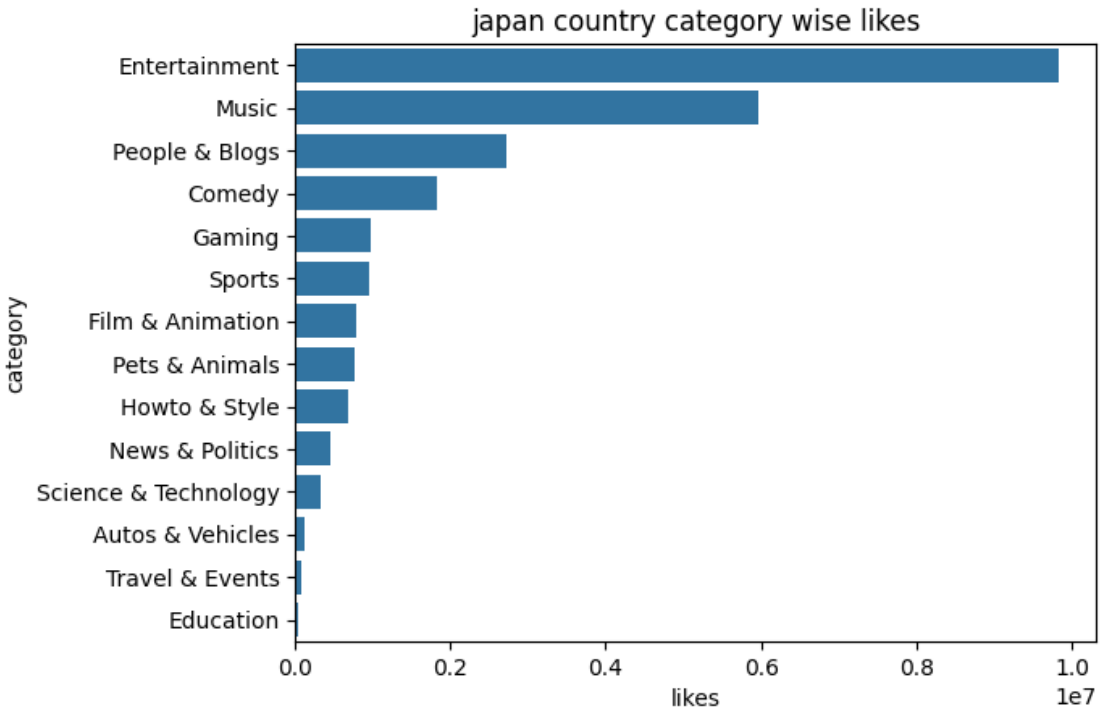
[225]:
```
#Users like videos from which category more country wise
```

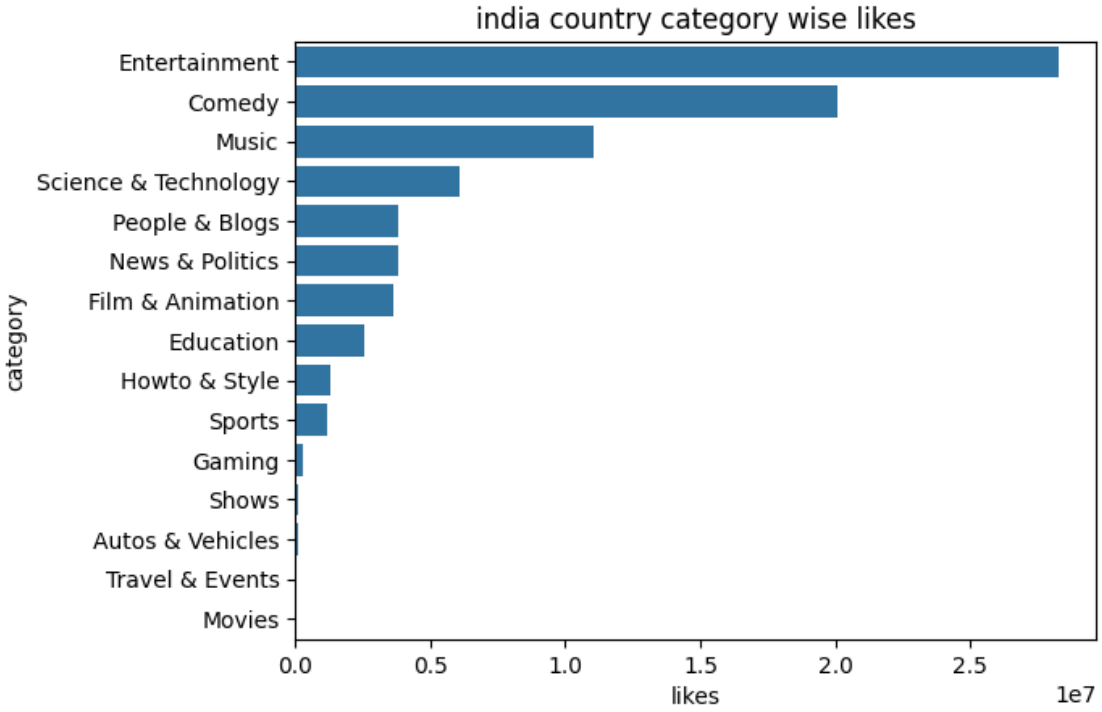
[226]:
```
plots=[]
for country in country_map:
    likes_by_country=merged_df[merged_df['country']==country_map[country]].
 ↪groupby('category')['likes'].sum().sort_values(ascending=False)
    df=pd.DataFrame(likes_by_country)
    sns.barplot(y='category',x='likes',data=df)
    plt.title('{} country category wise likes'.format(country_map[country]))
    plt.show()
```

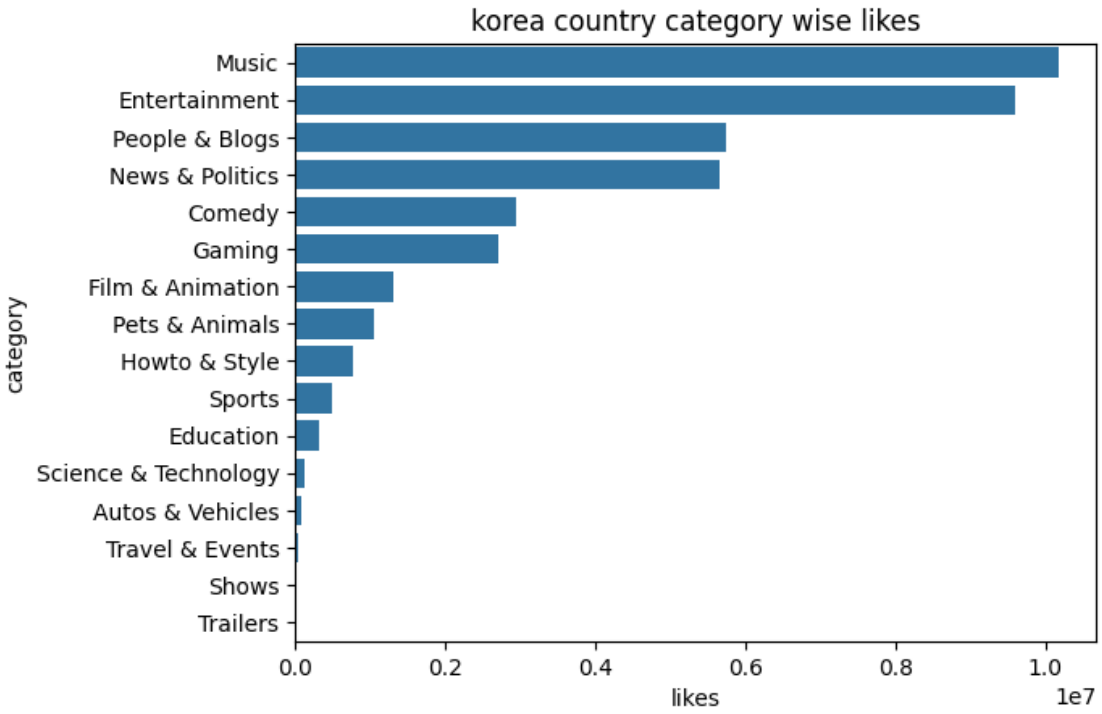


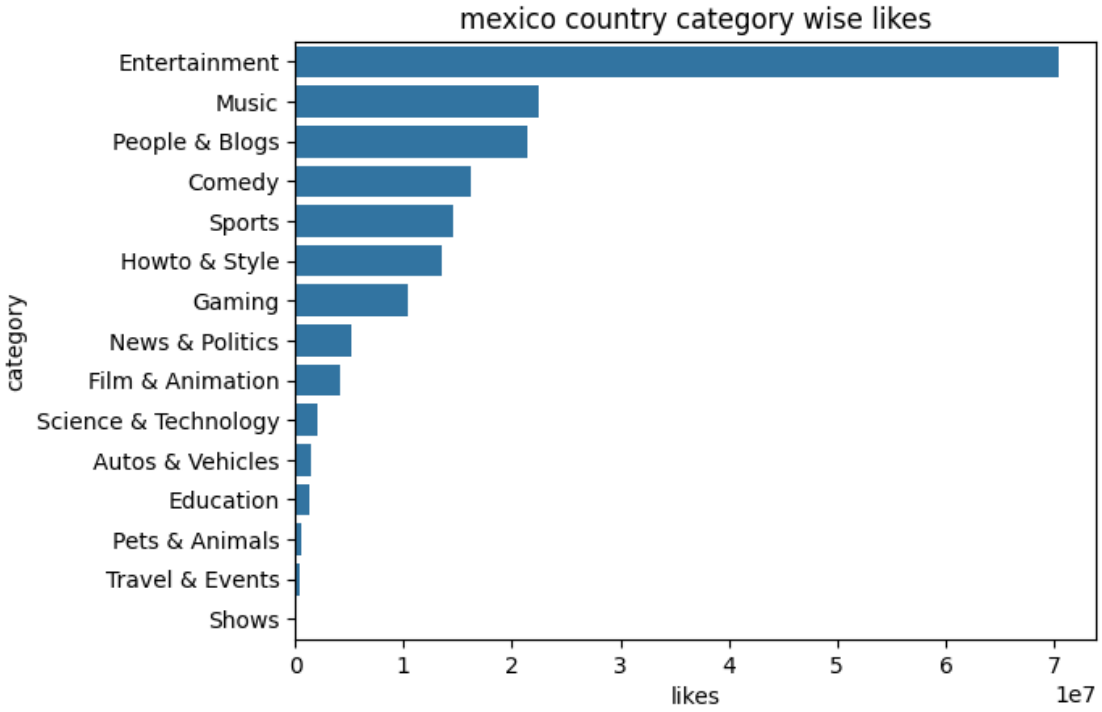

france country category wise likes
Music
Entertainment
Comedy
Sports
People & Blogs
Howto & Style
Gaming
News & Politics
Film & Animation
Science & Technology
Education
Autos & Vehicles
Pets & Animals
Travel & Events
Shows
Movies
Trailers
category
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
likes
1e7

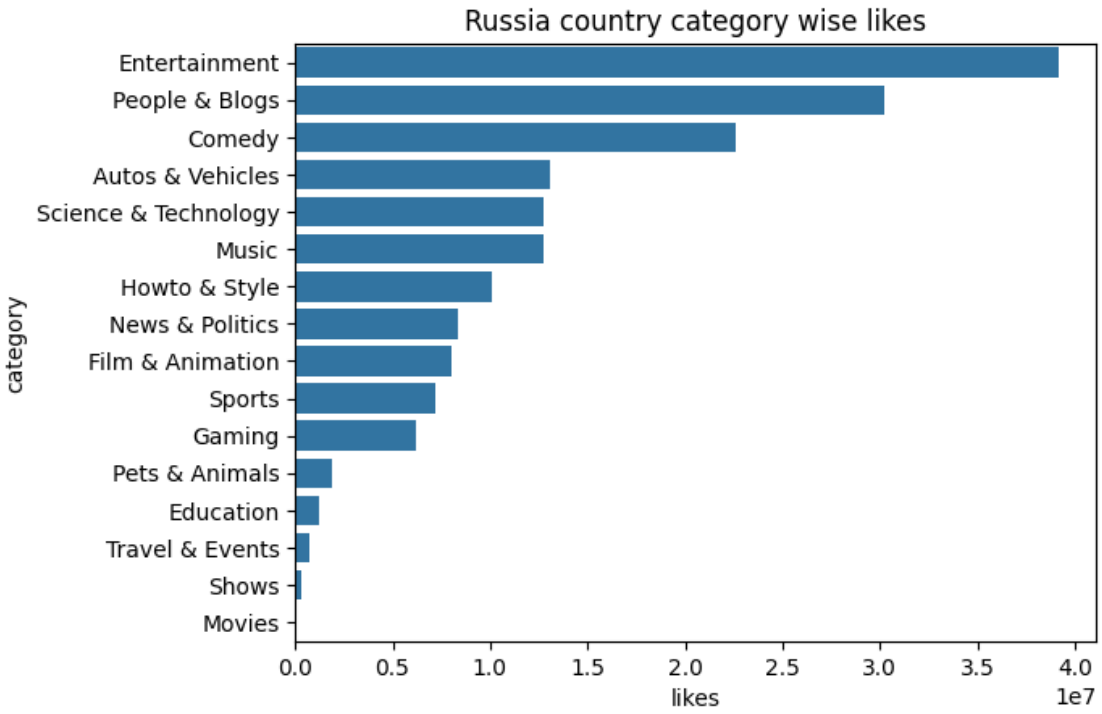

britan country category wise likes
Music
Entertainment
People & Blogs
Comedy
Gaming
Sports
Howto & Style
Film & Animation
Science & Technology
Pets & Animals
Education
News & Politics
Autos & Vehicles
Travel & Events
category
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
likes
1e7

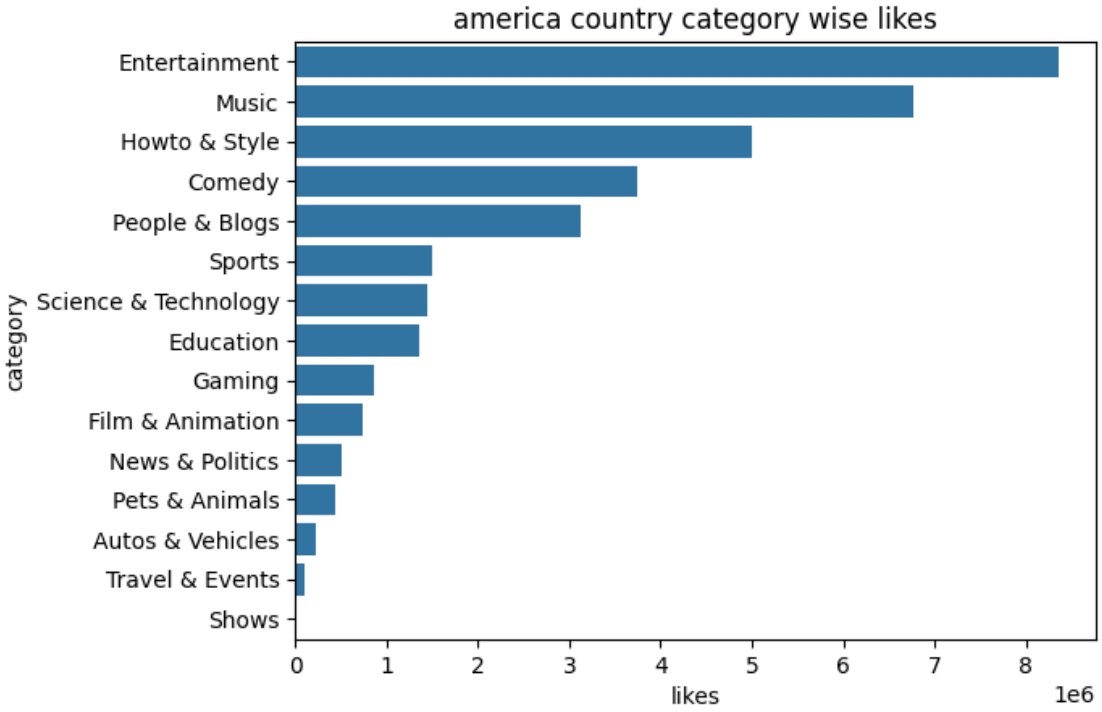

japan country category wise likes
Entertainment
Music
People & Blogs
Comedy
Gaming
Sports
Film & Animation
Pets & Animals
Howto & Style
News & Politics
Science & Technology
Autos & Vehicles
Travel & Events
Education
category
0.0
0.2
0.4
0.6
0.8
1.0
likes
1e7

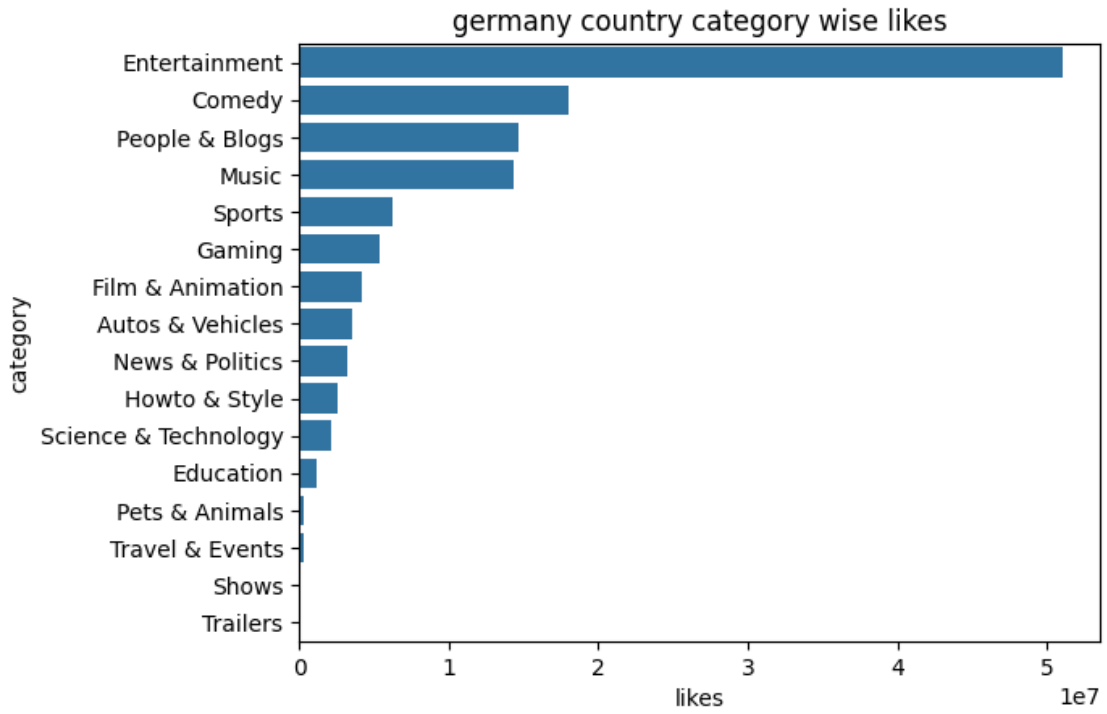

india country category wise likes
Entertainment
Comedy
Music
Science & Technology
People & Blogs
News & Politics
Film & Animation
Education
Howto & Style
Sports
Gaming
Shows
Autos & Vehicles
Travel & Events
Movies
category
0.0
0.5
1.0
1.5
2.0
2.5
likes
1e7


korea country category wise likes
Music
Entertainment
People & Blogs
News & Politics
Comedy
Gaming
Film & Animation
Pets & Animals
Howto & Style
Sports
Education
Science & Technology
Autos & Vehicles
Travel & Events
Shows
Trailers
category
0.0
0.2
0.4
0.6
0.8
1.0
likes
1e7


mexico country category wise likes
Entertainment
Music
People & Blogs
Comedy
Sports
Howto & Style
Gaming
News & Politics
Film & Animation
Science & Technology
Autos & Vehicles
Education
Pets & Animals
Travel & Events
Shows
category
0
1
2
3
4
5
6
7
likes
1e7


Russia country category wise likes
Entertainment
People & Blogs
Comedy
Autos & Vehicles
Science & Technology
Music
Howto & Style
News & Politics
Film & Animation
Sports
Gaming
Pets & Animals
Education
Travel & Events
Shows
Movies
category
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
likes
1e7


america country category wise likes
Entertainment
Music
Howto & Style
Comedy
People & Blogs
Sports
Science & Technology
Education
Gaming
Film & Animation
News & Politics
Pets & Animals
Autos & Vehicles
Travel & Events
Shows
category
0
1
2
3
4
5
6
7
8
likes
1e6



[227]:
```
#Top 5 videos that are on trending in each country?
merged_df.columns
```

[227]:
```
Index(['id', 'channelId', 'category', 'assignable', 'video_id', 'title_y',
       'channel_title', 'tags', 'views', 'likes', 'dislikes', 'comment_count',
       'thumbnail_link', 'comments_disabled', 'ratings_disabled',
       'video_error_or_removed', 'country', 'trending_date', 'published_date',
       'published_time'],
      dtype='object')
```

[228]:
```
# Top 5 videos that are on trending in each country
```

[229]:
```
for country in country_map:
    trending=merged_df[merged_df['country']==country_map[country]].
 ↪sort_values(by=['views'],ascending=False)
    df=pd.DataFrame(trending[['video_id','views','category','likes','views']])

    print('Top 5 videos that are on trending in',country_map[country])
    print(df.head(5))
    print()
```

```
Top 5 videos that are on trending in canada
          video_id     views       category    likes     views
2766   7C2z4GqqS5E  39349927          Music  3880074  39349927
```

```
11180  6ZfuNTqbHE8  37736281  Entertainment  1735931  37736281
1995   wfWkmURBNv8  37597115          Music   769384  37597115
1400   2Vv-BfVoq4g  33523622          Music  1634130  33523622
1493   xmU0s2QtaEY  24939458          Music   529906  24939458

Top 5 videos that are on trending in france
          video_id     views       category   likes     views
4522   AhQcNVyndSM  31791115          Music  299378  31791115
21527  Ed-A73MxMgo  12118553  Entertainment   50708  12118553
2774   ZgTKmRPWMq4   6451953          Music  136776   6451953
3609   Iwz4P8HfGVM   6389033          Music  112962   6389033
4600   tKO87MJtngk   5859790          Music   95516   5859790

Top 5 videos that are on trending in britan
        video_id     views category    likes     views
887  WtE011iVx1Q  47669287    Music   396337  47669287
711  VTzD0jNdrmo  39118664    Music   383030  39118664
882  i0p1bmr0EmE  38873543    Music  1111595  38873543
752  0XElmYomloA  27801810    Music   104931  27801810
655  sD9_l3oDOag  24843733    Music   372300  24843733

Top 5 videos that are on trending in japan
          video_id    views              category   likes    views
12810  aBr2kKAHN6M  6408303  Science & Technology  165892  6408303
9451   KiBS-dbv_x0  4885887         Entertainment  158270  4885887
7318   tR6M8FjHnvE  4719554                Comedy   18080  4719554
8733   s2qyL1DCDcM  3909683         Entertainment   37528  3909683
2216   JspVs6ILBKE  3798438        Pets & Animals   15313  3798438

Top 5 videos that are on trending in india
         video_id     views          category   likes     views
1     ePO5M5DE01I  35885754  Film & Animation  829362  35885754
402   YjMSttRJrhA  15590958  Film & Animation  255106  15590958
1545  3SWc5G8Gx7E  13541025             Music  261457  13541025
1063  h8bvxiE3KCo  13301387             Music  117639  13301387
1253  I7nbSzLCtEE  12374054             Music  134647  12374054

Top 5 videos that are on trending in korea
          video_id     views              category   likes     views
15281  bjn0nYBmS5g  13432309         Howto & Style   20041  13432309
1232   cajMxG9Y-Dk   7664810                 Music  198185   7664810
15671  gmK27rbFTpk   6950397  Science & Technology   11942   6950397
5763   kFgRKoLA1Us   6583116        People & Blogs    6951   6583116
8053   mgd-dSDfb4E   5639375         Entertainment  124908   5639375

Top 5 videos that are on trending in mexico
          video_id     views          category   likes     views
2463   OZLkv6cMKl4  11731087             Music  113378  11731087
```

```
13902  ummI3h6Lbm0   8940772   People & Blogs     131   8940772
14417  Y0M6LIPhrPg   6868690   People & Blogs    6279   6868690
3791   5aU4RQoNzXY   6520083            Music   27643   6520083
29078  3P-Ats8NLXg   4841490  News & Politics    3544   4841490

Top 5 videos that are on trending in Russia
          video_id     views          category  likes     views
2168   baODA54mAtc  10699657  Film & Animation   5194  10699657
30764  M26uJlp9ASI   9055150     Howto & Style   6559   9055150
23333  8FVYKlp1nkA   5557520     Entertainment   9431   5557520
32862  SwI33b2w7ag   5167531             Shows  50414   5167531
23962  dRoQzDc0y2o   3862756     Entertainment      0   3862756

Top 5 videos that are on trending in america
         video_id     views              category   likes     views
1696  hYirFqEc8Tg  12941382                Sports  377740  12941382
6165  ktafrbsKeZw  10008678  Science & Technology   23091  10008678
1008  u6unJQownW4   9129695                 Music  332748   9129695
4318  JqfuKsoEEms   8244248         Entertainment    2697   8244248
6332  J2m7EUwnqGg   8045394  Science & Technology   68126   8045394

Top 5 videos that are on trending in germany
          video_id     views         category   likes     views
3095   y0PMqR7IbAQ  26137768            Music  290314  26137768
25482  0AcBTKITQsQ  11870763  News & Politics   28739  11870763
28443  tMizgzhcTPk   9279003        Education       0   9279003
3422   OaqvfQ5ggas   8344248            Music   27457   8344248
3637   gi80-tmEduI   8186205            Music  216046   8186205
```

[230]:
```
#Is the most liked video also the most trending video?
```

[231]:
```
for country in country_map:
    trending=merged_df[merged_df['country']==country_map[country]].
 ↪sort_values(by=['likes'],ascending=False)
    df=pd.DataFrame(trending[['video_id','views','category','likes','views']])

    print('Top 5 videos that are on trending in',country_map[country])
    print(df.head(5))
    print()
```

```
Top 5 videos that are on trending in canada
          video_id     views       category    likes     views
2766   7C2z4GqqS5E  39349927          Music  3880074  39349927
1489   kTlv5_Bs8aw  13945717          Music  2055166  13945717
2195   OK3GJ0WIQ8s  10695328          Music  2050563  10695328
2667   p8npDG2ulKQ  10666323          Music  1956230  10666323
11180  6ZfuNTqbHE8  37736281  Entertainment  1735931  37736281
```

```
Top 5 videos that are on trending in france
          video_id     views       category    likes     views
3499   8O_MwlZ2dEg   5275672          Music  1401947   5275672
4219   28XC2KRE-DE   2248202          Music   401604   2248202
3479   cD8SYW8rjaQ   2191988          Music   354931   2191988
4522   AhQcNVyndSM  31791115          Music   299378  31791115
17786  YDLNnw7gWLo   2351873  Entertainment   259078   2351873

Top 5 videos that are on trending in britan
         video_id     views       category    likes     views
2479  _5d-sQ7Fh5M  11511155  Entertainment  1133321  11511155
882   i0p1bmr0EmE  38873543          Music  1111595  38873543
1485  x4CF5pruJz4  15919643         Gaming   915212  15919643
1272  Vm4Xn5vjqRA   3990953         Sports   889008   3990953
644   qtTM2YV3bI8  19878085          Music   874521  19878085

Top 5 videos that are on trending in japan
         video_id    views        category   likes    views
1675  lOf16ZQDDdE  2951703           Music  488444  2951703
1753  K4Melso7MPU  2384801           Music  416032  2384801
1209  Ip7eVuFxiEE  1478220           Music  232359  1478220
8633  sJGolNWtHFI   690491   Entertainment  210294   690491
7007  flPqsVWTnT0   854949  People & Blogs  166673   854949

Top 5 videos that are on trending in india
          video_id     views          category    likes     views
10041  T9WN2_ikz6Q   7633571     Entertainment  1075454   7633571
1      ePO5M5DE01I  35885754  Film & Animation   829362  35885754
7635   PoJ73OXEEy0   6731648     Entertainment   666922   6731648
10014  f6NJvg-kWJ8   6312526     Entertainment   495044   6312526
4041   Wje3zkc7syE   5691239            Comedy   482157   5691239

Top 5 videos that are on trending in korea
         video_id    views category   likes    views
1708  6tJP3YOQhV4  3886012    Music  738506  3886012
1697  ZaOvpvw1NwA  2185092    Music  410325  2185092
1001  HQsyxzOCuWA  2577475    Music  398281  2577475
1667  Hoz5OQjXEA8  2731984    Music  379813  2731984
1210  XcdLVKYsHHU  2417642    Music  352937  2417642

Top 5 videos that are on trending in mexico
          video_id    views        category   likes    views
18882  q_j0vVn9v0g  3077739   Entertainment  786410  3077739
18243  TNGOGOOVnPM  2238589   Entertainment  591326  2238589
11719  osgnc2jNvL8  3042116  People & Blogs  569268  3042116
10898  0UWEipTg19I  2827578  People & Blogs  532302  2827578
3387   3GLPjp_A1tE  3080856           Music  492854  3080856
```

```
Top 5 videos that are on trending in Russia
          video_id    views       category   likes    views
5205   IHv_6HP65VQ  1766961          Music  319116  1766961
4874   jBGeIEE4IME  1877459          Music  290930  1877459
23659  aaZ6MakieSY  2589777  Entertainment  277683  2589777
4526   kqHkpW0DINc  2024592          Music  272029  2024592
20714  7qckwhd_lfw  2724180  Entertainment  269178  2724180

Top 5 videos that are on trending in america
         video_id     views       category   likes     views
438   qooQd8AA7_M   5626588          Music  408618   5626588
1696  hYirFqEc8Tg  12941382         Sports  377740  12941382
4127  xf-dywd3zx0   1801695  Entertainment  369820   1801695
608   FITSPSA8gQs   6062169          Music  363560   6062169
810   floMqK_yHf8   1835185          Music  342655   1835185

Top 5 videos that are on trending in germany
          video_id    views              category   likes    views
20114  LTvlDklQHbs  3548264         Entertainment  506437  3548264
13395  PM47LAAT3j8  2497718                Comedy  421493  2497718
20546  0ImCUnvas8M  2517453         Entertainment  413750  2517453
28988  51wobJ4jOU4  1513232  Science & Technology  383360  1513232
8308   WzcOFPkL1yc  1840845        People & Blogs  382415  1840845
```

[232]:
```
merged_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 182299 entries, 0 to 6334
Data columns (total 20 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   id                      182299 non-null  int32
 1   channelId               182299 non-null  object
 2   category                182299 non-null  object
 3   assignable              182299 non-null  bool
 4   video_id                182299 non-null  object
 5   title_y                 182299 non-null  object
 6   channel_title           182299 non-null  object
 7   tags                    182299 non-null  object
 8   views                   182299 non-null  int64
 9   likes                   182299 non-null  int64
 10  dislikes                182299 non-null  int64
 11  comment_count           182299 non-null  int64
 12  thumbnail_link          182299 non-null  object
 13  comments_disabled       182299 non-null  bool
 14  ratings_disabled        182299 non-null  bool
```

```
 15  video_error_or_removed  182299 non-null  bool
 16  country                 182299 non-null  object
 17  trending_date           182299 non-null  datetime64[ns]
 18  published_date          182299 non-null  object
 19  published_time          182299 non-null  object
dtypes: bool(4), datetime64[ns](1), int32(1), int64(4), object(10)
memory usage: 23.6+ MB
```

[233]:
```
# Maximum number of days to trending status for a video
```

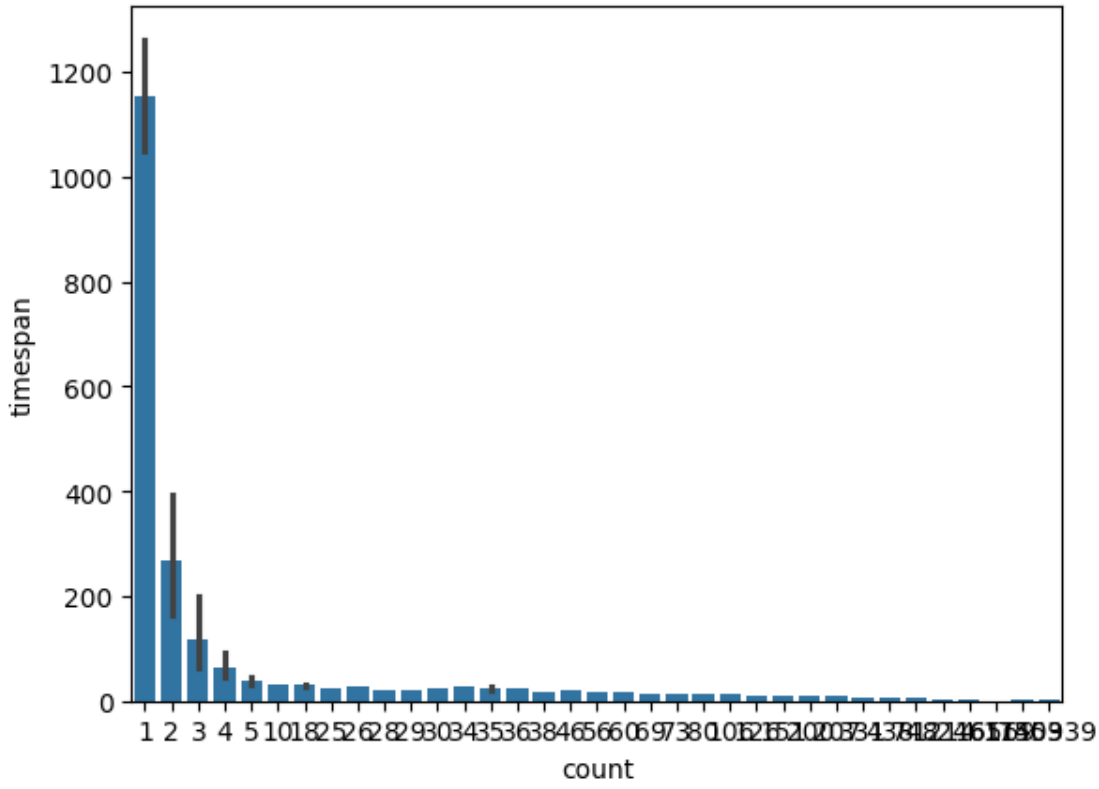
[234]:
```
merged_df['published_date']= pd.
  ↪to_datetime(merged_df['published_date'],format='%d-%m-%Y')
merged_df['timespan'] = (merged_df['trending_date'] -␣
  ↪merged_df['published_date']).dt.days
trending=pd.DataFrame(merged_df['timespan'].value_counts())
trending
```

[234]:
```
           count
timespan
1         140939
2          17553
0          16569
3           2461
4           1214
…           …
2184           1
1235           1
472            1
748            1
1568           1

[433 rows x 1 columns]
```

[235]:
```
sns.barplot(x='count',y='timespan',data=trending)
```

[235]:
```
<Axes: xlabel='count', ylabel='timespan'>
```



[236]:
```
#Users comment on which category the most?
```

[237]:
```
comment_count=merged_df.groupby('category')['comment_count'].sum().
  ↪sort_values(ascending=False)
comment_count=pd.DataFrame(comment_count)
comment_count
```

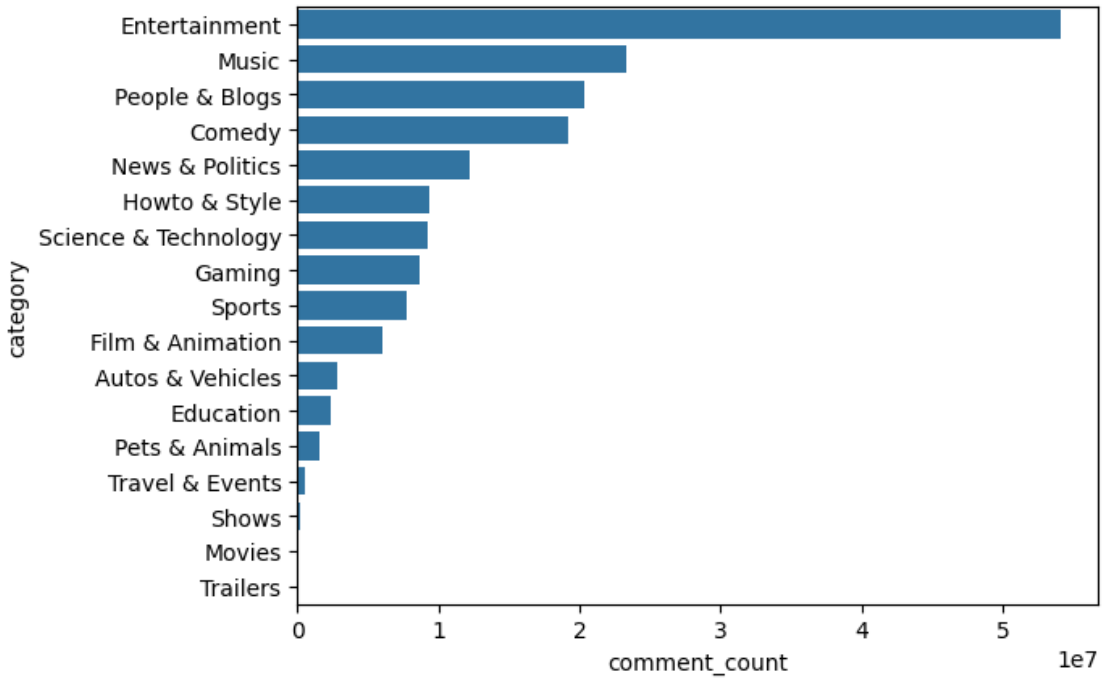
[237]:

category	comment_count
Entertainment	54061612
Music	23381489
People & Blogs	20394603
Comedy	19205867
News & Politics	12176381
Howto & Style	9387440
Science & Technology	9216968
Gaming	8695125
Sports	7740339
Film & Animation	6069950
Autos & Vehicles	2822884
Education	2334799

```
Pets & Animals              1586750
Travel & Events              549794
Shows                        199338
Movies                         1046
Trailers                          1
```

[238]:
```
sns.barplot(x='comment_count',y='category',data=comment_count)
```

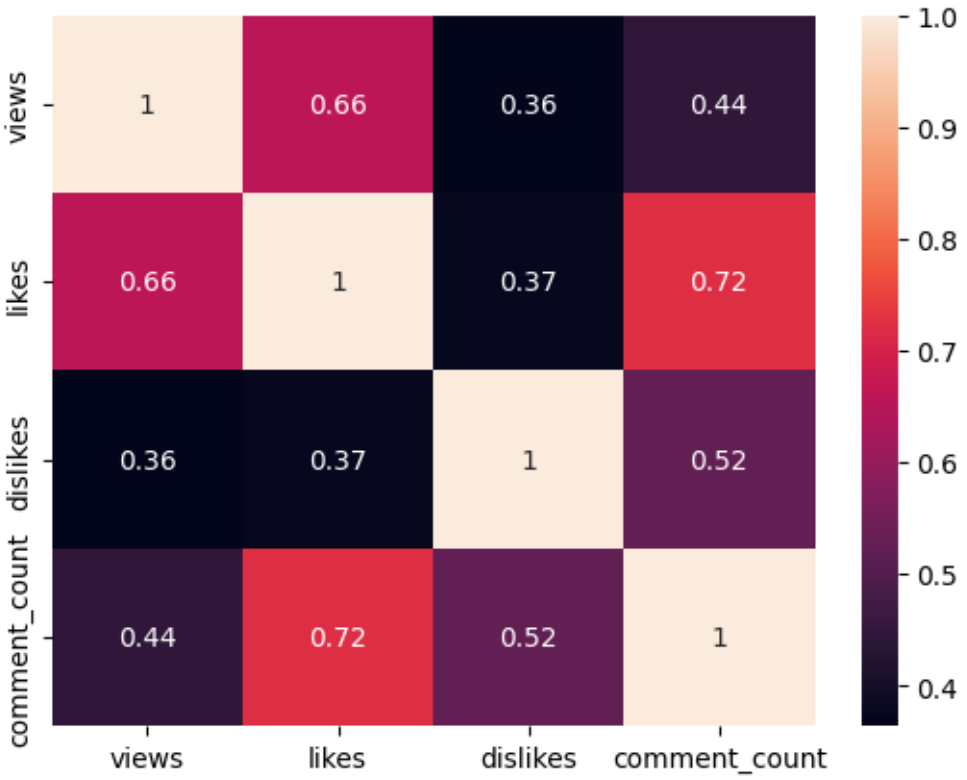
[238]:
```
<Axes: xlabel='comment_count', ylabel='category'>
```



[239]:
```
#Correlation between views, likes, dislikes, and comments
```

[240]:
```
columns=['views','likes','dislikes','comment_count']
sns.heatmap(merged_df[columns].corr(),annot=True)
```

[240]:
```
<Axes: >
```



[241]:
```
#which you tube channel has the most likes
```

[242]:
```
merged_df['channel_title'].unique()
```

[242]:
```
array([' ', 'Rooster Teeth', 'Lokdhun Punjabi', …, 'To Scale:',
       'Deep Look', 'Mrwhosetheboss'], dtype=object)
```

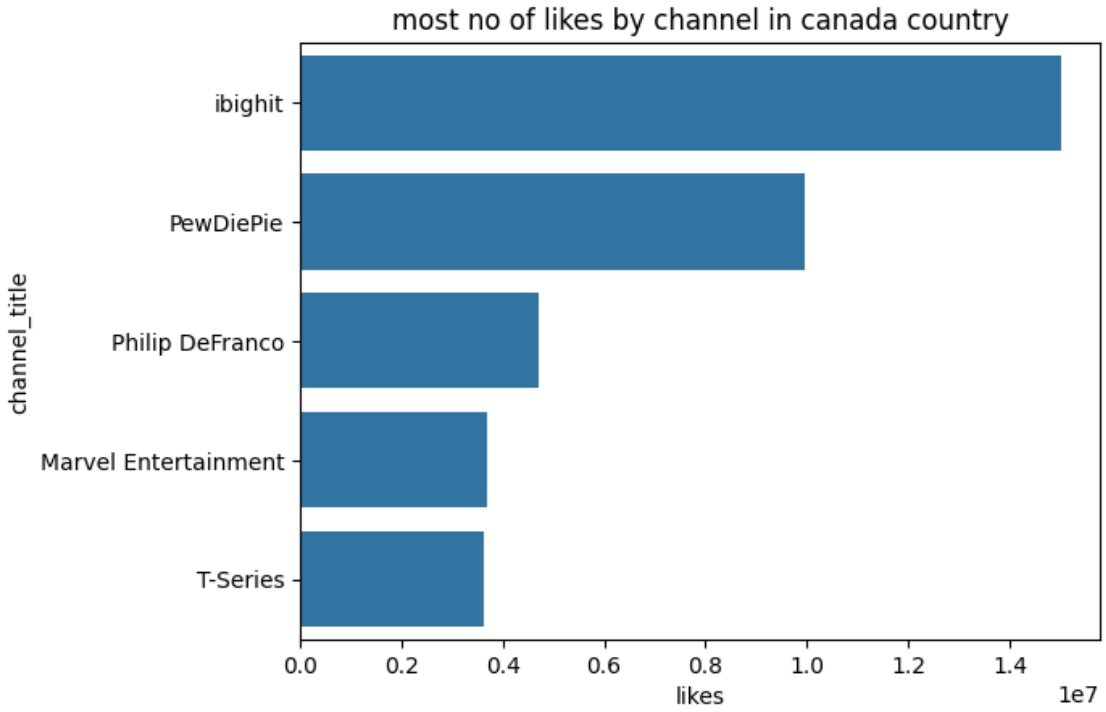
[243]:
```
channel_likes=merged_df.groupby('channel_title')['likes'].sum().
 ↪sort_values(ascending=False)
channel_likes=pd.DataFrame(channel_likes)
channel_likes.head()
```

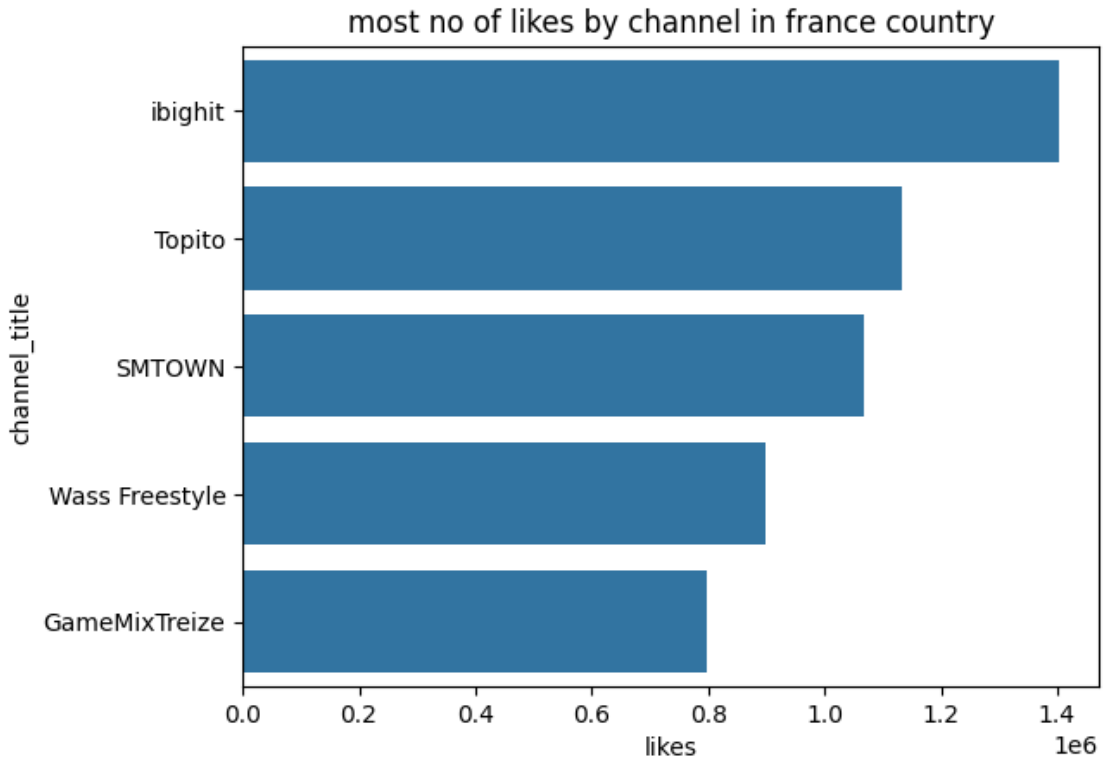
[243]:

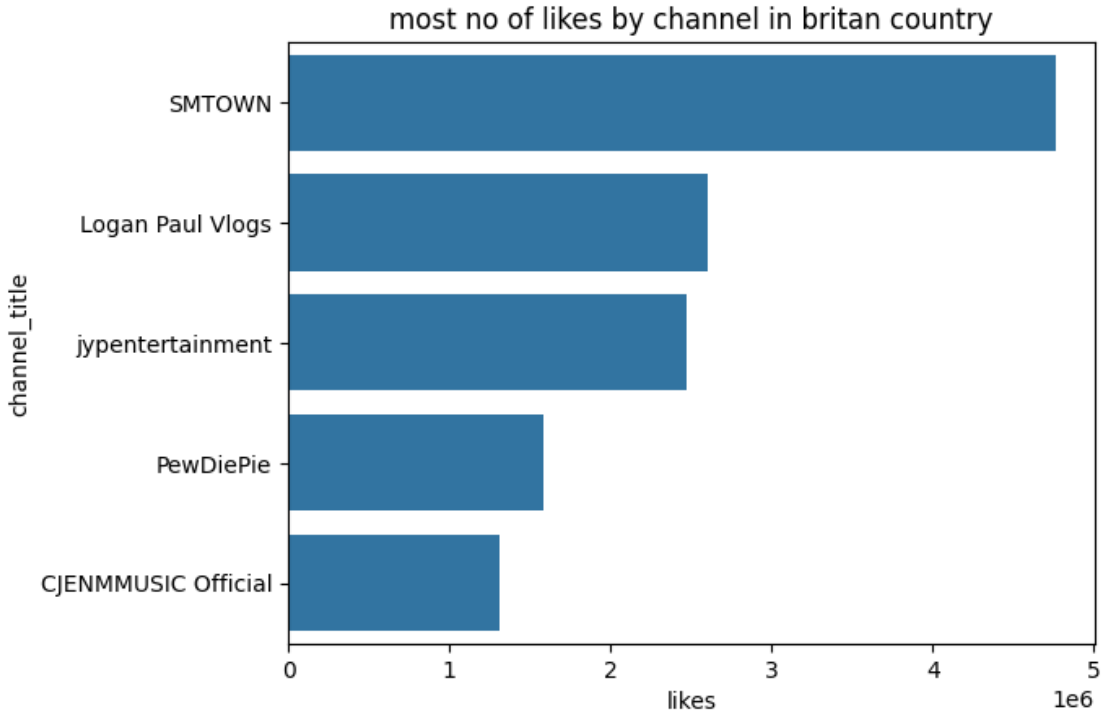
channel_title	likes
ibighit	16872642
PewDiePie	12481932
SMTOWN	11522512
BANGTANTV	8950098
Cracks	7645419

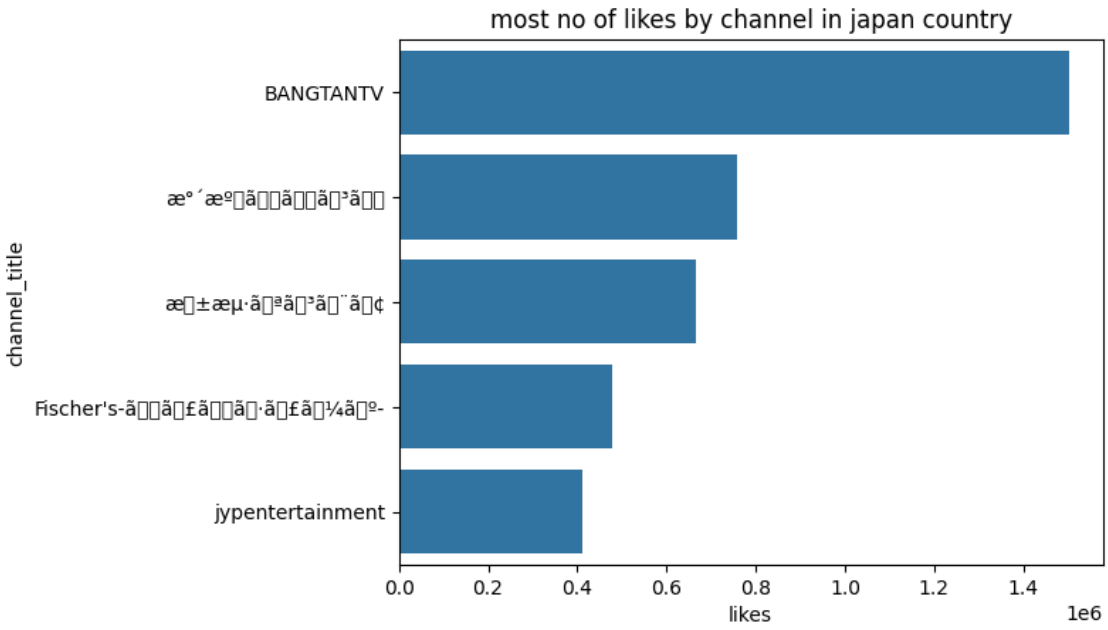
[244]:
```
#which you tube channel has the most likes country wise
```

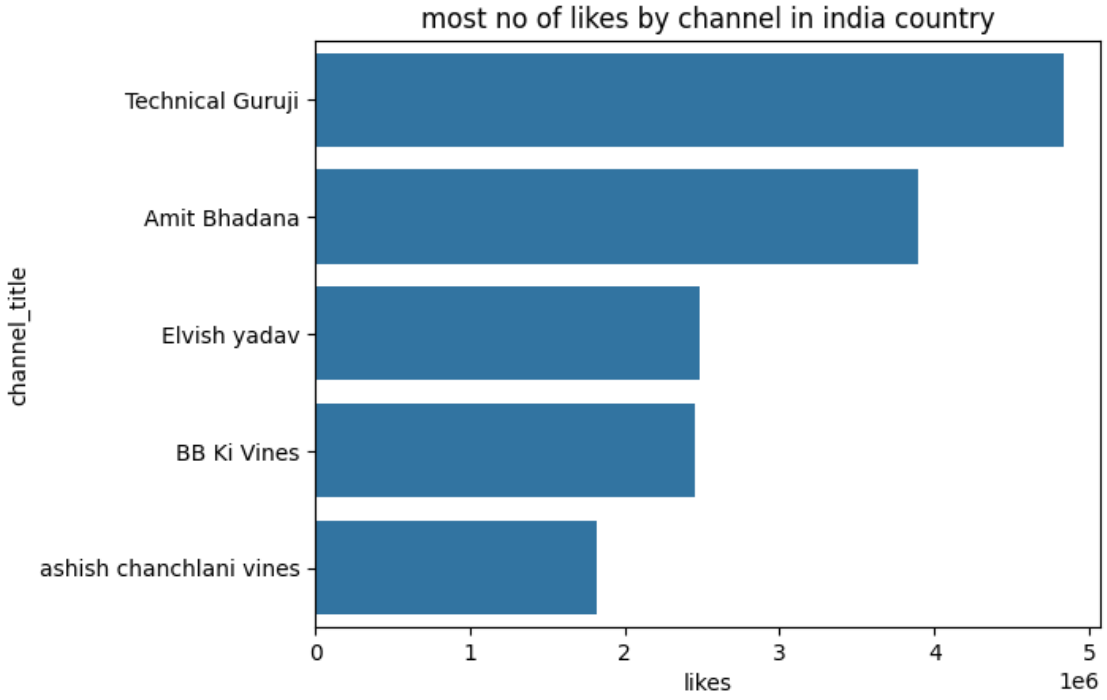
[245]:
```
for country in country_map:
    channel_likes=merged_df[merged_df['country']==country_map[country]].
 ↪groupby('channel_title')['likes'].sum().sort_values(ascending=False)
    channel_likes=pd.DataFrame(channel_likes)
    channel_likes=channel_likes.head(5)
    sns.barplot(x='likes',y='channel_title',data=channel_likes)
    plt.title('most no of likes by channel in {} country'.
 ↪format(country_map[country]))
    plt.show()
```

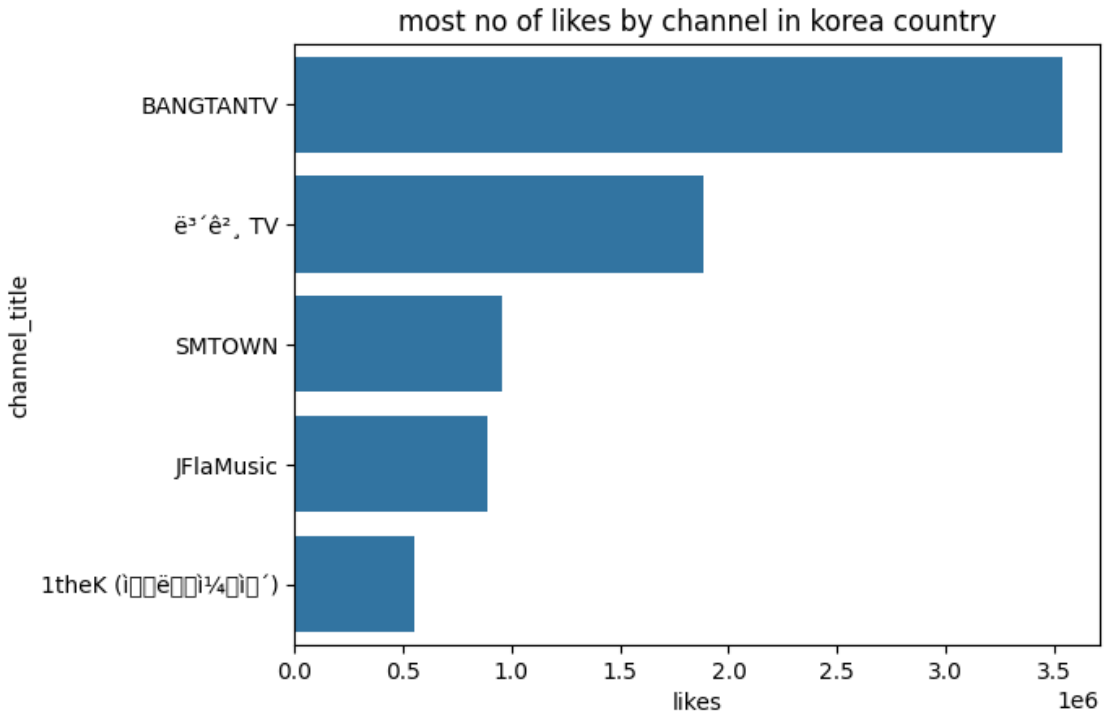


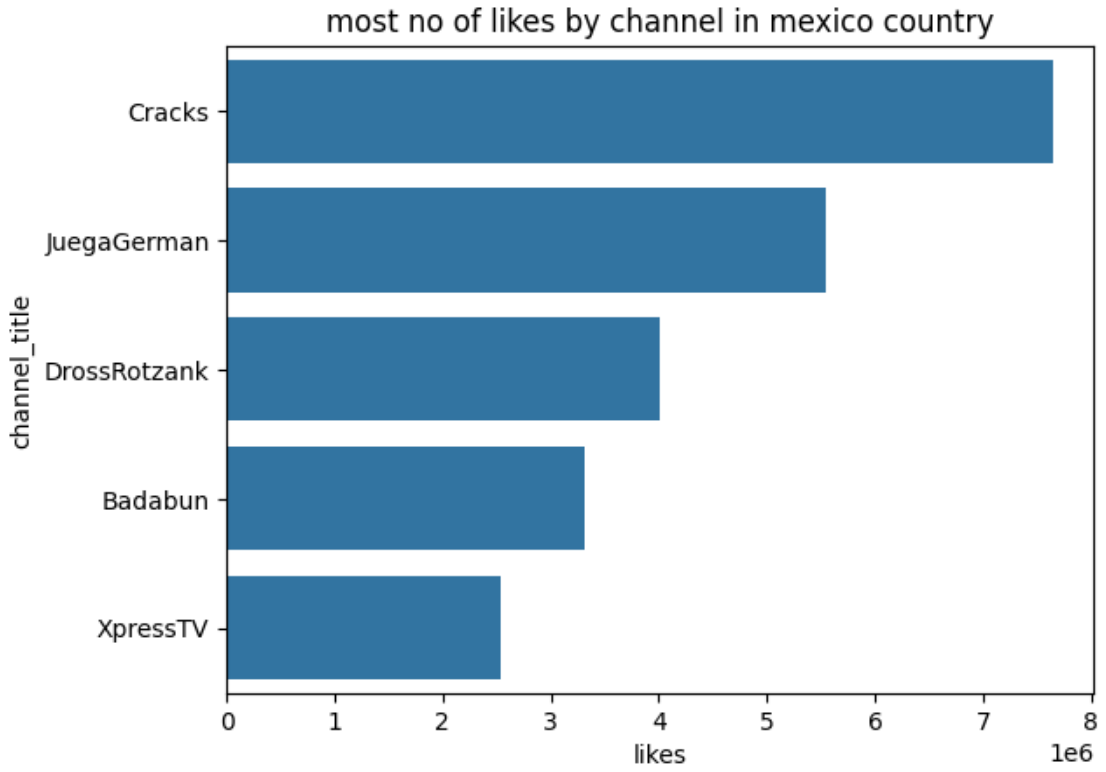

most no of likes by channel in france country
ibighit
Topito
SMTOWN
Wass Freestyle
GameMixTreize
channel_title
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
likes
1e6

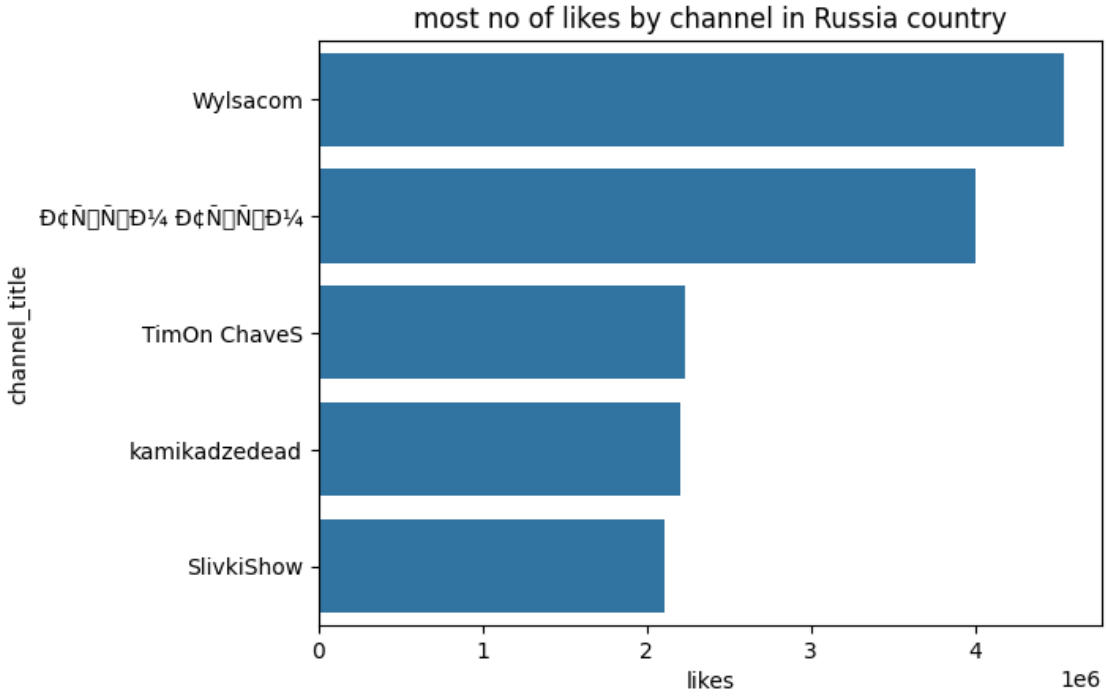

most no of likes by channel in britan country
SMTOWN
Logan Paul Vlogs
jypentertainment
PewDiePie
CJENMMUSIC Official
channel_title
0
1
2
3
4
5
likes
1e6

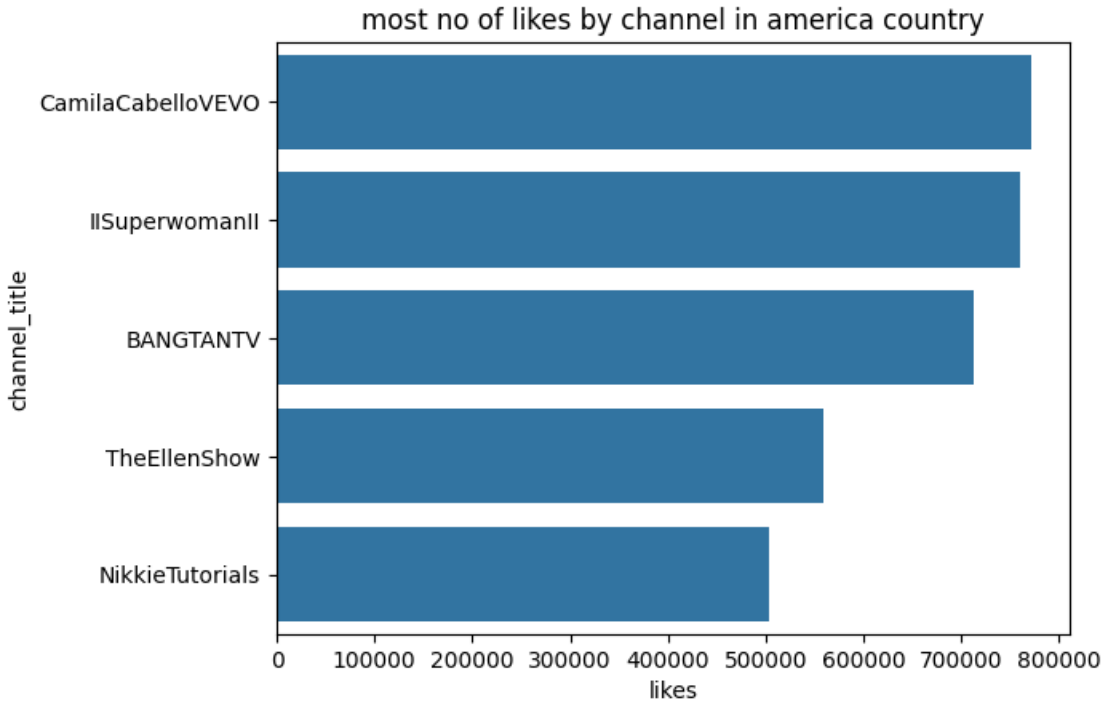

most no of likes by channel in japan country
BANGTANTV
æ°´æº□ã□□ã□□ã□³ã□□
æ□±æµ·ã□ªã□³ã□¨ã□¢
Fischer's-ã□□ã□£ã□□ã□·ã□£ã□¼ã□º-
jypentertainment
channel_title
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
likes
1e6

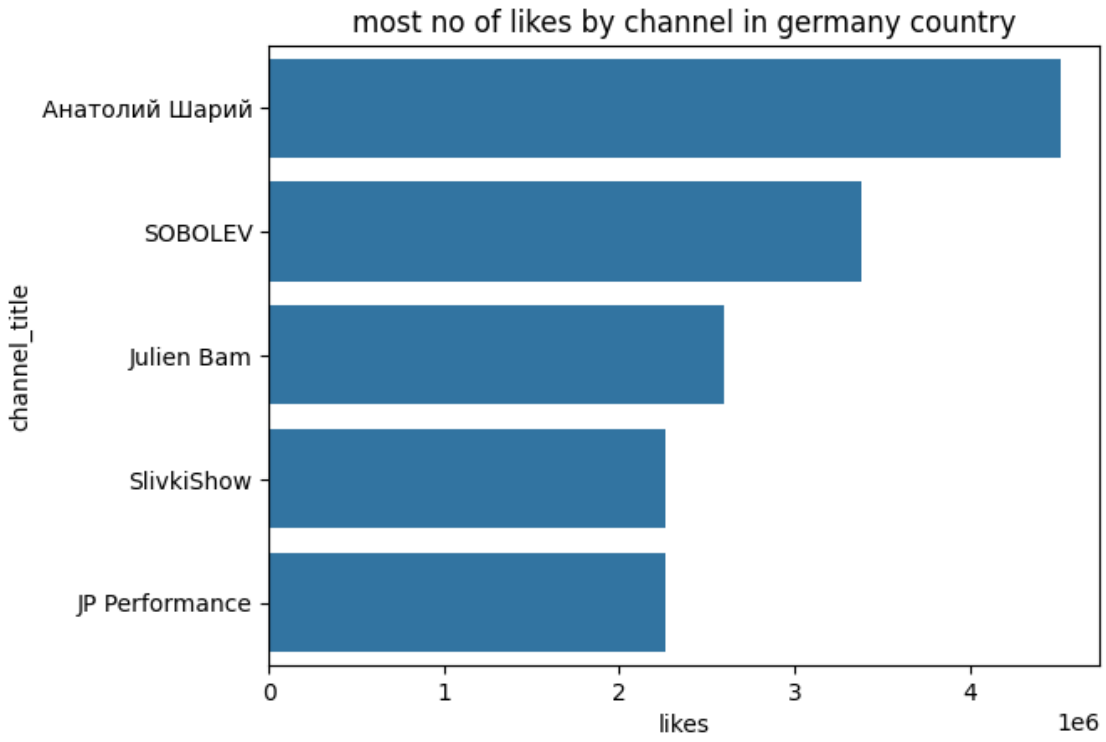

most no of likes by channel in india country
Technical Guruji
Amit Bhadana
Elvish yadav
BB Ki Vines
ashish chanchlani vines
channel_title
0
1
2
3
4
5
likes
1e6


most no of likes by channel in korea country
BANGTANTV
ë³´ê²¸ TV
SMTOWN
JFlaMusic
1theK (ì□□ë□□ì¼□ì□´)
channel_title
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
likes
1e6


most no of likes by channel in mexico country
Cracks
JuegaGerman
DrossRotzank
Badabun
XpressTV
channel_title
0
1
2
3
4
5
6
7
8
likes
1e6

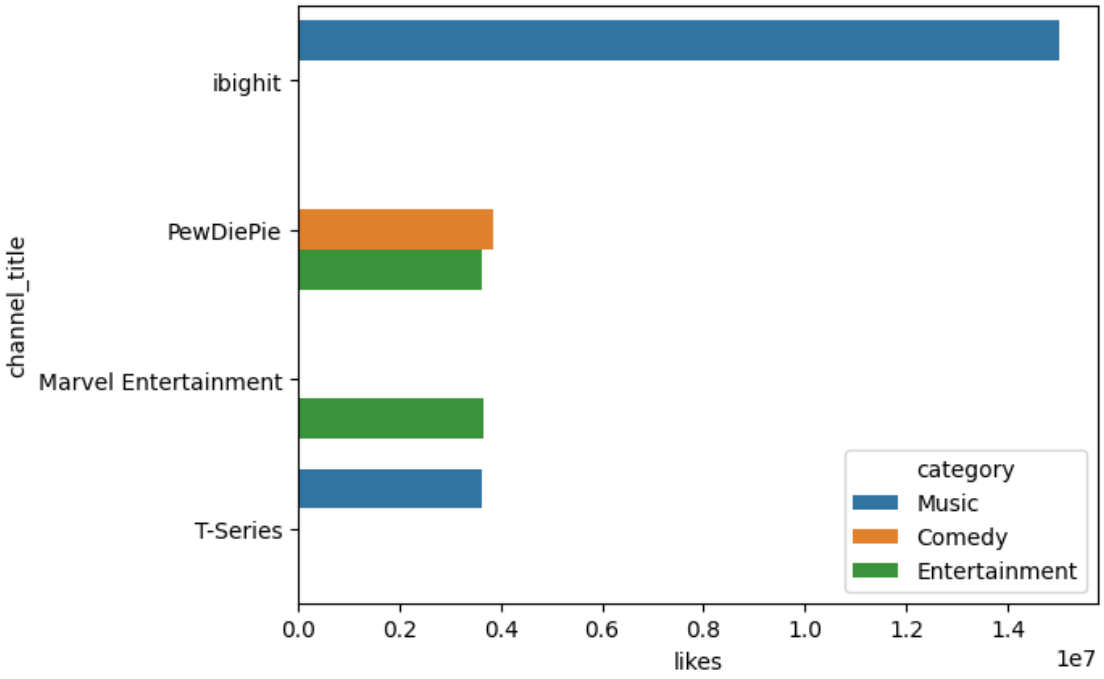

most no of likes by channel in Russia country
Wylsacom
Ð¢ÑÑÐ¼ Ð¢ÑÑÐ¼
TimOn ChaveS
kamikadzedead
SlivkiShow
channel_title
0
1
2
3
4
likes
1e6

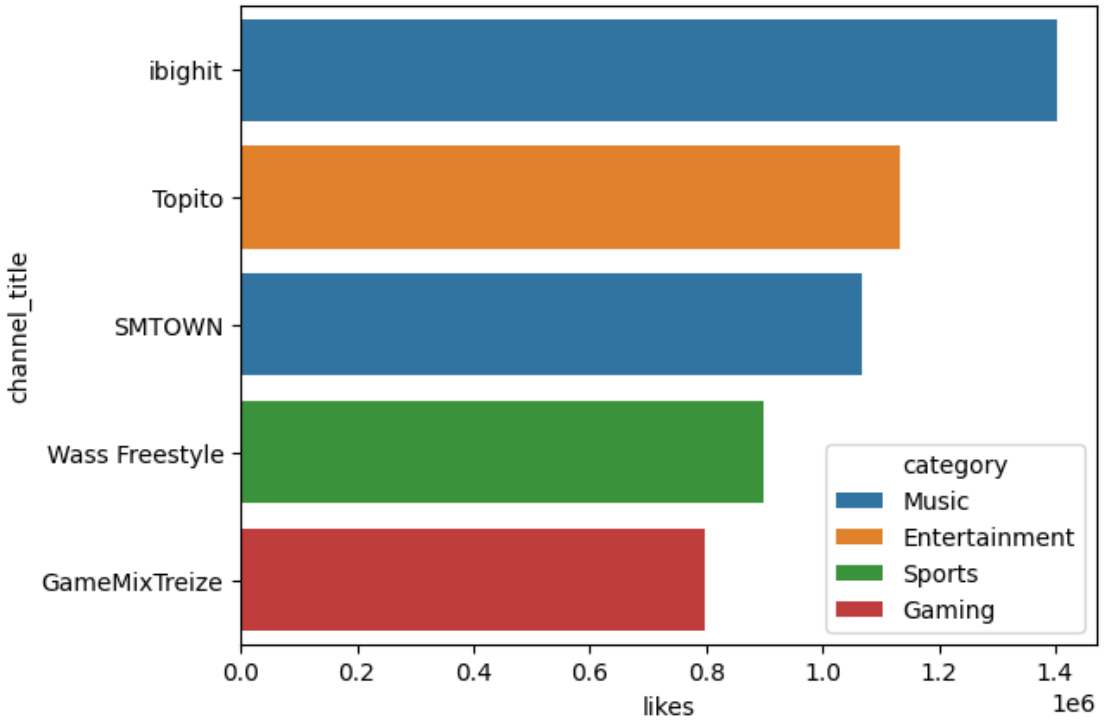

most no of likes by channel in america country
CamilaCabelloVEVO
IISuperwomanII
BANGTANTV
TheEllenShow
NikkieTutorials
channel_title
0
100000
200000
300000
400000
500000
600000
700000
800000
likes

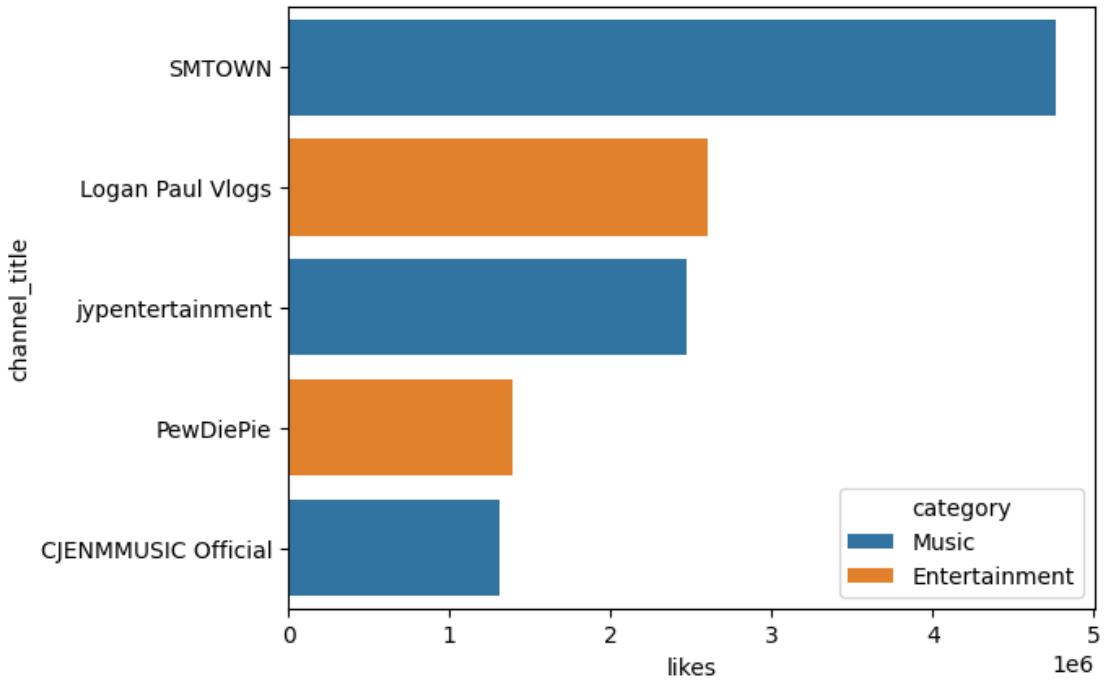


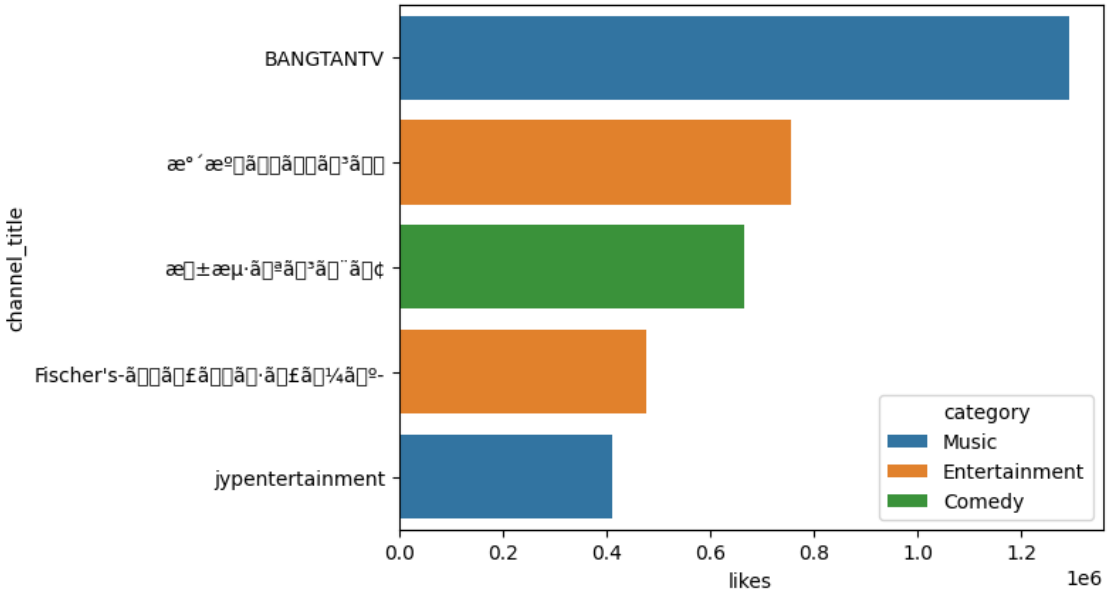
[246]:
```
#country wise channel most liked videos on category
```

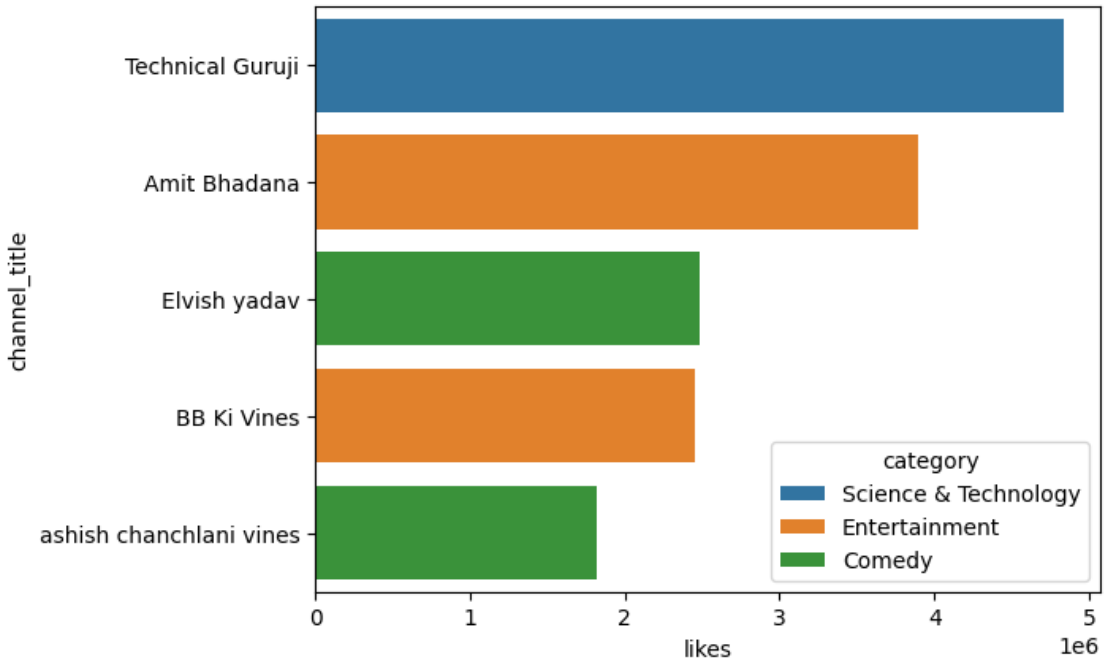
[247]:
```
for country in country_map:
    channel_likes=merged_df[merged_df['country']==country_map[country]].
 ↪groupby(['channel_title','category',])['likes'].sum().
 ↪sort_values(ascending=False)
    channel_likes=pd.DataFrame(channel_likes)
    channel_likes=channel_likes.head(5)
    #print(channel_likes)
    sns.barplot(x='likes',y='channel_title',data=channel_likes,hue='category')
    plt.show()
```

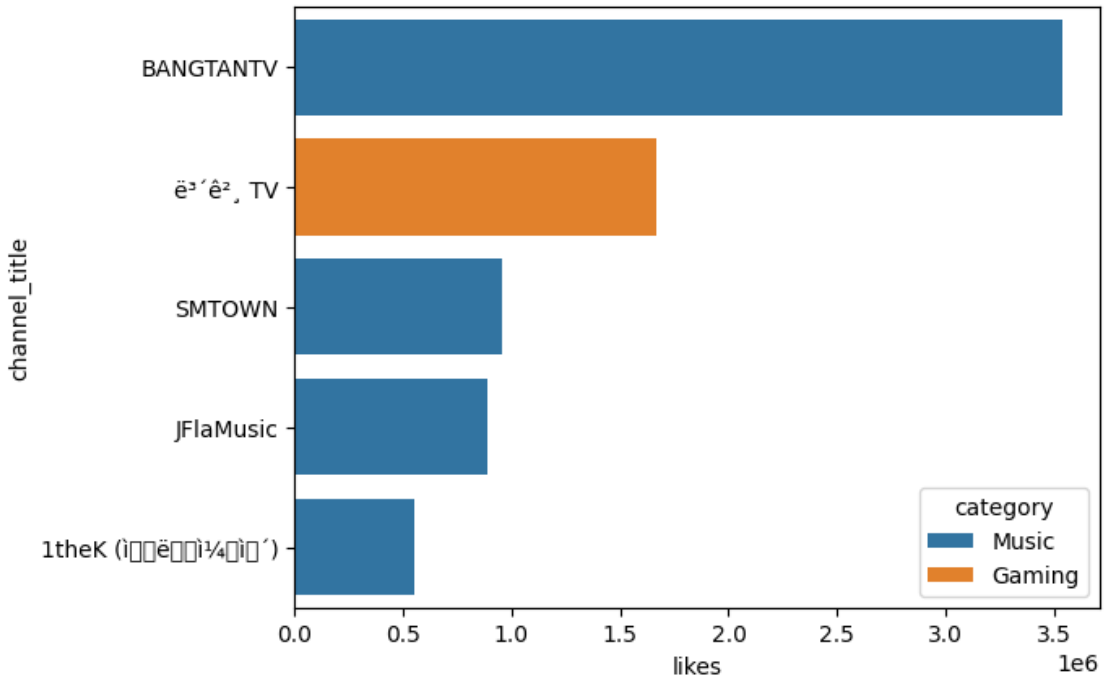

ibighit
PewDiePie
Marvel Entertainment
T-Series
channel_title
category
Music
Comedy
Entertainment
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
likes
1e7

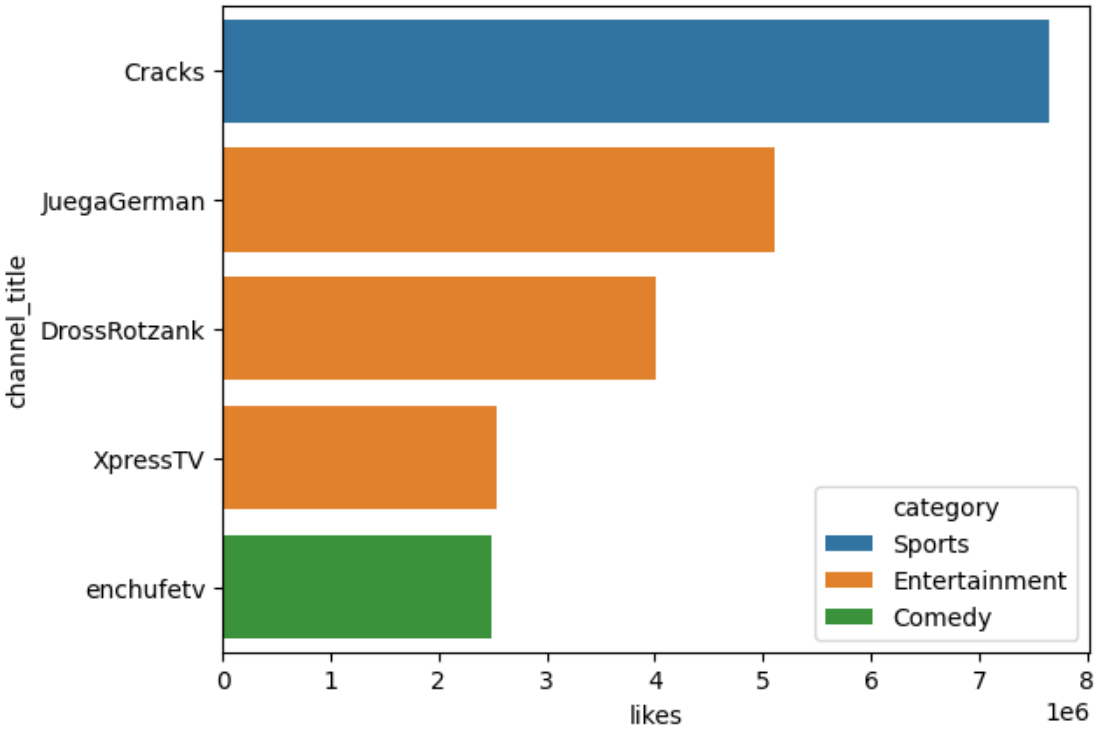

ibighit
Topito
SMTOWN
Wass Freestyle
GameMixTreize
channel_title
category
Music
Entertainment
Sports
Gaming
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
likes
1e6

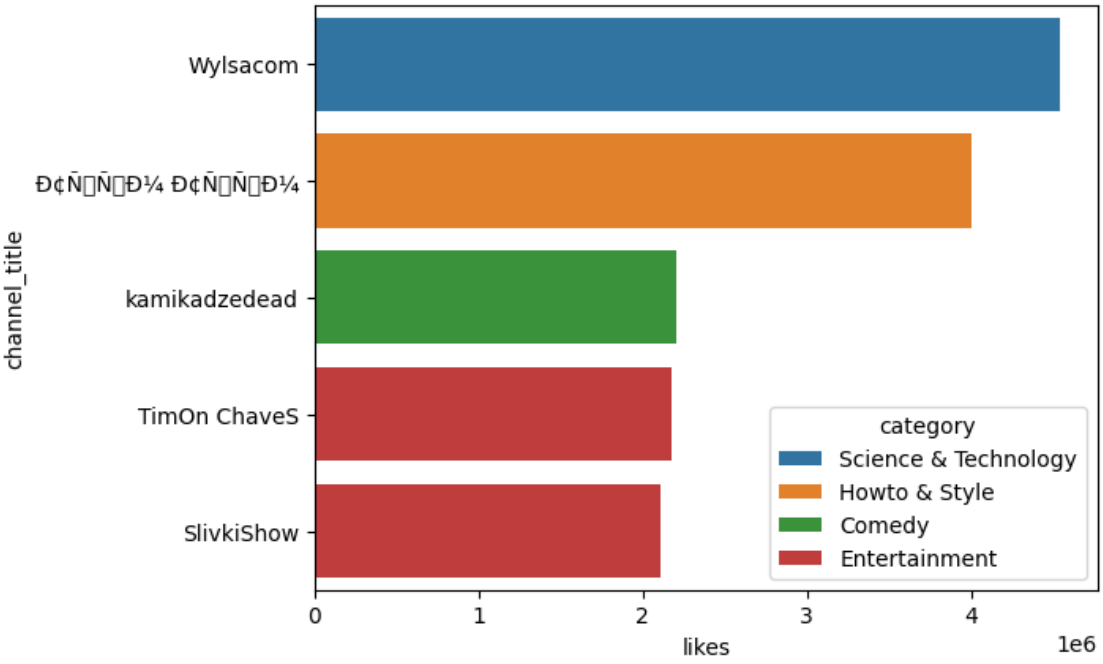

SMTOWN
Logan Paul Vlogs
jypentertainment
PewDiePie
CJENMMUSIC Official
channel_title
category
Music
Entertainment
0
1
2
3
4
5
likes
1e6

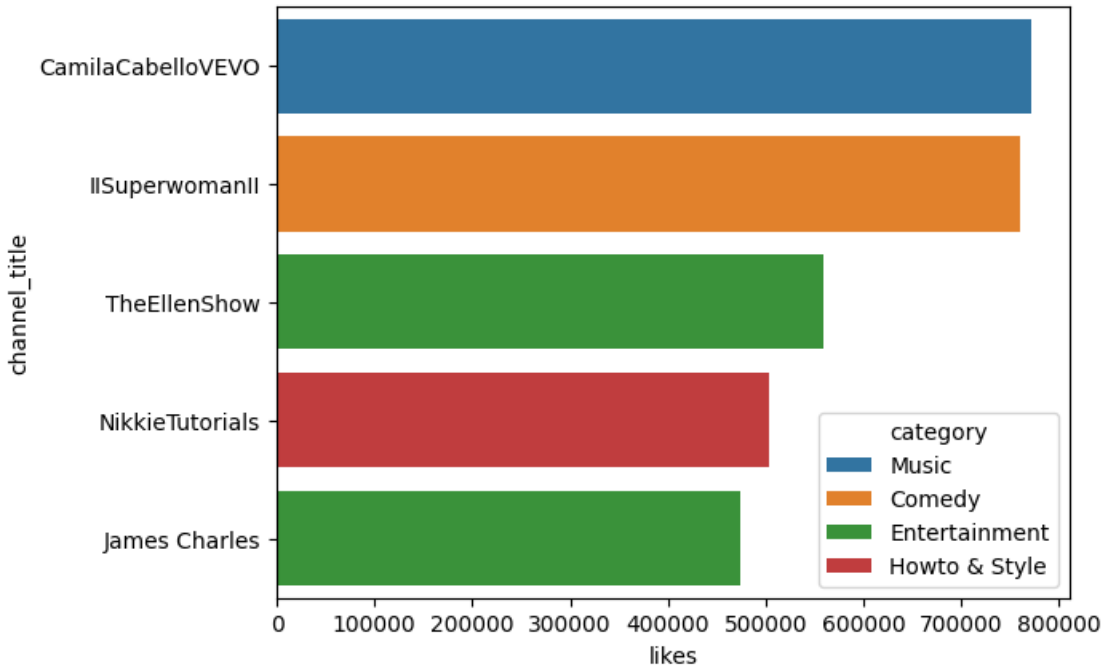

BANGTANTV
æ°´æºã□□ã□□ã□³ã□□
æ□±æµ·ã□ªã□³ã□¨ã□¢
Fischer's-ã□□ã□£ã□□ã□·ã□£ã□¼ã□º-
jypentertainment
channel_title
category
Music
Entertainment
Comedy
0.0
0.2
0.4
0.6
0.8
1.0
1.2
likes
1e6

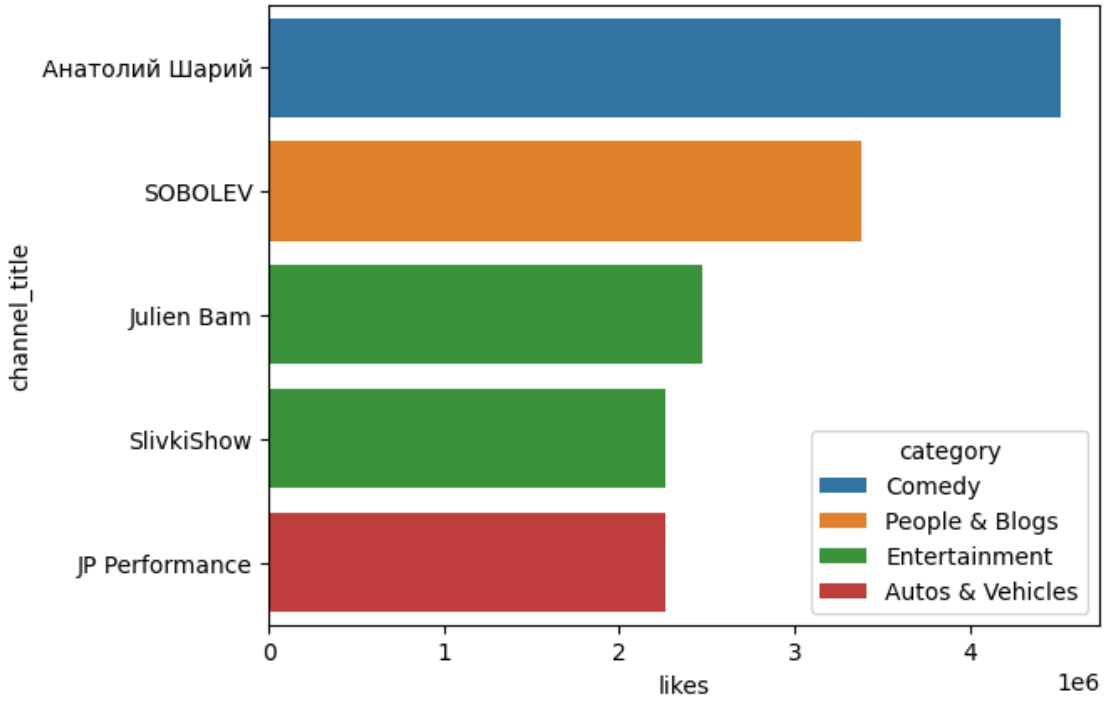

Technical Guruji
Amit Bhadana
Elvish yadav
BB Ki Vines
ashish chanchlani vines
channel_title
category
Science & Technology
Entertainment
Comedy
0
1
2
3
4
5
likes
1e6


BANGTANTV
ë³´ê²¸ TV
SMTOWN
JFlaMusic
1theK (ì□□ë□□ì¼□ì□´)
channel_title
category
Music
Gaming
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
likes
1e6


Cracks
JuegaGerman
DrossRotzank
XpressTV
enchufetv
channel_title
category
Sports
Entertainment
Comedy
0
1
2
3
4
5
6
7
8
likes
1e6


Wylsacom
Ð¢ÑÑÐ¼ Ð¢ÑÑÐ¼
kamikadzedead
TimOn ChaveS
SlivkiShow
channel_title
category
Science & Technology
Howto & Style
Comedy
Entertainment
0
1
2
3
4
likes
1e6





[248]:
```
#which category videos are the most disliked videos
```

[249]:
```
dislikes=merged_df.groupby('category')['dislikes'].sum().
  ↪sort_values(ascending=False)
dislikes=pd.DataFrame(dislikes)
dislikes=dislikes.head()
```

[250]:
```
plt.pie(dislikes['dislikes'],labels=dislikes.index,autopct="%.2f")
plt.show()
```