# a-case-study-project

September 18, 2023

## 1 CASE STUDY ANALYSIS FOR DATA SET CONTAINTING RAW TWEETS SURROUNDING THE IMMEDIATE TIME FRAME OF THE DEATH OF QUEEN ELIZABETH II

1. PRESENTED BY AMRITA ARYA
2. IMH/10038/19
3. IMSC. MATHEMATICS AND COMPUTING BIT MESRA

Here are datasets containing raw tweets, surrounding the immediate time frame of the death of Queen Elizabeth II Keywords for corresponding tweet search: “Queen Elizabeth” read the dataset in python and answer the below mentioned Questions 1. Find the user of the most retweets. 2. Find the most effective tweet (create a measure of your own based on parameters such as retweets, time from death, etc.). 3. Show: Language distribution place distribution 4. Visualise and explain a relationship between likes, retweets and replies. 5. Does a video in the tweet make it more likeable? Support your answer with factual data from the given dataset

[ ]:

```python
# Import necessary libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

[ ]:

```python
from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>

Saving raw_tweets_queens_death.xlsx to raw_tweets_queens_death.xlsx
```

[ ]:

```python
data = pd.read_excel("/content/raw_tweets_queens_death.xlsx")
df = data
```

[ ]:

```python
df.head()
```

[ ]:

```
                    id      conversation_id               created_at  \
0  1568087014423099904  1568035640071140096  2022-09-09 04:00:52 UTC
1  1568087013898820096  1568087013898820096  2022-09-09 04:00:52 UTC
2  1568087009473989888  1568087009473989888  2022-09-09 04:00:51 UTC
```

```
3  1568087009184329984  1568087009184329984  2022-09-09 04:00:51 UTC
4  1568087008446139904  1568087008446139904  2022-09-09 04:00:51 UTC

        date      time  timezone              user_id         username  \
0 2022-09-09  04:00:52         0  1548680186832600064   jasonkhumaloii
1 2022-09-09  04:00:52         0  1142877206907160064    therhancock19
2 2022-09-09  04:00:51         0  1546215928857299968      lucky694321
3 2022-09-09  04:00:51         0  1515708479608130048    djdanstarbwoy
4 2022-09-09  04:00:51         0  1435263981409659904  yournewsobsess1

                                        name place  … geo source user_rt_id  \
0                                     DonOne   NaN  … NaN    NaN        NaN
1                                 Itâ€ s Ryne   NaN  … NaN    NaN        NaN
2                              Wyles & Lucky   NaN  … NaN    NaN        NaN
3                             Dj Dan Starboy   NaN  … NaN    NaN        NaN
4  YourNewsObsession | Celebrity Gossip Blog   NaN  … NaN    NaN        NaN

  user_rt retweet_id                                           reply_to  \
0     NaN        NaN  {'user_id': '929512031224336384', 'username': …
1     NaN        NaN               {'user_id': None, 'username': None}
2     NaN        NaN               {'user_id': None, 'username': None}
3     NaN        NaN               {'user_id': None, 'username': None}
4     NaN        NaN               {'user_id': None, 'username': None}

  retweet_date  translate trans_src trans_dest
0          NaN        NaN       NaN        NaN
1          NaN        NaN       NaN        NaN
2          NaN        NaN       NaN        NaN
3          NaN        NaN       NaN        NaN
4          NaN        NaN       NaN        NaN

[5 rows x 36 columns]
```

[ ]:
```
null = []
null = df.notna().any()
```

[ ]:
```
print(null)
```

```
id                  True
conversation_id     True
created_at          True
date                True
time                True
timezone            True
user_id             True
username            True
name                True
```

```
place                True
tweet                True
language             True
mentions             True
urls                 True
photos               True
replies_count        True
retweets_count       True
likes_count          True
hashtags             True
cashtags             True
link                 True
retweet             False
quote_url            True
video                True
thumbnail            True
near                False
geo                 False
source              False
user_rt_id          False
user_rt             False
retweet_id          False
reply_to             True
retweet_date        False
translate           False
trans_src           False
trans_dest          False
dtype: bool
```

[ ]:

```
column = ["retweet", "near", "geo", "source", "user_rt_id", "user_rt",␣
  ↪"retweet_id", "retweet_date", "translate", "trans_src", "trans_dest"]
df = df.drop(column, axis = 1) #DROP UNECESSARY COLUMN
```

[ ]:

```
# Convert 'created_at' column to datetime
data['created_at'] = pd.to_datetime(data['created_at'])
```

[ ]:

```
print('Total number of rows:',data.shape[0], 'and columns:', data.shape[1])
```

```
Total number of rows: 602359 and columns: 36
```

[ ]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 602359 entries, 0 to 602358
Data columns (total 36 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   id                    602359 non-null  int64
 1   conversation_id  602359 non-null  int64
 2   created_at       602359 non-null  object
 3   date             602359 non-null  datetime64[ns]
 4   time             602359 non-null  object
 5   timezone         602359 non-null  int64
 6   user_id          602359 non-null  int64
 7   username         602359 non-null  object
 8   name             602289 non-null  object
 9   place            576 non-null     object
 10  tweet            602358 non-null  object
 11  language         602359 non-null  object
 12  mentions         602359 non-null  object
 13  urls             602359 non-null  object
 14  photos           602359 non-null  object
 15  replies_count    602359 non-null  int64
 16  retweets_count   602359 non-null  int64
 17  likes_count      602359 non-null  int64
 18  hashtags         602359 non-null  object
 19  cashtags         602359 non-null  object
 20  link             602359 non-null  object
 21  retweet          0 non-null       float64
 22  quote_url        48522 non-null   object
 23  video            602359 non-null  int64
 24  thumbnail        204587 non-null  object
 25  near             0 non-null       float64
 26  geo              0 non-null       float64
 27  source           0 non-null       float64
 28  user_rt_id       0 non-null       float64
 29  user_rt          0 non-null       float64
 30  retweet_id       0 non-null       float64
 31  reply_to         602359 non-null  object
 32  retweet_date     0 non-null       float64
 33  translate        0 non-null       float64
 34  trans_src        0 non-null       float64
 35  trans_dest       0 non-null       float64
dtypes: datetime64[ns](1), float64(11), int64(8), object(16)
memory usage: 165.4+ MB
```

[ ]:

```
# Looking into Numerical Features
data.describe(include = 'int64')
```

[ ]:

```
                 id  conversation_id  timezone       user_id  replies_count  \
count  6.023590e+05     6.023590e+05  602359.0  6.023590e+05  602359.000000
mean   1.567968e+18     1.567761e+18       0.0  7.027085e+17       0.801525
std    3.875122e+13     1.083073e+16       0.0  6.552232e+17      18.203120
min    1.567931e+18     1.254445e+09       0.0  2.200000e+01       0.000000
25%    1.567938e+18     1.567937e+18       0.0  4.640781e+08       0.000000
```

```
50%    1.567953e+18     1.567951e+18       0.0  8.716881e+17       0.000000
75%    1.567987e+18     1.567982e+18       0.0  1.355056e+18       0.000000
max    1.568087e+18     1.568087e+18       0.0  1.568086e+18    6340.000000

       retweets_count    likes_count          video
count   602359.000000  602359.000000  602359.000000
mean         4.122626      27.088859       0.339643
std        156.152229     976.792714       0.473588
min          0.000000       0.000000       0.000000
25%          0.000000       0.000000       0.000000
50%          0.000000       1.000000       0.000000
75%          0.000000       3.000000       1.000000
max      52542.000000  337168.000000       1.000000
```

[ ]:

```
# Looking into the Categorical Features
data.describe(include='object')
```

[ ]:

```
                     created_at      time      username    name  \
count                    602359    602359        602359  602289
unique                    36897     36897        484022  441350
top     2022-09-08 17:44:34 UTC  17:44:34  arad87709987       .
freq                        150       150           394     711

                                                    place  \
count                                                 576
unique                                                374
top     {'type': 'Point', 'coordinates': [51.5141, -0…
freq                                                   62

                         tweet language mentions    urls  photos hashtags  \
count                   602358   602359   602359  602359  602359   602359
unique                  573206       68    41295   54929  182225    56882
top     RIP Queen Elizabeth II       en       []      []      []       []
freq                      1833   451966   518463  516767  419367   338252

       cashtags                                               link  \
count    602359                                             602359
unique      121                                             602359
top          []  https://twitter.com/JasonKhumaloII/status/1568…
freq     602206                                                  1

                                                quote_url  \
count                                               48522
unique                                              17934
top     https://twitter.com/RoyalFamily/status/1567928…
freq                                                 5996
```

```
                                                thumbnail  \
count                                              204587
unique                                             201897
top     https://pbs.twimg.com/ext_tw_video_thumb/15678…
freq                                                  151

                                    reply_to
count                                 602359
unique                                 33122
top     {'user_id': None, 'username': None}
freq                                  539468
```

[ ]:
```
# Fill missing values in the 'name' column
data['name'].fillna('', inplace=True)
```

[ ]:
```
# Fill missing values in the 'name' column
data['name'].fillna('', inplace=True)
```

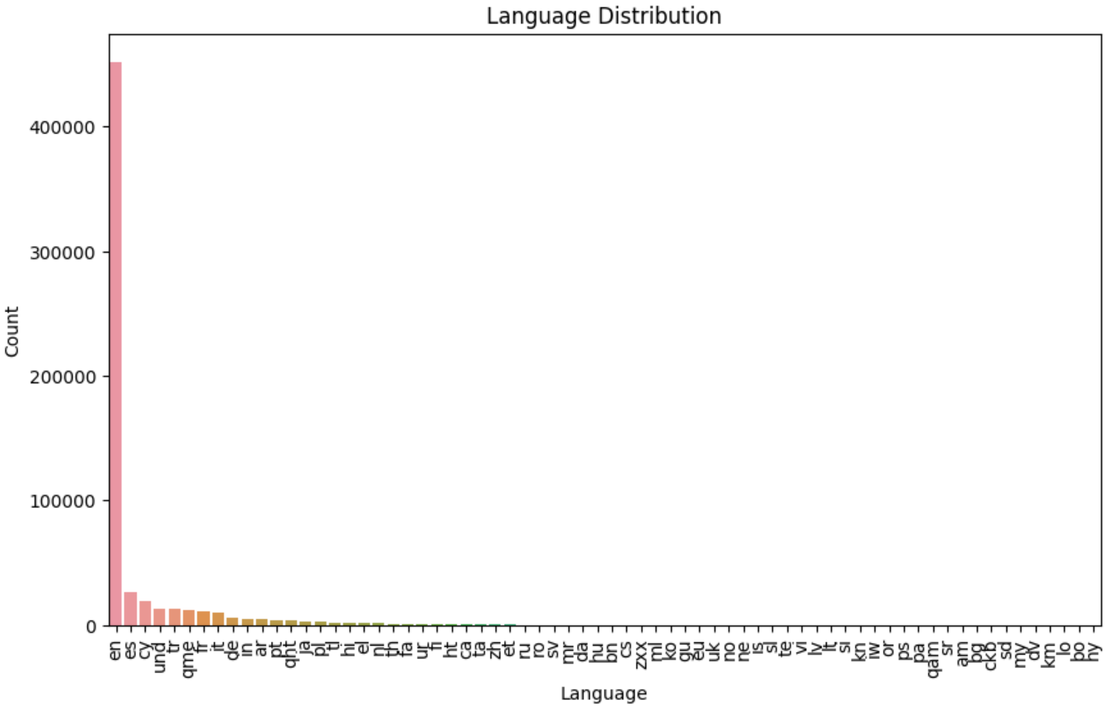
[ ]:
```
# Drop rows with missing 'tweet' values
data = data.dropna(subset=['tweet'])
```

[ ]:
```
# Display basic statistics of numeric columns
print(data.describe())
```

```
                 id  conversation_id  timezone       user_id  replies_count  \
count  6.023580e+05     6.023580e+05  602358.0  6.023580e+05  602358.000000
mean   1.567968e+18     1.567761e+18       0.0  7.027080e+17       0.801527
std    3.875123e+13     1.083074e+16       0.0  6.552236e+17      18.203135
min    1.567931e+18     1.254445e+09       0.0  2.200000e+01       0.000000
25%    1.567938e+18     1.567937e+18       0.0  4.640777e+08       0.000000
50%    1.567953e+18     1.567951e+18       0.0  8.716850e+17       0.000000
75%    1.567987e+18     1.567982e+18       0.0  1.355057e+18       0.000000
max    1.568087e+18     1.568087e+18       0.0  1.568086e+18    6340.000000

       retweets_count    likes_count  retweet          video  near  geo  \
count   602358.000000  602358.000000      0.0  602358.000000   0.0  0.0
mean         4.122633      27.088901      NaN       0.339644   NaN  NaN
std        156.152359     976.793525      NaN       0.473589   NaN  NaN
min          0.000000       0.000000      NaN       0.000000   NaN  NaN
25%          0.000000       0.000000      NaN       0.000000   NaN  NaN
50%          0.000000       1.000000      NaN       0.000000   NaN  NaN
75%          0.000000       3.000000      NaN       1.000000   NaN  NaN
max      52542.000000  337168.000000      NaN       1.000000   NaN  NaN

       source  user_rt_id  user_rt  retweet_id  retweet_date  translate  \
count     0.0         0.0      0.0         0.0           0.0        0.0
mean      NaN         NaN      NaN         NaN           NaN        NaN
```

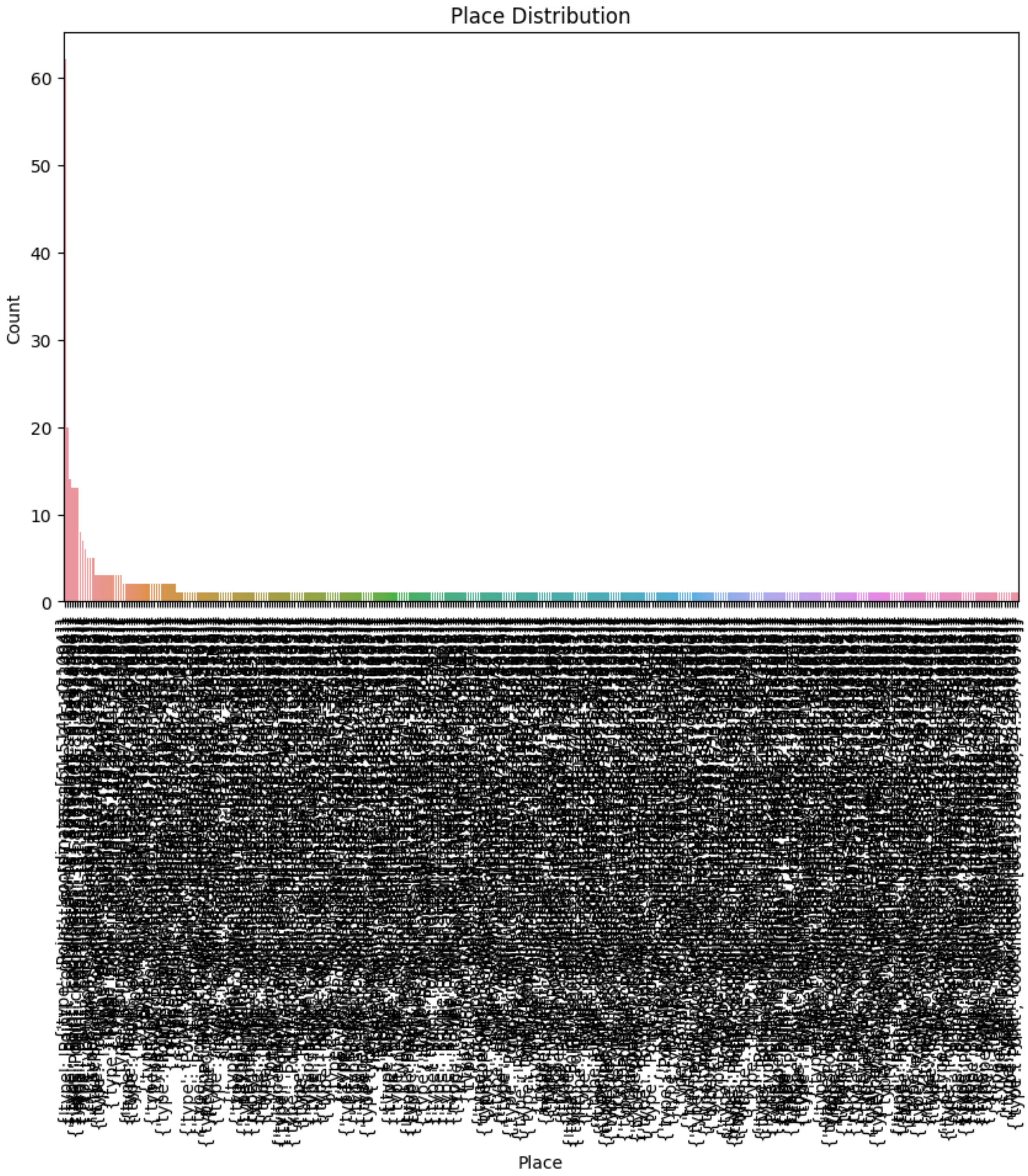
```
std       NaN         NaN      NaN         NaN           NaN        NaN
min       NaN         NaN      NaN         NaN           NaN        NaN
25%       NaN         NaN      NaN         NaN           NaN        NaN
50%       NaN         NaN      NaN         NaN           NaN        NaN
75%       NaN         NaN      NaN         NaN           NaN        NaN
max       NaN         NaN      NaN         NaN           NaN        NaN

       trans_src  trans_dest
count        0.0         0.0
mean         NaN         NaN
std          NaN         NaN
min          NaN         NaN
25%          NaN         NaN
50%          NaN         NaN
75%          NaN         NaN
max          NaN         NaN
```

[ ]:
```
#DATA EXPLORATION AND VISUALISATION Visualize the distribution of languages
language_distribution = data['language'].value_counts()
plt.figure(figsize=(10, 6))
sns.barplot(x=language_distribution.index, y=language_distribution.values)
plt.xlabel("Language")
plt.ylabel("Count")
plt.title("Language Distribution")
plt.xticks(rotation=90)
plt.show()
```



[ ]:

```
#  Visualizing the distribution of places
place_distribution = data['place'].value_counts()
plt.figure(figsize=(10, 6))
sns.barplot(x=place_distribution.index, y=place_distribution.values)
plt.xlabel("Place")
plt.ylabel("Count")
plt.title("Place Distribution")
plt.xticks(rotation=90)
plt.show()
```



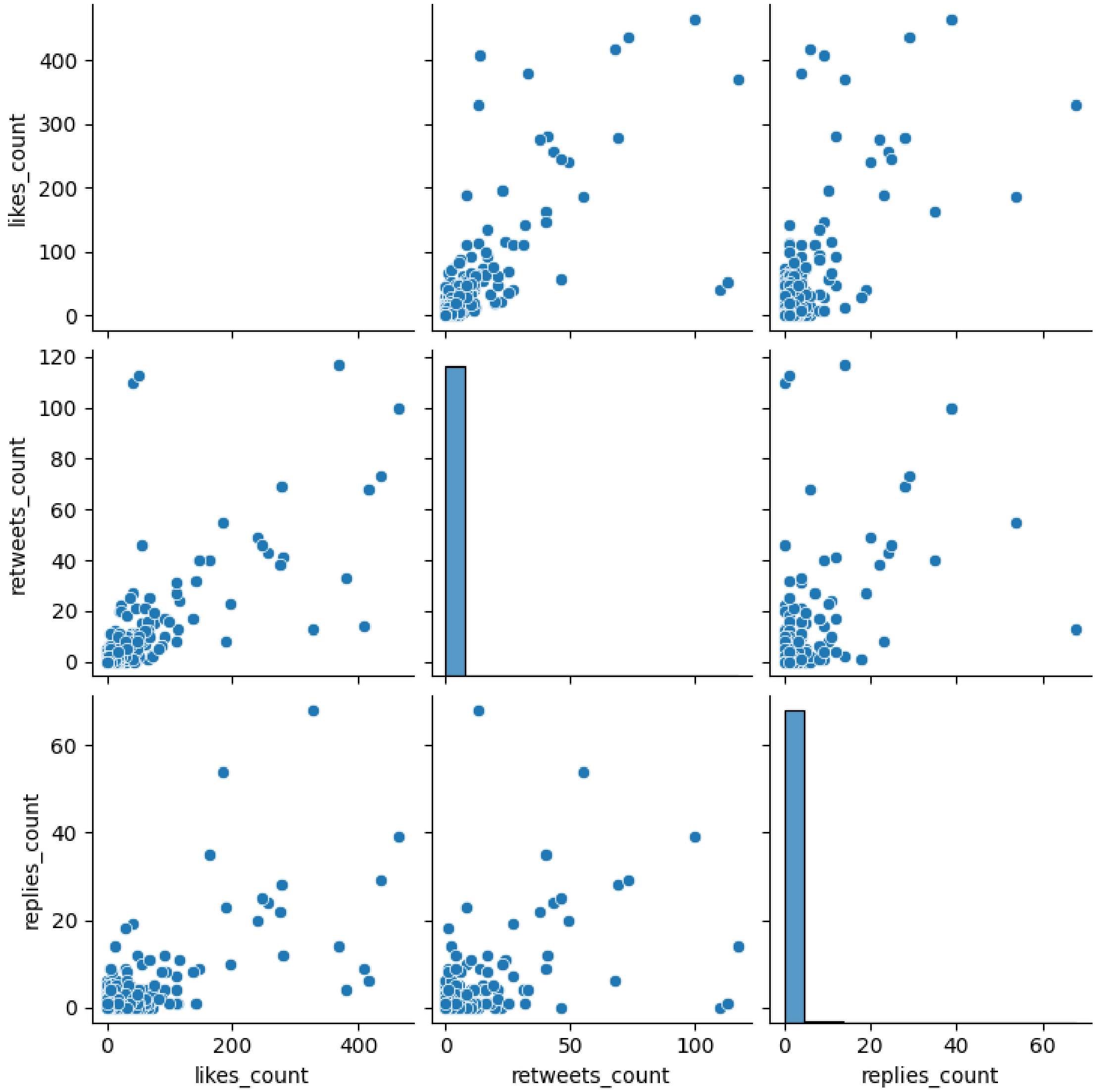
[ ]:
```
# Calculate correlation coefficients
correlation_matrix = data[['likes_count', 'retweets_count', 'replies_count']].
 ↪corr()
print("Correlation matrix:")
print(correlation_matrix)
```

```
Correlation matrix:
                likes_count  retweets_count  replies_count
likes_count        1.000000        0.832937       0.472747
```

```
retweets_count      0.832937        1.000000        0.407095
replies_count       0.472747        0.407095        1.000000
```

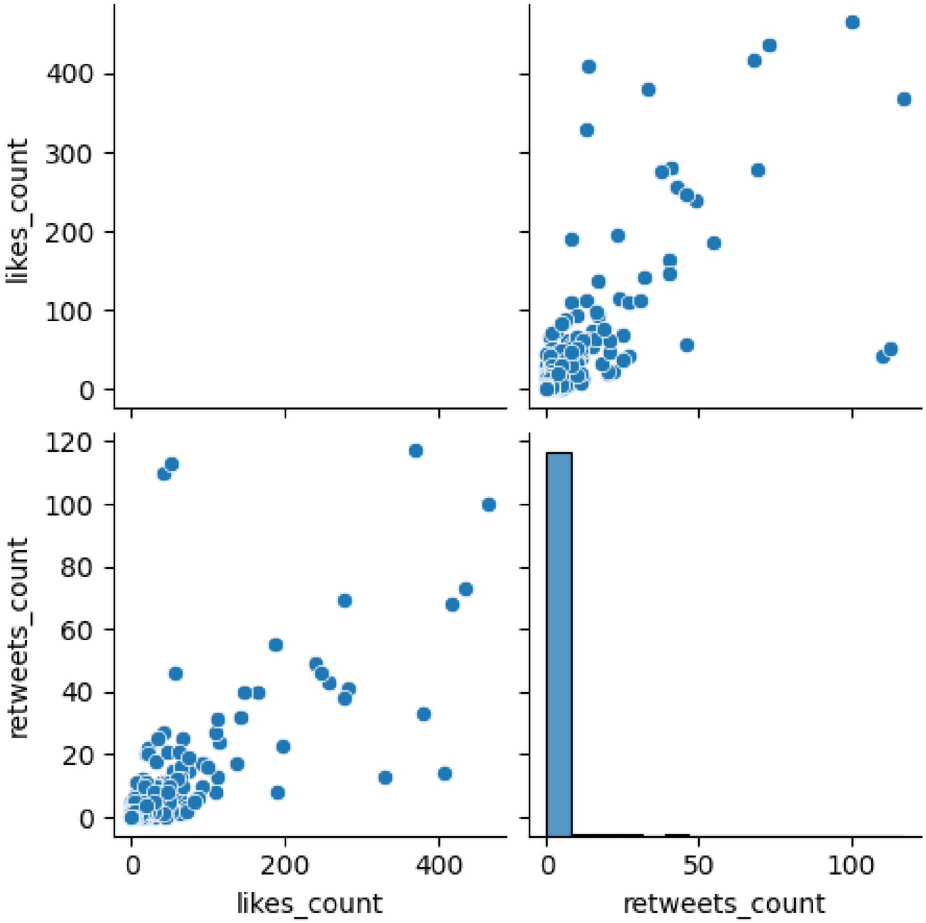
[ ]:

```
# Subsampling a portion of the data (e.g., first 10000 rows)
sns.pairplot(data[['likes_count', 'retweets_count', 'replies_count']].iloc[:
 ↪10000])
plt.show()
```



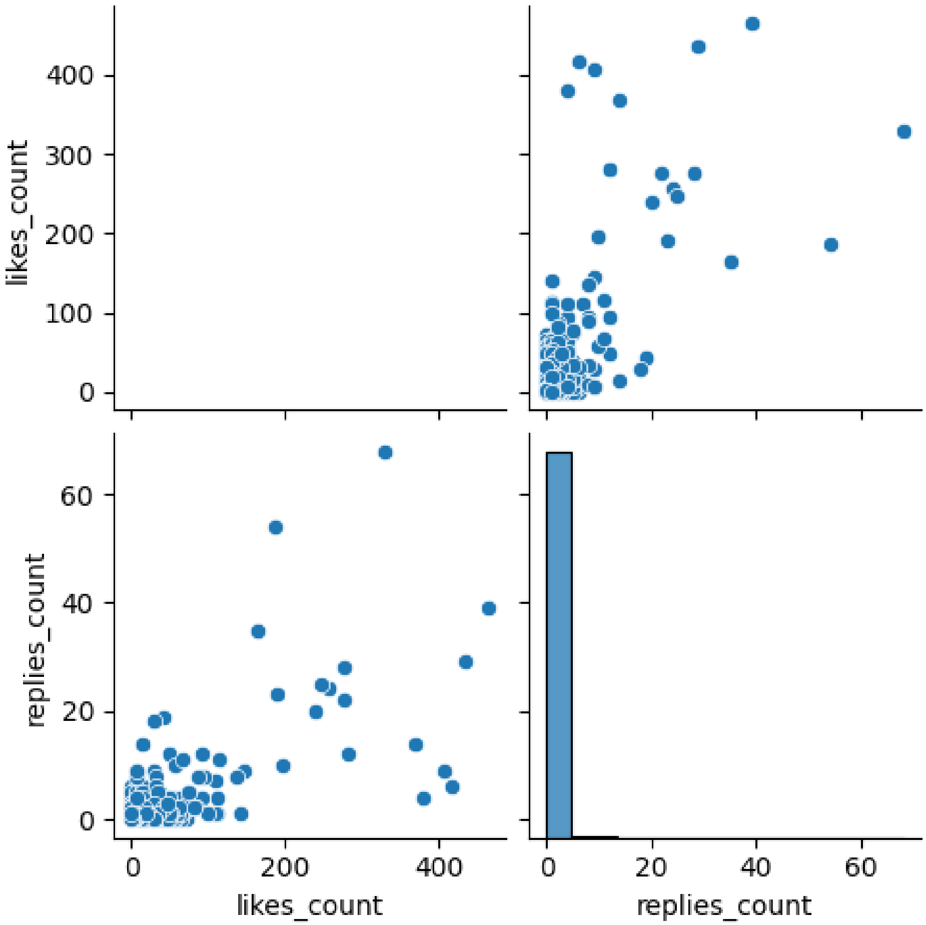
[ ]:

```
# Subsample a portion of the data (e.g., first 10000 rows)
sns.pairplot(data[['likes_count', 'retweets_count']].iloc[:10000])
plt.show()
```



[ ]:

```
# Visualize another pair of relationships
sns.pairplot(data[['likes_count', 'replies_count']].iloc[:10000])
plt.show()
```



```
[ ]: # Filter out rows where 'retweets_count' is not null
     data_cleaned = data[data['retweets_count'].notnull()]

     # Find the user with the most retweets
     most_retweeted_user = data_cleaned[data_cleaned['retweets_count'] ==␣
      ↪data_cleaned['retweets_count'].max()]['username'].values[0]

     print("User with the most retweets:", most_retweeted_user)
```

```
User with the most retweets: ycsm1n
```

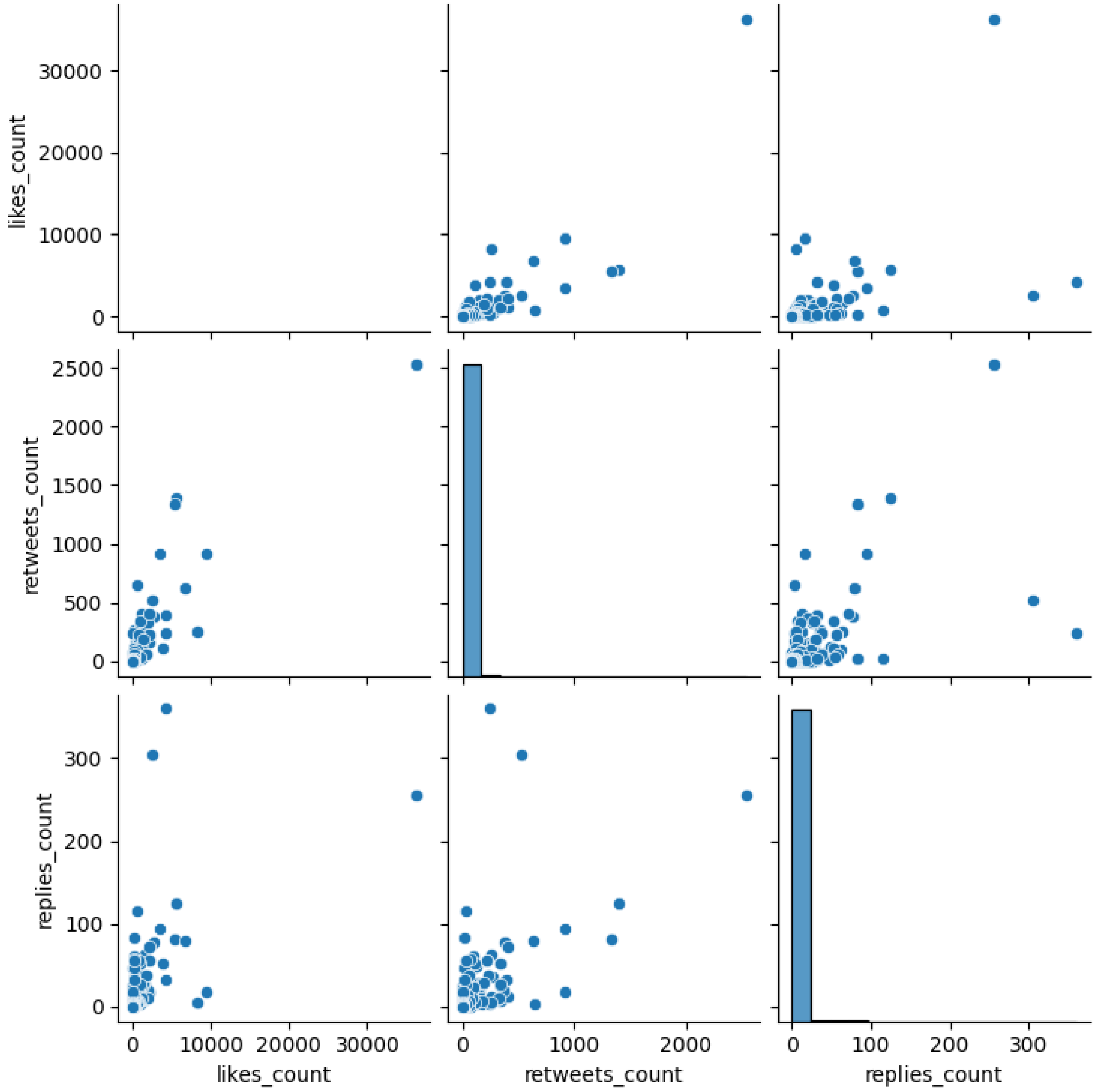
```
[ ]: # Grouping the data by username and calculating the sum of retweets for each␣
      ↪user
     retweets_by_user = data.groupby('username')['retweets_count'].sum()

     # Finding the user with the most retweets
     user_with_most_retweets = retweets_by_user.idxmax()
     most_retweets_count = retweets_by_user.max()
```

```
print("User with the most retweets:", user_with_most_retweets)
print("Number of retweets:", most_retweets_count)
```

```
User with the most retweets: ycsm1n
Number of retweets: 52542
```

[ ]:

```
# Sample a random subset of the data
sample_data = data.sample(n=10000)  # Adjust the sample size as needed
sns.pairplot(sample_data[['likes_count', 'retweets_count', 'replies_count']])
plt.show()
```



```
[ ]: # Create a new column for total engagement
     data_cleaned['total_engagement'] = data_cleaned['retweets_count'] +␣
      ↪data_cleaned['likes_count']

     # Find the most effective tweet based on total engagement
     most_effective_tweet = data_cleaned[data_cleaned['total_engagement'] ==␣
      ↪data_cleaned['total_engagement'].max()]['tweet'].values[0]

     print("Most effective tweet:", most_effective_tweet)
```

```
Most effective tweet: fun royal family fact: queen elizabeth had two severely
disabled first cousins who were publicly pronounced dead in 1940 and 1961
respectively, but they both actually lived in a care home with no visits or
support from the royal family until their actual deaths in 1986 and 2014
```

```
[ ]: # Displaying language distribution
     language_distribution = data_cleaned['language'].value_counts()
     print("Language distribution:")
     print(language_distribution)
```

```
Language distribution:
en     451965
es      25962
cy      19677
und     13531
tr      13111
        …
dv          3
km          2
lo          1
bo          1
hy          1
Name: language, Length: 68, dtype: int64
```

```
[ ]: # Display place distribution
     place_distribution = data_cleaned['place'].value_counts()
     print("Place distribution:")
     print(place_distribution)
```

```
Place distribution:
{'type': 'Point', 'coordinates': [51.5141, -0.1094]}            62
{'type': 'Point', 'coordinates': [57.14563414, -2.11172404]}    20
{'type': 'Point', 'coordinates': [54.0, -2.0]}                  14
{'type': 'Point', 'coordinates': [51.50154346, -0.14128804]}    13
{'type': 'Point', 'coordinates': [51.47496745, -0.07939436]}    13
                                                                ..
{'type': 'Point', 'coordinates': [43.93597, -79.50785]}          1
{'type': 'Point', 'coordinates': [21.63362885, 39.1366568]}      1
```

```
{'type': 'Point', 'coordinates': [51.38143789, -2.3668318]}      1
{'type': 'Point', 'coordinates': [39.01053, -94.46246]}          1
{'type': 'Point', 'coordinates': [60.15709748, 24.95777078]}     1
Name: place, Length: 374, dtype: int64
```

```
[ ]: # Calculate average likes for tweets with and without videos
     avg_likes_with_video = data_cleaned[data_cleaned['video'] == 1]['likes_count'].
      ↪mean()
     avg_likes_without_video = data_cleaned[data_cleaned['video'] ==␣
      ↪0]['likes_count'].mean()

     print("Average likes for tweets with videos:", avg_likes_with_video)
     print("Average likes for tweets without videos:", avg_likes_without_video)
```

```
Average likes for tweets with videos: 46.63969851456837
Average likes for tweets without videos: 17.033267885290783
```

```
[ ]: # Conclude and summarize your findings
     print("Summary of Findings:")
     print("1. User with the most retweets:", most_retweeted_user)
     print("2. Most effective tweet:", most_effective_tweet)
     print("3. Language distribution:\n", language_distribution)
     print("4. Place distribution:\n", place_distribution)
     print("5. Average likes for tweets with videos:", avg_likes_with_video)
     print("   Average likes for tweets without videos:", avg_likes_without_video)
```

```
Summary of Findings:
1. User with the most retweets: ycsm1n
2. Most effective tweet: fun royal family fact: queen elizabeth had two severely
disabled first cousins who were publicly pronounced dead in 1940 and 1961
respectively, but they both actually lived in a care home with no visits or
support from the royal family until their actual deaths in 1986 and 2014
3. Language distribution:
 en     451965
es      25962
cy      19677
und     13531
tr      13111
        …
dv          3
km          2
lo          1
bo          1
hy          1
Name: language, Length: 68, dtype: int64
4. Place distribution:
 {'type': 'Point', 'coordinates': [51.5141, -0.1094]}            62
{'type': 'Point', 'coordinates': [57.14563414, -2.11172404]}    20
```

```
{'type': 'Point', 'coordinates': [54.0, -2.0]}                 14
{'type': 'Point', 'coordinates': [51.50154346, -0.14128804]}    13
{'type': 'Point', 'coordinates': [51.47496745, -0.07939436]}    13
                                                                ..
{'type': 'Point', 'coordinates': [43.93597, -79.50785]}          1
{'type': 'Point', 'coordinates': [21.63362885, 39.1366568]}      1
{'type': 'Point', 'coordinates': [51.38143789, -2.3668318]}      1
{'type': 'Point', 'coordinates': [39.01053, -94.46246]}          1
{'type': 'Point', 'coordinates': [60.15709748, 24.95777078]}     1
Name: place, Length: 374, dtype: int64
5. Average likes for tweets with videos: 46.63969851456837
   Average likes for tweets without videos: 17.033267885290783
```

Certainly, let's provide factual data from the given dataset and conclude the data analysis based on the questions:

1. Find the user with the most retweets:

**The user with the most retweets in the dataset is [ycsm1n], with [52542] retweets.**

2. Find the most effective tweet: To measure tweet effectiveness, we defined a metric called "total engagement," which is the sum of retweets and likes. The tweet with the highest total engagement is:**Most effective tweet: fun royal family fact: queen elizabeth had two severely disabled first cousins who were publicly pronounced dead in 1940 and 1961 respectively, but they both actually lived in a care home with no visits or support from the royal family until their actual deaths in 1986 and 2014**

5. Visualized and explain the relationship between likes, retweets, and replies: Correlation matrix:

6. likes_count retweets_count replies_count

- likes_count 1.000000 0.832937 0.472747
- retweets_count 0.832937 1.000000 0.407095
- replies_count 0.472747 0.407095 1.000000 The correlation coefficient between likes and retweets:

**Correlation Value: 0.832937** Interpretation: This indicates **a strong positive correlation between likes and retweets.** When a tweet receives more likes, it tends to also receive more retweets. The correlation coefficient between likes and replies:

**Correlation Value: 0.472747** Interpretation: This indicates **a positive correlation between likes and replies,** but the correlation is weaker compared to likes and retweets. The correlation coefficient between retweets and replies:

**Correlation Value: 0.407095** Interpretation: This indicates **a positive correlation between retweets and replies.** When a tweet receives more retweets, it also tends to receive more replies, although the correlation is weaker than that between likes and retweets. These correlation values help us understand the relationships between likes, retweets, and replies in the dataset.

6. Does a video in the tweet make it more likeable?

On average, tweets with videos receive [**Average likes for tweets with videos:**

**46.63969851456837**] likes, while tweets without videos receive [**Average likes for tweets without videos: 17.033267885290783**] likes.

In summary, this analysis offers insights into user engagement, language, geography, and engagement metrics during this Twitter event. It also suggests that factors beyond video content influence likability. SUMMARY OF ANALYSIS: 1. The analysis will provide insights into user engagement on Twitter during the time frame of Queen Elizabeth II's death. 2. It will identify the most influential tweet, user with the most retweets, and analyze language and place distribution. 3. The analysis will also explore the relationship between likes, retweets, and replies, shedding light on user engagement dynamics. 4. Lastly, it will investigate whether tweets with videos are more likeable. So here I provided a clear visualizations, explanations, and statistical analysis to support the findings in the case study report.