```
!pip install snscrape
```

```
import snscrape.modules.twitter as sntwitter
import pandas as pd
```

```
!pip install transformers
```

```
!pip install scipy
```

```
#Extracting tweets that include 'Graham Potter' btween 29 August and 11th Sepetember

query = "Graham Potter lang:en until:2022-09-11 since:2022-08-29"
tweets = []
limit = 100000
for tweet in sntwitter.TwitterSearchScraper(query).get_items():
  if len(tweets) == limit:
      break
  else:
    tweets.append([tweet.date, tweet.username, tweet.content])

df = pd.DataFrame(tweets, columns=['Date', 'User', 'Tweet'])
print(df.head())

df.to_csv('tweets.csv')
```

```
df['Label']=''
df['Score']=''
```

```
#In as much as out limit was 100000 tweets, the total tweets extracted was 48458 tweets
df.tail()
```

```
import datetime
from datetime import datetime
```

```
#Running the converted date so as to extract just the date without the time

df['New_Date']=pd.to_datetime(df['Date'],format="%Y-%m-%d %H:%M:%S")
df['New_Date'] = [d.date() for d in df['New_Date']]
df.head()
```

```
del df['Date']
df.tail()
```

```
from transformers import AutoTokenizer, AutoModelForSequenceClassification
from scipy.special import softmax
```

```
The cache for model files in Transformers v4.22.0 has been updated. Migrating your o
Moving 0 files to the new cache system
```

```
0/0 [00:00<?, ?it/s]
```

```
roberta = "cardiffnlp/twitter-roberta-base-sentiment"

model = AutoModelForSequenceClassification.from_pretrained(roberta)
tokenizer = AutoTokenizer.from_pretrained(roberta)

labels = ['Negative', 'Neutral', 'Positive']
```

```
#Iterating throught each tweet to detrmine what is a username and a url
for x in range(len(df)):
  tweet = df['Tweet'][x]

  tweet_words = []

  for word in tweet.split(' '):
    if word.startswith('@') and len(word) > 1:
        word = '@user'

    elif word.startswith('http'):
        word = "http"
    tweet_words.append(word)

    tweet_proc = " ".join(tweet_words)
    df['Tweet'][x]=tweet_proc

  # sentiment analysis on each tweet
  encoded_tweet = tokenizer(tweet_proc, return_tensors='pt')

  output = model(**encoded_tweet)

  scores = output[0][0].detach().numpy()
  scores = softmax(scores)

  # saving the hihest score for each tweet in the dataframe along with its label
  max_scores = 0
  for i in range(len(scores)):
    if scores[i]>max_scores:
      max_scores= scores[i]
      label = labels[i]
  df['Label'][x]=label
  df['Score'][x]=max_scores
```

```
df['Label'].describe()
```

```
#This was to check for any null values
df.info()
```

```
#Finally saving the dataframe as a csv file to be create a visualisation in tableau
df.to_csv('tweets_fin.csv')
```

```
#Project aided by tutorials from Mehranshakarami on youtube
```