NAME : G.SRIKAR

REGISTRATION NO. : 19BCE7081

SUBJECT : Advanced Data Analytics

(CSE-4029)

SLOT : L21 + L22

** *Lab Assignment-4* **

Twitter Data Analysis

AIM: Ukraine Russia War Twitter data Sentiment Analysis using Python

Reading & Understanding the Data :

PYTHON-CODE :-

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
import nltk
import re
from nltk.corpus import stopwords
import string


data = pd.read_csv("filename.csv")
print(data.head())
```

OUTPUT :-

```
/usr/local/lib/python3.7/dist-packages/nltk/twitter/__init__.py:20: UserWarning: The twyth
  warnings.warn("The twython library has not been installed. "
                   id      conversation_id               created_at  \
0  1508562464916582400  1508532968867909640  2022-03-28 21:51:34 UTC
1  1508562464476209154  1508562464476209154  2022-03-28 21:51:33 UTC
2  1508562460445454338  1508562460445454338  2022-03-28 21:51:32 UTC
3  1508562459476561926  1508562459476561926  2022-03-28 21:51:32 UTC
4  1508562459149451265  1508460376924557316  2022-03-28 21:51:32 UTC

         date      time  timezone             user_id        username  \
0  2022-03-28  21:51:34         0            91818503     ttmionzinho
1  2022-03-28  21:51:33         0           305508890  gianluca_rubeo
2  2022-03-28  21:51:32         0  1490181377588871169         ul_mika
3  2022-03-28  21:51:32         0          2591497068    canadageneva
4  2022-03-28  21:51:32         0  1279828257693732870        beletifk

                           name place  ... geo source user_rt_id user_rt  \
0                   Victor Coelho   NaN  ... NaN    NaN        NaN     NaN
1                  Gianluca Rubeo   NaN  ... NaN    NaN        NaN     NaN
2  MIKA UL YARMYAOHUH HAOLULYAOHUH   NaN  ... NaN    NaN        NaN     NaN
3              Canada in Geneva 🍁   NaN  ... NaN    NaN        NaN     NaN
4                              ☸   NaN  ... NaN    NaN        NaN     NaN

  retweet_id                                           reply_to  retweet_date  \
0        NaN  [{'screen_name': 'CarlosPortSPFC', 'name': 'Ca...           NaN
1        NaN                                                 []           NaN
2        NaN                                                 []           NaN
3        NaN                                                 []           NaN
4        NaN  [{'screen_name': 'Amb_Ulyanov', 'name': 'Mikha...           NaN
```

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

[5 rows x 36 columns]
```

Printing all the column names of the dataset:

- **PYTHON-CODE :-**

```
print(data.columns)
```

- **OUTPUT :-**

```
print(data.columns)

Index(['id', 'conversation_id', 'created_at', 'date', 'time', 'timezone',
       'user_id', 'username', 'name', 'place', 'tweet', 'language', 'mentions',
       'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count',
       'hashtags', 'cashtags', 'link', 'retweet', 'quote_url', 'video',
       'thumbnail', 'near', 'geo', 'source', 'user_rt_id', 'user_rt',
       'retweet_id', 'reply_to', 'retweet_date', 'translate', 'trans_src',
       'trans_dest'],
      dtype='object')
```

Need three columns for this task (username, tweet, and language); I will only select these columns and move forward:

- **PYTHON-CODE :-**

```
data = data[["username", "tweet", "language"]]
```

Cheking whether any of these columns contains any null values or not:

- **PYTHON-CODE :**

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

- **OUTPUT :-**

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

username    0
tweet       0
language    0
dtype: int64
```

So none of the columns has null values, checking for how many tweets are posted in each language:

PYTHON-CODE :-

```
data["language"].value_counts()
```

OUTPUT :-

```
data["language"].value_counts()

en     8761
it      382
pt      324
und     194
ru       53
in       39
fr       38
es       31
de       19
ca       19
ja       18
tr       16
nl       15
pl       13
ar       13
hi       10
cs        9
tl        7
et        6
uk        6
fi        5
cy        4
ro        4
zh        3
```

```
sr          3
hu          2
ht          1
no          1
ko          1
da          1
lv          1
bg          1
ur          1
Name: language, dtype: int64
```

Most of the tweets are in English. Preparing this data for the task of sentiment analysis. Here I will remove all the links, punctuation, symbols and other language errors from the tweets:

PYTHON-CODE :

```
nltk.download('stopwords')
stemmer = nltk.SnowballStemmer("english")
stopword=set(stopwords.words('english'))


def clean(text):
    text = str(text).lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    text = [word for word in text.split(' ') if word not in stopword]
    text=" ".join(text)
    text = [stemmer.stem(word) for word in text.split(' ')]
```

```
    text=" ".join(text)
    return text
data["tweet"] = data["tweet"].apply(clean)
```

OUTPUT :-

```
nltk.download('stopwords')
stemmer = nltk.SnowballStemmer("english")
stopword=set(stopwords.words('english'))

def clean(text):
    text = str(text).lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    text = [word for word in text.split(' ') if word not in stopword]
    text=" ".join(text)
    text = [stemmer.stem(word) for word in text.split(' ')]
    text=" ".join(text)
    return text
data["tweet"] = data["tweet"].apply(clean)
```

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

looking at the wordcloud of the tweets, which will show the most frequently used words in the tweets by people sharing their feelings and updates about the Ukraine and Russia war.

PYTHON-CODE :

```
text = " ".join(i for i in data.tweet)
stopwords = set(STOPWORDS)
wordcloud = WordCloud(stopwords=stopwords,
background_color="white").generate(text)
plt.figure( figsize=(15,10))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

OUTPUT :

```
text = " ".join(i for i in data.tweet)
stopwords = set(STOPWORDS)
wordcloud = WordCloud(stopwords=stopwords, background_color="white").generate(text)
plt.figure( figsize=(15,10))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

PYTHON-CODE :

```
nltk.download('vader_lexicon')
sentiments = SentimentIntensityAnalyzer()
data["Positive"] = [sentiments.polarity_scores(i)["pos"] for i in data["tweet"]]
data["Negative"] = [sentiments.polarity_scores(i)["neg"] for i in data["tweet"]]
data["Neutral"] = [sentiments.polarity_scores(i)["neu"] for i in data["tweet"]]
data = data[["tweet", "Positive", "Negative", "Neutral"]]
print(data.head())
```

OUTPUT :

```
nltk.download('vader_lexicon')
sentiments = SentimentIntensityAnalyzer()
data["Positive"] = [sentiments.polarity_scores(i)["pos"] for i in data["tweet"]]
data["Negative"] = [sentiments.polarity_scores(i)["neg"] for i in data["tweet"]]
data["Neutral"] = [sentiments.polarity_scores(i)["neu"] for i in data["tweet"]]
data = data[["tweet", "Positive", "Negative", "Neutral"]]
print(data.head())
```

```
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
[nltk_data]   Package vader_lexicon is already up-to-date!
                                               tweet  Positive  Negative  \
0  carlosportspfc ministro stf que seu mito coloc...       0.0       0.0
1  la russia inizia la ritirata intorno kiev russ...       0.0       0.0
2  presid joe biden visit europ last week europea...       0.0       0.0
3  absent russia militari action foodsecur would ...       0.0       0.0
4  ambulyanov russia must provid annual money rus...       0.0       0.0

   Neutral
0      1.0
1      1.0
2      1.0
3      1.0
4      1.0
```

Looking at the most frequent words used by people with positive sentiments:

PYTHON-CODE :

```
positive =' '.join([i for i in data['tweet'][data['Positive'] > data["Negative"]]])

stopwords = set(STOPWORDS)

wordcloud = WordCloud(stopwords=stopwords,
background_color="white").generate(positive)

plt.figure( figsize=(15,10))

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis("off")

plt.show()
```

OUTPUT :

```
positive =' '.join([i for i in data['tweet'][data['Positive'] > data["Negative"]]])
stopwords = set(STOPWORDS)
wordcloud = WordCloud(stopwords=stopwords, background_color="white").generate(positive)
plt.figure( figsize=(15,10))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

Looking at the most frequent words used by people with negative sentiments:

PYTHON-CODE :

```
negative =' '.join([i for i in data['tweet'][data['Negative'] > data["Positive"]]])

stopwords = set(STOPWORDS)

wordcloud = WordCloud(stopwords=stopwords,
background_color="white").generate(negative)

plt.figure( figsize=(15,10))

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis("off")

plt.show()
```

OUTPUT :

```
negative =' '.join([i for i in data['tweet'][data['Negative'] > data["Positive"]]])
stopwords = set(STOPWORDS)
wordcloud = WordCloud(stopwords=stopwords, background_color="white").generate(negative)
plt.figure( figsize=(15,10))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```