STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

Author:- *Vinay Rajesh Gor*

CWID:- *10446180*

Subject:- *EM-626 Applied AI/ML for Systems & Enterprises*

Topic:- *Analyze tweets*

Instructor:- *Dr. Carlo Lipizzi*

Analyze Tweets

Introduction

The tweets.csv file contains 20,000 tweets composed by Sender, Timestamp, and Text. Tweets have been collected during a recent presidential debate, using “trump” as a keyword. That means they are most likely related to people's reaction to Trump speech during the debate.

Project Goals

The purpose of the project is to analyze,

1. Analyze the 5 most active senders
2. Analyze the 10 most retweeted tweets
3. Analyze the 5 most cited screen-names
4. Analyze the most 10 popular hashtags words
5. Create Word Cloud for 5 subset

Business Understanding

The objective of this study is to analyze *tweets containing keyword trump* as a *Data Scientist* for the Government to determine the general sentiment of tweets and to visualize the time series virtual conversation happened to the Trump speech during the debate.

The data set will be extracted from a csv file, will be imported to a *Python* file and will be analyzed using the combination of libraries such as *Pandas, Numpy, Matplotlib.py and Wordcloud*. The project is implemented in *Spyder (Anaconda)*

Data Understanding

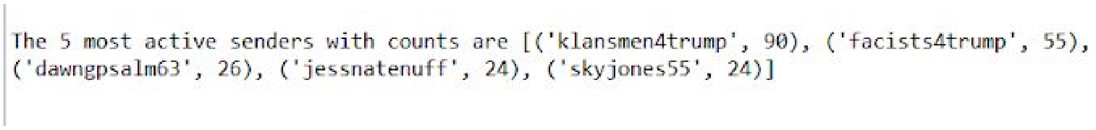
To understand the data and its characteristics the data was first explored in it’s basic format .csv (comma separated values). Using the *sheet* and visualizing it was explored that the dataset contains *2000 rows* and *3 columns*.

Data Preparation

Initially data is read in .csv format. The columns are created as ‘Sender’, ‘Time’ and ‘Tweet’ which are then converted to list. These are called as list_sender, list_time and list_tweets. The screen name, hashtags, list_split, str1, str2, str2, str3 etc. are also some variables which are created for convenience which will be used in program for various operations.

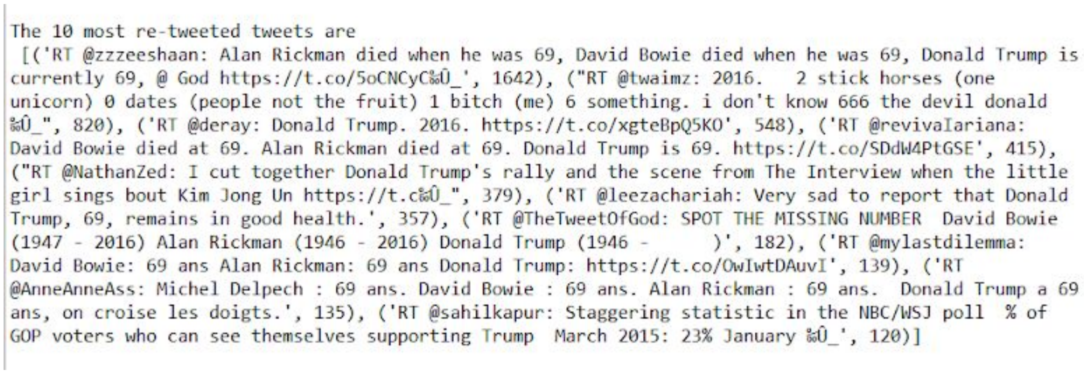
Results

The 5 most active senders



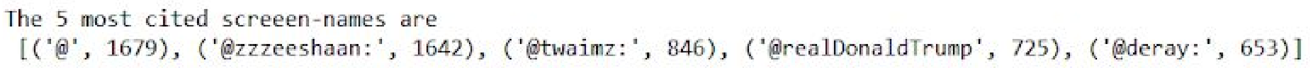
```
The 5 most active senders with counts are [('klansmen4trump', 90), ('facists4trump', 55),
('dawngpsalm63', 26), ('jessnatenuff', 24), ('skyjones55', 24)]
```

The 10 most retweeted tweets



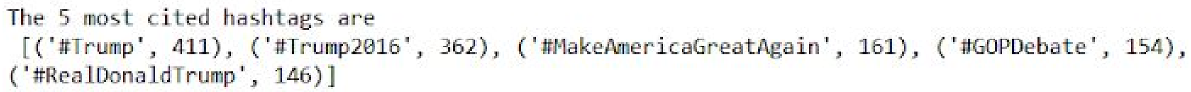
```
The 10 most re-tweeted tweets are
 [('RT @zzzeeshaan: Alan Rickman died when he was 69, David Bowie died when he was 69, Donald Trump is
currently 69, @ God https://t.co/5oCNCyC‰Û_', 1642), ("RT @twaimz: 2016.   2 stick horses (one
unicorn) 0 dates (people not the fruit) 1 bitch (me) 6 something. i don't know 666 the devil donald
‰Û_", 820), ('RT @deray: Donald Trump. 2016. https://t.co/xgteBpQ5KO', 548), ('RT @revivalariana:
David Bowie died at 69. Alan Rickman died at 69. Donald Trump is 69. https://t.co/SDdW4PtGSE', 415),
("RT @NathanZed: I cut together Donald Trump's rally and the scene from The Interview when the little
girl sings bout Kim Jong Un https://t.c‰Û_", 379), ('RT @leezachariah: Very sad to report that Donald
Trump, 69, remains in good health.', 357), ('RT @TheTweetOfGod: SPOT THE MISSING NUMBER  David Bowie
(1947 - 2016) Alan Rickman (1946 - 2016) Donald Trump (1946 -      )', 182), ('RT @mylastdilemma:
David Bowie: 69 ans Alan Rickman: 69 ans Donald Trump: https://t.co/OwIwtDAuvI', 139), ('RT
@AnneAnneAss: Michel Delpech : 69 ans. David Bowie : 69 ans. Alan Rickman : 69 ans.  Donald Trump a 69
ans, on croise les doigts.', 135), ('RT @sahilkapur: Staggering statistic in the NBC/WSJ poll  % of
GOP voters who can see themselves supporting Trump  March 2015: 23% January ‰Û_', 120)]
```

The 5 most cited screen-names



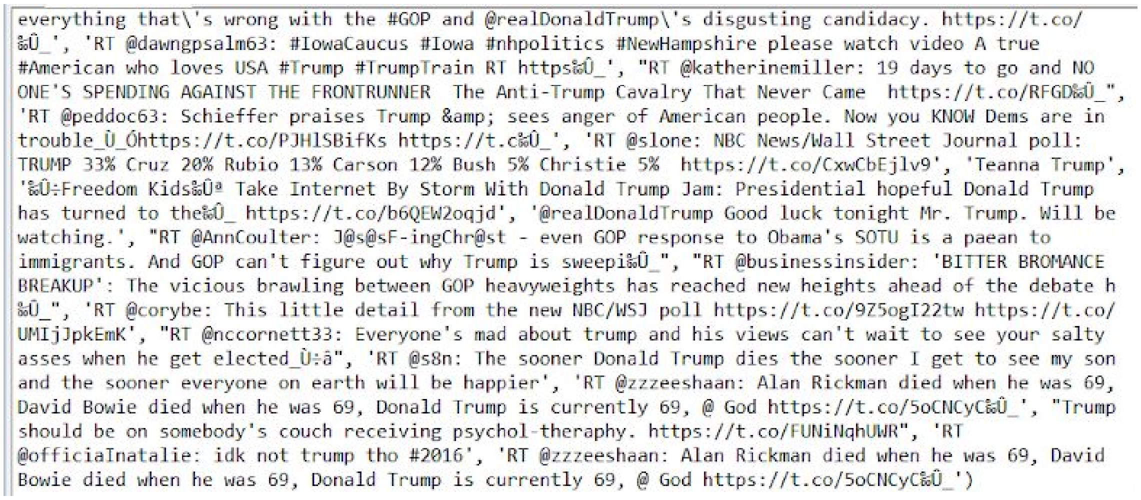
```
The 5 most cited screeen-names are
 [('@', 1679), ('@zzzeeshaan:', 1642), ('@twaimz:', 846), ('@realDonaldTrump', 725), ('@deray:', 653)]
```

The 10 most popular hashtags words



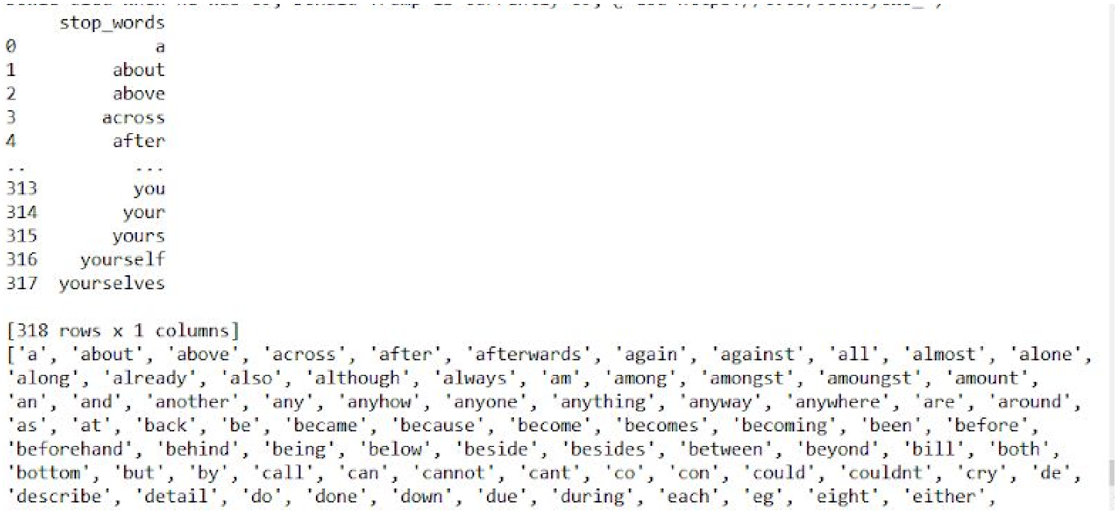
```
The 5 most cited hashtags are
 [('#Trump', 411), ('#Trump2016', 362), ('#MakeAmericaGreatAgain', 161), ('#GOPDebate', 154),
('#RealDonaldTrump', 146)]
```

Sorting Tweets by Time and splitting in 5 subsets



```
everything that\'s wrong with the #GOP and @realDonaldTrump\'s disgusting candidacy. https://t.co/
‰Û_', 'RT @dawngpsalm63: #IowaCaucus #Iowa #nhpolitics #NewHampshire please watch video A true
#American who loves USA #Trump #TrumpTrain RT https‰Û_', "RT @katherinemiller: 19 days to go and NO
ONE'S SPENDING AGAINST THE FRONTRUNNER  The Anti-Trump Cavalry That Never Came  https://t.co/RFGD‰Û_",
'RT @peddoc63: Schieffer praises Trump &amp; sees anger of American people. Now you KNOW Dems are in
trouble_Û_Óhttps://t.co/PJHlSBifKs https://t.c‰Û_', 'RT @slone: NBC News/Wall Street Journal poll:
TRUMP 33% Cruz 20% Rubio 13% Carson 12% Bush 5% Christie 5%  https://t.co/CxwCbEjlv9', 'Teanna Trump',
'‰Û÷Freedom Kids‰Ûª Take Internet By Storm With Donald Trump Jam: Presidential hopeful Donald Trump
has turned to the‰Û_ https://t.co/b6QEW2oqjd', '@realDonaldTrump Good luck tonight Mr. Trump. Will be
watching.', "RT @AnnCoulter: J@s@sF-ingChr@st - even GOP response to Obama's SOTU is a paean to
immigrants. And GOP can't figure out why Trump is sweepi‰Û_", "RT @businessinsider: 'BITTER BROMANCE
BREAKUP': The vicious brawling between GOP heavyweights has reached new heights ahead of the debate h
‰Û_", 'RT @corybe: This little detail from the new NBC/WSJ poll https://t.co/9Z5ogI22tw https://t.co/
UMIjJpkEmK', "RT @nccornett33: Everyone's mad about trump and his views can't wait to see your salty
asses when he get elected_Û÷â", 'RT @s8n: The sooner Donald Trump dies the sooner I get to see my son
and the sooner everyone on earth will be happier', 'RT @zzzeeshaan: Alan Rickman died when he was 69,
David Bowie died when he was 69, Donald Trump is currently 69, @ God https://t.co/5oCNCyC‰Û_', "Trump
should be on somebody's couch receiving psychol-theraphy. https://t.co/FUNiNqhUWR", 'RT
@officiaInatalie: idk not trump tho #2016', 'RT @zzzeeshaan: Alan Rickman died when he was 69, David
Bowie died when he was 69, Donald Trump is currently 69, @ God https://t.co/5oCNCyC‰Û_')
```

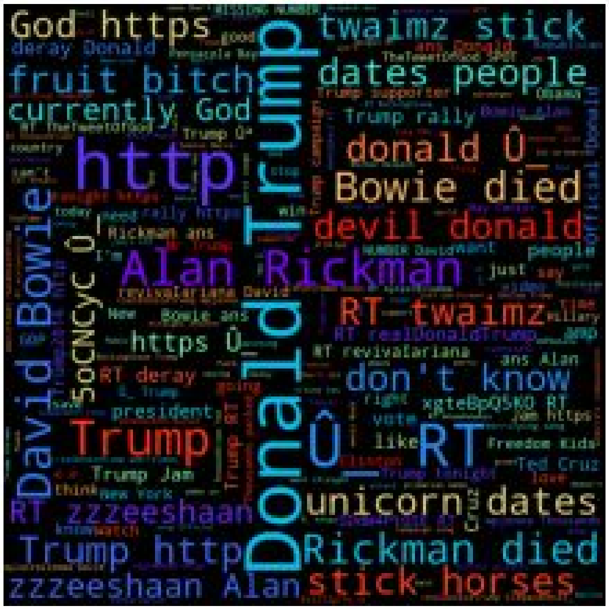
Importing Stop Words from “stopwords_en.txt” file and converting it to list to remove from tweets



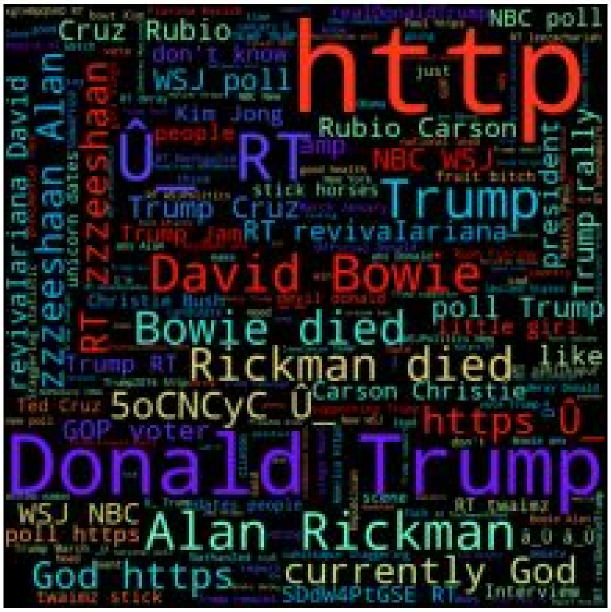
```
     stop_words
0             a
1         about
2         above
3        across
4         after
..          ...
313         you
314        your
315       yours
316    yourself
317  yourselves

[318 rows x 1 columns]
['a', 'about', 'above', 'across', 'after', 'afterwards', 'again', 'against', 'all', 'almost', 'alone',
'along', 'already', 'also', 'although', 'always', 'am', 'among', 'amongst', 'amoungst', 'amount',
'an', 'and', 'another', 'any', 'anyhow', 'anyone', 'anything', 'anyway', 'anywhere', 'are', 'around',
'as', 'at', 'back', 'be', 'became', 'because', 'become', 'becomes', 'becoming', 'been', 'before',
'beforehand', 'behind', 'being', 'below', 'beside', 'besides', 'between', 'beyond', 'bill', 'both',
'bottom', 'but', 'by', 'call', 'can', 'cannot', 'cant', 'co', 'con', 'could', 'couldnt', 'cry', 'de',
'describe', 'detail', 'do', 'done', 'down', 'due', 'during', 'each', 'eg', 'eight', 'either',
```

Word Clouds for each Subset

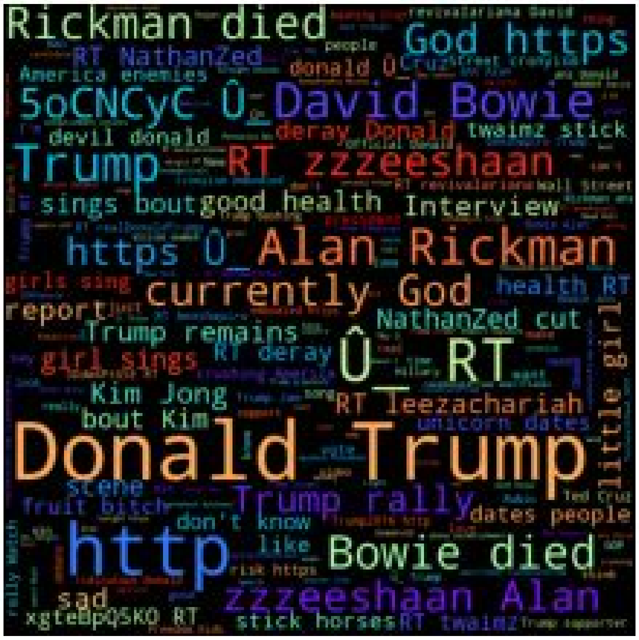
Word Cloud for String 1



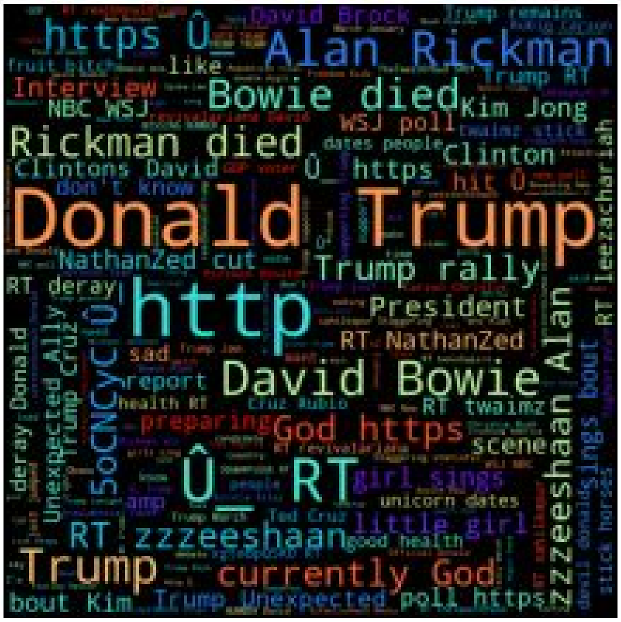
Word Cloud for String 2



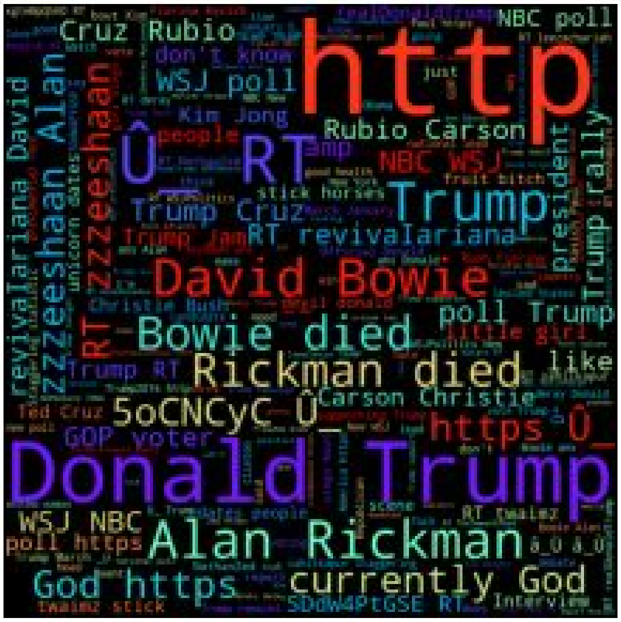
Word Cloud for String 3



Word Cloud for String 4



Word Cloud for String 5



Virtual Conversation of Topics

From the overall project it seems that people are mostly talking about Donald Trump, Crua Rubio, Alan Rickman and David Bowie. The conversation also leads to David and Bowie who died. Topics related to God, NBC poll, Rally, WSJ poll, Gop voters are also discussed.

Deployment

The results created can be used to analyze the general time virtual conversation during the presidential election to understand the most hashtags topics, most retweeted statements, etc.

Conclusions

To conclude tweets.csv was imported in python. Various operations were carried out to know most cited screen names, senders, tweets, hashtag words and virtual conversations.

Python Programming for tweets.csv file

```
#import the file
import pandas as pd
import numpy as np

#read the file
df = pd.read_csv('tweets.csv')
print(df)

#create column named as Sender, Time and Tweet
column_names=['Sender', 'Time', 'Tweet']
df = pd.read_csv('tweets.csv', names=column_names)
print(df)

#create Sender column to list
list_sender = df.Sender.to_list()
print(list_sender)

print("\n")

from collections import Counter

#top 5 most active senders are
counts = Counter(list_sender)
print("The 5 most active senders with counts are", counts.most_common(5))

print("\n")

#create tweets column to list
list_tweets = df.Tweet.to_list()
print(list_tweets)
print("\n")

#top 5 most retweeted are
for i in list_tweets:
    if i.startswith('RT'):
        counts = Counter(list_tweets)
print("The 10 most re-tweeted tweets are\n", counts.most_common(10))

#2 for loop for screen_name
screenname=list()
for i in list_tweets:
    for j in i.split():
        if j.startswith('@'):
            screenname.append(j)
            print(screenname)

#5 most cited screen-names
print('\n')
counts =Counter(screenname)
print("The 5 most cited screeen-names are\n", counts.most_common(5))

#2 for loop for hashtags
hashtags=list()
for h in list_tweets:
    for k in h.split():
        if k.startswith('#'):
            hashtags.append(k)
            print(hashtags)

#5 most popular hashtags
print('\n')
counts =Counter(hashtags)
```

```
print("The 5 most cited hashtags are\n", counts.most_common(5))

#create tweets column to list
list_time = df.Time.to_list()
print(list_time)

#sorting the data set by timestamp
list_time, list_tweets = zip((*sorted(zip(list_time,list_tweets))))
print(list_tweets)

sorting_by_tweets = list_tweets
print(sorting_by_tweets)

#splitting the tweets into 5 and converting in to strings
list_split = np.array_split(sorting_by_tweets,5)
for array in list_split:
    print(list(array))

#creating subset of strings
x=list_split[0]
print(x)

#printing 1st strings acting as a words of bags
str0 = ' '.join(map(str, x))
print(str0)

y=list_split[1]
print(y)

#printing 2nd strings acting as a words of bags
str1 = ' '.join(map(str, y))
print(str1)
```

```
z=list_split[2]
print(z)

#printing 2nd strings acting as a words of bags
str2 = ' '.join(map(str, z))
print(str2)

a=list_split[3]
print(a)

#printing 4th strings acting as a words of bags
str3 = ' '.join(map(str, a))
print(str3)

b=list_split[4]
print(b)

#printing 5th strings acting as a words of bags
str4 = ' '.join(map(str, b))
print(str4)

#import stopwords from text file
column_names_stopwords = ['stop_words']
sw = pd.read_csv('stopwords.txt', names=column_names_stopwords)
print(sw)

#create Stopwords column to list
list_stop_words = sw.stop_words.to_list()
print(list_stop_words)

from wordcloud import WordCloud
import matplotlib.pyplot as plt
```

```
#wordcloud for string0
cloud1=WordCloud(width=800,
height=800,background_color='black',colormap='rainbow',
stopwords=list_stop_words).generate(str0)
plt.imshow(cloud1)
plt.axis("off")
plt.show()

#wordcloud for string1
cloud2=WordCloud(width=800,
height=800,background_color='black',colormap='rainbow',
stopwords=list_stop_words).generate(str1)
plt.imshow(cloud2)
plt.axis("off")
plt.show()

#wordcloud for string2
cloud3=WordCloud(width=800,
height=800,background_color='black',colormap='rainbow',
stopwords=list_stop_words).generate(str2)
plt.imshow(cloud3)
plt.axis("off")
plt.show()

#wordcloud for string3
cloud4=WordCloud(width=800,
height=800,background_color='black',colormap='rainbow',
stopwords=list_stop_words).generate(str3)
plt.imshow(cloud4)
plt.axis("off")
plt.show()

#wordcloud for string4
```

```
cloud5=WordCloud(width=800,
height=800,background_color='black',colormap='rainbow',
stopwords=list_stop_words).generate(str4)
plt.imshow(cloud5)
plt.axis("off")
plt.show()
```