

```
In [80]: # Import the Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import string
import spacy

from matplotlib.pyplot import imread
from wordcloud import WordCloud, STOPWORDS
%matplotlib inline
```

```
In [81]: tweet=pd.read_csv("C:\\Users\\nishi\\Desktop\\Assignments\\Text_Mining\\Elon_musk\\tweet.csv")
tweet.drop(['Unnamed: 0'],inplace=True,axis=1)
tweet
```

Out[81]:

	Text
0	@kunalb11 I m an alien
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...
2	@joerogan @Spotify Great interview!
3	@gtera27 Doge is underestimated
4	@teslacn Congratulations Tesla China for amazi...
...	...
1994	@flcnhvy True, it sounds so surreal, but the n...
1995	@PPathole Make sure to read ur terms & con...
1996	@TeslaGong @PPathole Samwise Gamgee
1997	@PPathole Altho Dumb and Dumber is <U+0001F525...
1998	Progress update August 28

1999 rows × 1 columns

```
In [82]: tweet=[Text.strip() for Text in tweet.Text] # remove both the leading and the trailing spaces
tweet=[Text for Text in tweet if Text] # removes empty strings, because they are not truthy
tweet[0:10]
```

```
Out[82]: ['@kunalb11 I\'x92m an alien',
 '@ID_AA_Carmack Ray tracing on Cyberpunk with HDR is next-level. Have you tried it?',
 '@joerogan @Spotify Great interview!',
 '@gtera27 Doge is underestimated',
 '@teslacn Congratulations Tesla China for amazing execution last year. Now on to the next for even more!!',
 'Happy New Year of the Ox! https://t.co/9WFKMYu2oj', (https://t.co/9WFKMYu2oj),
 'Frodo was the underdoge,\nAll thought he would fail,\nHimself most of all. https://t.co/zGxJFDzzrM', (https://t.co/zGxJFDzzrM),
 '@OwenSparks_ @flcnhvy @anonyx10 Haha thanks :)',
 '@flcnhvy @anonyx10 Indeed! Tweets definitely do not represent real-world time allocation.',
 'The most entertaining outcome is the most likely']
```

```
In [83]: tweet_text=' '.join(tweet)
tweet_text
```

```
Out[83]: '@kunalb11 I\'x92m an alien @ID_AA_Carmack Ray tracing on Cyberpunk with HDR is next-level. Have you tried it? @joerogan @Spotify Great interview! @gtera27 Doge is underestimated @teslacn Congratulations Tesla China for amazing execution last year. Now on to the next for even more!! Happy New Year of the Ox! https://t.co/9WFKMYu2oj (https://t.co/9WFKMYu2oj) Frodo was the underdoge,\nAll thought he would fail,\nHimself most of all. https://t.co/zGxJFDzzrM (https://t.co/zGxJFDzzrM) @OwenSparks_ @flcnhvy @anonyx10 Haha thanks :) @flcnhvy @anonyx10 Indeed! Tweets definitely do not represent real-world time allocation. The most entertaining outcome is the most likely @GiveDirectly Just sent some Just agree to do Clubhouse with @kanyewest https://t.co/3rWE9uHSTS (https://t.co/3rWE9uHSTS) @geoffkeighley @UnrealEngine It\'x92s getting real Bought some Dogecoin for lil X, so he can be a toddler hodler @JoshManMode He definitely has issues, but the sentencing seems a bit high @freewalletorg Thanks for fixing @freewalletorg Please unlock my account @AstroJordy <U+0001F923><U+0001F923> This is true power haha https://t.co/Fc9uhQSd70 (https://t.co/Fc9uhQSd70) @freewalletorg Any crypto wallet that won\'x92t give you your private keys should be avoided at all costs @freewalletorg Your app sucks RT @SpaceX: NASA has selected Falcon Heavy to launch the first two elements of the lunar Gateway together on one mission! https://t.co/3pwt (https://t.co/3pwt) @ajtour
```

```
from nltk.tokenize import TweetTokenizer
tknwr = TweetTokenizer(strip_handles=True)
tweet_tokens=tknwr.tokenize(tweet_text)
print(tweet_tokens)
```

```
[ 'I', '\x92', 'm', 'an', 'alien', 'Ray', 'tracing', 'on', 'Cyberpunk', 'wit  
h', 'HDR', 'is', 'next-level', '.', 'Have', 'you', 'tried', 'it', '?', 'Grea  
t', 'interview', '!', 'Doge', 'is', 'underestimated', 'Congratulations', 'Tes  
la', 'China', 'for', 'amazing', 'execution', 'last', 'year', '.', 'Now', 'o  
n', 'to', 'the', 'next', 'for', 'even', 'more', '!', '!', 'Happy', 'New', 'Ye  
ar', 'of', 'the', 'Ox', '!', 'https://t.co/9WFKMYu2oj', 'Frodo', 'was', 'th  
e', 'underdog', ',', 'All', 'thought', 'he', 'would', 'fail', ',', 'Himsel  
f', 'most', 'of', 'all', '.', 'https://t.co/zGxJFDzzrM', 'Haha', 'thanks',  
':)'), 'Indeed', '!', 'Tweets', 'definitely', 'do', 'not', 'represent', 'real-  
world', 'time', 'allocation', '.', 'The', 'most', 'entertaining', 'outcome',  
'is', 'the', 'most', 'likely', 'Just', 'sent', 'some', 'Just', 'agree', 'to',  
'do', 'Clubhouse', 'with', 'https://t.co/3rWE9uHSTS', 'It', '\x92', 's', 'get  
ting', 'real', 'Bought', 'some', 'Dogecoin', 'for', 'lil', 'X', ',', 'so', 'h  
e', 'can', 'be', 'a', 'toddler', 'hodler', 'He', 'definitely', 'has', 'issue  
s', ',', 'but', 'the', 'sentencing', 'seems', 'a', 'bit', 'high', 'Thanks',  
'for', 'fixing', 'Please', 'unlock', 'my', 'account', '<U+0001F923>', '<U+000  
1F923>', 'This', 'is', 'true', 'power', 'haha', 'https://t.co/Fc9uhQSd70', 'A  
ny', 'crypto', 'wallet', 'that', 'won', '\x92', 't', 'give', 'you', 'your',  
'private', 'keys', 'should', 'be', 'avoided', 'at', 'all', 'costs', 'Your',
```

```
tweet_tokens_text=' '.join(tweet_tokens)
tweet_tokens_text
```

'I \x92 m an alien Ray tracing on Cyberpunk with HDR is next-level . Have you tried it ? Great interview ! Doge is underestimated Congratulations Tesla China for amazing execution last year . Now on to the next for even more ! ! Happy New Year of the Ox ! <https://t.co/9WFKMYu2oj> (<https://t.co/9WFKMYu2oj>) Frodo was the underdog , All thought he would fail , Himself most of all . <http://t.co/zGxJFDzzrM> (<https://t.co/zGxJFDzzrM>) Haha thanks :) Indeed ! Tweets definitely do not represent real-world time allocation . The most entertaining outcome is the most likely Just sent some Just agree to do Clubhouse with <https://t.co/3rWE9uHSTS> (<https://t.co/3rWE9uHSTS>) It \x92 s getting real Bought some Dogecoin for lil X , so he can be a toddler hodler He definitely has issues , but the sentencing seems a bit high Thanks for fixing Please unlock my account <U+0001F923> <U+0001F923> This is true power haha <https://t.co/Fc9uhQSd70> (<https://t.co/Fc9uhQSd70>) Any crypto wallet that won \x92 t give you your private keys should be avoided at all costs Your app sucks RT : NASA has selected Falcon Heavy to launch the first two elements of the lunar Gateway together on one mission ! <https://t.co/3pWt> (<https://t.co/3pWt>) Yes Once we can predict cash flow reasonably well , Starlink will IPO Starlink is a staggeringly difficult technical & economic endeavor . However , if we don <https://t.co/9Z8Ac6skqx> (<https://t.co/9Z8Ac6skqx>) SpaceX needs to pass through a deep

```
In [86]: no_punc_text=tweet_tokens_text.translate(str.maketrans('','',string.punctuation))
no_punc_text
```

```
Out[86]: 'I \x92 m an alien Ray tracing on Cyberpunk with HDR is nextlevel Have you t
ried it Great interview Doge is underestimated Congratulations Tesla China
for amazing execution last year Now on to the next for even more Happy New
Year of the Ox httpstco9WFKMYu2oj Frodo was the underdodge All thought he wo
uld fail Himself most of all httpstcozGxJFDzzrM Haha thanks Indeed Tweets
definitely do not represent realworld time allocation The most entertaining
outcome is the most likely Just sent some Just agree to do Clubhouse with htt
pstco3rWE9uHSTS It \x92 s getting real Bought some Dogecoin for lil X so he
can be a toddler hodler He definitely has issues but the sentencing seems a
bit high Thanks for fixing Please unlock my account U0001F923 U0001F923 This
is true power haha httpstcoFc9uhQSd70 Any crypto wallet that won \x92 t give
you your private keys should be avoided at all costs Your app sucks RT NASA
has selected Falcon Heavy to launch the first two elements of the lunar Gatew
ay together on one mission httpstco3pWt Yes Once we can predict cash flow re
asonably well Starlink will IPO Starlink is a staggeringly difficult technic
al economic endeavor However if we don httpstco9Z8Ac6skqx SpaceX needs to
pass through a deep chasm of negative cash flow over the next year or httpstc
o7J1c92hdjl Lowest cost per ton of carbon sequestered net of value of any pr
oduct made Must be scalable to g httpstcoXMyI7qWSgw It \x92 s meant to be th
```

```
In [87]: import re
no_url_text=re.sub(r'http\S+', '', no_punc_text)
no_url_text
```

```
Out[87]: 'I \x92 m an alien Ray tracing on Cyberpunk with HDR is nextlevel Have you t
ried it Great interview Doge is underestimated Congratulations Tesla China
for amazing execution last year Now on to the next for even more Happy New
Year of the Ox Frodo was the underdodge All thought he would fail Himself
most of all Haha thanks Indeed Tweets definitely do not represent realwor
ld time allocation The most entertaining outcome is the most likely Just sen
t some Just agree to do Clubhouse with It \x92 s getting real Bought some Do
gecoin for lil X so he can be a toddler hodler He definitely has issues but
the sentencing seems a bit high Thanks for fixing Please unlock my account U0
001F923 U0001F923 This is true power haha Any crypto wallet that won \x92 t
give you your private keys should be avoided at all costs Your app sucks RT
NASA has selected Falcon Heavy to launch the first two elements of the lunar
Gateway together on one mission Yes Once we can predict cash flow reasonabl
y well Starlink will IPO Starlink is a staggeringly difficult technical eco
nomic endeavor However if we don SpaceX needs to pass through a deep chasm
of negative cash flow over the next year or Lowest cost per ton of carbon se
questered net of value of any product made Must be scalable to g It \x92 s
meant to be the same price in all countries Only difference should be taxes
shipping This is intended for Earth but there may be some ideas that apply
```

```
import nltk
from nltk.tokenize import word_tokenize
text_tokens=word_tokenize(no_url_text)
print(text_tokens)
```

```
[ 'I', '\x92', 'm', 'an', 'alien', 'Ray', 'tracing', 'on', 'Cyberpunk', 'wit
h', 'HDR', 'is', 'nextlevel', 'Have', 'you', 'tried', 'it', 'Great', 'intervi
ew', 'Doge', 'is', 'underestimated', 'Congratulations', 'Tesla', 'China', 'fo
r', 'amazing', 'execution', 'last', 'year', 'Now', 'on', 'to', 'the', 'next',
'for', 'even', 'more', 'Happy', 'New', 'Year', 'of', 'the', 'Ox', 'Frodo', 'w
as', 'the', 'underdoge', 'All', 'thought', 'he', 'would', 'fail', 'Himself',
'most', 'of', 'all', 'Haha', 'thanks', 'Indeed', 'Tweets', 'definitely', 'd
o', 'not', 'represent', 'realworld', 'time', 'allocation', 'The', 'most', 'en
tertaining', 'outcome', 'is', 'the', 'most', 'likely', 'Just', 'sent', 'som
e', 'Just', 'agree', 'to', 'do', 'Clubhouse', 'with', 'It', '\x92', 's', 'get
ting', 'real', 'Bought', 'some', 'Dogecoin', 'for', 'lil', 'X', 'so', 'he',
'can', 'be', 'a', 'toddler', 'hodler', 'He', 'definitely', 'has', 'issues',
'but', 'the', 'sentencing', 'seems', 'a', 'bit', 'high', 'Thanks', 'for', 'fi
xing', 'Please', 'unlock', 'my', 'account', 'U0001F923', 'U0001F923', 'This',
'is', 'true', 'power', 'haha', 'Any', 'crypto', 'wallet', 'that', 'won', '\x9
2', 't', 'give', 'you', 'your', 'private', 'keys', 'should', 'be', 'avoided',
'at', 'all', 'costs', 'Your', 'app', 'sucks', 'RT', 'NASA', 'has', 'selecte
d', 'Falcon', 'Heavy', 'to', 'launch', 'the', 'first', 'two', 'elements', 'o
f', 'the', 'lunar', 'Gateway', 'together', 'on', 'one', 'mission', 'Yes', 'On
```

```
import nltk
nltk.download('punkt')
nltk.download('stopwords')
```

```
[nltk_data] Error loading punkt: <urlopen error [Errno 11001]
[nltk_data]      getaddrinfo failed>
[nltk_data] Error loading stopwords: <urlopen error [Errno 11001]
[nltk_data]      getaddrinfo failed>
```

```
Out[89]: False
```

```
len(text_tokens)
```

Out[90]: 17847

```
In [91]: from nltk.corpus import stopwords
my_stop_words=stopwords.words('english')

sw_list = ['\x92','rt','ye','yeah','haha','Yes','U0001F923','I']
my_stop_words.extend(sw_list)

no_stop_tokens=[word for word in text_tokens if not word in my_stop_words]
print(no_stop_tokens)
```

```
['alien', 'Ray', 'tracing', 'Cyberpunk', 'HDR', 'nextlevel', 'Have', 'tried',
'Great', 'interview', 'Doge', 'underestimated', 'Congratulations', 'Tesla',
'China', 'amazing', 'execution', 'last', 'year', 'Now', 'next', 'even', 'Happ
y', 'New', 'Year', 'Ox', 'Frodo', 'underdoge', 'All', 'thought', 'would', 'fa
il', 'Himself', 'Haha', 'thanks', 'Indeed', 'Tweets', 'definitely', 'represen
t', 'realworld', 'time', 'allocation', 'The', 'entertaining', 'outcome', 'lik
ely', 'Just', 'sent', 'Just', 'agree', 'Clubhouse', 'It', 'getting', 'real',
'Bought', 'Dogecoin', 'lil', 'X', 'toddler', 'hodler', 'He', 'definitely', 'i
ssues', 'sentencing', 'seems', 'bit', 'high', 'Thanks', 'fixing', 'Please',
'unlock', 'account', 'This', 'true', 'power', 'Any', 'crypto', 'wallet', 'giv
e', 'private', 'keys', 'avoided', 'costs', 'Your', 'app', 'sucks', 'RT', 'NAS
A', 'selected', 'Falcon', 'Heavy', 'launch', 'first', 'two', 'elements', 'lun
ar', 'Gateway', 'together', 'one', 'mission', 'Once', 'predict', 'cash', 'flo
w', 'reasonably', 'well', 'Starlink', 'IPO', 'Starlink', 'staggeringly', 'dif
ficult', 'technical', 'economic', 'endeavor', 'However', 'SpaceX', 'needs',
'pass', 'deep', 'chasm', 'negative', 'cash', 'flow', 'next', 'year', 'Lowes
t', 'cost', 'per', 'ton', 'carbon', 'sequestered', 'net', 'value', 'product',
'made', 'Must', 'scalable', 'g', 'It', 'meant', 'price', 'countries', 'Only',
'difference', 'taxes', 'shipping', 'This', 'intended', 'Earth', 'may', 'idea
', 'ideas', 'apply', 'mars', 'xprize', 'team', 'manage', '100m', 'carbo
n', 'capture', 'prize', 'everyone', 'tesla', 'receives', 'stock', 'my', 'comp',
'stock', 'options', 'take', 'table', 'that', 'missing', 'back', 'work', 'go',
'does', 'seem', 'bit', 'high', 'doge', 'appears', 'inflationary', 'meaningfull
y', 'fixed', 'coins', 'per', 'unit', 'time', 'whereas', 'wow', '1', 'orbital',
'launch', 'tower', 'stack', '2', 'enough', 'raptors', 'orbit', 'booster']
```

```
In [92]: lower_words=[Text.lower() for Text in no_stop_tokens]
print(lower_words[100:200])
```

```
['once', 'predict', 'cash', 'flow', 'reasonably', 'well', 'starlink', 'ipo', 's
tarlink', 'staggeringly', 'difficult', 'technical', 'economic', 'endeavor', 'ho
wever', 'spacex', 'needs', 'pass', 'deep', 'chasm', 'negative', 'cash', 'flow',
'next', 'year', 'lowest', 'cost', 'per', 'ton', 'carbon', 'sequestered', 'net',
'value', 'product', 'made', 'must', 'scalable', 'g', 'it', 'meant', 'price', 'c
ountries', 'only', 'difference', 'taxes', 'shipping', 'this', 'intended', 'eart
h', 'may', 'ideas', 'apply', 'mars', 'xprize', 'team', 'manage', '100m', 'carbo
n', 'capture', 'prize', 'everyone', 'tesla', 'receives', 'stock', 'my', 'comp',
'stock', 'options', 'take', 'table', 'that', 'missing', 'back', 'work', 'go',
'does', 'seem', 'bit', 'high', 'doge', 'appears', 'inflationary', 'meaningfull
y', 'fixed', 'coins', 'per', 'unit', 'time', 'whereas', 'wow', '1', 'orbital',
'launch', 'tower', 'stack', '2', 'enough', 'raptors', 'orbit', 'booster']
```

```
In [93]: from nltk.stem import PorterStemmer
ps=PorterStemmer()
stemmed_tokens=[ps.stem(word) for word in lower_words]
print(stemmed_tokens[100:200])
```

```
['onc', 'predict', 'cash', 'flow', 'reason', 'well', 'starlink', 'ipo', 'starli
nk', 'staggeringli', 'difficult', 'technic', 'econom', 'endeavor', 'howev', 'sp
acex', 'need', 'pass', 'deep', 'chasm', 'neg', 'cash', 'flow', 'next', 'year',
'lowest', 'cost', 'per', 'ton', 'carbon', 'sequest', 'net', 'valu', 'product',
'made', 'must', 'scalabl', 'g', 'it', 'meant', 'price', 'countri', 'onli', 'dif
fer', 'tax', 'ship', 'thi', 'intend', 'earth', 'may', 'idea', 'appli', 'mar',
'xprize', 'team', 'manag', '100m', 'carbon', 'captur', 'prize', 'everyon', 'tes
la', 'receiv', 'stock', 'my', 'comp', 'stock', 'option', 'take', 'tabl', 'tha
t', 'miss', 'back', 'work', 'go', 'doe', 'seem', 'bit', 'high', 'doge', 'appea
r', 'inflationari', 'meaning', 'fix', 'coin', 'per', 'unit', 'time', 'wherea',
'wow', '1', 'orbit', 'launch', 'tower', 'stack', '2', 'enough', 'raptor', 'orbi
t', 'booster']
```

```
In [94]: import spacy
nlp=spacy.load('en_core_web_sm')#python -m spacy download en_core_web_sm cmd prom
doc=nlp(' '.join(lower_words))
print(doc)
```

```
C:\Users\nishi\anaconda3\lib\site-packages\spacy\util.py:833: UserWarning: [W
095] Model 'en_core_web_sm' (3.0.0) was trained with spaCy v3.0 and may not b
e 100% compatible with the current version (3.2.1). If you see errors or degr
aded performance, download a newer compatible model or retrain your custom mo
del with the current spaCy version. For more details and available updates, r
un: python -m spacy validate
  Errors.E970.format(str_command=cmd_str, tool=cmd_list[0])
```

```
In [95]: from nltk.tokenize import TweetTokenizer
tknznr = TweetTokenizer(strip_handles=True)
tweet_tokens=tknznr.tokenize(tweet_text)
print(tweet_tokens)
```

```
['I', '\x92', 'm', 'an', 'alien', 'Ray', 'tracing', 'on', 'Cyberpunk', 'wit
h', 'HDR', 'is', 'next-level', '.', 'Have', 'you', 'tried', 'it', '?', 'Grea
t', 'interview', '!', 'Doge', 'is', 'underestimated', 'Congratulations', 'Tes
la', 'China', 'for', 'amazing', 'execution', 'last', 'year', '.', 'Now', 'o
n', 'to', 'the', 'next', 'for', 'even', 'more', '!', '!', 'Happy', 'New', 'Ye
ar', 'of', 'the', 'Ox', '!', 'https://t.co/9WFKMYu2oj', 'Frodo', 'was', 'th
e', 'underdoge', ',', 'All', 'thought', 'he', 'would', 'fail', ',', 'Himself',
'most', 'of', 'all', '.', 'https://t.co/zGxJFDzzrM', 'Haha', 'thanks',
':)', 'Indeed', '!', 'Tweets', 'definitely', 'do', 'not', 'represent', 'real-
world', 'time', 'allocation', '.', 'The', 'most', 'entertaining', 'outcome',
'is', 'the', 'most', 'likely', 'Just', 'sent', 'some', 'Just', 'agree', 'to',
'do', 'Clubhouse', 'with', 'https://t.co/3rWE9uHSTS', 'It', '\x92', 's', 'get
ting', 'real', 'Bought', 'some', 'DogeCoin', 'for', 'lil', 'X', ',', 'so', 'h
e', 'can', 'be', 'a', 'toddler', 'hodler', 'He', 'definitely', 'has', 'issue
s', ',', 'but', 'the', 'sentencing', 'seems', 'a', 'bit', 'high', 'Thanks',
'for', 'fixing', 'Please', 'unlock', 'my', 'account', '<U+0001F923>', '<U+000
1F923>', 'This', 'is', 'true', 'power', 'haha', 'https://t.co/Fc9uhQsD70', 'A
ny', 'crypto', 'wallet', 'that', 'won', '\x92', 't', 'give', 'you', 'your',
'private', 'keys', 'should', 'be', 'avoided', 'at', 'all', 'costs', 'Your',
```

```
In [96]: lemmas=[token.lemma_ for token in doc]
print(lemmas)
```

```
['alien', 'ray', 'tracing', 'cyberpunk', 'hdr', 'nextlevel', 'have', 'try',
'great', 'interview', 'doge', 'underestimate', 'congratulation', 'tesla', 'ch
ina', 'amazing', 'execution', 'last', 'year', 'now', 'next', 'even', 'happy',
'new', 'year', 'ox', 'frodo', 'underdoge', 'all', 'think', 'would', 'fail',
'himself', 'haha', 'thank', 'indeed', 'tweet', 'definitely', 'represent', 're
alworld', 'time', 'allocation', 'the', 'entertaining', 'outcome', 'likely',
'just', 'send', 'just', 'agree', 'clubhouse', 'it', 'get', 'real', 'buy', 'do
gecoin', 'lil', 'x', 'toddler', 'hodler', 'he', 'definitely', 'issue', 'sente
ncing', 'seem', 'bit', 'high', 'thank', 'fix', 'please', 'unlock', 'account',
'this', 'true', 'power', 'any', 'crypto', 'wallet', 'give', 'private', 'key',
'avoid', 'cost', 'your', 'app', 'suck', 'rt', 'nasa', 'select', 'falcon', 'he
avy', 'launch', 'first', 'two', 'element', 'lunar', 'gateway', 'together', 'o
ne', 'mission', 'once', 'predict', 'cash', 'flow', 'reasonably', 'well', 'sta
rlink', 'ipo', 'starlink', 'staggeringly', 'difficult', 'technical', 'economi
c', 'endeavor', 'however', 'spacex', 'need', 'pass', 'deep', 'chasm', 'negati
ve', 'cash', 'flow', 'next', 'year', 'low', 'cost', 'per', 'ton', 'carbon',
'sequester', 'net', 'value', 'product', 'make', 'must', 'scalable', 'g', 'i
t', 'mean', 'price', 'country', 'only', 'difference', 'taxis', 'ship', 'thi
s', 'intend', 'earth', 'may', 'idea', 'apply', 'mar', 'xprize', 'team', 'mana
```



```
In [97]: clean_tweets=' '.join(lemmas)
clean_tweets
```

```
Out[97]: 'alien ray tracing cyberpunk hdr nextlevel have try great interview doge unde  
restimate congratulation tesla china amazing execution last year now next eve  
n happy new year ox frodo underdoge all think would fail himself haha thank i  
ndeed tweet definitely represent realworld time allocation the entertaining o  
utcome likely just send just agree clubhouse it get real buy dogecoin lil x t  
oddler hodler he definitely issue sentencing seem bit high thank fix please u  
nlock account this true power any crypto wallet give private key avoid cost y  
our app suck rt nasa select falcon heavy launch first two element lunar gatew  
ay together one mission once predict cash flow reasonably well starlink ipo s  
tarlink staggeringly difficult technical economic endeavor however spacex nee  
d pass deep chasm negative cash flow next year low cost per ton carbon seques  
ter net value product make must scalable g it mean price country only differe  
nce taxis ship this intend earth may idea apply mar xprize team manage 100 m  
carbon capture prize everyone tesla receive stock my comp stock option take t  
able that miss back work go do seem bit high doge appear inflationary meaning  
fully fix coin per unit time whereas wow 1 orbital launch tower stack 2 enoug  
h raptor orbit booster 3 improve ship booster mass back work tonight ð ðogeco  
in instructional video the people speak so cute u0001f495 extremely misleadin  
g image reflect true time cost people rain pain u0001f3b6 who let doge u0001f
```

```
In [98]: from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer()
tweetscv=cv.fit_transform(lemmas)
```

```
In [99]: print(cv.vocabulary_)
```

```
{'alien': 194, 'ray': 2348, 'tracing': 3009, 'cyberpunk': 764, 'hdr': 1358, 'nextlevel': 1958, 'have': 1353, 'try': 3039, 'great': 1306, 'interview': 1555, 'doge': 888, 'underestimate': 3173, 'congratulation': 672, 'tesla': 2918, 'china': 581, 'amazing': 216, 'execution': 1055, 'last': 1639, 'year': 3390, 'now': 1986, 'next': 1957, 'even': 1029, 'happy': 1340, 'new': 1955, 'ox': 2087, 'frodo': 1205, 'underdoge': 3172, 'all': 196, 'think': 2946, 'would': 3376, 'fail': 1089, 'himself': 1401, 'haha': 1329, 'thank': 2927, 'indeed': 1496, 'tweet': 3054, 'definitely': 802, 'represent': 2415, 'realworld': 2363, 'time': 2969, 'allocation': 198, 'the': 2932, 'entertaining': 999, 'outcome': 2066, 'likely': 1693, 'just': 1597, 'send': 2557, 'agree': 179, 'clubhouse': 611, 'it': 1574, 'get': 1264, 'real': 2358, 'buy': 499, 'dogecoin': 890, 'lil': 1694, 'toddler': 2980, 'hodler': 1407, 'he': 1359, 'issue': 1573, 'sentencing': 2562, 'seem': 2549, 'bit': 416, 'high': 1393, 'fix': 1150, 'please': 2182, 'unlock': 3197, 'account': 137, 'this': 2949, 'true': 3034, 'power': 2218, 'any': 239, 'crypto': 747, 'wallet': 3281, 'give': 1276, 'private': 2256, 'key': 1609, 'avoid': 336, 'cost': 711, 'your': 3401, 'app': 250, 'suck': 2819, 'rt': 2485, 'nasa': 1934, 'select': 2550, 'falcon': 1094, 'heavy': 1369, 'launch': 1648, 'first': 1146, 'two': 3060, 'element': 968, 'lunar': 1750, 'gateway': 1248, 'together': 2981, 'one': 2033, 'mission': 1869, 'once': 2032, 'immediately': 2220, 'possibly': 528, 'flavor': 1161, 'incredibly': 2267, 'well': 2212}
```

```
In [100]: print(cv.get_feature_names()[100:200])
```

```
['74', '78', '7th', '90', '9007', '922', '948', '95', '99', 'aber', 'able', 'ab  
o', 'aboard', 'abort', 'about', 'above', 'absence', 'absolute', 'absolutely',  
'absorb', 'absorption', 'absurd', 'absurdly', 'ac', 'academia', 'accel', 'accel  
era', 'accelerate', 'acceleration', 'accelerator', 'accept', 'acceptable', 'acc  
ess', 'accessible', 'accident', 'accidental', 'accommodate', 'account', 'accur  
a', 'accuracy', 'accurate', 'ace', 'achieve', 'achievement', 'achy', 'acquisiti  
on', 'across', 'action', 'active', 'activity', 'actual', 'actual', 'actually',  
'actuary', 'adagio', 'add', 'additive', 'address', 'administer', 'adult', 'adva  
nc', 'advance', 'advanced', 'advantage', 'adventure', 'advertise', 'advice', 'a  
dvice', 'aero', 'afb', 'affair', 'affect', 'affordable', 'africa', 'after', 'af  
ternoon', 'age', 'ago', 'agony', 'agree', 'ah', 'ahead', 'ahem', 'ai', 'aim',  
'air', 'aircraft', 'airplane', 'ak', 'aka', 'alert', 'alexander', 'algo', 'algo  
rithm', 'alien', 'align', 'all', 'allocati', 'allocation', 'allow']
```

```
In [101]: print(tweetscv.toarray()[100:200])
```

```
[[0 0 0 ... 0 0 0]  
 [0 0 0 ... 0 0 0]  
 [0 0 0 ... 0 0 0]  
 ...  
 [0 0 0 ... 0 0 0]  
 [0 0 0 ... 0 0 0]  
 [0 0 0 ... 0 0 0]]
```

```
In [102]: print(tweetscv.toarray().shape)
```

```
(11486, 3410)
```

```
In [103]: cv_ngram_range=CountVectorizer(analyzer='word',ngram_range=(1,3),max_features=100000)  
bow_matrix_ngram=cv_ngram_range.fit_transform(lemmas)
```

```
In [104]: print(cv_ngram_range.get_feature_names())
print(bow_matrix_ngram.toarray())
```

```
['actually', 'also', 'back', 'big', 'booster', 'car', 'come', 'complete', 'cool', 'could', 'crew', 'day', 'design', 'do', 'dragon', 'earth', 'engine', 'even', 'ever', 'exactly', 'falcon', 'first', 'flight', 'fsd', 'future', 'get', 'go', 'good', 'great', 'haha', 'hard', 'high', 'if', 'it', 'just', 'land', 'launch', 'like', 'look', 'lot', 'love', 'make', 'many', 'mar', 'maybe', 'mission', 'model', 'month', 'much', 'need', 'new', 'next', 'no', 'not', 'ok', 'one', 'part', 'people', 'point', 'pretty', 'probably', 'production', 'right', 'rocket', 'rt', 'say', 'seem', 'soon', 'space', 'spacex', 'starlink', 'starship', 'still', 'super', 'sure', 'take', 'tesla', 'test', 'thank', 'that', 'the', 'there', 'they', 'think', 'this', 'time', 'true', 'try', 'ufe0f', 'use', 'way', 'we', 'week', 'well', 'will', 'work', 'would', 'yeah', 'year', 'you']
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
```

```
In [105]: from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_vngram_max_features=TfidfVectorizer(norm='l2',analyzer='word',ngram_range=(
tfidf_matix_ngram=tfidf_vngram_max_features.fit_transform(lemmas)
print(tfidf_vngram_max_features.get_feature_names())
print(tfidf_matix_ngram.toarray())
```

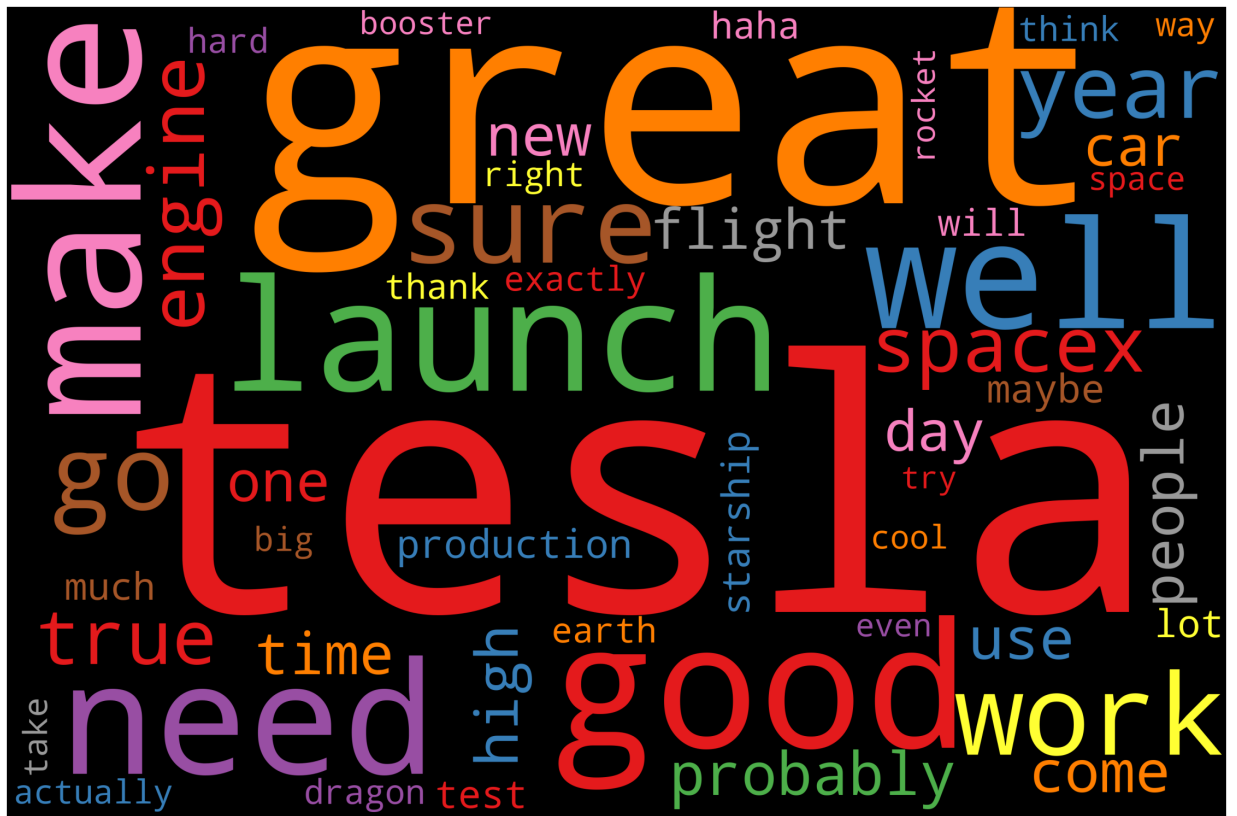
```
['10', '100', '13', '20', '2020', '30', '300', '40', '4d', '60', 'able', 'abs
olutely', 'access', 'accurate', 'actually', 'add', 'ago', 'agree', 'ai', 'ai
m', 'air', 'all', 'almost', 'already', 'also', 'although', 'always', 'amazin
g', 'and', 'another', 'appear', 'apply', 'appreciate', 'approval', 'arm', 'ar
ound', 'article', 'as', 'astronaut', 'at', 'autopilot', 'awesome', 'back', 'b
ad', 'bar', 'base', 'battery', 'be', 'become', 'berlin', 'beta', 'big', 'bi
t', 'boca', 'body', 'booster', 'brain', 'bs', 'build', 'burn', 'but', 'buy',
'cake', 'call', 'camera', 'can', 'car', 'cargo', 'catch', 'cell', 'ch4', 'cha
nce', 'change', 'china', 'civilization', 'close', 'cold', 'color', 'come', 'c
ompany', 'complete', 'computer', 'confirm', 'congratulation', 'control', 'coo
l', 'cost', 'could', 'course', 'cov', 'create', 'crew', 'critical', 'cureva
c', 'cybertruck', 'datum', 'day', 'deep', 'definitely', 'delivery', 'design',
'detail', 'difference', 'different', 'do', 'dock', 'doge', 'dojo', 'dragon',
'drive', 'droneship', 'due', 'dumb', 'early', 'earth', 'easy', 'electric', 'e
nable', 'end', 'energy', 'engine', 'engineer', 'engineering', 'enough', 'espe
cially', 'essential', 'esthetic', 'et', 'even', 'ever', 'every', 'everyone',
'everything', 'exactly', 'expect', 'extreme', 'extremely', 'fail', 'fair', 'f
alcon', 'far', 'fast', 'favorite', 'feel', 'fine', 'finish', 'fire', 'first',
'fix', 'flap', 'flight', 'flow', 'fly', 'for', 'fsd', 'full', 'fully', 'fun',
'fundamental', 'future', 'game', 'gas', 'gene', 'get', 'giant', 'giga', 'giv
e', 'glad', 'go', 'good', 'government', 'great', 'haha', 'half', 'happen', 'h
appy', 'hard', 'have', 'he', 'head', 'header', 'hear', 'heart', 'heavy', 'hel
p', 'high', 'hold', 'home', 'hop', 'hope', 'hopefully', 'hot', 'hour', 'huma
n', 'idea', 'if', 'imo', 'important', 'impressive', 'improve', 'improvement',
'in', 'increase', 'indeed', 'inference', 'irony', 'it', 'just', 'km', 'know',
'la', 'lab', 'land', 'landing', 'large', 'last', 'later', 'launch', 'least',
'leave', 'leg', 'less', 'let', 'level', 'life', 'liftoff', 'like', 'likely',
'limit', 'liquid', 'literally', 'little', 'live', 'long', 'longterm', 'look',
'lose', 'lot', 'love', 'low', 'machine', 'major', 'make', 'many', 'mar', 'mas
s', 'matter', 'max', 'may', 'maybe', 'mean', 'might', 'mile', 'min', 'mind',
'minute', 'mission', 'model', 'money', 'month', 'moon', 'most', 'mostly', 'mo
tor', 'mount', 'move', 'movie', 'much', 'must', 'my', 'name', 'need', 'net',
'neuralink', 'never', 'new', 'news', 'next', 'night', 'no', 'nosecone', 'no
t', 'note', 'nothing', 'now', 'number', 'of', 'ok', 'old', 'one', 'only', 'op
en', 'operational', 'option', 'orbit', 'orbital', 'original', 'our', 'pad',
'part', 'pass', 'pcr', 'people', 'per', 'person', 'physics', 'place', 'play',
'please', 'pm', 'point', 'positive', 'possible', 'potential', 'power', 'press
ure', 'pretty', 'price', 'primarily', 'probability', 'probably', 'problem',
'product', 'production', 'progress', 'propellant', 'prototype', 'provide', 'p
ublic', 'push', 'put', 'quite', 'range', 'raptor', 'reach', 'read', 'real',
'really', 'release', 'require', 'return', 'review', 'right', 'rise', 'roadste
r', 'rocket', 'rt', 'safe', 'safety', 'satellite', 'say', 'scale', 'second',
'see', 'seem', 'sell', 'send', 'sense', 'separation', 'service', 'set', 'seve
ral', 'ship', 'short', 'shot', 'should', 'show', 'signal', 'single', 'site',
'slightly', 'slow', 'small', 'sn5', 'sn8', 'sn9', 'so', 'software', 'solar',
'solution', 'solve', 'some', 'something', 'song', 'soon', 'sorry', 'sound',
'south', 'space', 'spacex', 'speak', 'speed', 'splashdown', 'stage', 'starlin
k', 'starship', 'start', 'static', 'station', 'steel', 'step', 'still', 'stoc
k', 'story', 'strange', 'super', 'support', 'sure', 'symptom', 'system', 'tak
e', 'talk', 'tank', 'target', 'tbh', 'team', 'technology', 'tesla', 'test',
```

```
'testing', 'texas', 'thank', 'that', 'the', 'there', 'they', 'thing', 'thin  
k', 'this', 'three', 'tile', 'time', 'to', 'today', 'tomorrow', 'ton', 'tonig  
ht', 'top', 'tough', 'track', 'traffic', 'training', 'travel', 'true', 'try',  
'tunnel', 'turn', 'twitter', 'two', 'u0001f3b6', 'u0001f440', 'u0001f495', 'u  
0001f525', 'u0001f5a4', 'u0435', 'u043b', 'u043e', 'u2192', 'ufe0f', 'underst  
and', 'update', 'upgrade', 'ur', 'use', 'useful', 'vaccine', 'value', 'vehicl  
e', 'version', 'very', 'video', 'view', 'volume', 'vs', 'wait', 'want', 'watc  
h', 'way', 'we', 'week', 'welcome', 'well', 'what', 'wide', 'wild', 'will',  
'without', 'word', 'work', 'world', 'worth', 'would', 'wow', 'yeah', 'year',  
'yes', 'yet', 'you', 'yup', 'zero']  
[[0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 ...  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]  
 [0. 0. 0. ... 0. 0. 0.]]
```

```
In [106]: # Defining a function to plot word cloud
def plot_cloud(wordcloud):
    plt.figure(figsize=(40,30))
    plt.imshow(wordcloud)
    plt.axis('off')

# Generate Word Cloud

STOPWORDS.add('pron')
STOPWORDS.add('rt')
STOPWORDS.add('yeah')
wordcloud=WordCloud(width=3000,height=2000,background_color='black',max_words=50,
                    colormap='Set1',stopwords=STOPWORDS).generate(clean_tweets)
plot_cloud(wordcloud)
```



```
In [107]: nlp=spacy.load('en_core_web_sm')  
  
one_block=clean_tweets  
doc_block=nlp(one_block)  
spacy.displacy.render(doc_block, style='ent', jupyter=True)
```

alien ray tracing cyberpunk hdr nextlevel have try great interview doge underestimate  
congratulation tesla china amazing execution last year **DATE** now next even happy new  
year ox frodo underdoge all think would fail himself haha thank indeed tweet definitely  
represent realworld time allocation the entertaining outcome likely just send just agree  
clubhouse it get real buy dogecoin lil x toddler hodler he definitely issue sentencing seem bit  
high thank fix please unlock account this true power any crypto wallet give private key avoid  
cost your app suck rt nasa select falcon heavy launch first **ORDINAL** two **CARDINAL**  
element lunar gateway together one **CARDINAL** mission once predict cash flow reasonably  
well starlink ipo starlink staggeringly difficult technical economic endeavor however spacex

```
In [108]: for token in doc_block[100:200]:  
          print(token,token.pos_)
```

```
once ADV  
predict VERB  
cash NOUN  
flow NOUN  
reasonably ADV  
well ADV  
starlink NOUN  
ipo NOUN  
starlink NOUN  
staggeringly ADV  
difficult ADJ  
technical ADJ  
economic ADJ  
endeavor NOUN  
however ADV  
spacex ADJ  
need AUX  
pass VERB  
deep ADJ  
chasm ADJ  
negative ADJ  
cash NOUN  
flow NOUN  
next ADJ  
year NOUN  
low ADJ  
cost NOUN  
per ADP  
ton NOUN  
carbon NOUN  
sequester NOUN  
net ADJ  
value NOUN  
product NOUN  
make VERB  
must AUX  
scalable VERB  
g NOUN  
it PRON  
mean VERB  
price NOUN  
country NOUN  
only ADJ  
difference NOUN  
taxi VERB  
ship NOUN  
this DET  
intend VERB  
earth NOUN  
may AUX  
idea AUX  
apply VERB  
mar NOUN
```



xprize NOUN  
team NOUN  
manage VERB  
100 NUM  
m NOUN  
carbon NOUN  
capture NOUN  
prize NOUN  
everyone PRON  
tesla NOUN  
receive VERB  
stock NOUN  
my PRON  
comp NOUN  
stock NOUN  
option NOUN  
take VERB  
table NOUN  
that DET  
miss VERB  
back ADP  
work NOUN  
go VERB  
do AUX  
seem VERB  
bit NOUN  
high ADJ  
doge ADJ  
appear VERB  
inflationary ADJ  
meaningfully ADV  
fix VERB  
coin NOUN  
per ADP  
unit NOUN  
time NOUN  
whereas SCONJ  
wow INTJ  
1 NUM  
orbital ADJ  
launch NOUN  
tower NOUN  
stack NOUN  
2 NUM  
enough ADJ  
raptor NOUN  
orbit NOUN

```
In [109]: nouns_verbs=[token.text for token in doc_block if token.pos_ in ('NOUN','VERB')]
print(nouns_verbs[100:200])
```

```
['work', 'go', 'seem', 'bit', 'appear', 'fix', 'coin', 'unit', 'time', 'launc
h', 'tower', 'stack', 'raptor', 'orbit', 'booster', 'improve', 'ship', 'booste
r', 'mass', 'work', 'tonight', 'dogecoin', 'video', 'people', 'speak', 'cute',
'image', 'reflect', 'time', 'cost', 'people', 'rain', 'pain', 'let', 'rainfore
st', 'come', 'fun', 'crypto', 'simplicity', 'genius', 'decade', 'work', 'look',
'note', 'email', 'text', 'lesson', 'learn', 'earth', 'mar', 'time', 'tell', 'st
ory', 'tesla', 'spacex', 'read', 'kingdom', 'haven', 'hear', 'name', 'year', 'd
ogecake', 'yolt', 'thing', 'restaurant', 'hang', 'sjm', 'currency', 'earth', 's
cratch', 'u0001f5a4', 'destiny', 'franz', 'say', 'ship', 'landing', 'burn', 'so
lution', 'greate', 'use', 'gas', 'maneuvering', 'rcs', 'thruster', 'seem', 'tur
bopumpfe', 'raptor', 'mu', 'falcon', 'launch', 'mission', '\x96', 'collect', 'd
atum', 'galaxy', 'explore', 'falcon', 'launch', 'starlink', 'satellite']
```

```
In [110]: from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer()

X=cv.fit_transform(nouns_verbs)
sum_words=X.sum(axis=0)

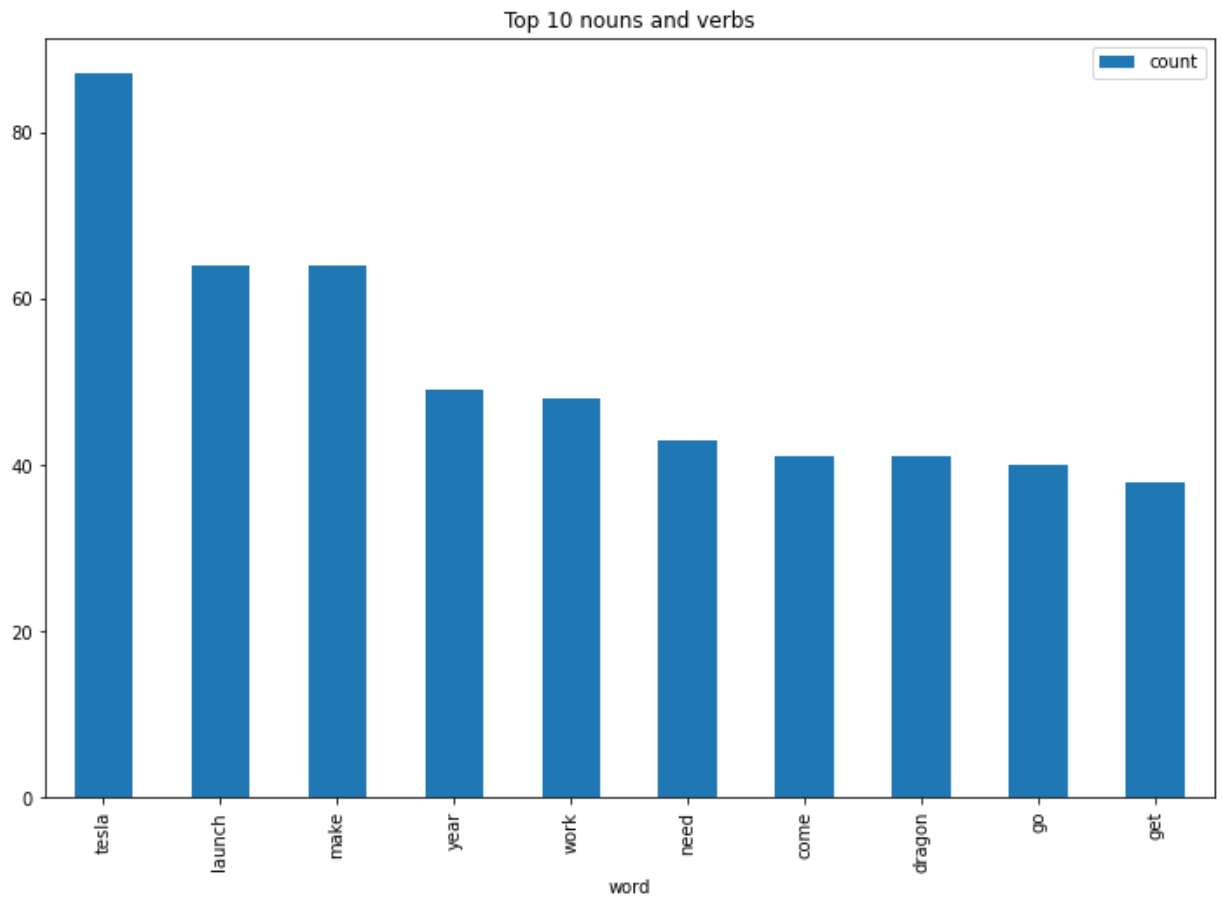
words_freq=[(word,sum_words[0,idx]) for word,idx in cv.vocabulary_.items()]
words_freq=sorted(words_freq, key=lambda x: x[1], reverse=True)

wd_df=pd.DataFrame(words_freq)
wd_df.columns=['word','count']
wd_df[0:10] # viewing top ten results
```

Out[110]:

	word	count
0	tesla	87
1	launch	64
2	make	64
3	year	49
4	work	48
5	need	43
6	come	41
7	dragon	41
8	go	40
9	get	38

```
In [111]: wd_df[0:10].plot.bar(x='word',figsize=(12,8),title='Top 10 nouns and verbs');
```



```
In [112]: from nltk import tokenize
          sentences=tokenize.sent_tokenize(' '.join(tweet))
          sentences
```

```
Out[112]: ['@kunalb11 I\>x92m an alien @ID_AA_Carmack Ray tracing on Cyberpunk with HDR  
is next-level.',  
          'Have you tried it?',  
          '@joerogan @Spotify Great interview!',  
          '@gtera27 Doge is underestimated @teslacn Congratulations Tesla China for am  
azing execution last year.',  
          'Now on to the next for even more!!',  
          'Happy New Year of the Ox!',  
          'https://t.co/9WFKMYu2oj Frodo was the underdodge,\nAll thought he would fai  
l,\nHimself most of all.',  
          'https://t.co/zGxJFDzzrM @OwenSparks_ @flcnhv @anonyx10 Haha thanks :) @flc  
nhv @anonyx10 Indeed!'  
          'Tweets definitely do not represent real-world time allocation.'  
          'The most entertaining outcome is the most likely @GiveDirectly Just sent so  
me Just agree to do Clubhouse with @kanyewest https://t.co/3rWE9uHSTS (http  
s://t.co/3rWE9uHSTS) @geoffkeighley @UnrealEngine It\>x92s getting real Bought  
some Dogecoin for lil X, so he can be a toddler hodler @JoshManMode He defini  
tely has issues, but the sentencing seems a bit high @freewalletorg Thanks fo  
r fixing @freewalletorg Please unlock my account @AstroJordy <U+0001F923><U+0  
001F923> This is t
```

```
In [113]: sent_df=pd.DataFrame(sentences,columns=['sentence'])
          sent_df
```

Out[113]:

	tweet	source	timestamp	language	sentiment	category	score	label
0	@kunalb11 I m an alien @ID_AA_Carmack Ray trac...							
1								Have you tried it?
2		@joerogan @Spotify						Great interview!
3	@gtera27 Doge is underestimated @teslacr Congr...							
4								Now on to the next for even more!!
...								...
919	@kenyanwalstreet Not actually a payout, just a...							
920								It may never pay out, as the stock can t b ht...
921								Details Aug 28.
922	Al symbiosis while u wait @vistacruiser7 @flcn...							
923	@TeslaGong @PPathole Samwise Gamgee @PPathole ...							

924 rows × 1 columns

```
In [114]: affin=pd.read_csv("C:\\Users\\nishi\\Desktop\\Assignments\\Text_Mining\\Afinn.csv")
affin
```

Out[114]:

	word	value
0	abandon	-2
1	abandoned	-2
2	abandons	-2
3	abducted	-2
4	abduction	-2
...	...	...
2472	yucky	-2
2473	yummy	3
2474	zealot	-2
2475	zealots	-2
2476	zealous	2

2477 rows × 2 columns

```
In [115]: affinity_scores=affin.set_index('word')['value'].to_dict()
affinity_scores
```

```
Out[115]: {'abandon': -2,
'abandoned': -2,
'abandons': -2,
'abducted': -2,
'abduction': -2,
'abductions': -2,
'abhor': -3,
'abhorred': -3,
'abhorrent': -3,
'abhors': -3,
'abilities': 2,
'ability': 2,
'aboard': 1,
'absentee': -1,
'absentees': -1,
'absolve': 2,
'absolved': 2,
'absolves': 2,
'absolving': 2,
'absorbed': 1,
```

```
In [116]: nlp=spacy.load('en_core_web_sm')
          sentiment_lexicon=affinity_scores

def calculate_sentiment(text:str=None):
    sent_score=0
    if text:
        sentence=nlp(text)
        for word in sentence:
            sent_score+=sentiment_lexicon.get(word.lemma_,0)
    return sent_score
```

C:\Users\nishi\anaconda3\lib\site-packages\spacy\util.py:833: UserWarning: [W095] Model 'en\_core\_web\_sm' (3.0.0) was trained with spaCy v3.0 and may not be 100% compatible with the current version (3.2.1). If you see errors or degraded performance, download a newer compatible model or retrain your custom model with the current spaCy version. For more details and available updates, run: python -m spacy validate  
Errors.E970.format(str\_command=cmd\_str, tool=cmd\_list[0])

```
In [117]: calculate_sentiment(text='great')
```

Out[117]: 3

```
In [118]: sent_df['sentiment_value']=sent_df['sentence'].apply(calculate_sentiment)
          sent_df['sentiment_value']
```

Out[118]: 0        0  
1        0  
2        3  
3        3  
4        0  
..  
919      0  
920     -4  
921      0  
922     -2  
923      0  
Name: sentiment\_value, Length: 924, dtype: int64

```
In [119]: sent_df['word_count']=sent_df['sentence'].str.split().apply(len)
sent_df['word_count']
```

```
Out[119]: 0      13
1       4
2       4
3      13
4       8
..
919    11
920    31
921     3
922    47
923    15
Name: word_count, Length: 924, dtype: int64
```

```
In [120]: sent_df.sort_values(by='sentiment_value')
```

```
Out[120]:
```

	sentence	sentiment_value	word_count
647	Very ba <a href="https://t.co/tJsh1Exz1Q">https://t.co/tJsh1Exz1Q</a> @justpaulinel...	-8	60
64	Also, the road to hell is mostly paved with ba...	-7	11
837	Cool Model 3 review by @iamjamiefoxx <a href="https://t.co/...">https://t.co/...</a>	-6	61
680	Occasional flights from land are ok, but frequ...	-4	39
920	It may never pay out, as the stock can t b ht...	-4	31
...	...	...	...
81	@teslaownersSV This is a good one @MrBeastYT I...	13	38
585	The open areas <a href="https://t.co/rabjKrtQlw">https://t.co/rabjKrtQlw</a> @Sav...	14	138
719	We just haven t observed the <a href="https://t.co/mez...">https://t.co/mez...</a>	15	72
36	@ajtourville @Erdayastronaut @SpaceX Yes, but ...	15	231
105	@Erdayastronaut @SpaceX Was also thinking that...	16	94

924 rows × 3 columns

```
In [121]: sent_df['sentiment_value'].describe()
```

```
Out[121]: count    924.000000
mean      1.352814
std       2.671421
min      -8.000000
25%       0.000000
50%       0.000000
75%       3.000000
max      16.000000
Name: sentiment_value, dtype: float64
```

```
In [122]: sent_df[sent_df['sentiment_value']<=0]
```

Out[122]:

	sentence	sentiment_value	word_count
0	@kunalb11 I m an alien @ID_AA_Carmack Ray trac...	0	13
1	Have you tried it?	0	4
4	Now on to the next for even more!!	0	8
6	<a href="https://t.co/9WFKMYu2oj">https://t.co/9WFKMYu2oj</a> Frodo was the underdog...	-2	14
8	Tweets definitely do not represent real-world ...	0	8
...	...	...	...
919	@kenyanwalstreet Not actually a payout, just a...	0	11
920	It may never pay out, as the stock can t b ht...	-4	31
921	Details Aug 28.	0	3
922	AI symbiosis while u wait @vistacruiser7 @flcn...	-2	47
923	@TeslaGong @PPathole Samwise Gamgee @PPathole ...	0	15

496 rows × 3 columns

```
In [123]: sent_df[sent_df['sentiment_value']>0]
```

Out[123]:

	sentence	sentiment_value	word_count
2	@joerogan @Spotify Great interview!	3	4
3	@gtera27 Doge is underestimated @teslacn Congr...	3	13
5	Happy New Year of the Ox!	3	6
7	<a href="https://t.co/zGxJFDzzrM">https://t.co/zGxJFDzzrM</a> @OwenSparks_ @flcnhvy ...	2	10
9	The most entertaining outcome is the most like...	3	109
...	...	...	...
911	He was one of the very best.	3	7
913	@Ali_Afshari In general, we need to improve ho...	4	87
915	@burakaydik True Wow, IHOP & GitHub are cl...	7	15
917	This is both great & terrifying.	3	6
918	Everything we ve ever sensed or thought has be...	2	17

428 rows × 3 columns



```
In [124]: sent_df['index']=range(0,len(sent_df))
sent_df
```

Out[124]:

	sentence	sentiment_value	word_count	index
0	@kunalb11 I m an alien @ID_AA_Carmack Ray trac...	0	13	0
1	Have you tried it?	0	4	1
2	@joerogan @Spotify Great interview!	3	4	2
3	@gtera27 Doge is underestimated @teslacn Congr...	3	13	3
4	Now on to the next for even more!!	0	8	4
...	...	...	...	...
919	@kenyanwalstreet Not actually a payout, just a...	0	11	919
920	It may never pay out, as the stock can t b ht...	-4	31	920
921	Details Aug 28.	0	3	921
922	AI symbiosis while u wait @vistacruiser7 @flcn...	-2	47	922
923	@TeslaGong @PPathole Samwise Gamgee @PPathole ...	0	15	923

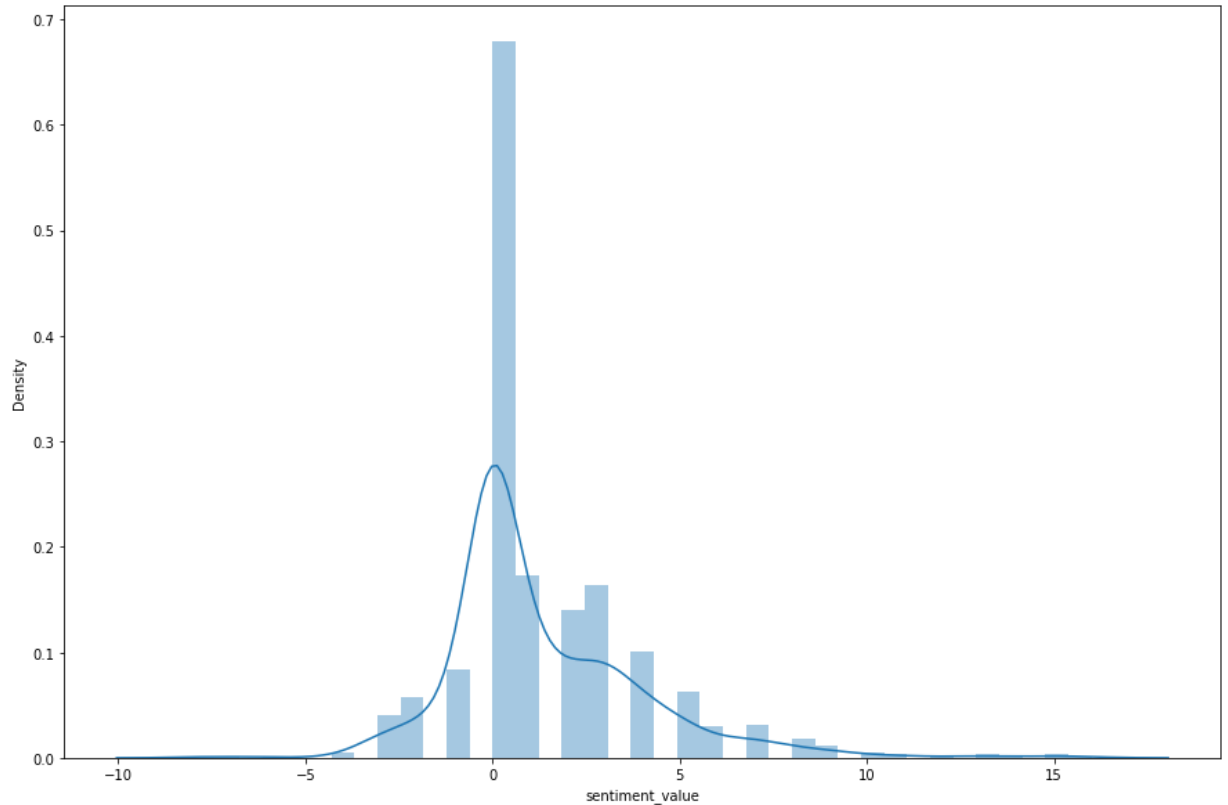
924 rows × 4 columns

```
In [125]: import seaborn as sns
plt.figure(figsize=(15,10))
sns.distplot(sent_df['sentiment_value'])
```

C:\Users\nishi\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

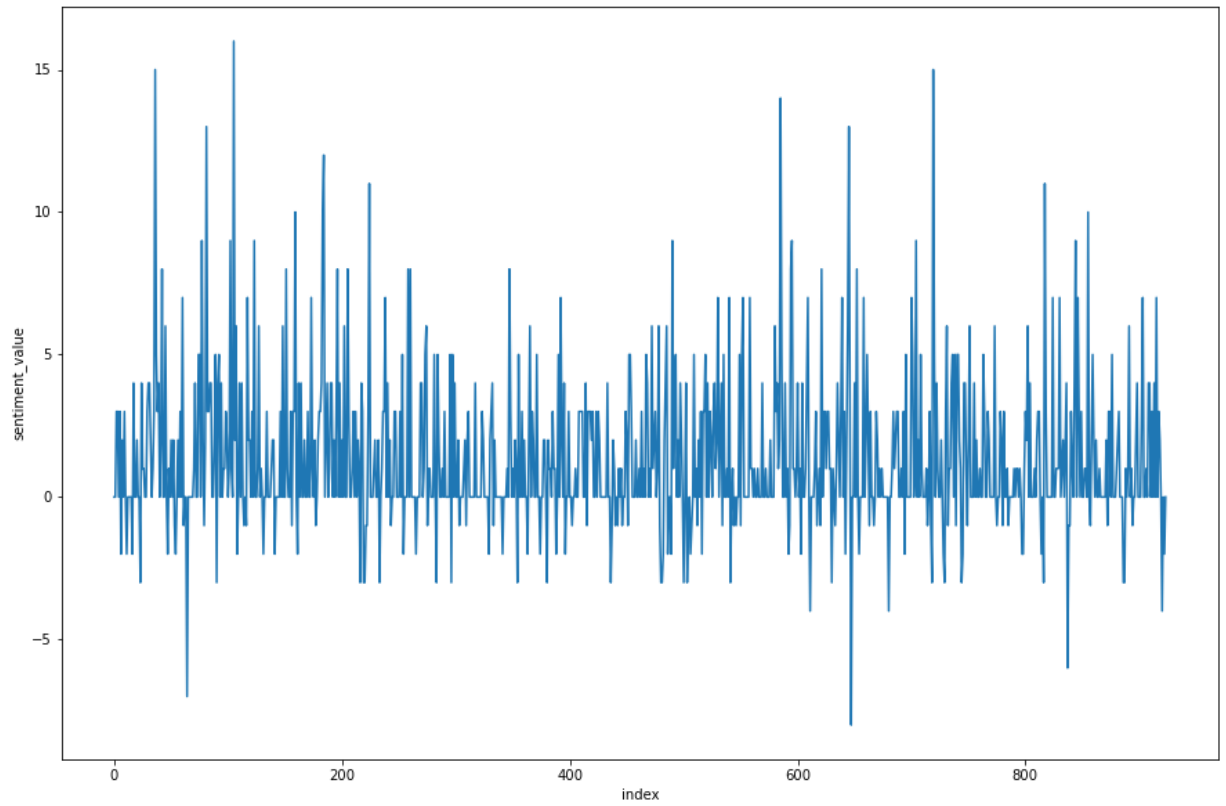
```
warnings.warn(msg, FutureWarning)
```

Out[125]: <AxesSubplot:xlabel='sentiment\_value', ylabel='Density'>



```
In [126]: plt.figure(figsize=(15,10))  
sns.lineplot(y='sentiment_value',x='index',data=sent_df)
```

```
Out[126]: <AxesSubplot:xlabel='index', ylabel='sentiment_value'>
```



```
In [127]: sent_df.plot.scatter(x='word_count',y='sentiment_value',figsize=(8,8),title='Sent
```

```
Out[127]: <AxesSubplot:title={'center':'Sentence sentiment value to sentence word coun  
t'}, xlabel='word_count', ylabel='sentiment_value'>
```

