



Problem Statement:

1) Perform sentimental analysis on the Elon-musk tweets (Exlon-musk.csv) TWO: *1)
Extract reviews of any product from ecommerce website like amazon *2) Perform emotion mining

1) Sentimental Analysis On Elon-Musk Tweets

1. Import Necessary Libraries

```
In [109]: # Import 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
```

2. Import Data

```
In [110]: # Load the dataset
tweets=pd.read_csv('Elon_musk.csv',encoding='Latin-1')
tweets.drop(['Unnamed: 0'],inplace=True,axis=1)
tweets
```

Out[110]:

	Text
0	@kunalb11 I'm an alien
1	@ID_AA_Carmack Ray tracing on Cyberpunk with HDR is next-level. Have you tried it?',
2	@joerogan @Spotify Great interview!',
3	@gtera27 Doge is underestimated',
4	@teslaln Congratulations Tesla China for amazing execution last year. Now on to the next for even more!!',
...	...
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

Text Preprocessing

```
In [111]: tweets=[Text.strip() for Text in tweets.Text] # remove both the leading and the trailing spaces
tweets=[Text for Text in tweets if Text] # removes empty strings, because they are not iterable
tweets[0:10]
```

```
Out[111]: ['@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',
 '@teslaln 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
 j'),)
 'Frodo was the underdog,\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 [112]: *# Joining the list into one string/text*

```
tweets_text=' '.join(tweets)
tweets_text
```

Out[112]: '@kunalb11 I\ux92m 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\ux92s 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\ux92t 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

In [113]: *# remove Twitter username handles from a given twitter text. (Removes @usernames)*

```
from nltk.tokenize import TweetTokenizer
tknzs = TweetTokenizer(strip_handles=True)
tweets_tokens=tknzs.tokenize(tweets_text)
print(tweets_tokens)
```

```
['I', '\ux92', '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', 'Frodo', 'was', 'the', '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', '\ux92', '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', 'Any', 'crypto', 'wallet', 'that', 'won', '\ux92', 't', 'give', 'you', 'your', 'private', 'keys', 'should', 'be', 'avoided', 'at', 'all', 'costs', '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', '@ajtour']
```

```
In [114]: # Again Joining the list into one string/text
tweets_tokens_text=' '.join(tweets_tokens)
tweets_tokens_text
```

```
Out[114]: '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 Chi
na for amazing execution last year . Now on to the next for even more ! ! Hap
py New Year of the Ox ! https://t.co/9WFKMYu2oj (https://t.co/9WFKMYu2oj) Fro
do was the underdoge , All thought he would fail , Himself most of all . http
s://t.co/zGxJFDzzrM (https://t.co/zGxJFDzzrM) Haha thanks :) Indeed ! Tweets
definitely do not represent real-world time allocation . The most entertaini
ng 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 Boug
ht 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/Fc9
uhQsd70 (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 ha
s 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 c
an predict cash flow reasonably well , Starlink will IPO Starlink is a stagge
ringly difficult technical & economic endeavor . However , if we don https://
t.co/9Z8Ac6skqx (https://t.co/9Z8Ac6skqx) SpaceX needs to pass through a deep
```

```
In [115]: # Remove Punctuations
no_punc_text=tweets_tokens_text.translate(str.maketrans('', '',string.punctuation))
no_punc_text
```

```
Out[115]: '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 underdoge 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 [116]: # remove https or url within text
import re
no_url_text=re.sub(r'http\S+', '', no_punc_text)
no_url_text
```

```
Out[116]: '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 underdoge 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
```

```
In [117]: 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
```

In [118]: *# Tokenization*

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

```
[nltk_data] Downloading package punkt to
[nltk_data] C:\Users\nandini\AppData\Roaming\nltk_data...
[nltk_data] Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\nandini\AppData\Roaming\nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

Out[118]: True

In [119]: *# Tokens count*

```
len(text_tokens)
```

Out[119]: 17847

In [120]: *# Remove Stopwords*

```
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)
```

```
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
s', 'apply', 'Mars', 'XPrize', 'team', 'manage', '100M', 'carbon', 'capture',
'prize', 'Everyone', 'Tesla', 'receives', 'stock', 'My', 'comp', 'stock', 'op
tions', 'take', 'table', 'That', 'missing', 'Back', 'work', 'go', 'Does', 'se
em', 'bit', 'high', 'Doge', 'appears', 'inflationary', 'meaningfully', 'fixe
d', 'coins', 'per', 'unit', 'time', 'whereas', 'Wow', '1', 'Orbital', 'launc
h', 'tower', 'stack', '2', 'Enough', 'Raptors', 'orbit', 'booster', '3', 'Imp
rove', 'ship', 'booster', 'mass', 'Back', 'work', 'tonight', 'D', 'Dogecoin',
'Instructional', 'video', 'The', 'people', 'spoken', 'So', 'cute', 'U0001F49
5', 'Extremely', 'misleading', 'image', 'reflect', 'true', 'time', 'cost', 'p
eople', 'rain', 'pain', 'U0001F3B6', 'Who', 'let', 'Doge', 'U0001F3B6', 'Hod
l', 'rainforests', 'So', 'finally', 'come', 'It', 'fun', 'crypto', 'Its', 'si
mplicity', 'genius', 'Yup', 'True', 'Not', 'easy', 'This', 'two', 'decades',
'intense', 'work', 'Have', 'look', 'old', 'notes', 'emails', 'texts', 'Lesson
s', 'learned', 'Of', 'Earth', 'Mars', 'Time', 'tell', 'story', 'Tesla', 'Spac
eX'. 'Have'. 'read'. 'It'. 'great'. 'The'. 'Second'. 'Last'. 'Kingdom'. 'Have
```

```
In [121]: # Normalize the data
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 [122]: # Stemming (Optional)
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 [123]: pip install spacy
```

```
Requirement already satisfied: spacy in c:\users\nandini\anaconda3\lib\site-packages (3.2.1)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.8 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (3.0.8)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (2.25.1)
Requirement already satisfied: pydantic!=1.8,!1.8.1,<1.9.0,>=1.7.4 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (1.8.2)
Requirement already satisfied: typer<0.5.0,>=0.3.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (0.4.0)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (1.0.6)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (1.0.1)
Requirement already satisfied: wasabi<1.1.0,>=0.8.1 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (0.9.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (4.59.0)
Requirement already satisfied: blis<0.8.0,>=0.4.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (0.7.5)
Requirement already satisfied: pathy>=0.3.5 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (0.6.1)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (2.0.6)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (2.0.6)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (3.3.0)
Requirement already satisfied: packaging>=20.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (20.9)
Requirement already satisfied: setuptools in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (52.0.0.post20210125)
Requirement already satisfied: jinja2 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (2.11.3)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (3.0.6)
Requirement already satisfied: thinc<8.1.0,>=8.0.12 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (8.0.13)
Requirement already satisfied: srsly<3.0.0,>=2.4.1 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (2.4.2)
Requirement already satisfied: numpy>=1.15.0 in c:\users\nandini\anaconda3\lib\site-packages (from spacy) (1.20.1)
Requirement already satisfied: pyparsing>=2.0.2 in c:\users\nandini\anaconda3\lib\site-packages (from packaging>=20.0->spacy) (2.4.7)
Requirement already satisfied: smart-open<6.0.0,>=5.0.0 in c:\users\nandini\anaconda3\lib\site-packages (from pathy>=0.3.5->spacy) (5.2.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\nandini\anaconda3\lib\site-packages (from pydantic!=1.8,!1.8.1,<1.9.0,>=1.7.4->spacy) (3.7.4.3)
Requirement already satisfied: idna<3,>=2.5 in c:\users\nandini\anaconda3\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\nandini\anaconda3\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (1.26.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\nandini\anaconda3\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2020.12.5)
```


Requirement already satisfied: chardet<5,>=3.0.2 in c:\users\nandini\anaconda3\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (4.0.0)
 Requirement already satisfied: click<9.0.0,>=7.1.1 in c:\users\nandini\anaconda3\lib\site-packages (from typer<0.5.0,>=0.3.0->spacy) (7.1.2)
 Requirement already satisfied: MarkupSafe>=0.23 in c:\users\nandini\anaconda3\lib\site-packages (from jinja2->spacy) (1.1.1)
 Note: you may need to restart the kernel to use updated packages.

In [124]: `import spacy`

In [125]: `# Lemmatization
 nlp=spacy.load('en_core_web_sm')
 doc=nlp(' '.join(lower_words))
 print(doc)`

alien ray tracing cyberpunk hdr nextlevel have tried great interview doge und
 erestimated congratulations tesla china amazing execution last year now next
 even happy new year ox frodo underdoge all thought would fail himself haha th
 anks indeed tweets definitely represent realworld time allocation the enterta
 ining outcome likely just sent just agree clubhouse it getting real bought do
 gecoin lil x toddler hodler he definitely issues sentencing seems bit high th
 anks fixing please unlock account this true power any crypto wallet give priv
 ate keys avoided costs your app sucks rt nasa selected falcon heavy launch fi
 rst two elements lunar gateway together one mission once predict cash flow re
 asonably well starlink ipo starlink staggeringly difficult technical economic
 endeavor however spacex needs pass deep chasm negative cash flow next year lo
 west cost per ton carbon sequestered net value product made must scalable g i
 t meant price countries only difference taxes shipping this intended earth ma
 y ideas apply mars xprize team manage 100m carbon capture prize everyone tesl
 a receives stock my comp stock options take table that missing back work go d
 oes seem bit high doge appears inflationary meaningfully fixed coins per unit
 time whereas wow 1 orbital launch tower stack 2 enough raptors orbit booster
 3 improve ship booster mass back work tonight ð ðogecoin instructional video
 the people spoken so cute u0001f495 extremely misleading image reflect true t



2. CountVectorizer with N-grams (Bigrams & Trigrams)

```
In [133]: cv_ngram_range=CountVectorizer(analyzer='word',ngram_range=(1,3),max_features=1000)
bow_matrix_ngram=cv_ngram_range.fit_transform(lemmas)
```

```
In [134]: print(cv_ngram_range.get_feature_names())
print(bow_matrix_ngram.toarray())
```

```
['actually', 'ai', 'also', 'back', 'big', 'booster', 'car', 'come', 'cool', 'co
uld', 'crew', 'day', 'design', 'do', 'dragon', 'earth', 'engine', 'even', 'eve
r', 'exactly', 'falcon', 'first', 'flight', 'fsd', 'future', 'get', 'go', 'goo
d', 'great', 'haha', 'hard', 'high', 'if', 'it', 'just', 'land', 'launch', 'lif
e', 'like', 'look', 'lot', 'love', 'make', 'many', 'mar', 'maybe', 'mission',
'model', 'much', 'need', 'new', 'next', 'no', 'not', 'ok', 'one', 'part', 'peop
le', 'point', 'pretty', 'probably', 'production', 'right', 'rocket', 'rt', 'sa
y', 'seem', 'soon', 'space', 'spacex', 'starlink', 'starship', 'still', 'supe
r', 'sure', 'take', 'tesla', 'test', 'thank', 'that', 'the', 'there', 'they',
'think', 'this', 'time', 'true', 'try', 'ufe0f', 'use', 'way', 'we', 'week', 'w
ell', '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]]
```

3. TF-IDF Vectorizer

```
In [135]: from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_ngram_max_features=TfidfVectorizer(norm='l2',analyzer='word',ngram_range=(
tfidf_matix_ngram=tfidf_ngram_max_features.fit_transform(lemmas)
```

```
In [136]: print(tfidf_v_ngram_max_features.get_feature_names())
print(tfidf_matrix_ngram.toarray())
```

```
['10', '100', '13', '20', '2020', '30', '300', '40', '4d', '60', 'able', 'absol
utely', 'access', 'accurate', 'actually', 'add', 'ago', 'agree', 'ai', 'aim',
'air', 'all', 'almost', 'already', 'also', 'although', 'always', 'amazing', 'an
d', 'another', 'appear', 'apply', 'appreciate', 'approval', 'arm', 'around', 'a
rticle', 'as', 'astronaut', 'at', 'autopilot', 'awesome', 'back', 'bad', 'bar',
'base', 'battery', 'be', 'become', 'berlin', 'beta', 'big', 'bit', 'boca', 'bod
y', 'booster', 'brain', 'bring', 'bs', 'build', 'burn', 'but', 'buy', 'cake',
'call', 'camera', 'can', 'car', 'cargo', 'cast', 'catch', 'cell', 'ch4', 'chanc
e', 'change', 'china', 'civilization', 'close', 'cold', 'color', 'come', 'compa
ny', 'complete', 'computer', 'confirm', 'congratulation', 'control', 'cool', 'c
ost', 'could', 'course', 'cov', 'crazy', 'create', 'crew', 'critical', 'cureva
c', 'cyberpunk', 'cybertruck', 'datum', 'day', 'deep', 'definitely', 'deliver
y', 'design', 'detail', 'difference', 'different', 'do', 'dock', 'doge', 'doj
o', 'dragon', 'drive', 'droneship', 'early', 'earth', 'easy', 'electric', 'enab
le', 'end', 'energy', 'engine', 'engineer', 'engineering', 'enough', 'entire',
'especially', 'essential', 'esthetic', 'et', 'even', 'ever', 'every', 'everyon
e', 'everything', 'exactly', 'expect', 'extreme', 'extremely', 'fair', 'falco
n', 'far', 'fast', 'favorite', 'feel', 'fine', 'fire', 'first', 'fix', 'flap',
'flight', 'flow', 'fly', 'for', 'fsd', 'ft', 'full', 'fully', 'fun', 'fundament
al', 'future', 'game', 'gas', 'gene', 'get', 'giant', 'giga', 'give', 'glad',
'go', 'goal', 'good', 'government', 'great', 'haha', 'half', 'happen', 'hard',
'have', 'he', 'head', 'hear', 'heavy', 'help', 'high', 'hold', 'home', 'hop',
'hope', 'hopefully', 'hot', 'hour', 'human', 'idea', 'if', 'imo', 'important',
'impressive', 'improve', 'improvement', 'in', 'increase', 'indeed', 'inferenc
e', 'interesting', 'irony', 'it', 'just', 'km', 'know', 'la', 'lab', 'land', 'l
anding', 'large', 'last', 'later', 'launch', 'least', 'leave', 'leg', 'less',
'let', 'level', 'life', 'liftoff', 'like', 'likely', 'limit', 'liquid', 'litera
lly', 'little', 'live', 'long', 'longterm', 'look', 'lose', 'lot', 'love', 'lo
w', 'machine', 'main', 'major', 'make', 'many', 'mar', 'mass', 'matter', 'may',
'maybe', 'mean', 'might', 'mile', 'million', 'min', 'mind', 'minute', 'missio
n', 'model', 'money', 'month', 'moon', 'most', 'mostly', 'motor', 'mount', 'mov
e', 'movie', 'much', 'must', 'my', 'name', 'nasa', 'need', 'net', 'neural', 'ne
uralink', 'never', 'new', 'news', 'next', 'night', 'no', 'nosecone', 'not', 'no
te', 'nothing', 'now', 'number', 'of', 'ok', 'old', 'one', 'only', 'open', 'ope
rational', 'option', 'orbit', 'orbital', 'original', 'pad', 'part', 'pass', 'pc
r', 'people', 'per', 'person', 'physics', 'place', 'play', 'please', 'pm', 'poi
nt', 'positive', 'possible', 'potential', 'power', 'pressure', 'pretty', 'pric
e', 'primarily', 'prior', 'probability', 'probably', 'problem', 'product', 'pro
duction', 'progress', 'propellant', 'prototype', 'provide', 'public', 'push',
'put', 'quite', 'range', 'raptor', 'reach', 'read', 'real', 'really', 'releas
e', 'require', 'return', 'review', 'right', 'rise', 'roadster', 'rocket', 'rt',
'safe', 'safety', 'satellite', 'say', 'scale', 'second', 'see', 'seem', 'sell',
'send', 'sense', 'separation', 'service', 'set', 'several', 'ship', 'short', 's
hot', 'should', 'show', 'single', 'site', 'slightly', 'slow', 'small', 'sn5',
'sn8', 'sn9', 'so', 'software', 'solar', 'solution', 'solve', 'some', 'somethin
g', 'song', 'soon', 'sorry', 'sound', 'south', 'space', 'spacecraft', 'spacex',
'speed', 'splashdown', 'stage', 'starlink', 'starship', 'start', 'static', 'sta
tion', 'steel', 'step', 'still', 'stock', 'story', 'super', 'support', 'sure',
'symptom', 'system', 'take', 'talk', 'tank', 'target', 'tbh', 'team', 'technica
lly', 'technology', 'tesla', 'test', 'testing', 'texas', 'thank', 'that', 'th
e', 'there', 'they', 'thing', 'think', 'this', 'three', 'thrust', 'time', 'to',
'today', 'tomorrow', 'ton', 'tonight', 'top', 'total', 'tough', 'traffic', 'tra
ining', 'travel', 'true', 'try', 'tunnel', 'turn', 'twitter', 'two', 'u0001f3b
6', 'u0001f440', 'u0001f495', 'u0001f525', 'u0001f5a4', 'u043b', 'u043e', 'u219
```

```

2', 'ufe0f', 'understand', 'update', 'upgrade', 'ur', 'use', 'useful', 'vaccin
e', 'value', 'vehicle', 'version', 'very', 'video', 'view', 'volume', 'vs', 'wa
it', 'want', 'watch', 'way', 'we', 'week', 'welcome', 'well', 'what', 'will',
'without', 'word', 'work', 'world', 'worth', 'would', 'wow', 'yeah', 'year', 'y
es', '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.]]

```

Generate Word Cloud

In [137]: `pip install wordcloud`

Requirement already satisfied: wordcloud in c:\users\nandini\anaconda3\lib\site-packages (1.8.1)Note: you may need to restart the kernel to use updated packages.

Requirement already satisfied: pillow in c:\users\nandini\anaconda3\lib\site-packages (from wordcloud) (8.2.0)

Requirement already satisfied: numpy>=1.6.1 in c:\users\nandini\anaconda3\lib\site-packages (from wordcloud) (1.20.1)

Requirement already satisfied: matplotlib in c:\users\nandini\anaconda3\lib\site-packages (from wordcloud) (3.3.4)

Requirement already satisfied: cycler>=0.10 in c:\users\nandini\anaconda3\lib\site-packages (from matplotlib->wordcloud) (0.10.0)

Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\users\nandini\anaconda3\lib\site-packages (from matplotlib->wordcloud) (2.4.7)

Requirement already satisfied: python-dateutil>=2.1 in c:\users\nandini\anaconda3\lib\site-packages (from matplotlib->wordcloud) (2.8.1)

Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\nandini\anaconda3\lib\site-packages (from matplotlib->wordcloud) (1.3.1)

Requirement already satisfied: six in c:\users\nandini\anaconda3\lib\site-packages (from cycler>=0.10->matplotlib->wordcloud) (1.15.0)

In [138]: `from wordcloud import WordCloud, STOPWORDS`

```
In [139]: # Define 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)
```



Named Entity Recognition (NER)


```
In [140]: # Parts Of Speech (POS) Tagging
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 trace cyberpunk hdr nextlevel have try great interview doge underestimate
congratulation tesla china GPE amazing execution last year DATE now next even happy
new year DATE ox frodo underdoge all thought 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 PRODUCT 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 ORG 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

```
In [141]: for token in doc_block[100:200]:  
          print(token,token.pos_)
```

```
once ADV  
predict VERB  
cash NOUN  
flow NOUN  
reasonably ADV  
well ADV  
starlink VERB  
ipo NOUN  
starlink NOUN  
staggeringly ADV  
difficult ADJ  
technical ADJ  
economic ADJ  
endeavor NOUN  
however ADV  
spacex VERB  
need NOUN  
pass VERB  
deep ADJ  
chasm NOUN  
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 ADV  
it PRON  
mean VERB  
price NOUN  
country NOUN  
only ADV  
difference NOUN  
taxi NOUN  
ship NOUN  
this PRON  
intend ADJ  
earth NOUN  
may AUX  
idea VERB  
apply VERB  
mars PROP  
xprize PROP
```



team NOUN
manage VERB
100 NUM
m VERB
carbon NOUN
capture NOUN
prize NOUN
everyone PRON
tesla ADV
receive VERB
stock NOUN
my PRON
comp NOUN
stock NOUN
option NOUN
take VERB
table NOUN
that PRON
miss VERB
back ADJ
work NOUN
go NOUN
do AUX
seem VERB
bit ADV
high ADJ
doge PROP
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 VERB
2 NUM
enough ADJ
raptor NOUN
orbit NOUN

```
In [142]: # Filtering nouns & verbs only
nouns_verbs = [token.text for token in doc_block if token.pos_ in ('NOUN', 'VERB')]
print(nouns_verbs[100:200])

['time', 'launch', 'tower', 'stack', 'raptor', 'orbit', 'booster', 'improve',
'ship', 'booster', 'mass', 'work', 'tonight', 'dogecoin', 'video', 'people', 's
peak', 'cute', 'image', 'reflect', 'time', 'cost', 'people', 'rain', 'pain', 'u
0001f3b6', 'let', 'come', 'fun', 'crypto', 'simplicity', 'genius', 'yup', 'deca
de', 'work', 'look', 'note', 'email', 'text', 'lesson', 'learn', 'earth', 'tim
e', 'tell', 'story', 'tesla', 'spacex', 'read', 'hear', 'name', 'year', 'dogeca
ke', 'yolt', 'thing', 'restaurant', 'get', 'hang', 'stranger', 'sjm', 'currenc
y', 'earth', 'scratch', 'u0001f5a4', 'destiny', 'franz', 'say', 'ship', 'landin
g', 'burn', 'solution', 'greate', 'use', 'gas', 'maneuvering', 'rcs', 'thruste
r', 'seem', 'turbopumpfed', 'raptor', 'falcon', 'launch', 'collect', 'galaxy',
'explore', 'launch', 'starlink', 'satellite', 'orbit', 'mission', 'pad', '39a',
'deck', 'default', 'engine', 'lever', 'arm', 'shut', 'engine', 'min', 'throttl
e']
```

```
In [143]: # Counting the noun & verb tokens
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)

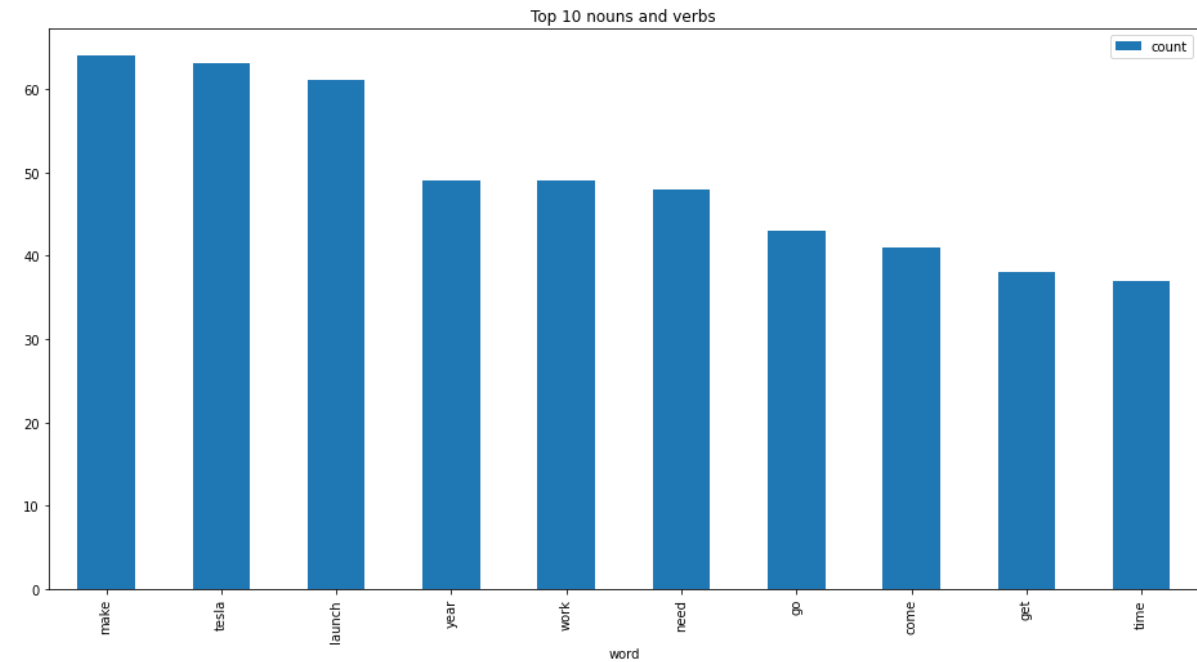
wd_df=pd.DataFrame(words_freq)
wd_df.columns=['word', 'count']
wd_df[0:10] # viewing top ten results
```

```
Out[143]:
```

	word	count
0	make	64
1	tesla	63
2	launch	61
3	year	49
4	work	49
5	need	48
6	go	43
7	come	41
8	get	38
9	time	37

Visualizing the results(Bar charts for top 10 nouns & verbs)

```
wd_df[0:10].plot.bar(x='word',figsize=(16,8),title='Top 10 nouns and verbs');
```



Emotion Mining | Sentimental Analysis

```
from nltk import tokenize
```

```
sentence = tokenize.sent_tokenize(''.join(tweets))
sentence
```

```
[ '@kunalb11 I\x92m an alien@ID_AA_Carmack Ray tracing on Cyberpunk with HDR i
s next-level.',
'Have you tried it?',
'@joerogan @Spotify Great interview!',
'@gtera27 Doge is underestimated@teslacr Congratulations Tesla China for ama
zing execution last year.',
'Now on to the next for even more!',
'!Happy New Year of the Ox!',
'https://t.co/9WFKMYu2ojFrodo was the underdoge,\nAll thought he would fai
l,\nHimself most of all.',
'https://t.co/zGxJFDzzrM@OwenSparks_ @flcnhvy @anonyx10 Haha thanks :)@flcnh
vy @anonyx10 Indeed!',
'Tweets definitely do not represent real-world time allocation.The most ente
rtaining outcome is the most likely@GiveDirectly Just sent someJust agree to
do Clubhouse with @kanyewesthttps://t.co/3rWE9uHSTS@geoffkeighley @UnrealEngi
ne It\x92s getting realBought some Dogecoin for lil X, so he can be a toddler
hodler@JoshManMode He definitely has issues, but the sentencing seems a bit h
igh@freewalletorg Thanks for fixing@freewalletorg Please unlock my account@As
troJordy <U+0001F923><U+0001F923>This is true power haha https://t.co/Fc9uhQS
```

```
In [151]: sent_df = pd.DataFrame(sentence, columns = ['sentence'])
sent_df
```

```
Out[151]:
```

	sentence
0	@kunalb11 I'm an alien@ID_AA_Carmack Ray traci...
1	Have you tried it?
2	@joerogan @Spotify Great interview!
3	@gtera27 Doge is underestimated@teslacr Congra...
4	Now on to the next for even more!
...	...
854	Everything we've ever sensed or thought has be...
855	@kenyanwalstreet Not actually a payout, just a...
856	It may never pay out, as the stock can't b... ht...
857	Details Aug 28.AI symbiosis while u wait@vista...
858	@TeslaGong @PPathole Samwise Gamgee@PPathole A...

859 rows × 1 columns

Emotion lexicon affin

```
In [153]: affin = pd.read_csv('Afinn.csv', sep=',', encoding='Latin-1')
affin
```

```
Out[153]:
```

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

```
Out[154]: {'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,
...}
```

Custom function: score each word in a sentence in lemmatised form, but calculate the score for the whole original sentence

```
In [161]: 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
```

Manual Testing

```
In [162]: calculate_sentiment(text = 'great')
```

```
Out[162]: 3
```

Calculate the sentimate value for each sentence


```
In [163]: sent_df['sentiment_value']=sent_df['sentence'].apply(calculate_sentiment)
sent_df['sentiment_value']
```

```
Out[163]: 0      0
          1      0
          2      0
          3      0
          4      0
          ..
         854     0
         855     0
         856     0
         857     0
         858     0
          Name: sentiment_value, Length: 859, dtype: int64
```

How many words are there in a sentence

```
In [164]: sent_df['word_count']=sent_df['sentence'].str.split().apply(len)
sent_df['word_count']
```

```
Out[164]: 0      12
          1       4
          2       4
          3      12
          4       8
          ..
         854     15
         855     11
         856     30
         857     46
         858     13
          Name: word_count, Length: 859, dtype: int64
```

```
In [165]: sent_df.sort_values(by='sentiment_value')
```

```
Out[165]:
```

	sentence	sentiment_value	word_count
576	Problems like that have been addressed.	-2	6
830	No expert installer required.	-1	4
666	Hard to see with naked eye.	-1	6
131	Stop the sneaky moves, Charlie Ergen!Entertain...	-1	22
74	No question that FSD should be viewed as reaso...	-1	14
...
69	Excited about offering low-cost access to o... h...	3	22
578	Good insight.	3	2
84	Good progress towards our "Hop in & go to ...	3	18
746	Best boss battle song ever.	3	5
230	Wow, roc... https://t.co/0SRIFg03Fg @TashaARK To ...	4	12

859 rows × 3 columns

Sentiment score of the whole review

```
In [166]: sent_df['sentiment_value'].describe()
```

```
Out[166]: count      859.000000
mean         0.055879
std          0.417169
min          -2.000000
25%          0.000000
50%          0.000000
75%          0.000000
max           4.000000
Name: sentiment_value, dtype: float64
```

Negative sentiment score of the whole review

```
In [167]: sent_df[sent_df['sentiment_value']<=0]
```

```
Out[167]:
```

	sentence	sentiment_value	word_count
0	@kunalb11 I'm an alien@ID_AA_Carmack Ray traci...	0	12
1	Have you tried it?	0	4
2	@joerogan @Spotify Great interview!	0	4
3	@gtera27 Doge is underestimated@teslacr Congra...	0	12
4	Now on to the next for even more!	0	8
...
854	Everything we've ever sensed or thought has be...	0	15
855	@kenyanwalstreet Not actually a payout, just a...	0	11
856	It may never pay out, as the stock can't b... ht...	0	30
857	Details Aug 28.AI symbiosis while u wait@vista...	0	46
858	@TeslaGong @PPathole Samwise Gamgee@PPathole A...	0	13

829 rows × 3 columns

Positive sentiment score of the whole review

```
In [168]: sent_df[sent_df['sentiment_value']>0]
```

```
Out[168]:
```

	sentence	sentiment_value	word_count
18	Improve ship & booster mass@michaelhodapp_...	2	12
69	Excited about offering low-cost access to o... h...	3	22
84	Good progress towards our "Hop in & go to ...	3	18
88	Help would be appreciated.	2	4
100	Hopefully.	2	1
113	Huge thanks to all those who made this possible.	1	9
135	Great game.	3	2
160	Hopefully launching tomorrow.	2	3
216	Improvement should be significant.	2	4
230	Wow, roc... https://t.co/0SRIFg03Fg@TashaARK To ...	4	12
264	Hopefully, more Teslaquila next month.	2	5
302	Hilarious that he has to clarify that he r... ht...	2	35

Adding index cloumn

```
In [169]: sent_df['index']=range(0,len(sent_df))
sent_df
```

```
Out[169]:
```

	sentence	sentiment_value	word_count	index
0	@kunalb11 I'm an alien@ID_AA_Carmack Ray traci...	0	12	0
1	Have you tried it?	0	4	1
2	@joerogan @Spotify Great interview!	0	4	2
3	@gtera27 Doge is underestimated@teslacn Congra...	0	12	3
4	Now on to the next for even more!	0	8	4
...
854	Everything we've ever sensed or thought has be...	0	15	854
855	@kenyanwalstreet Not actually a payout, just a...	0	11	855
856	It may never pay out, as the stock can't b... ht...	0	30	856
857	Details Aug 28.AI symbiosis while u wait@vista...	0	46	857
858	@TeslaGong @PPathole Samwise Gamgee@PPathole A...	0	13	858

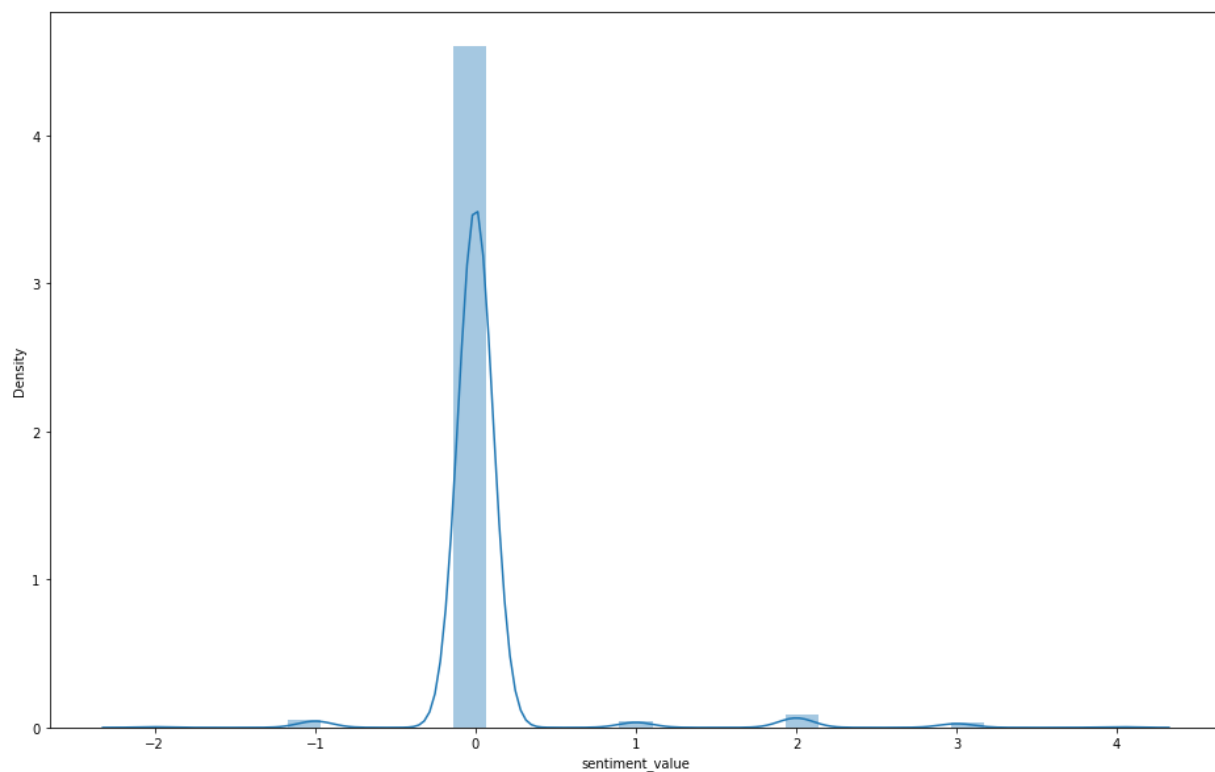
859 rows × 4 columns

Ploting sentiment value for the whole review

```
In [174]: import warnings
warnings.filterwarnings('ignore')
```

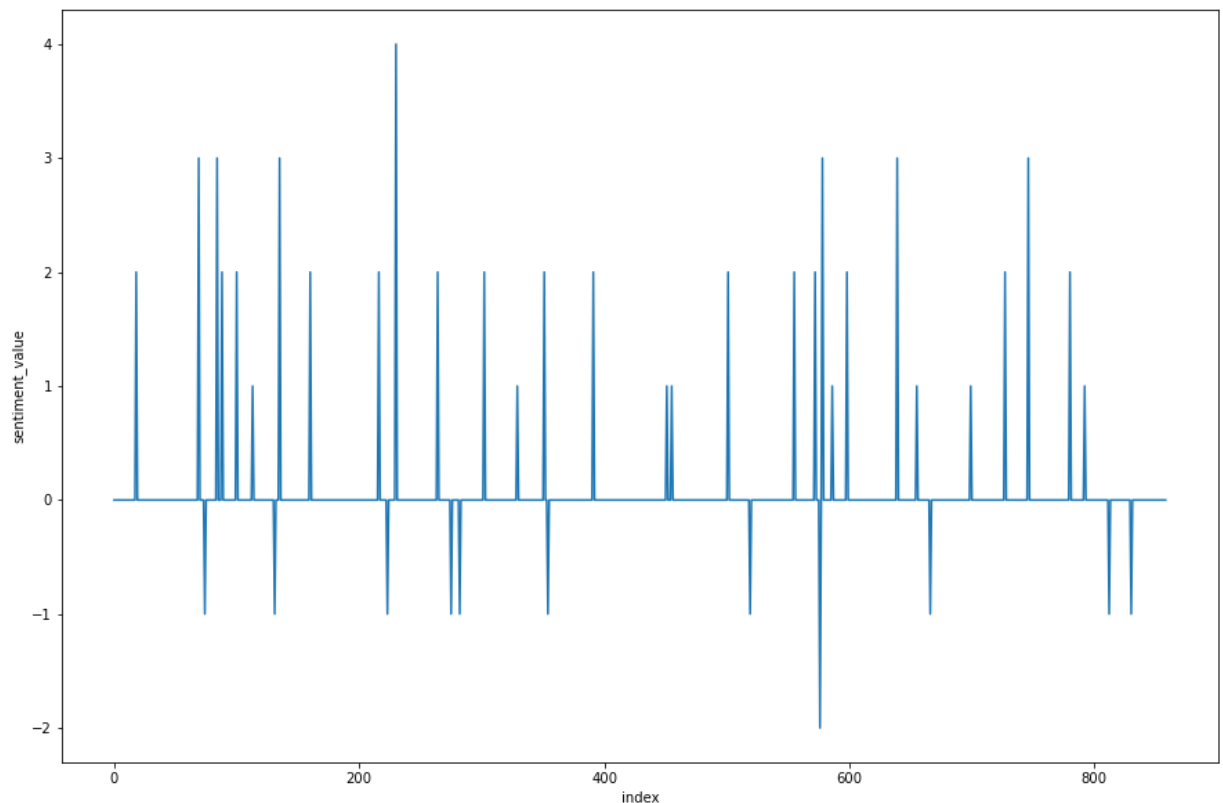


```
In [178]: plt.figure(figsize=(16,10))  
sns.distplot(sent_df['sentiment_value'])  
plt.show()
```



Line plot for the sentiment value of whole review

```
In [181]: plt.figure(figsize=(15,10))  
sns.lineplot(y='sentiment_value', x='index', data = sent_df)  
plt.show()
```



Correlation Analysis

```
In [185]: sent_df.plot.scatter(x='word_count', y='sentiment_value', figsize=(15,10), title  
plt.show()
```



THE END