

# Assignment-11-Text-Mining-01 (Elon-Musk)

In [1]:

```
# Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import string
import spacy

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

In [2]:

```
# Load the dataset
tweets=pd.read_csv("C:/Users/LENOVO/Documents/Custom Office Templates/Elon_musk.csv", encoding='utf-8')
tweets
```

Out[2]:

Unnamed: 0		Text
0	1	@kunalb11 I'm an alien
1	2	@ID_AA_Carmack Ray tracing on Cyberpunk with H...
2	3	@joerogan @Spotify Great interview!
3	4	@gtera27 Doge is underestimated
4	5	@teslacr Congratulations Tesla China for amazi...
...	...	...
1994	1995	@flcnhvy True, it sounds so surreal, but the n...
1995	1996	@PPathole Make sure to read ur terms & con...
1996	1997	@TeslaGong @PPathole Samwise Gamgee
1997	1998	@PPathole Altho Dumb and Dumber is <U+0001F525...
1998	1999	Progress update August 28

1999 rows × 2 columns

In [3]:

```
tweets.drop(['Unnamed: 0'],inplace=True,axis=1)
tweets
```

Out[3]:

	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

## Text Preprocessing

In [4]:

```
tweets = [Text.strip() for Text in tweets.Text] # remove both the leading and the trailing
tweets = [Text for Text in tweets if Text] # removes empty strings, because they are consid
tweets[0:10]
```

Out[4]:

```
['@kunalb11 I\'x92m an alien',
 '@ID_AA_Carmack Ray tracing on Cyberpunk with HDR is next-level. Have you t
ried 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 underdodge,\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 t
ime allocation.',
 'The most entertaining outcome is the most likely']
```

In [5]:

```
# Joining the list into one string/text
tweets_text = ' '.join(tweets)
tweets_text
```

Out[5]:

'@kunalb11 I\ x2m an alien @ID\_AA\_Carmack Ray tracing on Cyberpunk with HD R is next-level. Have you tried it? @joerogan @Spotify Great interview! @tera27 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\ x2s 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\ x2t give you your private keys should be avoided at all costs @freewalletorg Your app sucks RT @SnaceX: NASA has selected Falcon H

In [6]:

```
# remove Twitter username handles from a given twitter text. (Removes @usernames)
from nltk.tokenize import TweetTokenizer
tknznr=TweetTokenizer(strip_handles=True)
tweets_tokens=tknznr.tokenize(tweets_text)
print(tweets_tokens)
```

['I', '\ x2', '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', '\ x2', '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', '\ x2t', 'give', 'you', 'your', 'private', 'keys', 'should', 'be', 'avoided', 'at', 'all', 'costs', 'Your', 'app', 'sucks', 'RT', '@SnaceX: NASA has selected Falcon H

In [7]:

```
# again joining the list into one string/text
tweets_tokens_text=' '.join(tweets_tokens)
tweets_tokens_text
```

Out[7]:

'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 . <https://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 techni

In [8]:

```
# Remove Punctuations
no_punc_text=tweets_tokens_text.translate(str.maketrans('', '',string.punctuation))
no_punc_text
```

Out[8]:

'I \x92 m an alien Ray tracing on Cyberpunk with HDR is nextlevel 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 httpstco9WFKMYu2oj Frodo was the underdog All thought he would 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 httpstco3rWE9uHSTS 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 Gateway together on one mission httpstco3pWt Yes Once we can predict cash flow reasonably well Starlink will IPO Starlink is a staggeringly difficult technical economic endeavor However if we don httpstco9Z8Ac6skqx SpaceX needs to pass through a deep chasm of negative cash flow over the next year or httpstco7J1c92hdil lowest cost ne

In [9]:

```
# remove https or url within text
import re
no_url_text=re.sub(r'https\S+', '',no_punc_text)
no_url_text
```

Out[9]:

'I \x92 m an alien Ray tracing on Cyberpunk with HDR is nextlevel Have yo  
u 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 Frodo was the underdoge All thought he would f  
ail Himself most of all Haha thanks Indeed Tweets definitely do not r  
epresent realworld time allocation The most entertaining outcome is the m  
ost likely Just sent some Just agree to do Clubhouse with It \x92 s getti  
ng real Bought some Dogecoin for lil X so he can be a toddler hodler He d  
efinitely has issues but the sentencing seems a bit high Thanks for fixin  
g Please unlock my account U0001F923 U0001F923 This is true power haha An  
y crypto wallet that won \x92 t give you your private keys should be avoid  
ed at all costs Your app sucks RT NASA has selected Falcon Heavy to launc  
h the first two elements of the lunar Gateway together on one mission Ye  
s Once we can predict cash flow reasonably well Starlink will IPO Starlin  
k is a staggeringly difficult technical economic 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 sequestered net of value  
of anv product made Must be scalable to g It \x92 s meant to be the same

In [10]:

```
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', 'wi  
th', 'HDR', 'is', 'nextlevel', 'Have', 'you', 'tried', 'it', 'Great', 'int  
erview', 'Doge', 'is', 'underestimated', 'Congratulations', 'Tesla', 'Chin  
a', 'for', 'amazing', 'execution', 'last', 'year', 'Now', 'on', 'to', 'th  
e', 'next', 'for', 'even', 'more', 'Happy', 'New', 'Year', 'of', 'the', 'O  
x', 'Frodo', 'was', 'the', 'underdoge', 'All', 'thought', 'he', 'would',  
'fail', 'Himself', 'most', 'of', 'all', 'Haha', 'thanks', 'Indeed', 'Tweet  
s', 'definitely', 'do', 'not', 'represent', 'realworld', 'time', 'allocati  
on', 'The', 'most', 'entertaining', 'outcome', 'is', 'the', 'most', 'likel  
y', 'Just', 'sent', 'some', 'Just', 'agree', 'to', 'do', 'Clubhouse', 'wit  
h', 'It', '\x92', 's', 'getting', 'real', 'Bought', 'some', 'Dogecoin', 'f  
or', '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', 'accou  
nt', 'U0001F923', 'U0001F923', 'This', 'is', 'true', 'power', 'haha', 'An  
y', 'crypto', 'wallet', 'that', 'won', '\x92', 't', 'give', 'you', 'your',  
'private', 'keys', 'should', 'be', 'avoided', 'at', 'all', 'costs', 'You  
r', 'app', 'sucks', 'RT', 'NASA', 'has', 'selected', 'Falcon', 'Heavy', 't  
o', 'launch', 'the', 'first', 'two', 'elements', 'of', 'the', 'lunar', 'Ga  
teway', 'together', 'on', 'one', 'mission', 'Yes', 'Once', 'we', 'can', 'predict', 'cash', 'flow', 'reasonably', 'well', 'Starlink', 'will', 'IPO', 'Starlin  
k', 'is', 'a', 'staggeringly', 'difficult', 'technical', 'economic', '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', 'sequestered', 'net', 'of', 'value', 'of', 'anv', 'product', 'made', 'Must', 'be', 'scalable', 'to', 'g', 'It', '\x92', 's', 'meant', 'to', 'be', 'the', 'same']



In [14]:

```
# 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',
'starlink', 'staggeringly', 'difficult', 'technical', 'economic', 'endeavo
r', 'however', 'spacex', 'needs', 'pass', 'deep', 'chasm', 'negative', 'cas
h', 'flow', 'next', 'year', 'lowest', 'cost', 'per', 'ton', 'carbon', 'seque
stered', 'net', 'value', 'product', 'made', 'must', 'scalable', 'g', 'it',
'meant', 'price', 'countries', 'only', 'difference', 'taxes', 'shipping', 't
his', 'intended', 'earth', 'may', 'ideas', 'apply', 'mars', 'xprize', 'tea
m', 'manage', '100m', 'carbon', 'capture', 'prize', 'everyone', 'tesla', 're
ceives', 'stock', 'my', 'comp', 'stock', 'options', 'take', 'table', 'that',
'missing', 'back', 'work', 'go', 'does', 'seem', 'bit', 'high', 'doge', 'app
ears', 'inflationary', 'meaningfully', 'fixed', 'coins', 'per', 'unit', 'tim
e', 'whereas', 'wow', '1', 'orbital', 'launch', 'tower', 'stack', '2', 'enou
gh', 'raptors', 'orbit', 'booster']
```

In [15]:

```
# Stemming
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', 'sta
rlink', 'staggeringli', 'difficult', 'technic', 'econom', 'endeavor', 'howe
v', 'spacex', 'need', 'pass', 'deep', 'chasm', 'neg', 'cash', 'flow', 'nex
t', 'year', 'lowest', 'cost', 'per', 'ton', 'carbon', 'sequest', 'net', 'val
u', 'product', 'made', 'must', 'scalabl', 'g', 'it', 'meant', 'price', 'coun
tri', 'onli', 'differ', 'tax', 'ship', 'thi', 'intend', 'earth', 'may', 'ide
a', 'appli', 'mar', 'xprize', 'team', 'manag', '100m', 'carbon', 'captur',
'prize', 'everyon', 'tesla', 'receiv', 'stock', 'my', 'comp', 'stock', 'opti
on', 'take', 'tabl', 'that', 'miss', 'back', 'work', 'go', 'doe', 'seem', 'b
it', 'high', 'doge', 'appear', 'inflationari', 'meaning', 'fix', 'coin', 'pe
r', 'unit', 'time', 'wherea', 'wow', '1', 'orbit', 'launch', 'tower', 'stac
k', '2', 'enough', 'raptor', 'orbit', 'booster']
```

In [16]:

```
# 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 underestimated congratulations tesla china amazing execution last year now next even happy new year ox frodo underdoge all thought would fail himself haha thanks indeed tweets definitely represent realworld time allocation the entertaining outcome likely just sent just agree clubhouse it getting real bought dogecoin lil x toddler hodler he definitely issues sentencing seems bit high thanks fixing please unlock account this true power any crypto wallet give private keys avoided costs your app sucks rt nasa selected falcon heavy launch first two elements lunar gateway together one mission once predict cash flow reasonably well starlink ipo starlink staggeringly difficult technical economic endeavor however 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 countries only difference taxes shipping this intended earth may ideas apply mars xprize team manage 100m carbon capture prize everyone tesla receives stock my comp stock options take table that missing back work go does 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 0 dogecoin instructional video the people spoken so c

In [17]:

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

['alien', 'ray', 'trace', 'cyberpunk', 'hdr', 'nextlevel', 'have', 'try', 'great', 'interview', 'doge', 'underestimate', 'congratulation', 'tesla', 'china', 'amazing', 'execution', 'last', 'year', 'now', 'next', 'even', 'happy', 'new', 'year', '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', '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', 'two', 'element', 'lunar', 'gateway', 'together', 'one', 'mission', 'once', 'predict', 'cash', 'flow', 'reasonably', 'well', 'starlink', 'ipo', 'starlink', 'staggeringly', 'difficult', 'technical', 'economic', 'endeavor', 'however', 'spacex', 'need', 'pass', 'deep', 'chasm', 'negative', 'cash', 'flow', 'next', 'year', 'low', 'cost', 'per', 'ton', 'carbon', 'sequester', 'net', 'value', 'product', 'make', 'must', 'scalable', 'g', 'it', 'mean', 'price', 'country', 'only', 'difference', 'taxi', 'ship', 'this', 'intend', 'earth', 'launch', 'tower', 'stack', '2', 'enough', 'raptors', 'orbit', 'booster', '3', 'improve', 'ship', 'booster', 'mass', 'back', 'work', 'tonight', '0', 'dogecoin', 'instructional', 'video', 'the', 'people', 'spoken', 'so', 'c']



In [18]:

```
clean_tweets=' '.join(lemmas)
clean_tweets
```

Out[18]:

'alien ray trace cyberpunk hdr nextlevel have try great interview doge und  
erestimate congratulation tesla china amazing execution last year now next  
even happy new year ox frodo underdoge all thought would fail himself haha  
thank indeed tweet definitely represent realworld time allocation the ente  
rtaining outcome likely just send just agree clubhouse it get real buy dog  
ecoin lil x toddler hodler he definitely issue sentencing seem bit high th  
ank fix please unlock account this true power any crypto wallet give priva  
te key avoid cost your app suck rt nasa select falcon heavy launch first t  
wo element lunar gateway together one mission once predict cash flow reaso  
nably well starlink ipo starlink staggeringly difficult technical economic  
endeavor however spacex need pass deep chasm negative 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 this intend earth may idea  
apply mars xprize team manage 100 m carbon capture prize everyone tesla re  
ceive stock my comp stock option take table that miss back work go do seem  
bit high doge appear inflationary meaningfully fix coin per unit time wher  
eas wow 1 orbital launch tower stack 2 enough raptor orbit booster 3 impro  
ve shin booster mass back work tonight ð ðððecoin instructional video the

## Feature Extaction

### 1. Using CountVectorizer

In [19]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer()
tweetscv = cv.fit_transform(lemmas)
```



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

In [24]:

```
cv_ngram_range = CountVectorizer(analyzer='word', ngram_range=(1,3), max_features = 100)
bow_matrix_ngram = cv_ngram_range.fit_transform(lemmas)
```

In [25]:

```
print(cv_ngram_range.get_feature_names())
print(bow_matrix_ngram.toarray())
```

```
['actually', 'ai', 'also', 'back', 'big', 'booster', 'car', 'come', 'complet
e', 'cool', 'could', 'crew', 'day', 'design', 'do', 'dragon', 'earth', 'engi
ne', 'even', 'ever', 'exactly', 'falcon', 'first', 'flight', 'fsd', 'futur
e', 'get', 'go', 'good', 'great', 'haha', 'hard', 'high', 'if', 'it', 'jus
t', 'land', 'launch', 'like', 'look', 'lot', 'love', 'make', 'many', 'mar',
'maybe', 'mission', 'model', 'month', 'much', 'need', 'new', 'next', 'no',
'not', 'one', 'part', 'people', 'point', 'pretty', 'probably', 'production',
'right', 'rocket', 'rt', 'say', 'seem', 'soon', 'space', 'spacex', 'starlin
k', 'starship', 'still', 'super', 'sure', 'take', 'tesla', 'test', 'thank',
'that', 'the', 'there', 'they', 'think', 'this', 'time', 'true', 'try', 'ufe
of', 'use', 'way', 'we', 'week', 'well', 'will', 'work', 'would', 'yeah', 'y
ear', '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 [26]:

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_v_ngram_max_features = TfidfVectorizer(norm="l2", analyzer='word', ngram_range=(1,3), m
tfidf_matrix_ngram_max_features = tfidf_v_ngram_max_features.fit_transform(lemmas)
```

In [27]:

```
print(tfidf_v_ngram_max_features.get_feature_names())
print(tfidf_matrix_ngram_max_features.toarray())
```

```
['10', '100', '13', '20', '2020', '30', '300', '40', '4d', '60', 'able', 'ab
solutely', 'access', 'accurate', 'actually', 'add', 'after', 'ago', 'agree',
'ai', 'aim', 'air', 'all', 'almost', 'already', 'also', 'although', 'always',
'amazing', 'and', 'another', 'appear', 'apply', 'appreciate', 'approval',
'arm', 'around', 'article', 'as', 'astronaut', 'at', 'autopilot', 'aweso
me', 'back', 'bad', 'bar', 'base', 'battery', 'be', 'become', 'berlin', 'bet
a', 'big', 'bit', 'boca', 'body', 'booster', 'brain', 'bring', 'bs', 'buil
d', 'burn', 'but', 'buy', 'cake', 'call', 'camera', 'can', 'car', 'cargo',
'cast', 'cell', 'ch4', 'chance', 'change', 'china', 'civilization', 'close',
'cold', 'color', 'come', 'company', 'complete', 'computer', 'confirm', 'cong
ratulation', 'control', 'cool', 'cost', 'could', 'course', 'cov', 'crazy',
'create', 'crew', 'critical', 'curevac', 'cyberpunk', 'cybertruck', 'datum',
'day', 'deep', 'definitely', 'delivery', 'design', 'detail', 'difference',
'different', 'do', 'dock', 'doge', 'dojo', 'dragon', 'drive', 'droneship',
'early', 'earth', 'easy', 'electric', 'enable', 'end', 'energy', 'engine',
'engineer', 'engineering', 'enough', 'entire', 'especially', 'essential', 'e
sthetic', 'et', 'even', 'ever', 'every', 'everyone', 'everything', 'exactl
y', 'expect', 'extreme', 'extremely', 'factory', 'fail', 'fair', 'falcon',
'far', 'fast', 'favorite', 'feel', 'fine', 'fire', 'first', 'fix', 'flap',
'flight', 'flow', 'fly', 'for', 'fsd', 'ft', 'full', 'fully', 'fun', 'fundam
ental', 'future', 'game', 'gas', 'gene', 'get', 'giant', 'giga', 'give', 'gl
ad', 'go', 'goal', 'good', 'government', 'great', 'haha', 'half', 'happen',
'happy', 'hard', 'have', 'he', 'head', 'header', 'hear', 'heavy', 'help', 'h
igh', 'hold', 'home', 'hop', 'hope', 'hopefully', 'hour', 'human', 'idea',
'if', 'imo', 'important', 'impressive', 'improve', 'improvement', 'in', 'inc
rease', 'indeed', 'inference', 'interesting', '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', 'longter
m', 'look', 'lose', 'lot', 'love', 'low', 'machine', 'main', 'major', 'mak
e', 'many', 'mar', 'mass', 'matter', 'max', 'may', 'maybe', 'mean', 'might',
'mile', 'min', 'mind', 'minute', 'mission', 'model', 'money', 'month', 'moo
n', 'most', 'mostly', 'move', 'movie', 'much', 'must', 'my', 'name', 'nasa',
'need', 'net', 'neural', 'neuralink', 'never', 'new', 'news', 'next', 'nigh
t', 'no', 'nosecone', 'not', 'note', 'nothing', 'now', 'number', 'of', 'ok',
'old', 'once', 'one', 'only', 'open', 'operational', 'option', 'orbit', 'orb
ital', 'original', 'pad', 'part', 'pass', 'pcr', 'people', 'per', 'person',
'place', 'play', 'please', 'pm', 'point', 'possible', 'potential', 'power',
'pressure', 'pretty', 'price', 'primarily', 'prior', 'probability', 'probabl
y', 'problem', 'product', 'production', 'progress', 'propellant', 'prototyp
e', 'provide', 'public', 'push', 'put', 'quite', 'range', 'raptor', 'rate',
'reach', 'read', 'real', 'really', 'release', 'require', 'return', 'review',
'right', 'rise', 'roadster', 'rocket', 'rt', 'safe', 'safety', 'satellite',
'say', 'scale', 'second', 'see', 'seem', 'sell', 'send', 'sense', 'separatio
n', 'service', 'set', 'several', 'ship', 'short', 'shot', 'should', 'show',
'signal', 'single', 'site', 'slightly', 'slow', 'small', 'sn5', 'sn8', 'sn
9', 'so', 'software', 'solar', 'solution', 'solve', 'some', 'something', 'so
ng', 'soon', 'sorry', 'sound', 'south', 'space', 'spacecraft', 'spacex', 'sp
eed', 'splashdown', 'stage', 'starlink', 'starship', 'start', 'static', 'sta
tion', 'steel', 'step', 'still', 'stock', 'story', 'super', 'support', 'sur
e', 'symptom', 'system', 'take', 'talk', 'tank', 'target', 'tbh', 'team', 't
echnically', 'technology', 'tesla', 'test', 'testing', 'texas', 'thank', 'th
at', 'the', 'there', 'they', 'thing', 'think', 'this', 'three', 'thrust', 't
ile', 'time', 'to', 'today', 'tomorrow', 'tonight', 'top', 'total', 'tough',
'traffic', 'training', 'travel', 'true', 'try', 'tunnel', 'turn', 'twitter',
'two', 'u0001f3b6', 'u0001f440', 'u0001f495', 'u0001f525', 'u0001f5a4', 'u04
```

```
3b', 'u043e', 'u2192', 'ufe0f', 'understand', 'update', 'upgrade', 'ur', 'us  
e', 'useful', 'vaccine', 'vehicle', 'version', 'very', 'video', 'view', 'vol  
ume', 'wait', 'want', 'watch', 'way', 'we', 'week', 'welcome', 'well', 'wha  
t', 'will', 'without', 'word', 'work', 'world', 'worth', 'would', 'wow', 'ye  
ah', '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.]]
```

## Generate Word Cloud



In [29]:

```
# 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 endeavor however energy need more clean stream negative cash flow

In [30]:

```
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 PROPN  
xprize PROPN  
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 [31]:

```
# Filtering the nouns and 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', 'improv
e', 'ship', 'booster', 'mass', 'work', 'tonight', 'dogecoin', 'video', 'peop
le', 'speak', 'cute', 'image', 'reflect', 'time', 'cost', 'people', 'rain',
'pain', 'u0001f3b6', 'let', 'come', 'fun', 'crypto', 'simplicity', 'genius',
'yup', 'decade', 'work', 'look', 'note', 'email', 'text', 'lesson', 'learn',
'earth', 'time', 'tell', 'story', 'tesla', 'spacex', 'read', 'hear', 'name',
'year', 'dogecake', 'yolt', 'thing', 'restaurant', 'get', 'hang', 'strange
n', 'sjm', 'currency', 'earth', 'scratch', 'u0001f5a4', 'destiny', 'franz',
'say', 'ship', 'landing', 'burn', 'solution', 'greate', 'use', 'gas', 'maneu
vering', 'rcs', 'thruster', 'seem', 'turbopumpfed', 'raptor', 'falcon', 'lau
nch', 'collect', 'galaxy', 'explore', 'launch', 'starlink', 'satellite', 'or
bit', 'mission', 'pad', '39a', 'deck', 'default', 'engine', 'lever', 'arm',
'shut', 'engine', 'min', 'throttle']
```

In [32]:

```
# Counting the noun and 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[32]:

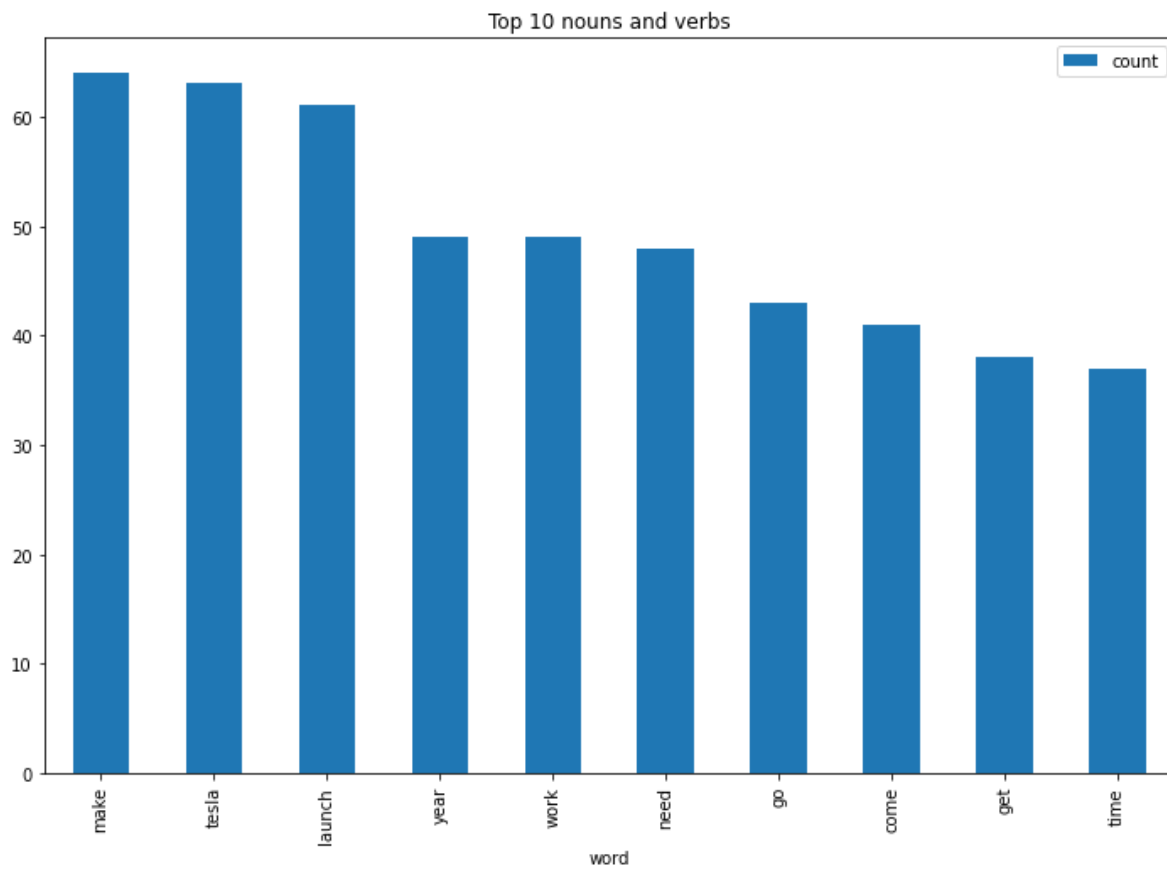
	word	count
0	make	64
1	tesla	63
2	launch	61
3	year	49
4	work	49
5	need	48
6	go	43
7	come	41
8	get	38
9	time	37

In [33]:

```
# Visualising results (Barchart for top 10 nouns + verbs)
wd_df[0:10].plot.bar(x='word',figsize=(12,8),title='Top 10 nouns and verbs')
```

Out[33]:

```
<AxesSubplot:title={'center':'Top 10 nouns and verbs'}, xlabel='word'>
```



## Emotion Mining - Sentiment Analysis

In [34]:

```
from nltk import tokenize
sentences=tokenize.sent_tokenize(' '.join(tweets))
sentences
```

Out[34]:

```
['@kunalb11 I\'m an alien @ID_AA_Carmack Ray tracing on Cyberpunk with H
DR 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 Frodo was the underdoge,\nAll thought he would f
ail,\nHimself most of all.',
 '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 (h
ttps://t.co/3rWE9uHSTS) @geoffkeighley @UnrealEngine It\'s getting real
Bought some Dogecoin for lil X, so he can be a toddler hodler @JoshManMod
e He definitely has issues. but the sentencing seems a bit high @freewalle
```

In [35]:

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

Out[35]:

	sentence
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 @teslacn 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	AI symbiosis while u wait @vistacruiser7 @flcn...
923	@TeslaGong @PPathole Samwise Gamgee @PPathole ...

924 rows × 1 columns

In [36]:

```
# Emotion Lexicon - Affin
affin=pd.read_csv("C:/Users/LENOVO/Documents/Custom Office Templates/Afinn.csv",sep=',',enc
affin
```

Out[36]:

	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 [37]:

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

Out[37]:

```
{'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.
```

In [38]:

```
# Custom function: Score each word in a sentence in Lemmatised form, but calculate the score
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
```

In [39]:

```
# Manual testing
calculate_sentiment(text='great')
```

Out[39]:

3

In [40]:

```
# Calculating sentiment value for each sentence
sent_df['sentiment_value'] = sent_df['sentence'].apply(calculate_sentiment)
sent_df['sentiment_value']
```

Out[40]:

```
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 [41]:

```
# how many words are in the sentence?
sent_df['word_count'] = sent_df['sentence'].str.split().apply(len)
sent_df['word_count']
```

Out[41]:

```
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 [42]:

```
sent_df.sort_values(by='sentiment_value')
```

Out[42]:

	sentence	sentiment_value	word_count
647	Very ba... https://t.co/tJsh1Exz1Q @justpaulinel...	-8	60
64	Also, the road to hell is mostly paved with ba...	-7	11
837	Cool Model 3 review by @iamjamiefoxx https://t...	-7	61
611	Then static fire, checkouts, static fire, fly ...	-4	12
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... https://t.co/rabjKrtQlw @Sav...	14	138
719	We just haven't observed the... https://t.co/mez...	15	72
36	@ajtourville @Erdyastronaut @SpaceX Yes, but ...	16	231
105	@Erdyastronaut @SpaceX Was also thinking that...	16	94

924 rows × 3 columns

In [43]:

```
# Sentiment score of the whole review
sent_df['sentiment_value'].describe()
```

Out[43]:

```
count      924.000000
mean        1.345238
std         2.684749
min        -8.000000
25%         0.000000
50%         0.000000
75%         3.000000
max        16.000000
Name: sentiment_value, dtype: float64
```

In [44]:

```
# negative sentiment score of the whole review
sent_df[sent_df['sentiment_value']<=0]
```

Out[44]:

	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
5	Happy New Year of the Ox!	0	6
6	<a href="https://t.co/9WFKMYu2oj">https://t.co/9WFKMYu2oj</a> Frodo was the underdog...	-2	14
...	...	...	...
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 [45]:

```
# Positive sentiment score of the whole review
sent_df[sent_df['sentiment_value']>0]
```

Out[45]:

	sentence	sentiment_value	word_count
2	@joerogan @Spotify Great interview!	3	4
3	@gtera27 Doge is underestimated @teslacn Congr...	3	13
7	https://t.co/zGxJFDzzrM @OwenSparks_ @flcnhvy ...	2	10
9	The most entertaining outcome is the most like...	3	109
17	Back to work I go @CapybaraSurfer @MattWallace...	4	38
...	...	...	...
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...	3	15
917	This is both great & terrifying.	3	6
918	Everything we've ever sensed or thought has be...	3	17

428 rows × 3 columns

In [46]:

```
# Adding index column
sent_df['index']=range(0,len(sent_df))
sent_df
```

Out[46]:

	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 [47]:

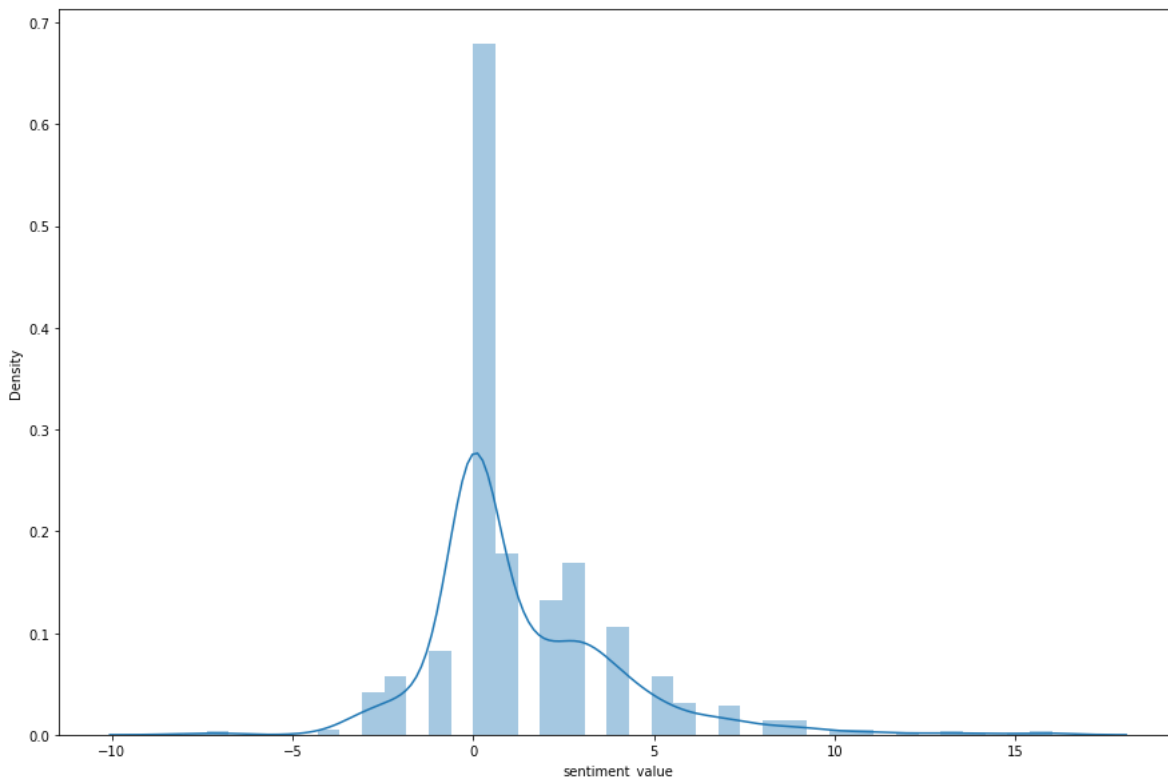
```
# Plotting the sentiment value for whole review
import seaborn as sns
plt.figure(figsize=(15,10))
sns.distplot(sent_df['sentiment_value'])
```

C:\Users\LENOVO\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[47]:

<AxesSubplot:xlabel='sentiment\_value', ylabel='Density'>

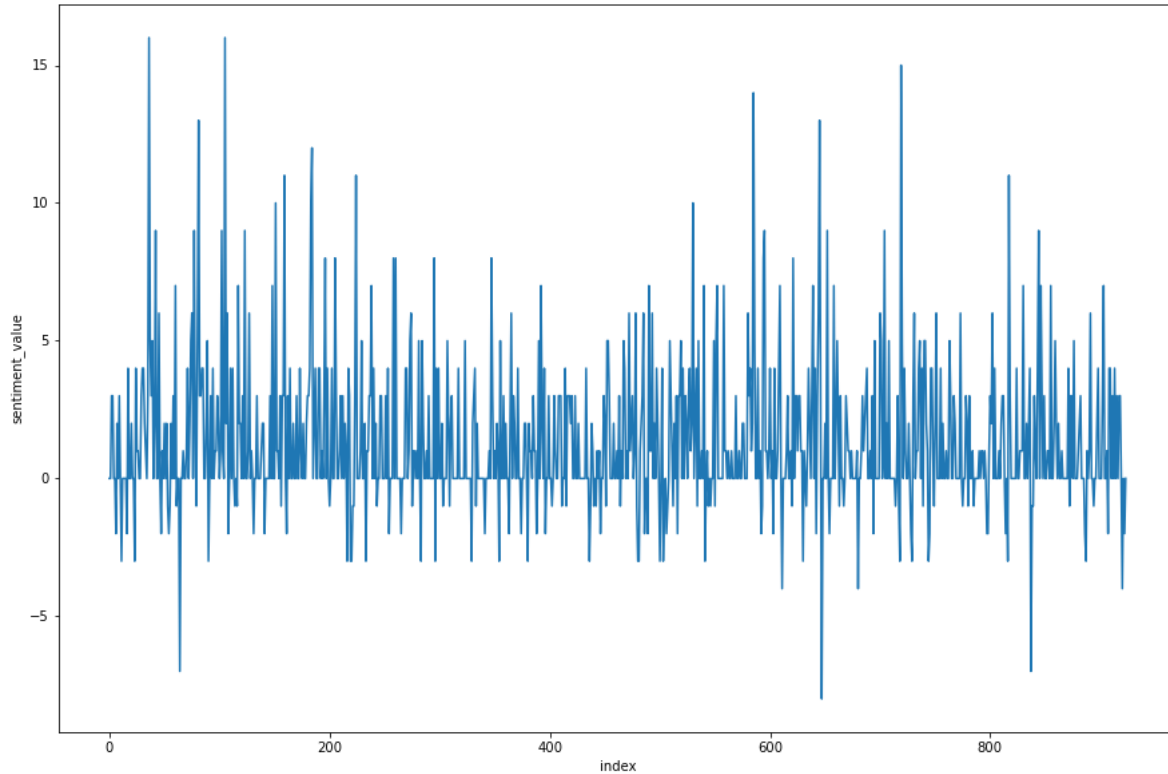


In [48]:

```
# Plotting the line plot for the sentiment value of whole review  
plt.figure(figsize=(15,10))  
sns.lineplot(y='sentiment_value',x='index',data=sent_df)
```

Out[48]:

<AxesSubplot:xlabel='index', ylabel='sentiment\_value'>



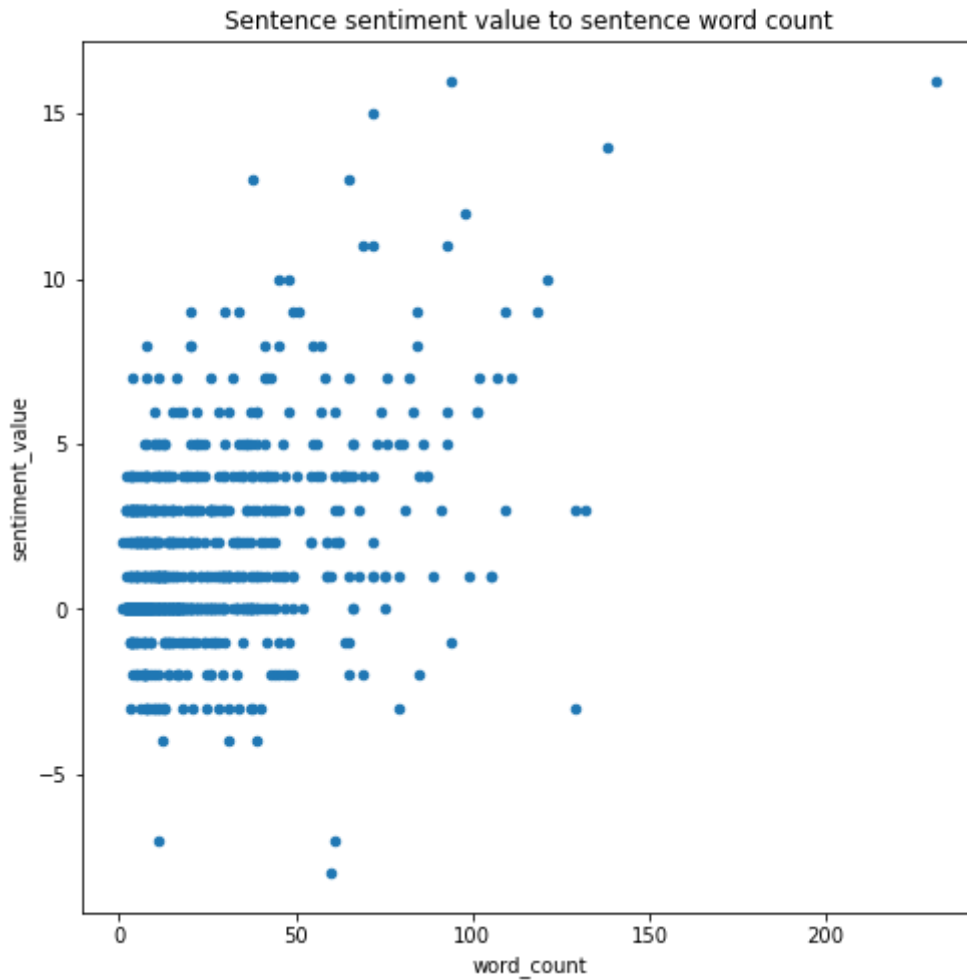
In [49]:

```
# Correlation analysis
```

```
sent_df.plot.scatter(x='word_count', y='sentiment_value',figsize=(8,8),title='Sentence sent
```

Out[49]:

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



In [ ]: