

# Text Mining\_Q1\_Elon Musk

1) Perform sentimental analysis on the Elon-musk tweets (Elon-musk.csv)

## 1. Import Libs

In [14]:

```
pip install -U spacy
```

...

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import string
import spacy
from wordcloud import WordCloud
import re
import warnings
warnings.filterwarnings('ignore')

from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.tokenize import sent_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
```

## 2. Import Data

In [3]:

```
tweets=pd.read_csv('Elon_musk.csv',encoding='Latin-1')
tweets
```

Out[3]:

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	@teslacn 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 [4]:

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

Out[4]:

		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

### 3. Data Preprocessing

remove both the leading and the trailing characters

In [5]:

```
tweets=[Text.strip() for Text in tweets.Text]
tweets=[Text for Text in tweets if Text] # removes empty strings.
tweets[0:10]
```

Out[5]:

```
['@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 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 t
ime allocation.',
 'The most entertaining outcome is the most likely']
```

Joining the list into one string/text

In [6]:

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

RationalEtienne @Adamklotz\_ Once we can predict cash flow reasonably well, Starlink will IPO @RationalEtienne @Adamklotz\_ Starlink is a staggeringly difficult technical & economic endeavor. However, if we don\x85 <http://t.co/9Z8Ac6skqx> (<https://t.co/9Z8Ac6skqx>) @RationalEtienne @Adamklotz\_ SpaceX needs to pass through a deep chasm of negative cash flow over the next year or\x85 <https://t.co/7J1c92hdjl> (<https://t.co/7J1c92hdjl>) @ID\_AA\_Carmack Lowest cost per ton of carbon sequestered, net of value of any product made. \n\nMust be scalable to g\x85 <https://t.co/XMyI7qWSgw> (<https://t.co/XMyI7qWSgw>) @Adamklotz\_ It\x92s meant to be the same price in all countries. Only difference should be taxes & shipping. @tobyliiiiiiiiiii This is intended for Earth, but there may be some ideas that apply to Mars too @CryptoShrikar @CoinDesk @Tesla @Dan\_Z\_Palmer <U+0001F923><U+0001F923> XPrize team will manage the \$100M carbon capture prize <https://t.co/fSw5IanL0r> (<https://t.co/fSw5IanL0r>) @SRuhle Everyone at Tesla receives stock. \n\nMy comp is all stock/options, which I do not take off the table. \n\nThat\x92s what you\x92re missing. Back to work I go @CapybaraSurfer @MattWallace888 Does seem a bit high @itsALLrisky Doge appears to be inflationary, but is not meaningfully so (fixed # of coins per unit time), whereas\x85 <https://t.co/4UH4rARC6T> (<https://t.co/4UH4rARC6T>) @MichaelGalanin @kevin

In [7]:

```
tweets_text = re.sub("[^A-Za-z" " ]+", " ", tweets_text).lower() #remove special character
tweets_text = re.sub("[0-9" " ]+", " ", tweets_text).lower() #remove numbers
tweets_text = re.sub(r'^https?:\\/\\.*[\\r\\n]*', '', tweets_text).lower() #remove hyperlink
tweets_text
```

tarlink satellites u https t co bjfjlcwzdk https t co ln reesbw off twitte  
r for a while it will be filled with graffiti art eiraum u c u b u u u b  
u u u c giga berlin progress https t co ekpg qcbui hamoon neuralink neural  
ink is working super hard to ensure implant safety amp is in close communi  
cation with the https t co yky llpumd rt spacex this mission enables acces  
s for everyday people who dream of going to space rt spacex announcing the  
first commercial astronaut mission to orbit earth aboard dragon u https t  
co mbesvnakad https t co uklsjffrjk if you ve worked on advanced wearables  
phones or robots those skills are needed neuralink feels weird helping mak  
e hopefully good version of cyberpunk come true please consider working at  
neuralink short term solve brain spine injuries long term human ai symbios  
is latte https t co qrc alads https t co xccokojytl on clubhouse tonight a  
t pm la time erdayastronaut tj cooney rocket spacex felixschlang marcusho  
usegame tom has some great stories about experim https t co teezkiwcmk rt  
commercial crew launch alert u f nasa and spacex are targeting no earlier  
than april for the launch of the second crew rotation mis erdayastronaut t  
j cooney rocket spacex felixschlang marcushousegame he became a big fan o  
f methane but was https t co mdv vdefyc tj cooney rocket spacex felixschl  
ang marcushousegame that sounds correct tom certainly deserves a lot of cr  
e https t co hdri hfa b tj cooney rocket spacex felixschlang marcushouseg

In [8]:

```
wordnet = WordNetLemmatizer()
```

In [9]:

```
text_tokens = word_tokenize(tweets_text)
```

In [10]:

```
tokens_without_sw = [word for word in text_tokens if not word in stopwords.words()]
```

In [11]:

```
tf = TfidfVectorizer()
```

In [12]:

```
text_tf = tf.fit_transform(tokens_without_sw)
```

## Create the DataFrame

In [13]:

```
feature_names = tf.get_feature_names()
dense = text_tf.todense()
denselist = dense.tolist()
df =pd.DataFrame(denselist, columns=feature_names)
```

In [14]:

```
df
```

Out[14]:

	aa	aayypcpp	abo	aboard	abort	absence	absolute	absolutely	absorb	absorption
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...
14512	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14513	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14514	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14515	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14516	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

14517 rows × 5099 columns

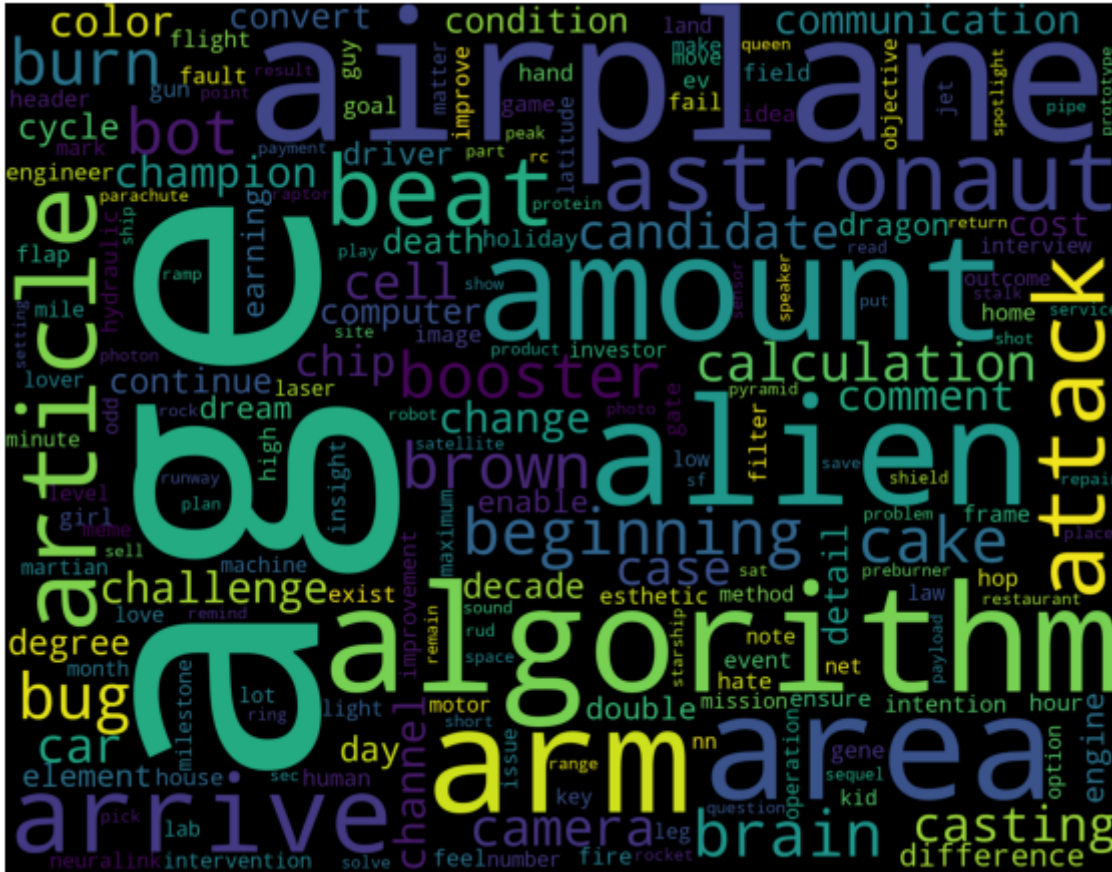
In [15]:

```
word_list = ' '.join(df)
```

In [47]:

```
wordcloud = WordCloud(background_color='black',width=1800,height=1400).generate(word_list)
```

```
plt.figure(figsize=(10,10))
plt.imshow(wordcloud)
plt.axis('off')
```

$$(-0.5, 1799.5, 1399.5, -0.5)$$


## 4. Sentimental Analysis

In [43]:

```
with open('positive-words.txt') as pw:  
    positive_words = pw.read().split("\n")
```

In [44]:

```
positive_words = positive_words[35:]  
positive_words
```

```
'adroit',  
'adroitly',  
'adulate',  
'adulation',  
'adulatory',  
'advanced',  
'advantage',  
'advantageous',  
'advantageously',  
'advantages',  
'adventuresome',  
'adventurous',  
'advocate',  
'advocated',  
'advocates',  
'affability',  
'affable',  
'affably',  
'affectation',  
'affection'
```

In [45]:

```
with open('negative-words.txt') as nw:  
    negative_words = nw.read().split("\n")
```

In [46]:

```
negative_words = negative_words[35:]  
negative_words
```

```
'ail',  
'ailing',  
'ailment',  
'aimless',  
'alarm',  
'alarmed',  
'alarming',  
'alarmingly',  
'alienate',  
'alienated',  
'alienation',  
'allegation',  
'allegations',  
'allege',  
'allergic',  
'allergies',  
'allergy',  
'aloof',  
'altercation',  
'ambiguity'
```

In [22]:

```
txt_pos_in_pw = ' '.join([word for word in df if word in positive_words])
```

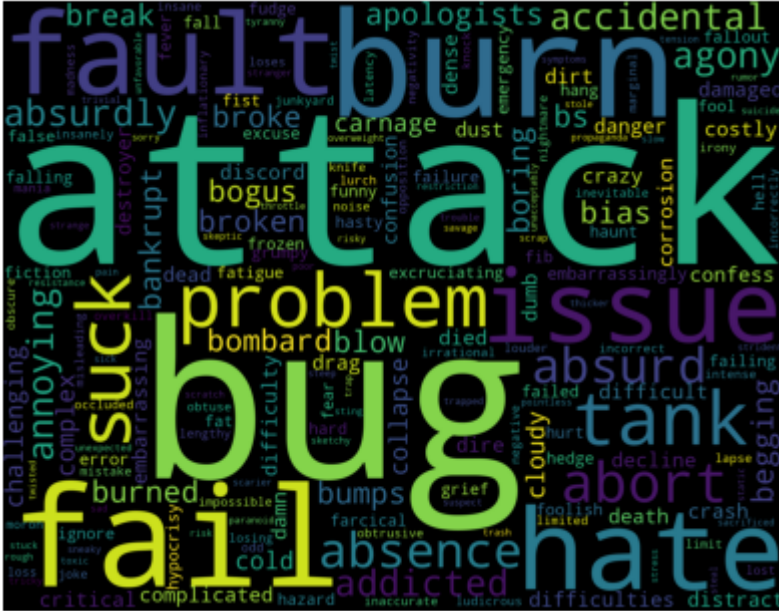
In [23]:

```
wordcloud_pos = WordCloud(background_color='black',width=1800,height=1400).generate(txt_pos
```





```
plt.figure(figsize=(7,15))
plt.imshow(wordcloud_neg)
plt.axis('off')
```

$$(-0.5, 1799.5, 1399.5, -0.5)$$


In [ ]: