In [1]:
```
import nltk
nltk.download('stopwords')
```

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

Out[1]:
```
True
```

In [2]:
```
nltk.download('wordnet')
```

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

Out[2]:
```
True
```

In [3]:
```
pip install -U textblob
```

```
Requirement already satisfied: textblob in c:\programdata\anaconda3\lib\site-pa
ckages (0.17.1)
Requirement already satisfied: nltk>=3.1 in c:\programdata\anaconda3\lib\site-p
ackages (from textblob) (3.6.1)
Requirement already satisfied: regex in c:\programdata\anaconda3\lib\site-packa
ges (from nltk>=3.1->textblob) (2021.4.4)
Requirement already satisfied: click in c:\programdata\anaconda3\lib\site-packa
ges (from nltk>=3.1->textblob) (7.1.2)
Requirement already satisfied: joblib in c:\programdata\anaconda3\lib\site-pack
ages (from nltk>=3.1->textblob) (1.0.1)
Requirement already satisfied: tqdm in c:\programdata\anaconda3\lib\site-packag
es (from nltk>=3.1->textblob) (4.59.0)
Note: you may need to restart the kernel to use updated packages.
```

In [4]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from nltk.corpus import stopwords
from textblob import TextBlob
```

In [5]:
```
data = pd.read_csv('Elon_musk.csv',encoding="latin-1")
data
```

Out[5]:

	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 [6]:
```
data.head()
```

Out[6]:

	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...

Number of Words

In [7]:
```
#Number of Words in single tweet
data['word_count'] = data['Text'].apply(lambda x: len(str(x).split(" ")))
data[['Text','word_count']]
```

Out[7]:

	Text	word_count
0	@kunalb11 I m an alien	4
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	13
2	@joerogan @Spotify Great interview!	4
3	@gtera27 Doge is underestimated	4
4	@teslacn Congratulations Tesla China for amazi...	17
...	...	...
1994	@flcnhvy True, it sounds so surreal, but the n...	23
1995	@PPathole Make sure to read ur terms & con...	12
1996	@TeslaGong @PPathole Samwise Gamgee	4
1997	@PPathole Altho Dumb and Dumber is <U+0001F525...	7
1998	Progress update August 28	4

1999 rows × 2 columns

In [8]:
```
#Number of Words in single tweet
data['word_count'] = data['Text'].apply(lambda x: len(str(x).split(" ")))
data[['Text','word_count']].head()
```

Out[8]:

	Text	word_count
0	@kunalb11 I m an alien	4
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	13
2	@joerogan @Spotify Great interview!	4
3	@gtera27 Doge is underestimated	4
4	@teslacn Congratulations Tesla China for amazi...	17

Number of Characters

In [9]:
```
#Number of characters in single tweet
data['char_count'] = data['Text'].str.len() ## this also includes spaces
data[['Text','char_count']]
```

Out[9]:

	Text	char_count
0	@kunalb11 I m an alien	22
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	82
2	@joerogan @Spotify Great interview!	35
3	@gtera27 Doge is underestimated	31
4	@teslacn Congratulations Tesla China for amazi...	104
...	...	...
1994	@flcnhvy True, it sounds so surreal, but the n...	144
1995	@PPathole Make sure to read ur terms & con...	77
1996	@TeslaGong @PPathole Samwise Gamgee	35
1997	@PPathole Altho Dumb and Dumber is <U+0001F525...	59
1998	Progress update August 28	25

1999 rows × 2 columns

In [10]:
```
#Number of characters in single tweet
data['char_count'] = data['Text'].str.len() ## this also includes spaces
data[['Text','char_count']].head()
```

Out[10]:

	Text	char_count
0	@kunalb11 I m an alien	22
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	82
2	@joerogan @Spotify Great interview!	35
3	@gtera27 Doge is underestimated	31
4	@teslacn Congratulations Tesla China for amazi...	104

Average Word Length

In [11]:
```
def avg_word(sentence):
  words = sentence.split()
  return (sum(len(word) for word in words)/len(words))

data['avg_word'] = data['Text'].apply(lambda x: avg_word(x))
data[['Text','avg_word']]
```

Out[11]:

	Text	avg_word
0	@kunalb11 I m an alien	4.750000
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	5.384615
2	@joerogan @Spotify Great interview!	8.000000
3	@gtera27 Doge is underestimated	7.000000
4	@teslacn Congratulations Tesla China for amazi...	5.176471
...	...	...
1994	@flcnhvy True, it sounds so surreal, but the n...	5.260870
1995	@PPathole Make sure to read ur terms & con...	5.500000
1996	@TeslaGong @PPathole Samwise Gamgee	8.000000
1997	@PPathole Altho Dumb and Dumber is <U+0001F525...	7.571429
1998	Progress update August 28	5.500000

1999 rows × 2 columns

In [12]:
```
def avg_word(sentence):
  words = sentence.split()
  return (sum(len(word) for word in words)/len(words))

data['avg_word'] = data['Text'].apply(lambda x: avg_word(x))
data[['Text','avg_word']].head()
```

Out[12]:

	Text	avg_word
0	@kunalb11 I m an alien	4.750000
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	5.384615
2	@joerogan @Spotify Great interview!	8.000000
3	@gtera27 Doge is underestimated	7.000000
4	@teslacn Congratulations Tesla China for amazi...	5.176471

Number of stopwords

In [13]:
```
stop = stopwords.words('english')

data['stopwords'] = data['Text'].apply(lambda x: len([x for x in x.split() if x i
data[['Text','stopwords']]
```

Out[13]:

	Text	stopwords
0	@kunalb11 I m an alien	1
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	4
2	@joerogan @Spotify Great interview!	0
3	@gtera27 Doge is underestimated	1
4	@teslacn Congratulations Tesla China for amazi...	5
...	...	...
1994	@flcnhvy True, it sounds so surreal, but the n...	11
1995	@PPathole Make sure to read ur terms & con...	2
1996	@TeslaGong @PPathole Samwise Gamgee	0
1997	@PPathole Altho Dumb and Dumber is <U+0001F525...	2
1998	Progress update August 28	0

1999 rows × 2 columns

In [14]:
```
stop = stopwords.words('english')

data['stopwords'] = data['Text'].apply(lambda x: len([x for x in x.split() if x i
data[['Text','stopwords']].head()
```

Out[14]:

	Text	stopwords
0	@kunalb11 I m an alien	1
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	4
2	@joerogan @Spotify Great interview!	0
3	@gtera27 Doge is underestimated	1
4	@teslacn Congratulations Tesla China for amazi...	5

Number of Special Characters

In [15]:
```
data['hastags'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.sta
data[['Text','hastags']]
```

Out[15]:

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

1999 rows × 2 columns

In [16]:
```
data['hastags'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.sta
data[['Text','hastags']].head()
```

Out[16]:

	Text	hastags
0	@kunalb11 I m an alien	1
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	1
2	@joerogan @Spotify Great interview!	2
3	@gtera27 Doge is underestimated	1
4	@teslacn Congratulations Tesla China for amazi...	1

Number of Numerics

In [17]:
```
data['numerics'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.is
data[['Text','numerics']]
```

Out[17]:

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

1999 rows × 2 columns

In [18]:
```
data['numerics'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.is
data[['Text','numerics']].head()
```

Out[18]:

	Text	numerics
0	@kunalb11 I m an alien	0
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	0
2	@joerogan @Spotify Great interview!	0
3	@gtera27 Doge is underestimated	0
4	@teslacn Congratulations Tesla China for amazi...	0

Number of Upper Case Words

In [19]:
```
data['upper'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.isupp
data[['Text','upper']]
```

Out[19]:

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

1999 rows × 2 columns

In [20]:
```
data['upper'] = data['Text'].apply(lambda x: len([x for x in x.split() if x.isupp
data[['Text','upper']].head()
```

Out[20]:

	Text	upper
0	@kunalb11 I m an alien	0
1	@ID_AA_Carmack Ray tracing on Cyberpunk with H...	1
2	@joerogan @Spotify Great interview!	0
3	@gtera27 Doge is underestimated	0
4	@teslacn Congratulations Tesla China for amazi...	0

Pre - Processing

Lower Case

In [21]:
```
data['Text'] = data['Text'].apply(lambda x: " ".join(x.lower() for x in x.split()
data['Text']
```

Out[21]:
```
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 &amp; con...
1996                  @teslagong @ppathole samwise gamgee
1997    @ppathole altho dumb and dumber is <u+0001f525...
1998                             progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [22]:
```
data['Text'] = data['Text'].apply(lambda x: " ".join(x.lower() for x in x.split()
data['Text'].head()
```

Out[22]:
```
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...
Name: Text, dtype: object
```

Removing Punctuation

In [23]:
```
data['Text'] = data['Text'].str.replace('[^\w\s]','')
data['Text']
```

```
<ipython-input-23-e207339d85ad>:1: FutureWarning: The default value of regex wi
ll change from True to False in a future version.
  data['Text'] = data['Text'].str.replace('[^\w\s]','')
```

Out[23]:
```
0                                    kunalb11 im an alien
1       id_aa_carmack ray tracing on cyberpunk with hd...
2                         joerogan spotify great interview
3                           gtera27 doge is underestimated
4       teslacn congratulations tesla china for amazin...
                              ...
1994    flcnhvy true it sounds so surreal but the nega...
1995    ppathole make sure to read ur terms amp condit...
1996                    teslagong ppathole samwise gamgee
1997    ppathole altho dumb and dumber is u0001f525u00...
1998                             progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [24]:
```
data['Text'] = data['Text'].str.replace('[^\w\s]','')
data['Text'].head()
```

```
<ipython-input-24-5a2099d0f9da>:1: FutureWarning: The default value of regex wi
ll change from True to False in a future version.
  data['Text'] = data['Text'].str.replace('[^\w\s]','')
```

Out[24]:
```
0                                kunalb11 im an alien
1    id_aa_carmack ray tracing on cyberpunk with hd...
2                     joerogan spotify great interview
3                       gtera27 doge is underestimated
4    teslacn congratulations tesla china for amazin...
Name: Text, dtype: object
```

Removal of Stop Words

In [25]:
```
stop = stopwords.words('english')
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text']
```

Out[25]:
```
0                                     kunalb11 im alien
1       id_aa_carmack ray tracing cyberpunk hdr nextle...
2                         joerogan spotify great interview
3                              gtera27 doge underestimated
4       teslacn congratulations tesla china amazing ex...
                              ...
1994    flcnhvy true sounds surreal negative propagand...
1995    ppathole make sure read ur terms amp condition...
1996                    teslagong ppathole samwise gamgee
1997        ppathole altho dumb dumber u0001f525u0001f525
1998                               progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [26]:
```
stop = stopwords.words('english')
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text'].head()
```

Out[26]:
```
0                                   kunalb11 im alien
1    id_aa_carmack ray tracing cyberpunk hdr nextle...
2                     joerogan spotify great interview
3                          gtera27 doge underestimated
4    teslacn congratulations tesla china amazing ex...
Name: Text, dtype: object
```

Common word removal

In [27]:
```
freq = pd.Series(' '.join(data['Text']).split()).value_counts()[:10]
freq
```

Out[27]:
```
spacex            239
amp               218
tesla             166
erdayastronaut    142
rt                127
ppathole          123
flcnhvy           114
yes                86
great              76
teslaownerssv      73
dtype: int64
```

In [28]:
```
freq = list(freq.index)
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text']
```

Out[28]:
```
0                                      kunalb11 im alien
1       id_aa_carmack ray tracing cyberpunk hdr nextle...
2                              joerogan spotify interview
3                             gtera27 doge underestimated
4       teslacn congratulations china amazing executio...
                              ...
1994    true sounds surreal negative propaganda still ...
1995    make sure read ur terms conditions clicking ac...
1996                               teslagong samwise gamgee
1997                  altho dumb dumber u0001f525u0001f525
1998                             progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [29]:
```
freq = pd.Series(' '.join(data['Text']).split()).value_counts()[:10]
freq
```

Out[29]:
```
wholemarsblog      68
teslarati          59
nasaspaceflight    55
haha               55
good               51
launch             49
sure               43
yeah               41
would              40
much               40
dtype: int64
```

In [30]:
```
freq = list(freq.index)
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text'].head()
```

Out[30]:
```
0                                  kunalb11 im alien
1    id_aa_carmack ray tracing cyberpunk hdr nextle...
2                           joerogan spotify interview
3                          gtera27 doge underestimated
4    teslacn congratulations china amazing executio...
Name: Text, dtype: object
```

Rare Words Removal

In [31]:
```
freq = pd.Series(' '.join(data['Text']).split()).value_counts()[-10:]
freq
```

Out[31]:
```
overweight           1
httpstcof8rwy4exee   1
held                 1
httpstcoczykfyo0ix   1
sam_lopezxx          1
propulsive           1
sup                  1
httpstco3fazzgss8c   1
transit              1
viktaur27            1
dtype: int64
```

In [32]:
```
freq = list(freq.index)
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text']
```

Out[32]:
```
0                                    kunalb11 im alien
1       id_aa_carmack ray tracing cyberpunk hdr nextle...
2                             joerogan spotify interview
3                            gtera27 doge underestimated
4       teslacn congratulations china amazing executio...
                              ...
1994    true sounds surreal negative propaganda still ...
1995        make read ur terms conditions clicking accept
1996                           teslagong samwise gamgee
1997                 altho dumb dumber u0001f525u0001f525
1998                           progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [33]:
```
freq = pd.Series(' '.join(data['Text']).split()).value_counts()[-10:]
freq
```

Out[33]:
```
httpstcodjdzxq4maz    1
cas                   1
rakyll                1
amplitudes            1
github                1
httpstco941a2odu5h    1
accelera              1
o2                    1
httpstcowbk7zz0fqx    1
channel               1
dtype: int64
```

In [34]:
```
freq = list(freq.index)
data['Text'] = data['Text'].apply(lambda x: " ".join(x for x in x.split() if x nc
data['Text'].head()
```

Out[34]:
```
0                                    kunalb11 im alien
1    id_aa_carmack ray tracing cyberpunk hdr nextle...
2                           joerogan spotify interview
3                          gtera27 doge underestimated
4    teslacn congratulations china amazing executio...
Name: Text, dtype: object
```

Spelling correction

In [35]:
```
data['Text'][:5].apply(lambda x: str(TextBlob(x).correct()))
```

Out[35]:
```
0                                    kunalb11 in alien
1    id_aa_carmack ray tracing cyberpunk her nextle...
2                            joerogan specify interview
3                          gtera27 done underestimated
4    teslacn congratulations china amazing executio...
Name: Text, dtype: object
```

Tokenization

In [36]:
```
nltk.download('punkt')
```

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

Out[36]:
```
True
```

In [37]:
```
TextBlob(data['Text'][1]).words
```

Out[37]:
```
WordList(['id_aa_carmack', 'ray', 'tracing', 'cyberpunk', 'hdr', 'nextlevel',
'tried'])
```

Stemming

In [38]:
```
from nltk.stem import PorterStemmer
st = PorterStemmer()
data['Text'][:5].apply(lambda x: " ".join([st.stem(word) for word in x.split()]))
```

Out[38]:
```
0                                   kunalb11 im alien
1    id_aa_carmack ray trace cyberpunk hdr nextleve...
2                           joerogan spotifi interview
3                              gtera27 doge underestim
4    teslacn congratul china amaz execut last year ...
Name: Text, dtype: object
```

Lemmatization

In [39]:
```
from textblob import Word
```

In [40]:
```
data['Text'] = data['Text'].apply(lambda x: " ".join([Word(word).lemmatize() for
data['Text']
```

Out[40]:
```
0                                     kunalb11 im alien
1       id_aa_carmack ray tracing cyberpunk hdr nextle...
2                              joerogan spotify interview
3                            gtera27 doge underestimated
4       teslacn congratulation china amazing execution...
                              ...
1994    true sound surreal negative propaganda still e...
1995          make read ur term condition clicking accept
1996                            teslagong samwise gamgee
1997                 altho dumb dumber u0001f525u0001f525
1998                            progress update august 28
Name: Text, Length: 1999, dtype: object
```

In [41]:
```
data['Text'] = data['Text'].apply(lambda x: " ".join([Word(word).lemmatize() for
data['Text'].head()
```

Out[41]:
```
0                                   kunalb11 im alien
1    id_aa_carmack ray tracing cyberpunk hdr nextle...
2                           joerogan spotify interview
3                         gtera27 doge underestimated
4    teslacn congratulation china amazing execution...
Name: Text, dtype: object
```

Advanced Text Processing

N-grams

In [42]:
```
TextBlob(data['Text'][0]).ngrams(2)
```

Out[42]:
```
[WordList(['kunalb11', 'im']), WordList(['im', 'alien'])]
```

Term frequency

In [43]:
```
tf1 = (data['Text'][1:2]).apply(lambda x: pd.value_counts(x.split(" "))).sum(axis
tf1.columns = ['words','tf']
tf1
```

Out[43]:

	words	tf
0	id_aa_carmack	1
1	tracing	1
2	tried	1
3	cyberpunk	1
4	ray	1
5	hdr	1
6	nextlevel	1

Inverse Document Frequency

In [44]:
```
for i,word in enumerate(tf1['words']):
  tf1.loc[i, 'idf'] = np.log(data.shape[0]/(len(data[data['Text'].str.contains(wo

tf1
```

Out[44]:

	words	tf	idf
0	id_aa_carmack	1	4.166415
1	tracing	1	7.600402
2	tried	1	5.808643
3	cyberpunk	1	5.115496
4	ray	1	5.035453
5	hdr	1	6.907255
6	nextlevel	1	6.907255

Term Frequency – Inverse Document Frequency (TF-IDF)

In [45]:
```
tf1['tfidf'] = tf1['tf'] * tf1['idf']
tf1
```

Out[45]:

	words	tf	idf	tfidf
0	id_aa_carmack	1	4.166415	4.166415
1	tracing	1	7.600402	7.600402
2	tried	1	5.808643	5.808643
3	cyberpunk	1	5.115496	5.115496
4	ray	1	5.035453	5.035453
5	hdr	1	6.907255	6.907255
6	nextlevel	1	6.907255	6.907255

In [46]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer(max_features=1000, lowercase=True, analyzer='word',
 stop_words= 'english',ngram_range=(1,1))
vect = tfidf.fit_transform(data['Text'])
vect
```

Out[46]:
```
<1999x1000 sparse matrix of type '<class 'numpy.float64'>'
        with 6978 stored elements in Compressed Sparse Row format>
```

Bag of Words

In [47]:
```
from sklearn.feature_extraction.text import CountVectorizer
bow = CountVectorizer(max_features=1000, lowercase=True, ngram_range=(1,1),analyz
data_bow = bow.fit_transform(data['Text'])
data_bow
```

Out[47]:
```
<1999x1000 sparse matrix of type '<class 'numpy.int64'>'
        with 7552 stored elements in Compressed Sparse Row format>
```

Sentiment Analysis

In [48]:
```
data['Text'][:5].apply(lambda x: TextBlob(x).sentiment)
```

Out[48]:
```
0                                  (-0.25, 0.75)
1                                     (0.0, 0.0)
2                                     (0.0, 0.0)
3                                     (0.0, 0.0)
4    (0.20000000000000004, 0.32222222222222224)
Name: Text, dtype: object
```

In [49]:
```
data['sentiment'] = data['Text'].apply(lambda x: TextBlob(x).sentiment[0] )
data[['Text','sentiment']]
```

Out[49]:

	Text	sentiment
0	kunalb11 im alien	-0.250000
1	id_aa_carmack ray tracing cyberpunk hdr nextle...	0.000000
2	joerogan spotify interview	0.000000
3	gtera27 doge underestimated	0.000000
4	teslacn congratulation china amazing execution...	0.200000
...	...	...
1994	true sound surreal negative propaganda still e...	0.152381
1995	make read ur term condition clicking accept	0.000000
1996	teslagong samwise gamgee	0.000000
1997	altho dumb dumber u0001f525u0001f525	-0.375000
1998	progress update august 28	0.000000

1999 rows × 2 columns

In [50]:
```
data['sentiment'] = data['Text'].apply(lambda x: TextBlob(x).sentiment[0] )
data[['Text','sentiment']].head()
```

Out[50]:

	Text	sentiment
0	kunalb11 im alien	-0.25
1	id_aa_carmack ray tracing cyberpunk hdr nextle...	0.00
2	joerogan spotify interview	0.00
3	gtera27 doge underestimated	0.00
4	teslacn congratulation china amazing execution...	0.20

Perform emotion mining

Read Data

In [51]:
```
! pip install future
```

```
Requirement already satisfied: future in c:\programdata\anaconda3\lib\site-pack
ages (0.18.2)
```

In [52]:
```
pip install -U future
```

```
Requirement already satisfied: future in c:\programdata\anaconda3\lib\site-pack
ages (0.18.2)
Note: you may need to restart the kernel to use updated packages.
```

In [53]:
```
import codecs
import re
import copy
import collections
import pandas as pd
import numpy as np
import nltk
from nltk.stem import PorterStemmer
from nltk.tokenize import WordPunctTokenizer
import matplotlib

%matplotlib inline
```

In [54]:
```
from __future__ import division
import os
from nltk.corpus import twitter_samples
```

In [55]:
```
nltk.download('stopwords')
```

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

Out[55]:
```
True
```

In [56]:
```
from nltk.corpus import stopwords
```

In [57]:
```python
with codecs.open("positive-words.txt", "r", encoding="utf-8") as p:
    pos = p.read()
    print(pos)
```

```
adaptable
adaptive
adequate
adjustable
admirable
admirably
admiration
admire
admirer
admiring
admiringly
adorable
adore
adored
adorer
adoring
adoringly
adroit
adroitly
adulate
```

In [58]:
```python
with codecs.open("negative-words.txt", "r", encoding="ISO-8859-1") as n:
    neg = n.read()
    print(neg)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;
; Opinion Lexicon: Negative
;
; This file contains a list of NEGATIVE opinion words (or sentiment words).
;
; This file and the papers can all be downloaded from
;    http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html (http://www.cs.u
ic.edu/~liub/FBS/sentiment-analysis.html)
;
; If you use this list, please cite one of the following two papers:
;
;   Minqing Hu and Bing Liu. "Mining and Summarizing Customer Reviews."
;       Proceedings of the ACM SIGKDD International Conference on Knowledge
;       Discovery and Data Mining (KDD-2004), Aug 22-25, 2004, Seattle,
;       Washington, USA,
;   Bing Liu, Minqing Hu and Junsheng Cheng. "Opinion Observer: Analyzing
;       and Comparing Opinions on the Web." Proceedings of the 14th
```

In [59]:
```
with codecs.open("stop.txt", "r", encoding="ISO-8859-1") as s:
    stop = s.read()
    print(stop)
```

```
a
a's
able
about
above
according
accordingly
across
actually
after
afterwards
again
against
ain't
all
allow
allows
almost
alone
```

In [61]:
```
import nltk
nltk.download('punkt')
```

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

Out[61]:
```
True
```

In [65]:
```
from nltk.corpus import twitter_samples

positive_tweets = twitter_samples.strings('positive_tweets.json')
negative_tweets = twitter_samples.strings('negative_tweets.json')
text = twitter_samples.strings('tweets.20150430-223406.json')
```

In [66]:
```
from nltk.corpus import twitter_samples

positive_tweets = twitter_samples.strings('positive_tweets.json')
negative_tweets = twitter_samples.strings('negative_tweets.json')
text = twitter_samples.strings('tweets.20150430-223406.json')
```

In [67]:
```
from nltk.corpus import twitter_samples

positive_tweets = twitter_samples.strings('positive_tweets.json')
negative_tweets = twitter_samples.strings('negative_tweets.json')
text = twitter_samples.strings('tweets.20150430-223406.json')
tweet_tokens = twitter_samples.tokenized('positive_tweets.json')[0]

print(tweet_tokens[0])
```

```
#FollowFriday
```

In [68]:
```
!pip3 install beautifulsoup4
```

```
Requirement already satisfied: beautifulsoup4 in c:\programdata\anaconda3\lib\s
ite-packages (4.9.3)
Requirement already satisfied: soupsieve>1.2 in c:\programdata\anaconda3\lib\si
te-packages (from beautifulsoup4) (2.2.1)
```

In [69]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import re
import time
from datetime import datetime
import matplotlib.dates as mdates
import matplotlib.ticker as ticker
from urllib.request import urlopen
from bs4 import BeautifulSoup
import requests
```

In [70]:

```
no_pages = 2

def get_data(pageNo):
    headers = {"User-Agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:66.0) G

    r = requests.get('https://www.amazon.in/gp/bestsellers/books/ref=zg_bs_pg_'+s
    content = r.content
    soup = BeautifulSoup(content)
    #print(soup)

    alls = []
    for d in soup.findAll('div', attrs={'class':'a-section a-spacing-none aok-rel
        #print(d)
        name = d.find('span', attrs={'class':'zg-text-center-align'})
        n = name.find_all('img', alt=True)
        #print(n[0]['alt'])
        author = d.find('a', attrs={'class':'a-size-small a-link-child'})
        rating = d.find('span', attrs={'class':'a-icon-alt'})
        users_rated = d.find('a', attrs={'class':'a-size-small a-link-normal'})
        price = d.find('span', attrs={'class':'p13n-sc-price'})

        all1=[]

        if name is not None:
            #print(n[0]['alt'])
            all1.append(n[0]['alt'])
        else:
            all1.append("unknown-product")

        if author is not None:
            #print(author.text)
            all1.append(author.text)
        elif author is None:
            author = d.find('span', attrs={'class':'a-size-small a-color-base'})
            if author is not None:
                all1.append(author.text)
            else:
                all1.append('0')

        if rating is not None:
            #print(rating.text)
            all1.append(rating.text)
        else:
            all1.append('-1')

        if users_rated is not None:
            #print(price.text)
            all1.append(users_rated.text)
        else:
            all1.append('0')

        if price is not None:
            #print(price.text)
            all1.append(price.text)
        else:
            all1.append('0')
```

```
        alls.append(all1)
    return alls
```

In [71]:
```
results = []
for i in range(1, no_pages+1):
    results.append(get_data(i))
flatten = lambda l: [item for sublist in l for item in sublist]
df = pd.DataFrame(flatten(results),columns=['Book Name','Author','Rating','Custom
df.to_csv('amazon_products.csv', index=False, encoding='utf-8')
```

In [73]:
```
df = pd.read_csv("amazon_products.csv")
df
```

Out[73]:

	Book Name	Author	Rating	Customers_Rated	Price
0	The Psychology of Money	Morgan Housel	4.6 out of 5 stars	23,931	₹225.00
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6 out of 5 stars	29,854	₹438.00
2	Ikigai: The Japanese secret to a long and happ...	Héctor García	4.6 out of 5 stars	27,983	₹285.00
3	Word Power Made Easy	Norman Lewis	4.4 out of 5 stars	33,693	₹91.00
4	Rich Dad Poor Dad: What the Rich Teach Their K...	Robert T. Kiyosaki	4.6 out of 5 stars	51,369	₹241.00
...	...	...	...	...	...
95	Indian Art and Culture for Civil Services and ...	Nitin Singhania	4.6 out of 5 stars	25	₹580.00
96	101 Brain Booster Activity Book: Fun Activity ...	Wonder House Books	4.4 out of 5 stars	3,605	₹139.00
97	Peppa Pig: Little Library	Peppa Pig	4.3 out of 5 stars	16,632	₹152.00
98	The Seven Husbands of Evelyn Hugo: A Novel	Taylor Jenkins Reid	4.6 out of 5 stars	21,416	₹276.00
99	My First 4 in 1 Alphabet Numbers Colours Shape...	Wonder House Books	4.3 out of 5 stars	2,416	₹158.00

100 rows × 5 columns

In [74]:
```
df.head()
```

Out[74]:

	Book Name	Author	Rating	Customers_Rated	Price
0	The Psychology of Money	Morgan Housel	4.6 out of 5 stars	23,931	₹225.00
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6 out of 5 stars	29,854	₹438.00
2	Ikigai: The Japanese secret to a long and happ...	Héctor García	4.6 out of 5 stars	27,983	₹285.00
3	Word Power Made Easy	Norman Lewis	4.4 out of 5 stars	33,693	₹91.00
4	Rich Dad Poor Dad: What the Rich Teach Their K...	Robert T. Kiyosaki	4.6 out of 5 stars	51,369	₹241.00

In [75]:
```
df.shape
```

Out[75]:
```
(100, 5)
```

In [76]:
```
df.head(61)
```

Out[76]:

	Book Name	Author	Rating	Customers_Rated	Price
0	The Psychology of Money	Morgan Housel	4.6 out of 5 stars	23,931	₹225.00
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6 out of 5 stars	29,854	₹438.00
2	Ikigai: The Japanese secret to a long and happ...	Héctor García	4.6 out of 5 stars	27,983	₹285.00
3	Word Power Made Easy	Norman Lewis	4.4 out of 5 stars	33,693	₹91.00
4	Rich Dad Poor Dad: What the Rich Teach Their K...	Robert T. Kiyosaki	4.6 out of 5 stars	51,369	₹241.00
...	...	...	...	...	...
56	Indian Polity For Civil Services and Other Sta...	M Laxmikanth	4.4 out of 5 stars	424	₹620.00
57	Current Affairs Yearly 2022	Arihant Experts	-1	0	₹99.00
58	One Arranged Murder	Chetan Bhagat	4.4 out of 5 stars	17,065	₹140.00
59	My First Complete Learning Library: Boxset of ...	Wonder House Books	4.6 out of 5 stars	8,179	₹669.00
60	Three Thousand Stitches: Ordinary People, Extr...	Sudha Murty	4.6 out of 5 stars	6,293	₹144.00

61 rows × 5 columns

In [77]:
```
df['Rating'] = df['Rating'].apply(lambda x: x.split()[0])
```

In [78]:
```
df['Rating'] = pd.to_numeric(df['Rating'])
```

In [79]:
```
df["Price"] = df["Price"].str.replace('₹', '')
```

In [80]:
```
df["Price"] = df["Price"].str.replace(',', '')
```

In [81]:
```
df['Price'] = df['Price'].apply(lambda x: x.split('.')[0])
```

In [82]:
```
df['Price'] = df['Price'].astype(int)
```

In [83]:
```
df["Customers_Rated"] = df["Customers_Rated"].str.replace(',', '')
```

In [84]:
```
df['Customers_Rated'] = pd.to_numeric(df['Customers_Rated'], errors='ignore')
```

In [85]:
```
df.head()
```

Out[85]:

	Book Name	Author	Rating	Customers_Rated	Price
0	The Psychology of Money	Morgan Housel	4.6	23931	225
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6	29854	438
2	Ikigai: The Japanese secret to a long and happ...	Héctor García	4.6	27983	285
3	Word Power Made Easy	Norman Lewis	4.4	33693	91
4	Rich Dad Poor Dad: What the Rich Teach Their K...	Robert T. Kiyosaki	4.6	51369	241

In [86]:
```
df.dtypes
```

Out[86]:
```
Book Name           object
Author              object
Rating             float64
Customers_Rated      int64
Price                int32
dtype: object
```

In [87]:
```
df.replace(str(0), np.nan, inplace=True)
df.replace(0, np.nan, inplace=True)
```

In [88]:
```
count_nan = len(df) - df.count()
```

In [89]:
```
count_nan
```

Out[89]:
```
Book Name          0
Author             0
Rating             0
Customers_Rated    2
Price              0
dtype: int64
```

In [90]:
```
df = df.dropna()
```

In [91]:
```
data = df.sort_values(["Price"], axis=0, ascending=False)[:15]
data
```

Out[91]:

	Book Name	Author	Rating	Customers_Rated	Price
72	Harry Potter Box Set: The Complete Collection ...	J.K. Rowling	4.7	29047.0	2417
86	Target High 6th Premium Edition	MUTHUVENKATACHALAM S.	4.6	510.0	1002
30	Oswaal CBSE Question Bank Chapterwise For Term...	Oswaal Editorial Board	4.9	173.0	810
59	My First Complete Learning Library: Boxset of ...	Wonder House Books	4.6	8179.0	669
55	Educart TERM 2 CBSE Question Bank Bundle - Mat...	Educart	4.9	72.0	663
56	Indian Polity For Civil Services and Other Sta...	M Laxmikanth	4.4	424.0	620
95	Indian Art and Culture for Civil Services and ...	Nitin Singhania	4.6	25.0	580
84	MTG Objective NCERT at your FINGERTIPS - Biolo...	MTG Editorial Board	4.6	6931.0	577
70	A Modern Approach To Verbal & Non-Verbal Reaso...	R.S. Aggarwal	4.4	10190.0	511
80	India that is Bharat: Coloniality, Civilisatio...	J Sai Deepak	4.8	2233.0	503
41	The Intelligent Investor (English) Paperback –...	Benjamin Graham	4.5	30011.0	448
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6	29854.0	438
16	Quantitative Aptitude for Competitive Examinat...	R S Aggarwal	4.4	22587.0	428
89	Coffee Can Investing: The Low Risk Road to Stu...	Saurabh Mukherjea	4.5	2590.0	374
8	DO EPIC SHIT	Ankur Warikoo	4.5	1064.0	363

In [92]:
```
from bokeh.models import ColumnDataSource
from bokeh.transform import dodge
import math
from bokeh.io import curdoc
curdoc().clear()
from bokeh.io import push_notebook, show, output_notebook
from bokeh.layouts import row
from bokeh.plotting import figure
from bokeh.transform import factor_cmap
from bokeh.models import Legend
output_notebook()
```

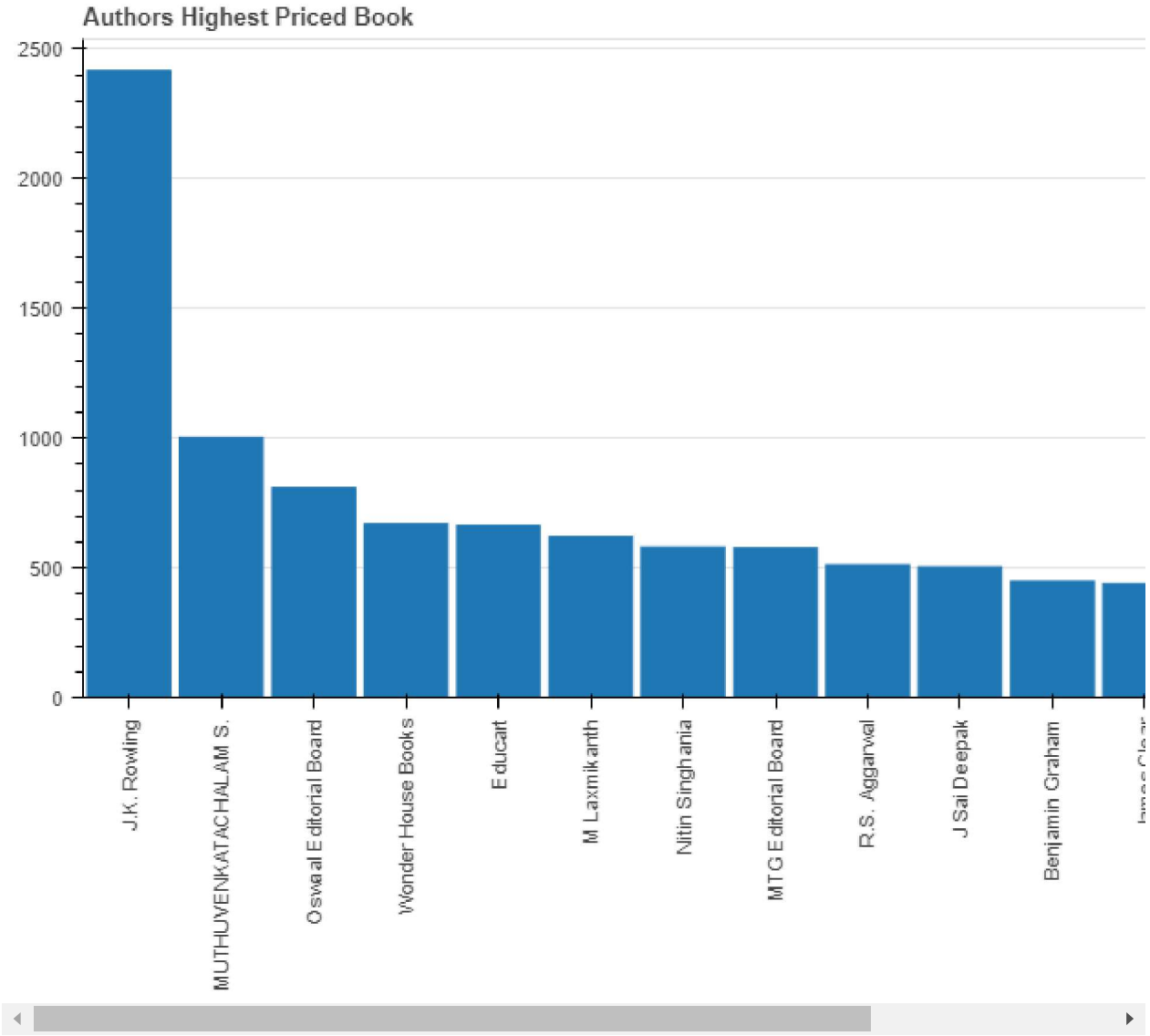
(https://bokeh.org)) BokehJS 2.3.2 successfully loaded.

In [93]:
```
p = figure(x_range=data.iloc[:,1], plot_width=800, plot_height=550, title="Author

p.vbar(x=data.iloc[:,1], top=data.iloc[:,4], width=0.9)

p.xgrid.grid_line_color = None
p.y_range.start = 0
p.xaxis.major_label_orientation = math.pi/2
```

In [94]:
```
show(p)
```



In [95]:
```
data = df[df['Customers_Rated'] > 1000]
```

In [96]:
```
data = data.sort_values(['Rating'],axis=0, ascending=False)[:15]
data
```

Out[96]:

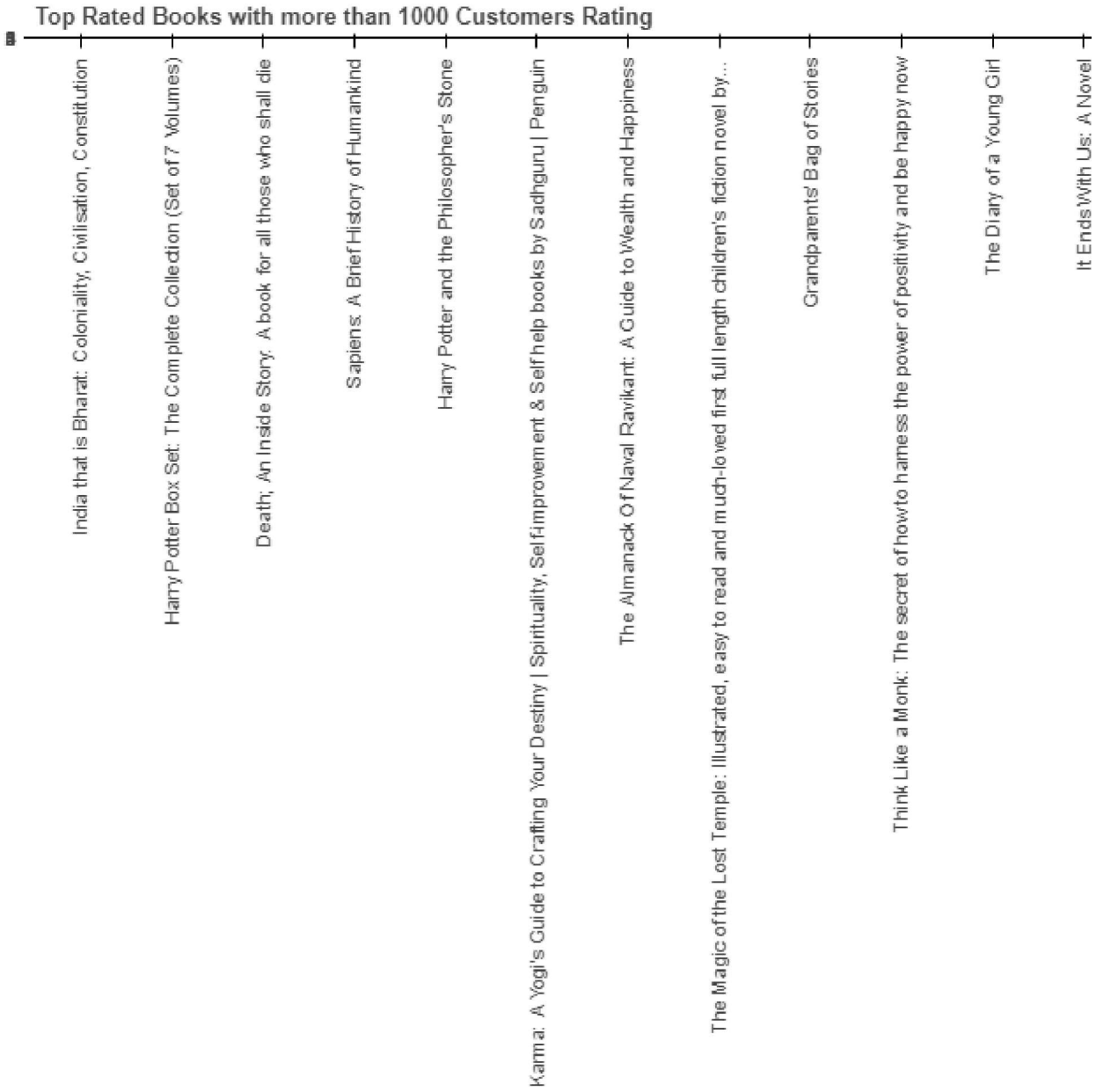
	Book Name	Author	Rating	Customers_Rated	Price
80	India that is Bharat: Coloniality, Civilisatio...	J Sai Deepak	4.8	2233.0	503
72	Harry Potter Box Set: The Complete Collection ...	J.K. Rowling	4.7	29047.0	2417
67	Death; An Inside Story: A book for all those w...	Sadhguru	4.7	8329.0	158
27	Sapiens: A Brief History of Humankind	Yuval Noah Harari	4.7	37581.0	336
91	Harry Potter and the Philosopher's Stone	J.K. Rowling	4.7	33221.0	275
31	Karma: A Yogi's Guide to Crafting Your Destiny...	Sadhguru	4.7	6959.0	143
9	The Almanack Of Naval Ravikant: A Guide to Wea...	Eric Jorgenson	4.7	2986.0	195
82	The Magic of the Lost Temple: Illustrated, eas...	Sudha Murty	4.7	5462.0	159
45	Grandparents' Bag of Stories	Sudha Murty	4.7	2457.0	137
63	Think Like a Monk: The secret of how to harnes...	Jay Shetty	4.6	19622.0	280
36	The Diary of a Young Girl	Anne Frank	4.6	17429.0	99
22	It Ends With Us: A Novel	Colleen Hoover	4.6	23065.0	341
60	Three Thousand Stitches: Ordinary People, Extr...	Sudha Murty	4.6	6293.0	144
29	Eat That Frog!: 21 Great Ways to Stop Procrast...	Brian Tracy	4.6	4605.0	162
75	The Theory of Everything: The Origin and Fate ...	Stephen Hawking	4.6	9426.0	129

In [98]:
```
p = figure(x_range=data.iloc[:,0], plot_width=800, plot_height=600, title="Top Ra

p.vbar(x=data.iloc[:,0], top=data.iloc[:,2], width=0.9)

p.xgrid.grid_line_color = None
p.y_range.start = 0
p.xaxis.major_label_orientation = math.pi/2
```

In [99]:
```
show(p)
```



In [100]:
```
p = figure(x_range=data.iloc[:,1], plot_width=800, plot_height=600, title="Top Ra

p.vbar(x=data.iloc[:,1], top=data.iloc[:,2], width=0.9)

p.xgrid.grid_line_color = None
p.y_range.start = 0
p.xaxis.major_label_orientation = math.pi/2
```

In [101]:
```
show(p)
```

```
ERROR:bokeh.core.validation.check:E-1019 (DUPLICATE_FACTORS): FactorRange must
specify a unique list of categorical factors for an axis: duplicate factors fou
nd: 'J.K. Rowling', 'Sadhguru', 'Sudha Murty'
```

In [102]:
```
data = df.sort_values(["Customers_Rated"], axis=0, ascending=False)[:20]
data
```

Out[102]:

	Book Name	Author	Rating	Customers_Rated	Price
10	The Alchemist	Paulo Coelho	4.6	66602.0	196
13	The Subtle Art of Not Giving a F*ck: A Counter...	Mark Manson	4.5	55167.0	279
14	How to Win Friends and Influence People	Dale Carnegie	4.5	54253.0	76
4	Rich Dad Poor Dad: What the Rich Teach Their K...	Robert T. Kiyosaki	4.6	51369.0	241
69	The Power of Your Subconscious Mind	Joseph Murphy	4.5	47008.0	99
12	The Power of Your Subconscious Mind	Joseph Murphy	4.5	47008.0	119
5	My First Library: Boxset of 10 Board Books for...	Wonder House Books	4.5	42942.0	349
27	Sapiens: A Brief History of Humankind	Yuval Noah Harari	4.7	37581.0	336
20	Man's Search For Meaning: The classic tribute ...	Viktor E Frankl	4.5	36829.0	182
3	Word Power Made Easy	Norman Lewis	4.4	33693.0	91
91	Harry Potter and the Philosopher's Stone	J.K. Rowling	4.7	33221.0	275
41	The Intelligent Investor (English) Paperback –...	Benjamin Graham	4.5	30011.0	448
1	Atomic Habits: The life-changing million copy ...	James Clear	4.6	29854.0	438
72	Harry Potter Box Set: The Complete Collection ...	J.K. Rowling	4.7	29047.0	2417
2	Ikigai: The Japanese secret to a long and happ...	Héctor García	4.6	27983.0	285
87	Pride & Prejudice	Jane Austen	4.5	24515.0	99
39	The Richest Man in Babylon	George S. Clason	4.5	24241.0	99
0	The Psychology of Money	Morgan Housel	4.6	23931.0	225
22	It Ends With Us: A Novel	Colleen Hoover	4.6	23065.0	341
16	Quantitative Aptitude for Competitive Examinat...	R S Aggarwal	4.4	22587.0	428

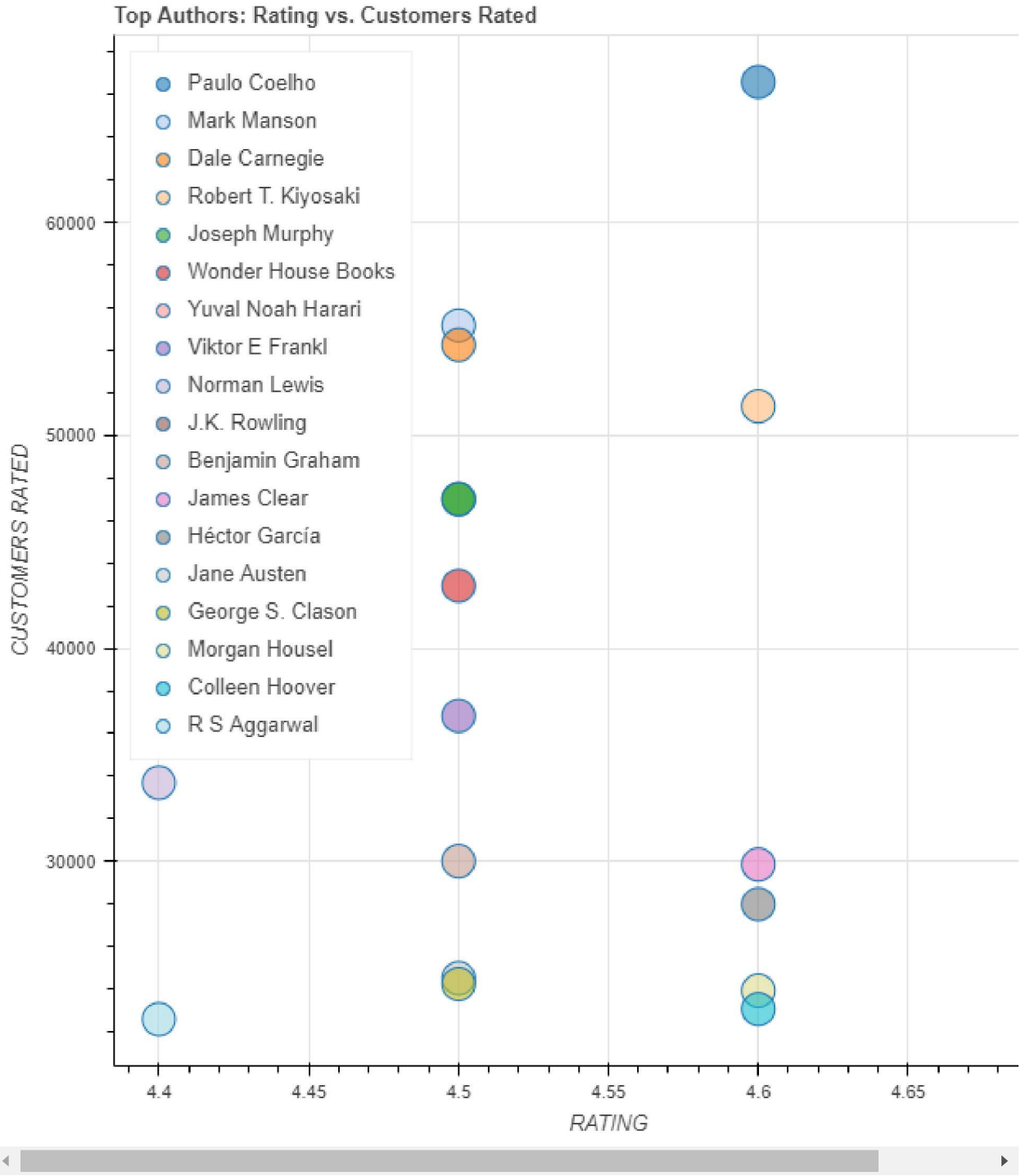
In [103]:
```
from bokeh.transform import factor_cmap
from bokeh.models import Legend
from bokeh.palettes import Dark2_5 as palette
import itertools
from bokeh.palettes import d3
#colors has a list of colors which can be used in plots
colors = itertools.cycle(palette)

palette = d3['Category20'][20]
```

In [104]:
```
index_cmap = factor_cmap('Author', palette=palette,
                         factors=data["Author"])
p = figure(plot_width=700, plot_height=700, title = "Top Authors: Rating vs. Cust
p.scatter('Rating','Customers_Rated',source=data,fill_alpha=0.6, fill_color=index
p.xaxis.axis_label = 'RATING'
p.yaxis.axis_label = 'CUSTOMERS RATED'
p.legend.location = 'top_left'
```

```
BokehDeprecationWarning: 'legend' keyword is deprecated, use explicit 'legend_l
abel', 'legend_field', or 'legend_group' keywords instead
```

In [105]:
```
show(p)
```



In [ ]:
```

```