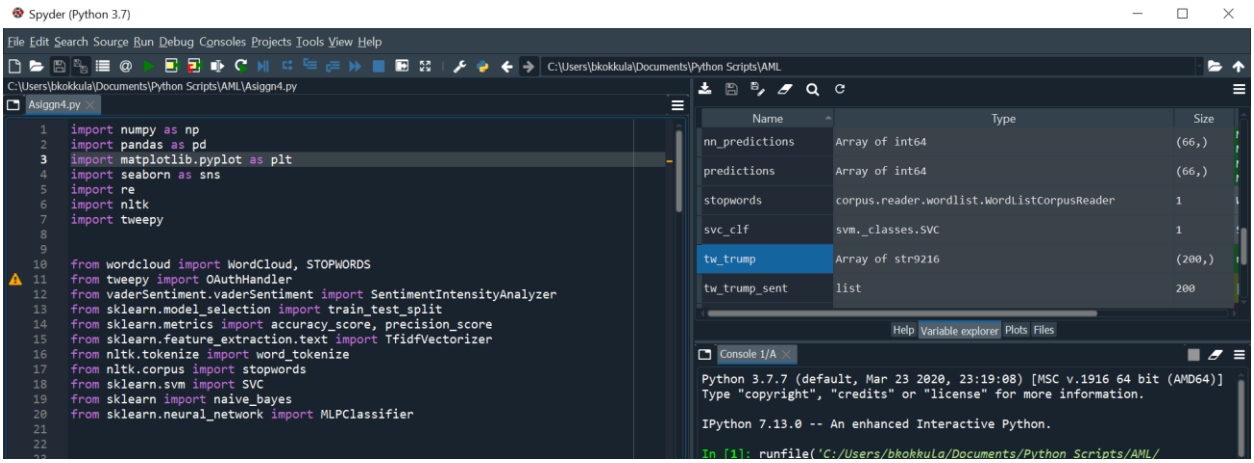
AML Major Assignment1: Twitter Data

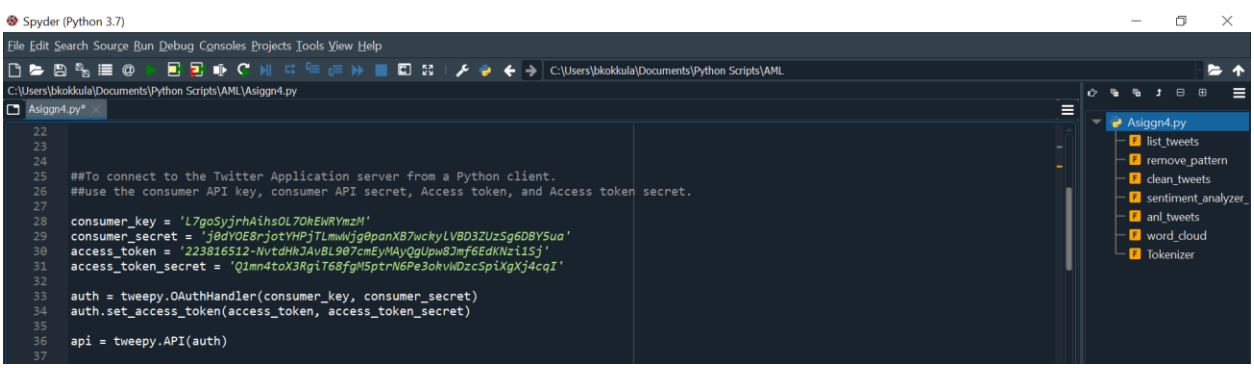
Scraping tweets from twitter Sentiment: Analysis is a special case of text classification where users opinions or sentiments regarding a product are classified into predefined categories such as positive, negative, neutral etc. Public sentiments can then be used for corporate decision making regarding a product which is being liked or disliked by the public.

Importing Libraries: Using Python for developing a sentiment analysis model, you need to import the required libraries. The following script does that:



In the script above, we import “Numpy”, “Pandas”, “Matplotlib” “seaborn” “tweepy” “NLTK” and “re” libraries.

Connecting Python Client Application to Twitter Server: To connect to the Twitter Application server from a Python client, use the consumer API key, consumer API secret, Access token, and Access token secret. Execute the following script:



Scraping Tweets: We have successfully connected to the Twitter API. The next step is to fetch tweets. Next, create an empty list alltweets which will contain the scraped tweets. In the search query specify the string “realDonaldTrump” which means that you want to search the tweets that contain the word “realDonaldTrump”.

```python
# Creating Twitter List

def list_tweets(user_id, count, prt=False):
    tweets = api.user_timeline(
        "@" + user_id, count=count, tweet_mode='extended')
    tw = []
    for t in tweets:
        tw.append(t.full_text)
        if prt:
            print(t.full_text)
            print()
    return tw

# trump user name in twitter.
user_id = 'realDonaldTrump'
count=200
```

Once you execute the script above, you will see 200 most recent tweets containing the string “realDonaldTrump” will be stored in the all tweets list and with that, we end the first part of the article.

Performing Vader Sentimental Analysis:

We have scraped live tweets from twitter. To create a vader sentimental analysis model using existing dataset and to use that model to predict sentiments for the 200 tweets that you scraped. Follow these steps to perform sentiment analysis on scraped tweets:

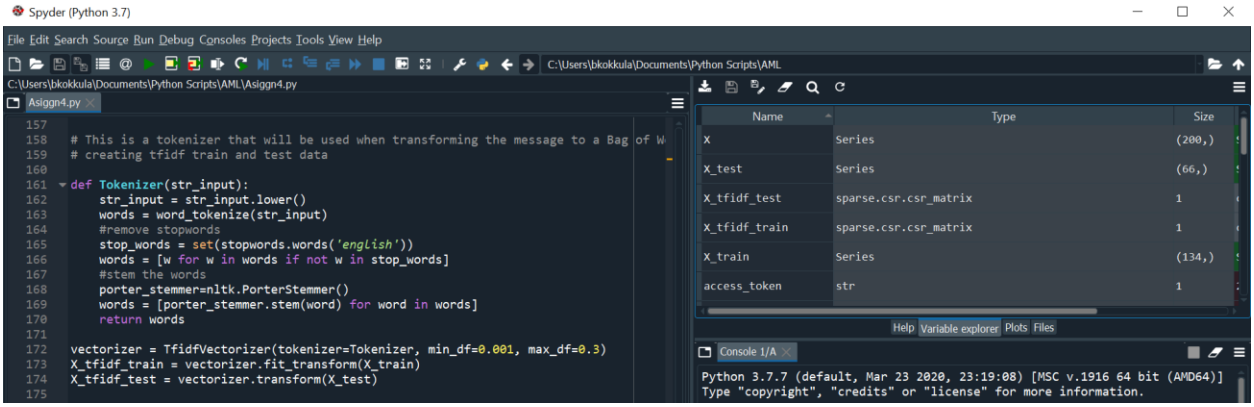
```python
# Performing vader sentiment analysis

analyser = SentimentIntensityAnalyzer()
analyser.polarity_scores("The movie is good")


def sentiment_analyzer_scores(text):
    score = analyser.polarity_scores(text)
    lb = score['compound']
    if lb >= 0.05:
        return 1
    elif (lb > -0.05) and (lb < 0.05):
        return 0
    else:
```

Data Preprocessing: we will divide the data into the label and feature set and then will remove special characters and empty spaces from the tweets. Execute the following script to do so:

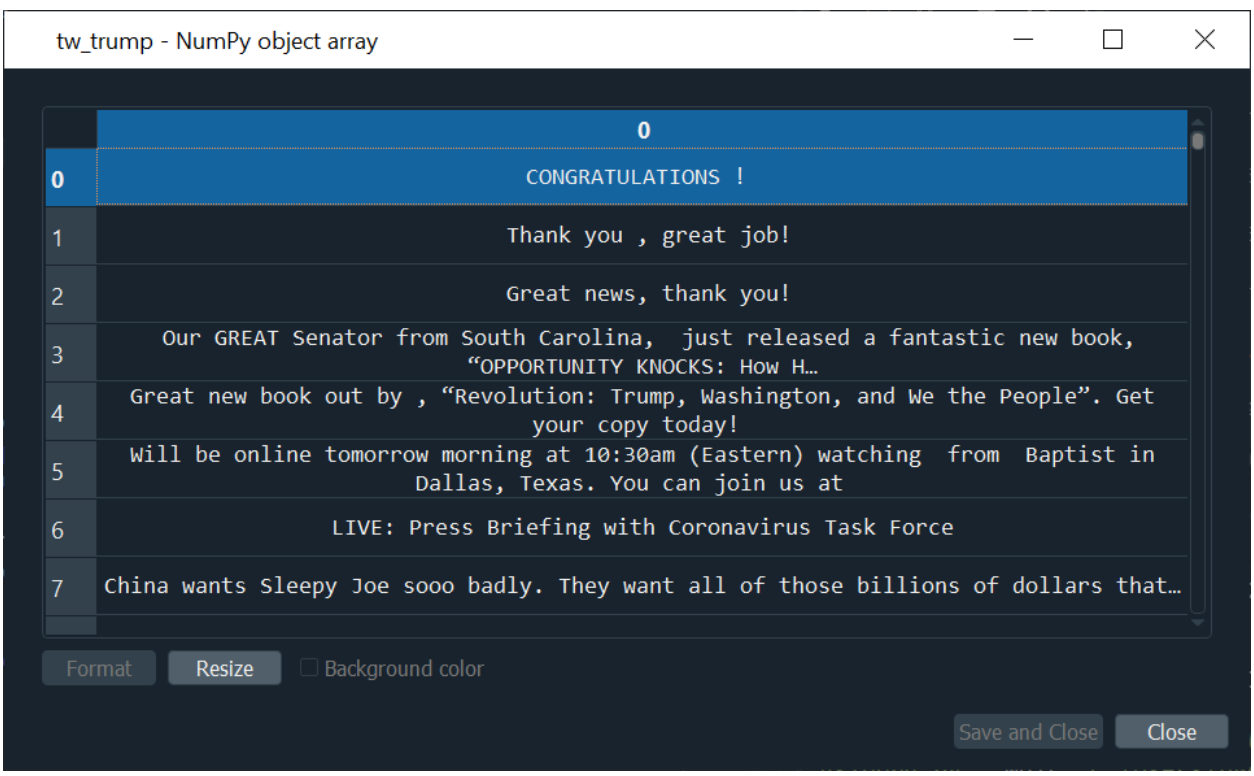
```python
#Cleaning Twitter Dataset


def remove_pattern(input_txt, pattern):
    r = re.findall(pattern, input_txt)
    for i in r:
        input_txt = re.sub(i, '', input_txt)
    return input_txt


def clean_tweets(lst):
    # remove twitter Return handles (RT @xxx:)
    lst = np.vectorize(remove_pattern)(lst, "RT @[\w]*:")
    # remove twitter handles (@xxx)
    lst = np.vectorize(remove_pattern)(lst, "@[\w]*")
    # remove URL links (httpxxx)
    lst = np.vectorize(remove_pattern)(lst, "https?://[A-Za-z0-9./]*")
    # remove special characters, numbers, punctuations (except for #)
    lst = np.core.defchararray.replace(lst, "[^a-zA-Z#]", " ")
    return lst
```

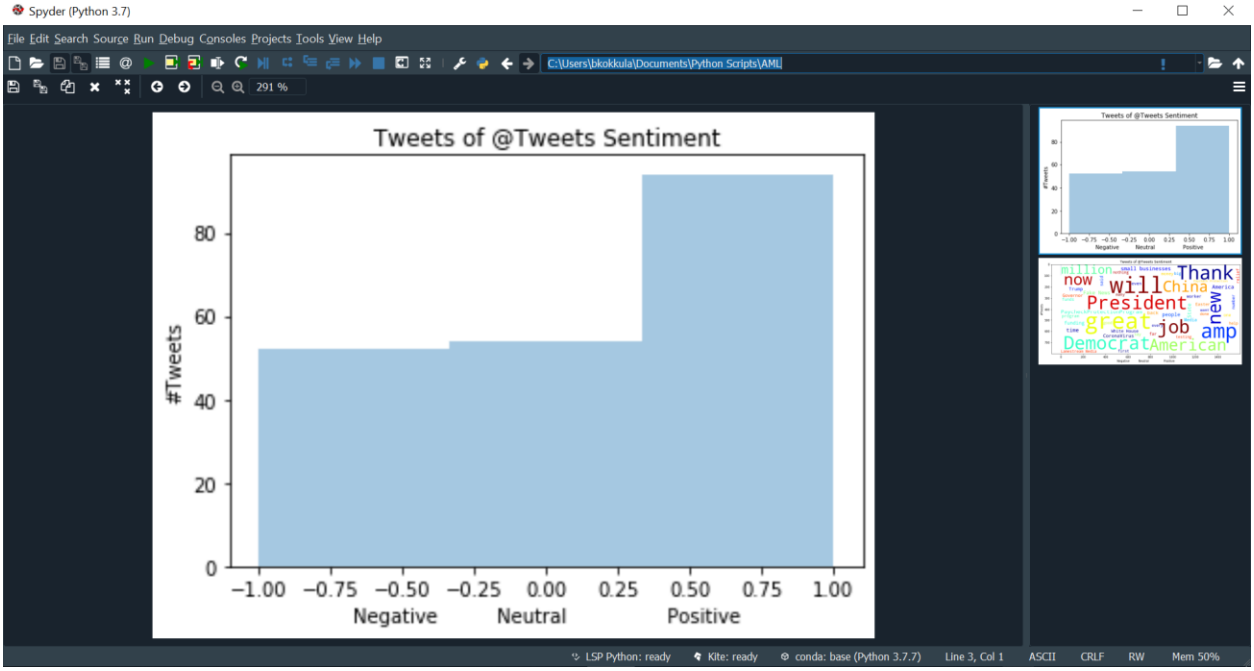
```
Python 3.7.7 (default, Mar 23 2020, 23:19:08) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.13.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/bkokkula/Documents/Python Scripts/AML/
Asiggn4.py', wdir='C:/Users/bkokkula/Documents/Python Scripts/AML')
```

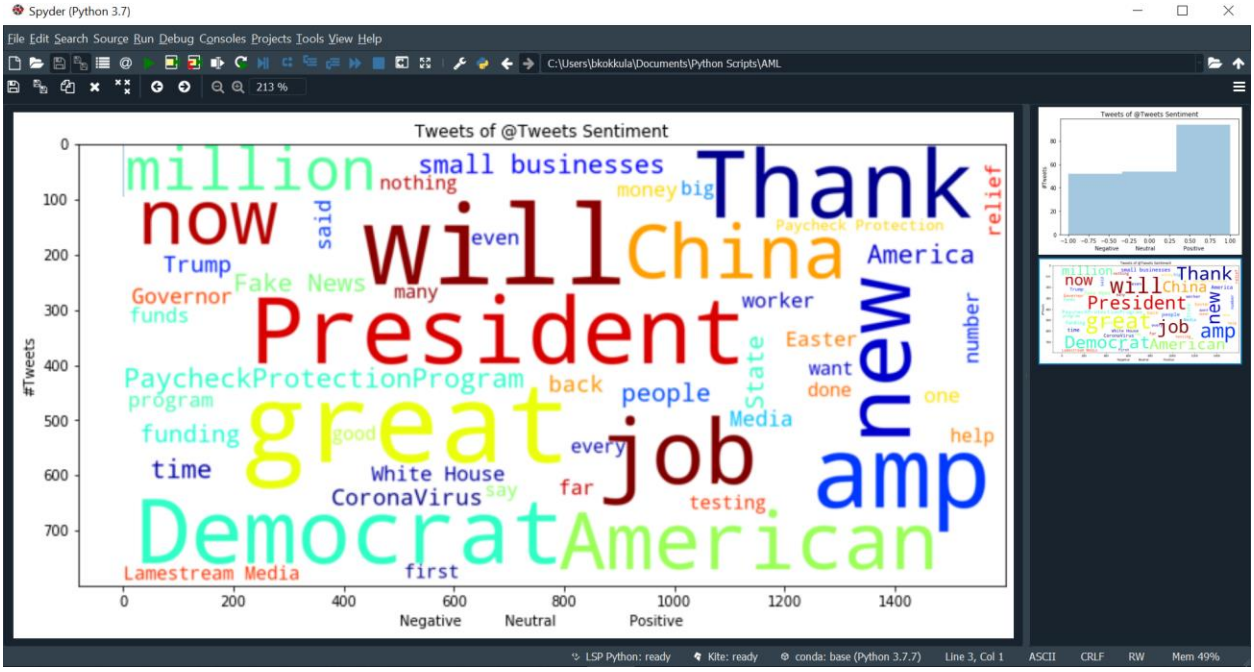
TF-IDF for Text to Numeric Conversion and creating transform train and test data sets: You can use the TFIDF scheme to convert text to numbers. The following script does that:



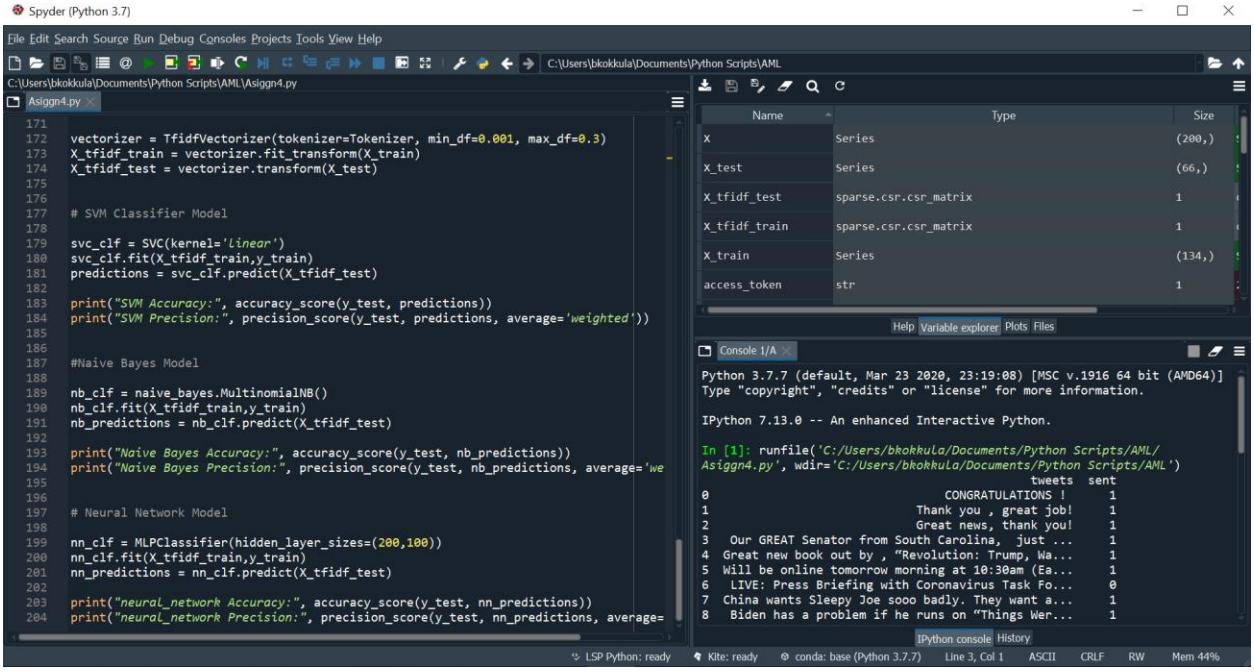
In the output, you will see each of the 200 scraped tweets along with its sentiment. A screenshot of the output from the Spyder console is shown below:

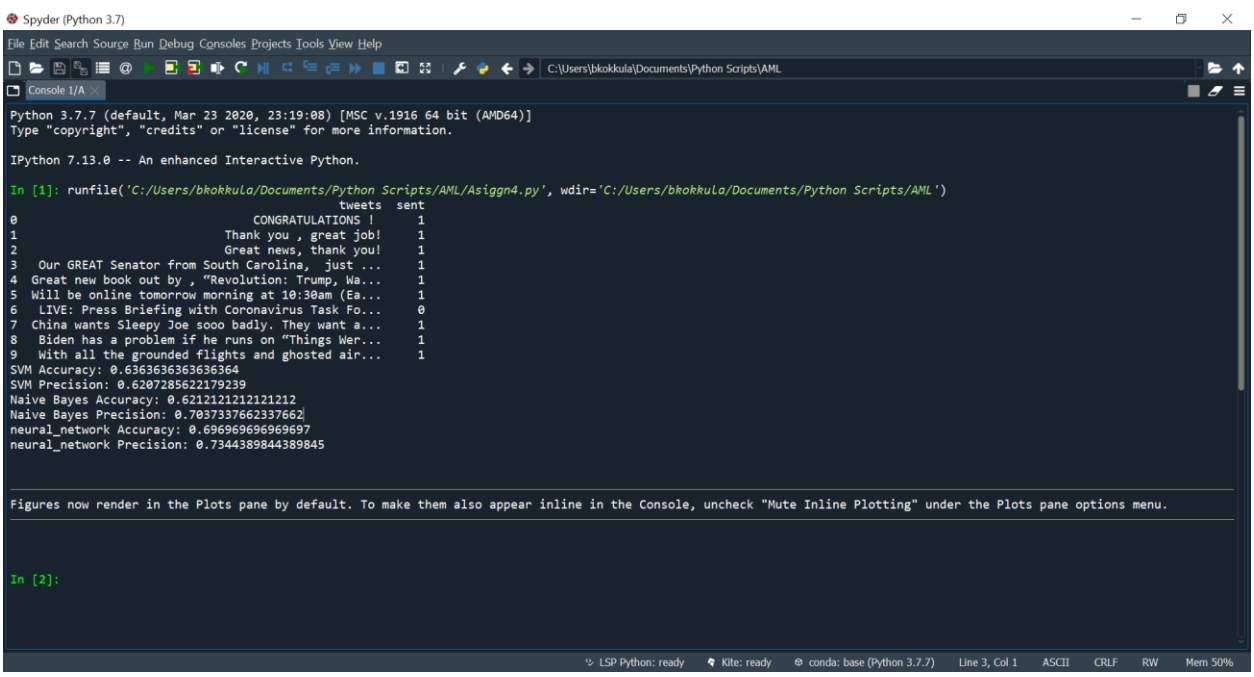



Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\bkokkula\Documents\Python Scripts\AML
Tweets of @Tweets Sentiment
#Tweets
0
20
40
60
80
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00
Negative
Neutral
Positive
LSP Python: ready
Kite: ready
conda: base (Python 3.7.7)
Line 3, Col 1
ASCII
CRLF
RW
Mem 50%


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\bkokkula\Documents\Python Scripts\AML
Tweets of @Tweets Sentiment
#Tweets
0
100
200
300
400
500
600
700
0 200 400 600 800 1000 1200 1400
Negative
Neutral
Positive
million
small businesses
nothing
money
big
Thank
now
said
will
even
China
Paycheck Protection
relief
Trump
America
Governor
Fake News
many
funds
worker
President
new
number
Easter
state
want
done
PaycheckProtectionProgram
back
people
one
program
Media
great
good
funding
every
job
help
time
White House
CoronaVirus
say
far
testing
amp
Democrat
American
Lamestream Media
first
LSP Python: ready
Kite: ready
conda: base (Python 3.7.7)
Line 3, Col 1
ASCII
CRLF
RW
Mem 49%

The performance of algorithms on the scraped tweets and see output from below:





Output Results: SVM Accuracy: 0.6363636363636364;

SVM Precision: 0.6207285622179239

Naive Bayes Accuracy: 0.6212121212121212

Naive Bayes Precision: 0.7037337662337662

neural _network Accuracy: 0.696969696969697

neural_network Precision: 0.7344389844389845

Conclusion: The sentimental analysis is one of the most important tasks in corporate decision making. Being aware of the public sentiment about a product can play a crucial role in the success or failure of the product. By see above accuracy of each model the neural network model performed best and followed by SVM model and NB model.