

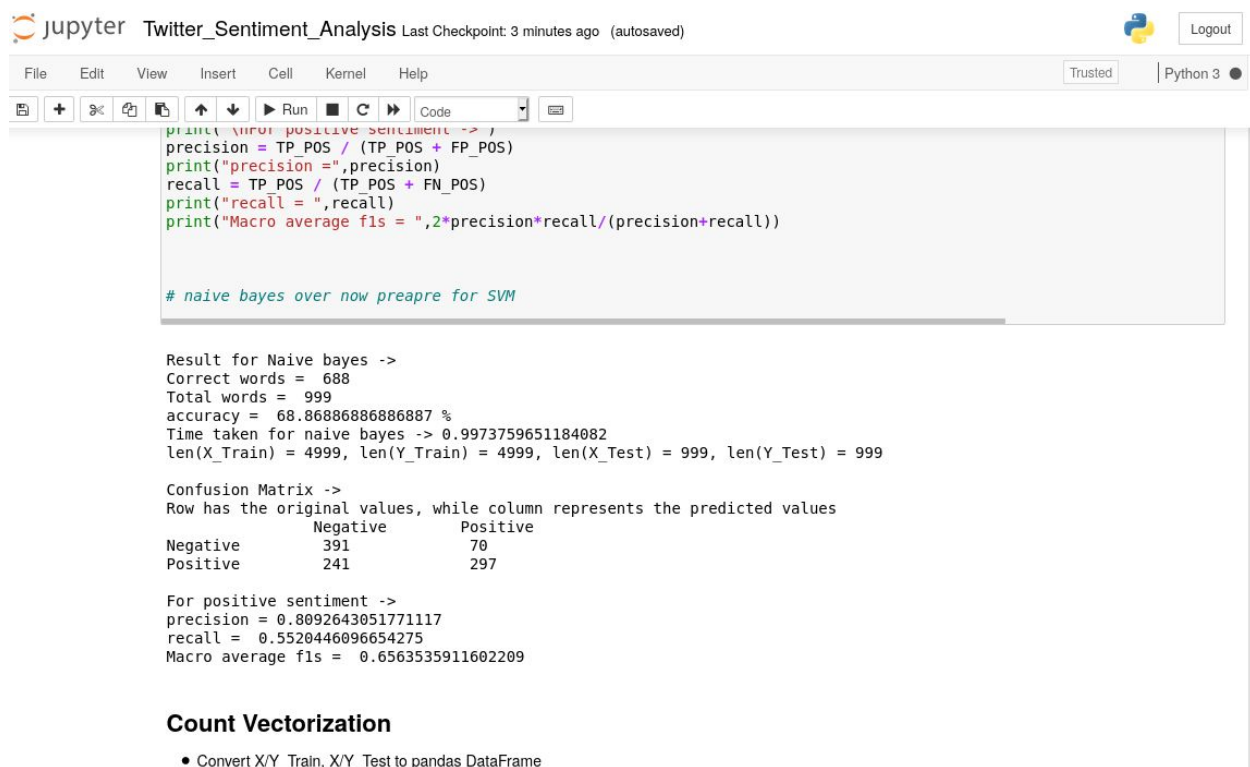
NLP : ASSIGNMENT - 3

OUTPUT

3a : Sentiment Analysis in Twitter

OUTPUT 1

The below screenshots are from the Twitter_Sentiment_Analysis.ipynb, in the conditions that the number of training & testing tweets were 5k & 1k respectively



The screenshot shows a Jupyter Notebook titled "Twitter_Sentiment_Analysis" with a last checkpoint 3 minutes ago. The code cell contains the following Python code:

```
print("\nFor positive sentiment -> ")
precision = TP_POS / (TP_POS + FP_POS)
print("precision =", precision)
recall = TP_POS / (TP_POS + FN_POS)
print("recall = ", recall)
print("Macro average f1s = ", 2*precision*recall/(precision+recall))

# naive bayes over now prepare for SVM
```

The output of the code is as follows:

```
Result for Naive bayes ->
Correct words = 688
Total words = 999
accuracy = 68.86886886886887 %
Time taken for naive bayes -> 0.9973759651184082
len(X_Train) = 4999, len(Y_Train) = 4999, len(X_Test) = 999, len(Y_Test) = 999

Confusion Matrix ->
Row has the original values, while column represents the predicted values
          Negative    Positive
Negative    391         70
Positive    241        297

For positive sentiment ->
precision = 0.8092643051771117
recall = 0.5520446096654275
Macro average f1s = 0.6563535911602209
```

Count Vectorization

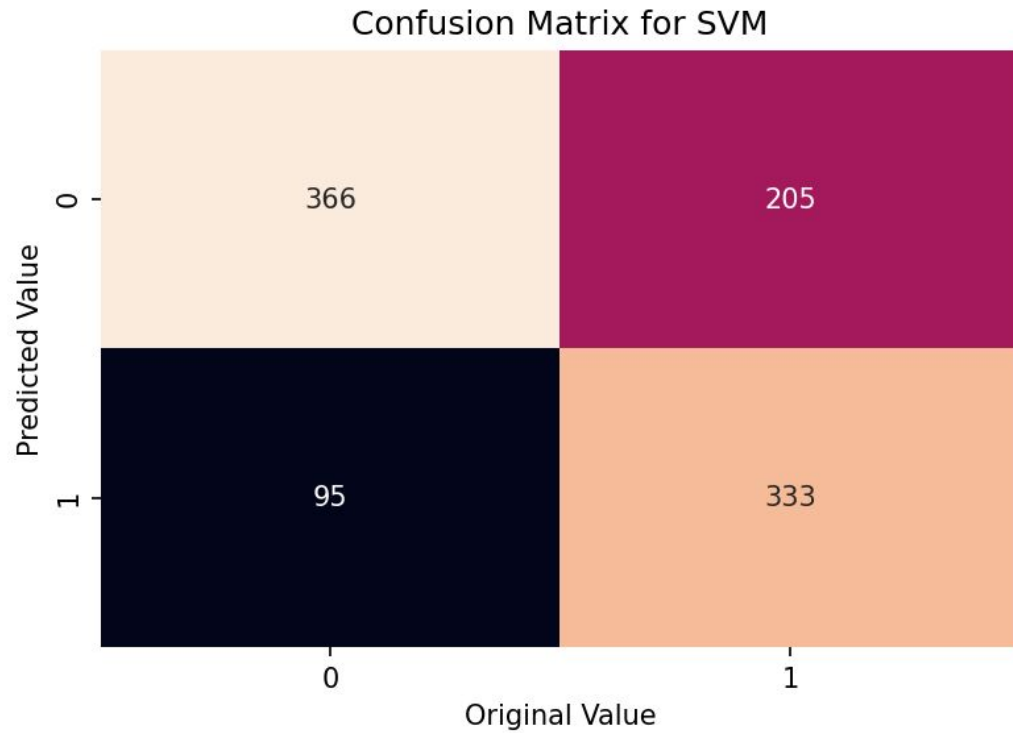
- Convert X/Y_Train, X/Y_Test to pandas DataFrame

This is the output for Naive Bayes - Bag Of Words approach

```

Now SVM model.....
No saved model... Creating one!
Time to train & save model -> 843.4613010883331
Accuracy by SVM -> 69.96996996996997 %
Time taken to predict -> 1167.4428622722626

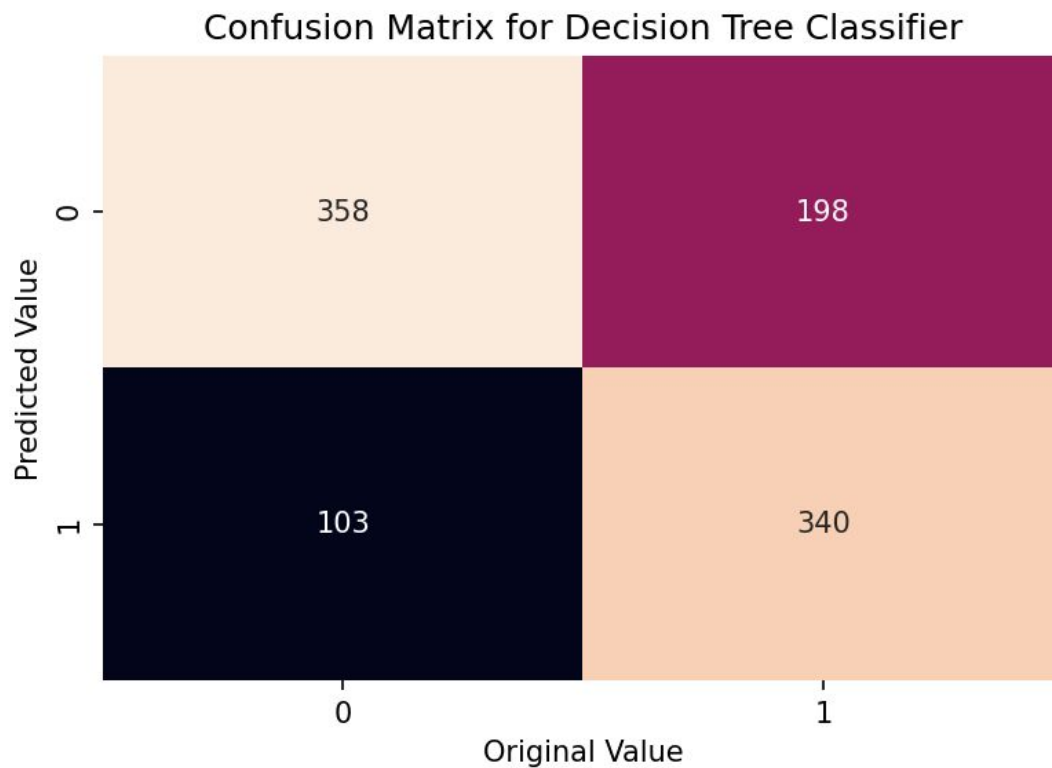
```



| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.64 | 0.79 | 0.71 | 461 |
| 1 | 0.78 | 0.62 | 0.69 | 538 |
| accuracy | | | 0.70 | 999 |
| macro avg | 0.71 | 0.71 | 0.70 | 999 |
| weighted avg | 0.71 | 0.70 | 0.70 | 999 |

This is the output for SVM

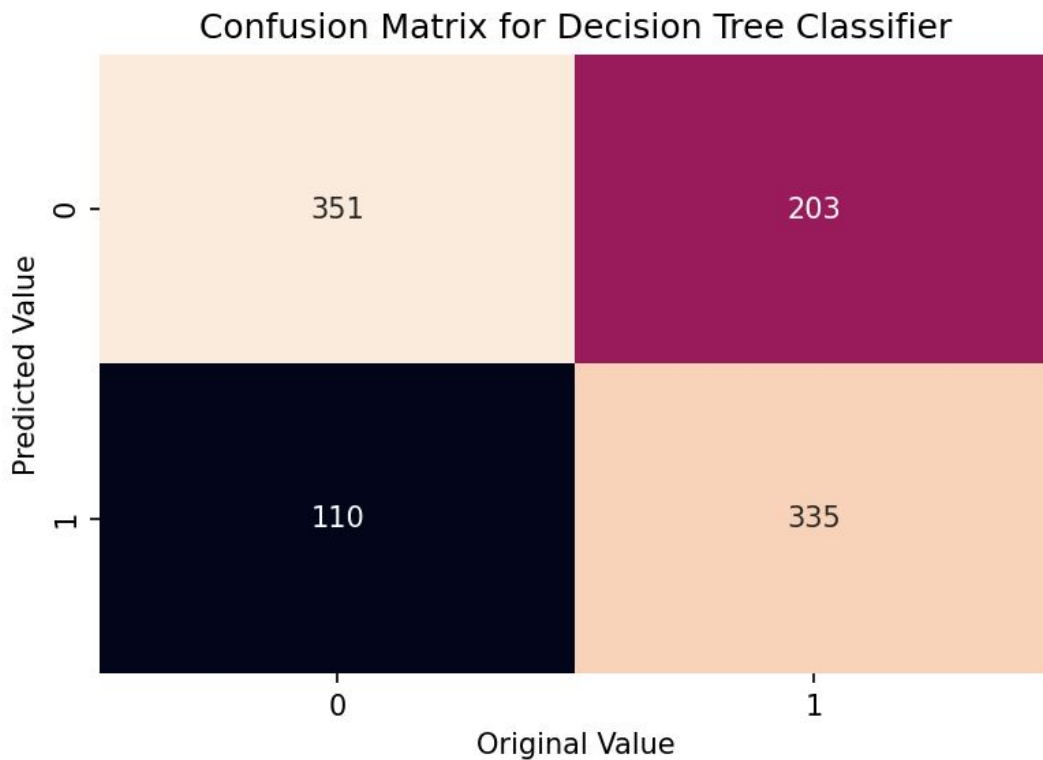
```
Now Decision Tree Classifier...
Accuracy by decision tree -> 69.86986986986987 %
Time taken for .fit & .predict -> 1172.3926916122437
```



| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.64 | 0.78 | 0.70 | 461 |
| 1 | 0.77 | 0.63 | 0.69 | 538 |
| accuracy | | | 0.70 | 999 |
| macro avg | 0.71 | 0.70 | 0.70 | 999 |
| weighted avg | 0.71 | 0.70 | 0.70 | 999 |

This is the output for Decision Tree Classifier

```
MLP Classifier now.....
Accuracy by MLP Classifier -> 68.66866866866866 %
Time taken for .fit & .predict -> 1205.3631665706635
```



| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.63 | 0.76 | 0.69 | 461 |
| 1 | 0.75 | 0.62 | 0.68 | 538 |
| accuracy | | | 0.69 | 999 |
| macro avg | 0.69 | 0.69 | 0.69 | 999 |
| weighted avg | 0.70 | 0.69 | 0.69 | 999 |

This is the output for MLP Classifier

As we can see, even for such a small amount of data (5k train & 1k test), the code took about 20 minutes, & 5GB RAM to run. The majority of time & RAM is consumed by SVM & Count Vectorization respectively.

Some other interesting stats -

Initially ran the code for 20k training set & 5k testing set -

```

1 Terminal 2 Firefox 3 Editor 4 Nautilus 10 Spotify
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS
File Edit View Search Terminal Help
accuracy = 70.6941382776556 %
Time taken for naive bayes -> 2.1225857734680176
len(X_Train) = 9999, len(Y_Train) = 9999, len(X_Test) = 4999, len(Y_Test) = 4999
Making DF & saving them...
Applying count vectorization!
Time to make & save DFs, apply count vectorization -> 23.84649668933594

Now Decision Tree Classifier...
Accuracy by decision tree -> 71.65433886617324 %
Time taken for .fit & .predict -> 51.869420289993286

MLP Classifier now....
Accuracy by MLP Classifier -> 58.138026085201846 %
Time taken for .fit & .predict -> 59.83747434616809
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS l
assets/      contractions.json      Twitter_Sentiment_Analysis.py
BingLiuNegativeLexicons.txt sentiment_test_5k.csv
BingLiuPositiveLexicons.txt sentiment_train_10k.csv
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS rm -rf assets/
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS rm sentiment_train_10k.csv
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS cd ..
ritik@ritik-VN: ~/Desktop/NLP/SVM/1
assets/      contractions.json      sentiment_test_5k.csv      sentiment_train.csv
BingLiuNegativeLexicons.txt NEW/      sentiment_test.csv      svm_sentiment.py
BingLiuPositiveLexicons.txt output.txt      sentiment_train_10k.csv
ritik@ritik-VN: ~/Desktop/NLP/SVM/head -20000 sentiment_train.csv > sentiment_train_20k.csv
ritik@ritik-VN: ~/Desktop/NLP/SVM/mv sentiment_train_20k.csv NEW/
ritik@ritik-VN: ~/Desktop/NLP/SVM/cd NEW/
ritik@ritik-VN: ~/Desktop/NLP/SVM/1
BingLiuNegativeLexicons.txt contractions.json      sentiment_train_20k.csv
BingLiuPositiveLexicons.txt sentiment_test_5k.csv      Twitter_Sentiment_Analysis.py
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS via Twitter_Sentiment_Analysis.py
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS python Twitter_Sentiment_Analysis.py
Started the stopwatch...
Time to read contractions & lexicons -> 0.0825916890548338844
Time to import + read + save training data -> 1.1293282588850898
Vocabulary not found! building new one
Saving the vocabulary in current dir...
Length of vocabulary -> 25466
Time to build vocabulary -> 1.3268392886029853
Making X_Train & Y_Train from features....
Time to make X_Train & Y_Train -> 3.0165493488311768
Now testing part starts...
Time to import + read + save testing data -> 3.282160759972168

Result for Naive Bayes ->
Correct words = 3686
Total words = 4999
accuracy = 72.13442688537788 %
Time taken for naive bayes -> 3.6848573604692283
len(X_Train) = 19999, len(Y_Train) = 19999, len(X_Test) = 4999, len(Y_Test) = 4999
Making DF & saving them...
Applying count vectorization!

ritik@ritik-VN: ~
File Edit View Search Terminal Help
1 [ 0.7] 6 [ 0.0] 11 [ 11.3] 16 [ 0.0]
2 [ 0.0] 7 [ 0.0] 12 [ 0.0] 17 [ 0.0]
3 [ 0.0] 8 [ 0.0] 13 [ 0.0] 18 [ 0.0]
4 [ 0.0] 9 [ 0.0] 14 [ 0.0] 19 [ 0.0]
5 [ 0.0] 10 [ 0.0] 15 [ 0.0] 20 [ 0.0]
Heat [ 0.0] 19 [ 0.0] 27.46/49.11 Tasks: 77, 326 thr: 1 running
Sup [ 0.0] 13.06/58.80 Load average: 1.00 1.00 0.82
Uptime: 110 days(1), 01:04:54

PID USER PR NI VIRT RES SHR S CPU% MEM% TIME Command
26675 ritik 20 0 42.5G 27.0G 936 S 0 11.3 54.9 1:38.90 python3.7 Twitter_Sentiment_Analysis.py
26701 ritik 20 0 42.5G 27.0G 936 S 0 0.54.0 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26702 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26703 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26713 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26712 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.08 python3.7 Twitter_Sentiment_Analysis.py
26710 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.05 python3.7 Twitter_Sentiment_Analysis.py
26697 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26698 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26699 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26700 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26695 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.03 python3.7 Twitter_Sentiment_Analysis.py
26696 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26704 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26705 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26706 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26707 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26708 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26678 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26680 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26683 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.10 python3.7 Twitter_Sentiment_Analysis.py
26676 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26677 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26679 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.04 python3.7 Twitter_Sentiment_Analysis.py
26681 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.03 python3.7 Twitter_Sentiment_Analysis.py
26682 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.03 python3.7 Twitter_Sentiment_Analysis.py
26684 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26685 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26686 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26688 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26689 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.02 python3.7 Twitter_Sentiment_Analysis.py
26690 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26691 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26692 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26693 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
26694 ritik 20 0 42.5G 27.0G 936 S 0 0.54.9 0:00.01 python3.7 Twitter_Sentiment_Analysis.py
9571 ritik 20 0 2951M 65272 25976 S 0 0.1 18.48.18 /usr/lib/firefox/firefox -marionette -hea
9596 ritik 20 0 2951M 65272 25976 S 0 0.1 0:00.13 /usr/lib/firefox/firefox -marionette -hea
9597 ritik 20 0 2951M 65272 25976 S 0 0.1 1:37.07 /usr/lib/firefox/firefox -marionette -hea
9576 ritik 20 0 2951M 65272 25976 S 0 0.1 2:25.71 /usr/lib/firefox/firefox -marionette -hea
9791 ritik 20 0 2951M 65272 25976 S 0 0.1 0:17.09 /usr/lib/firefox/firefox -marionette -hea
File Edit View Search Terminal Help

```

As you can see, it took more than 40GB RAM, & was running endlessly, I had to force terminate it

Then I ran it for 10k training & 5k testing ->


```

ritik@ritik-VM:~/Desktop/NLP/SVM$ cat output.txt
Begin.06:30.675 NotebookApp]
Time to read contractions & lexicons -> 0.0026280879974365234
Time to import+read+save training data -> 0.6834373474121094
Vocabulary not found! building new one /jupyter/runtime/nbserver-27842-open.
Saving the vocabulary in current dir!
Length of vocabulary -> 160350 /? token=8f328abc2f3616786bbe5223010ebc396bfa9b
Time to build vocabulary -> 0.0078514313697814942f3616786bbe5223010ebc396bfa9b
Making X_Train & Y_Train from features. (127.0.0.1) 0.48ms
Time to make X_Train & Y_Train -> 0.6033999919891357ent_Sentiment_Analysis.ipynb is no
Now testing part starts!App] 404 GET /nbextensions/widgets/notebook/js/extension
Time to import+read+save testing data -> 0.1924849510192871er_Sentiment_A
[I 21:06:48.835 NotebookApp] Kernel started: 1095a610-f6e9-48e7-bb55-60ae89e
Using Naive bayes ->bookApp] Saving file at /Twitter_Sentiment_Analysis.ipyn
Correct words = 13539bookApp] Saving file at /Twitter_Sentiment_Analysis.ipyn
Total words = 4999
accuracy = 0.7079415883176635K%kernel interrupted: 1095a610-f6e9-48e7-bb55-60a
Time taken for naive bayes -> 2.315837860107422 for 1095a610-f6e9-48e7-bb55-
4999-9999-4999-9999
Making DF & saving them!App] Kernel restarted: 1095a610-f6e9-48e7-bb55-60ae8
Applying count vectorization!Restoring connection for 1095a610-f6e9-48e7-bb5
Time to make & save DFs, apply count vectorization -> 25.233197689056396
[I 21:26:40.825 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipyn
Now SVM model!App] Kernel interrupted: 1095a610-f6e9-48e7-bb55-60a
No saved model!... Creating one!arting buffering for 1095a610-f6e9-48e7-bb55-
Time to train & save model -> 5286.2617082595825
Accuracy by SVM -> 0.7203440688137628arted: 1095a610-f6e9-48e7-bb55-60ae8
Time taken to predict -> 47820.864698171616nection for 1095a610-f6e9-48e7-bb5
edd9bac27083a5f731f
Now Decision Tree Classifier.Replaying 3 buffered messages
Accuracy by decision tree -> 0.71874374874975Twitter_Sentiment_Analysis.ipyn
Time taken for .fit & .predict -> 7840.198331832886r_Sentiment_Analysis.ipyn
[I 21:30:49.777 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipyn
MLP Classifier now!App] Saving file at /Twitter_Sentiment_Analysis.ipyn
Accuracy by MLP Classifier -> 0.6983396679335867tter_Sentiment_Analysis.ipyn
Time taken for .fit & .predict -> 8034.783812999725r_Sentiment_Analysis.ipyn

```

As you can see, it took about 2 hours & 30 minutes to run the entire code. Also, it consumed upto 24GB of RAM

Another thing to note MLP classifier runs on multicore, unlike Decision Trees or SVM, as we can see in the screenshot below

The image displays two terminal windows. The top window shows a Jupyter Notebook session titled 'Twitter_Sentiment_Analysis.ipynb'. It includes a GET request to a local host, kernel startup logs, and file saving operations. The bottom window shows a system resource monitor with columns for PID, USER, CPU, MEM, and COMMAND. It lists various processes, including 'ipykernel_launcher' and 'mongod', along with their resource usage and commands.

```

1 Terminal 2 Firefox 3 Editor 4 Nautilus 10 Spotify
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS
File Edit View Search Terminal Help
[W 21:06:47.786 NotebookApp] Notebook Twitter_Sentiment_Analysis.ipynb is not trusted
[W 21:06:47.816 NotebookApp] 404 GET /nbextensions/widgets/notebook/js/extension.js?v=20281107210629 (127.0.0.1) 1.68ms referer=http://localhost:50808/notebooks/Twitter_Sentiment_Analysis.ipynb
[I 21:06:48.835 NotebookApp] Kernel started: 1095a610-f6e9-48e7-bb55-68ae89e30fdc, name: python3
[I 21:08:48.847 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb
[I 21:15:23.649 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb

[I 21:26:28.299 NotebookApp] Kernel interrupted: 1095a610-f6e9-48e7-bb55-68ae89e30fdc
[I 21:26:39.996 NotebookApp] Starting buffering for 1095a610-f6e9-48e7-bb55-68ae89e30fdc:97a3657046984cd
09bac27083a5f731f
[I 21:26:40.113 NotebookApp] Kernel restarted: 1095a610-f6e9-48e7-bb55-68ae89e30fdc
[I 21:26:45.247 NotebookApp] Restoring connection for 1095a610-f6e9-48e7-bb55-68ae89e30fdc:97a3657046984cd
cdd9bac27083a5f731f
[I 21:26:48.825 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb
[I 21:28:24.855 NotebookApp] Kernel interrupted: 1095a610-f6e9-48e7-bb55-68ae89e30fdc
[I 21:28:28.967 NotebookApp] Starting buffering for 1095a610-f6e9-48e7-bb55-68ae89e30fdc:97a3657046984cd
09bac27083a5f731f
[I 21:28:29.510 NotebookApp] Kernel restarted: 1095a610-f6e9-48e7-bb55-68ae89e30fdc
[I 21:28:29.584 NotebookApp] Restoring connection for 1095a610-f6e9-48e7-bb55-68ae89e30fdc:97a3657046984cd
cdd9bac27083a5f731f
[I 21:28:29.585 NotebookApp] Replaying 3 buffered messages
[I 21:28:48.802 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb
[I 21:28:49.938 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb
[I 21:30:49.777 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb
[I 21:44:49.813 NotebookApp] Saving file at /Twitter_Sentiment_Analysis.ipynb

ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS
File Edit View Search Terminal Help
12K assets/Y_Test_DF
96K assets/Y_Train_DF
40K assets/Y_Train_DF
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS du -sh assets/**
1021M assets/SVM_model
428K assets/TS_BB_Train
176K assets/vocabulary.pickle
344M assets/X_Test_DF
152K assets/X_Train_DF
1.7G assets/X_Train_DF
12K assets/Y_Test_DF
96K assets/Y_Train_DF
40K assets/Y_Train_DF
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS du -sh assets/**
1021M assets/SVM_model
428K assets/TS_BB_Train
176K assets/vocabulary.pickle
344M assets/X_Test_DF
152K assets/X_Train_DF
1.7G assets/X_Train_DF
12K assets/Y_Test_DF
96K assets/Y_Train_DF
40K assets/Y_Train_DF
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS du -sh assets/
3.1G assets/
ritik@ritik-VN: ~/Desktop/NLP/SVM/NEWS

```

| PID | USER | CPU | MEM | COMMAND | | | | | |
|-------|--------|------|------|---|------|------|------|------|-------|
| 28170 | ritik | 20 | 0 | 7937M 5557M 47272 R 98.7 11.0 23.53.22 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28204 | ritik | 20 | 0 | 7937M 5557M 47272 R 99.1 11.0 0:11.11 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28209 | ritik | 20 | 0 | 7937M 5557M 47272 R 99.1 11.0 0:09.99 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28208 | ritik | 20 | 0 | 7937M 5557M 47272 R 99.1 11.0 0:10.14 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28210 | ritik | 20 | 0 | 7937M 5557M 47272 R 81.4 11.0 0:09.83 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28212 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.9 11.0 0:09.76 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28199 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.9 11.0 0:10.44 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28208 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:10.63 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28198 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:09.61 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28201 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:09.81 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28202 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:10.67 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28197 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:09.76 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28203 | ritik | 20 | 0 | 7937M 5557M 47272 R 74.2 11.0 0:10.04 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28206 | ritik | 20 | 0 | 7937M 5557M 47272 R 73.5 11.0 0:10.31 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28196 | ritik | 20 | 0 | 7937M 5557M 47272 R 73.5 11.0 0:09.62 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28207 | ritik | 20 | 0 | 7937M 5557M 47272 R 73.5 11.0 0:09.61 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28205 | ritik | 20 | 0 | 7937M 5557M 47272 R 73.5 11.0 0:09.68 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28195 | ritik | 20 | 0 | 7937M 5557M 47272 R 66.3 11.0 0:09.68 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28194 | ritik | 20 | 0 | 7937M 5557M 47272 R 65.7 11.0 0:09.45 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28211 | ritik | 20 | 0 | 7937M 5557M 47272 R 65.0 11.0 0:09.27 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28229 | ritik | 20 | 0 | 7937M 5557M 47272 R 38.2 11.0 0:03.42 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28228 | ritik | 20 | 0 | 7937M 5557M 47272 R 29.5 11.0 0:03.37 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28222 | ritik | 20 | 0 | 7937M 5557M 47272 R 26.3 11.0 0:03.14 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28225 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.6 11.0 0:03.36 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28231 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.6 11.0 0:02.72 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28213 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.6 11.0 0:03.24 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28224 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.6 11.0 0:03.42 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28215 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.6 11.0 0:03.25 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28227 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.0 11.0 0:03.11 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28226 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.0 11.0 0:03.13 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28223 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.0 11.0 0:03.11 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28230 | ritik | 20 | 0 | 7937M 5557M 47272 R 25.0 11.0 0:02.83 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28217 | ritik | 20 | 0 | 7937M 5557M 47272 R 24.3 11.0 0:03.11 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28221 | ritik | 20 | 0 | 7937M 5557M 47272 R 17.7 11.0 0:03.06 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28218 | ritik | 20 | 0 | 7937M 5557M 47272 R 17.7 11.0 0:03.06 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28216 | ritik | 20 | 0 | 7937M 5557M 47272 R 17.7 11.0 0:02.88 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28219 | ritik | 20 | 0 | 7937M 5557M 47272 R 17.7 11.0 0:02.67 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28220 | ritik | 20 | 0 | 7937M 5557M 47272 R 16.4 11.0 0:02.81 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28214 | ritik | 20 | 0 | 7937M 5557M 47272 R 16.4 11.0 0:02.97 /usr/bin/python3.7 -m ipykernel_launcher | | | | | |
| 28266 | ritik | 20 | 0 | 26516 4388 140 R 0.7 0.0 0:02.75 http | | | | | |
| 28769 | mongod | 20 | 0 | 1542M 34272 7420 S 0.0 0.1 24:53.22 /usr/bin/mongod --config /etc/mongod.conf | | | | | |
| 28786 | mongod | 20 | 0 | 1542M 34272 7420 S 0.0 0.1 19:44.35 /usr/bin/mongod --config /etc/mongod.conf | | | | | |
| 28779 | mongod | 20 | 0 | 1542M 34272 7420 S 0.0 0.1 1:52.92 /usr/bin/mongod --config /etc/mongod.conf | | | | | |
| 28772 | mongod | 20 | 0 | 1542M 34272 7420 S 0.0 0.1 0:30.78 /usr/bin/mongod --config /etc/mongod.conf | | | | | |
| F100 | F200 | F300 | F400 | F500 | F600 | F700 | F800 | F900 | F1000 |

3b : Emotion Intensity Prediction

OUTPUT 2

The below screenshots are from Emotion_Intensity_JOY.ipynb

SVM

```
In [10]: ## SVM
print("\nResult for SVM ->")
model_prediction(SVR(), X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by SVM model ->",time() - start_time)
```

Result for SVM ->
 Results of sklearn.metrics:
 MAE: 0.14406197351666597
 MSE: 0.0323679201410181
 RMSE: 0.17991086721212285
 R-Squared: 0.3135262200437178

pearson corr. , p valve = (0.5630474325682381, 5.9210556367657214e-61)
 SpearmanrResult(correlation=0.5713068409028125, pvalue=4.2709173140663443e-63)
 Time taken by SVM model -> 12.120771408081055

This is the output for JOY by SVM

Decision Tree Classifier

```
In [11]: ## Using Decision Tree
print("\nResult Decision Tree ->")
model_prediction(DecisionTreeRegressor(max_depth = 5), X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by Decision Tree ->",time() - start_time)
```

Result Decision Tree ->
 Results of sklearn.metrics:
 MAE: 0.1538318236993331
 MSE: 0.037988457554869993
 RMSE: 0.19490627890057824
 R-Squared: 0.19432327011481154

pearson corr. , p valve = (0.4784480070703477, 4.00490638315593e-42)
 SpearmanrResult(correlation=0.4766555956767596, pvalue=8.852764699823259e-42)
 Time taken by Decision Tree -> 12.420202016830444

This is the output for JOY by Decision Tree Classifier

MLP Regressor

```
In [12]: ## using MLP
print("\nResult for MLP ->")
clf = MLPRegressor(solver='lbfgs', alpha=1e-5, hidden_layer_sizes=(5, 2), random_state=1, max_iter=1000)
model_prediction(clf, X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by MLP ->", time() - start_time)
```

Result for MLP ->
 Results of sklearn.metrics:
 MAE: 0.18284210257614578
 MSE: 0.047417972355903115
 RMSE: 0.21775668154135505
 R-Squared: -0.005662229120769302

pearson corr. , p valve = (nan, nan)
 SpearmanrResult(correlation=nan, pvalue=nan)
 Time taken by MLP -> 13.100698947906494

This is the output for JOY by MLP Classifier

As you can see, the code takes no more than 13 seconds to run, so no need to save the models

OUTPUT 3

The below screenshots are from Emotion_Intensity_ANGER.ipynb

SVM

```
In [10]: ## SVM
print("\nResult for SVM ->")
model_prediction(SVR(), X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by SVM model ->", time() - start_time)
```

Result for SVM ->
 Results of sklearn.metrics:
 MAE: 0.11319175362244595
 MSE: 0.019879849385711378
 RMSE: 0.14099591974845008
 R-Squared: 0.3262405365895137

pearson corr. , p valve = (0.5790818107120931, 2.9132508996171585e-69)
 SpearmanrResult(correlation=0.5689982163886088, pvalue=2.06129836924723e-66)
 Time taken by SVM model -> 11.63190221786499

This is the output for ANGER by SVM

Decision Tree Classifier

```
In [11]: ## Using Decision Tree
print("\nResult Decision Tree ->")
model_prediction(DecisionTreeRegressor(max_depth = 5), X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by Decision Tree ->",time() - start_time)
```

Result Decision Tree ->
Results of sklearn.metrics:
MAE: 0.1154059323323111
MSE: 0.020980108851885355
RMSE: 0.14484512022117058
R-Squared: 0.2889510072194118

pearson corr. , p valve = (0.547594469013182, 1.1151218338857375e-60)
SpearmanrResult(correlation=0.5507191215289514, pvalue=1.7222153245789713e-61)
Time taken by Decision Tree -> 11.959733486175537

This is the output for ANGER by Decision Tree Classifier

MLP Regressor

```
In [12]: ## using MLP
print("\nResult for MLP ->")
clf = MLPRegressor(solver='lbfgs', alpha=1e-5,hidden_layer_sizes=(5, 2), random_state=1,max_iter=1000)
model_prediction(clf, X_Train_DF, Y_Train, X_Test_DF, Y_Test)
print("Time taken by MLP ->",time() - start_time)
```

Result for MLP ->
Results of sklearn.metrics:
MAE: 0.11389595058964681
MSE: 0.020303883056515853
RMSE: 0.14249169469311485
R-Squared: 0.311869366417845

pearson corr. , p valve = (0.5831974578708614, 1.872611882396523e-70)
SpearmanrResult(correlation=0.5732742289854856, pvalue=1.3114177982316639e-67)
Time taken by MLP -> 18.60454535484314

In []:

This is the output for ANGER by MLP Classifier