**SENTIMENT CLASSIFICATION TASK**

This task implementation needs implementation of 2 papers which were shared

Out of which 1) Sentiment Classification using Machine Learning

2) Improvements of accuracy by removing objective sentences and retaining only Subjective sentences

**First Paper :**

It was implemented successfully by considering all the nitty-gritties presented in the paper.

But accuracies we got were quite more than that of paper even though we consider 3-fold cross validation.

Here is the list of accuracies we got with the tabulated features ..

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **freq/presence** | **Naive bayes** | **Logistic Regression** | **SVM** |
| Unigrams | frequency | 83.6 | 83.8 | 82.0 |
| Unigrams | presence | 83.6 | 83.0 | **85.6** |
| Unigrams+Bigrams | presence | 84.5 | 85.6 | **85.8** |
| Bigrams | presence | 83.3 | 82.4 | 82.3 |
| Unigrams+POS tagging | presence | 81.2 | 84.5 | 83.0 |
| Top 2633 unigrams | presence | 80.8 | 81.7 | 79.6 |

We have considered 1000 negative + 1000 positive files that were shared.

The accuracies we presented are the genuine results we got.

The accuracies we got are quite more than that of paper… We thought the reason might be increased data ( because they have considered only 700+700 reviews )

|  |
| --- |
| Instructions to Run our code :  1 : Run this code in the same folder we have attached  2: >>>python paper1.py |

Note : As training data is more, it can consume upto **3 min** to complete execution.

**Second Paper :**

It was improvement of basic sentiment classifier, by elimimnating objective sentences by building Subjectivity Detectors.

Each review was considered, a graph was drawn using “**networkx**” framework and partitioned it using min\_cut, by following the same procedure mentioned in paper.

Sentence level dataset to build subjective classifier was downloaded from

[www.cs.cornell.edu/people/pabo/movie-review-data/](http://www.cs.cornell.edu/people/pabo/movie-review-data/)

Here is the list of accuracies we got, ( **after extracting subjective sentences** ) with the tabulated features ..

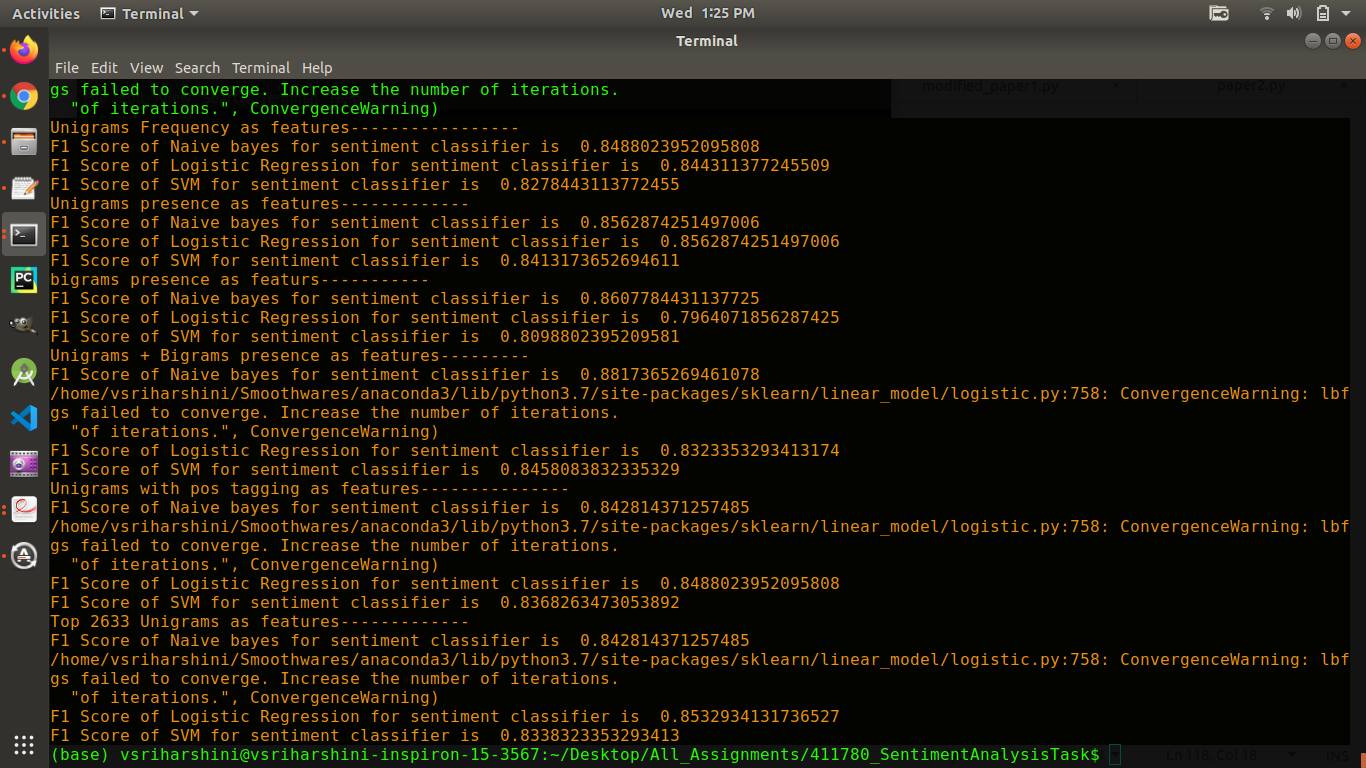
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **freq/presence** | **Naive bayes** | **Logistic Regression** | **SVM** |
| Unigrams | frequency | 84.8 | 84.4 | 82.7 |
| Unigrams | presence | 85.6 | 85.6 | 84.1 |
| Unigrams+Bigrams | presence | **88.17** | 83.23 | 84.4 |
| Bigrams | presence | 86.0 | 79.6 | 80.6 |
| Unigrams+POS tagging | presence | 84.4 | 84.7 | 83.8 |
| Top 2633 unigrams | presence | 84.28 | 85.4 | 83.38 |

So finally, we were able to acheive the accuracy of 88.17% for Sentiment Analysis of given dataset. ( Under 3 – fold cross validation )

|  |
| --- |
| Instructions to Run this code :  1 : Run this code in the same folder we have attached  2: >>>python paper2.py  3: Executing above step automatically creates 2 modified folders named ‘new\_neg’ , ‘new\_pos’ in your working directory. Now run the paper1 on these modified files.  4: >>>python improved\_paper1.py |

Note : It will take upto **1 min** , to complete the execution

The presented code is well commented and results produced are genuine.



Task Submitted by :

1) 411780 ( V. Sri Harshini ) 2) 411783 ( A. Sai )