COL-774 ASSIGNMENT-2

Name : Gattu Karthik
Entry No : 2019CS10348

Text Classification :

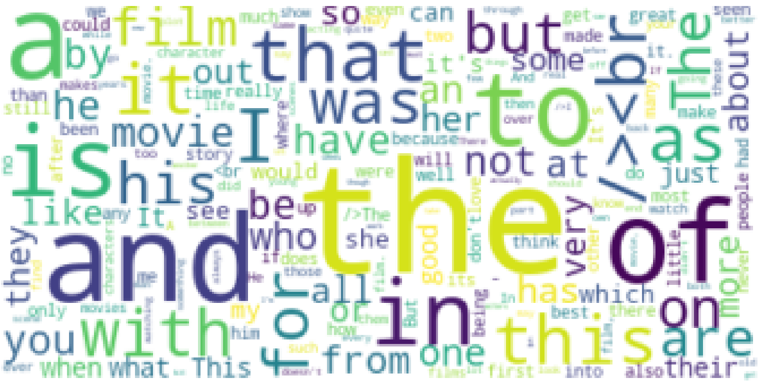
a_i :

Naive Bayes:

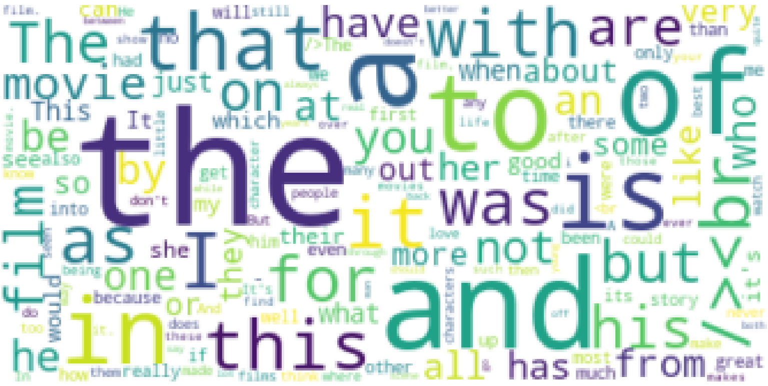
Accuracy over Training set: 95.04 %
Accuracy over Test set: 80.73%

a_ii :

Word cloud for positive class



Word cloud for negative class



b_i :

Random Prediction:

Accuracy over Test set: 50.07 %

b_ii :

Simple positive only Prediction:

Accuracy over Test set: 66.67%

b_iii :

The amount of improvement obtained in algorithm over these models are around 30%   and 14% respectively.

c_i :

[[tn,fn]
[fp,tp]]

confusion matrix (a_i)

```
[[4379 2269]
[ 621 7731]]
```

confusion matrix (b_i)

Random prediction

```
[[2543 4937]
 [2457 5063]]
```

Majority prediction

```
[[    0     0]
 [ 5000 10000]]
```

c_ii :

In all the three confusion matrices above, the positive category has the highest value of the diagonal entry. The more positive corrections may be due to the fact that the data has a more positive category.

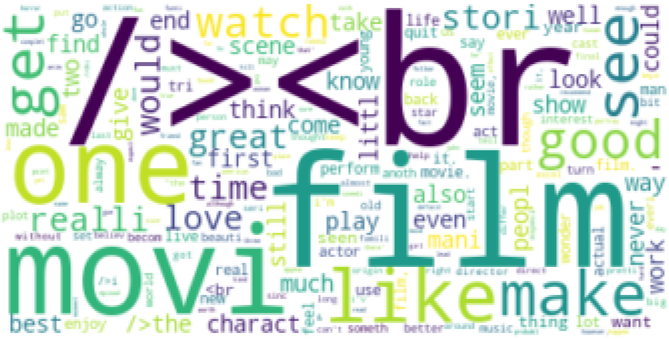
c_iii :

More number of positives are predicted correctly,because the data_set size of positives is large.Same behavior can be expected for negative cases also.

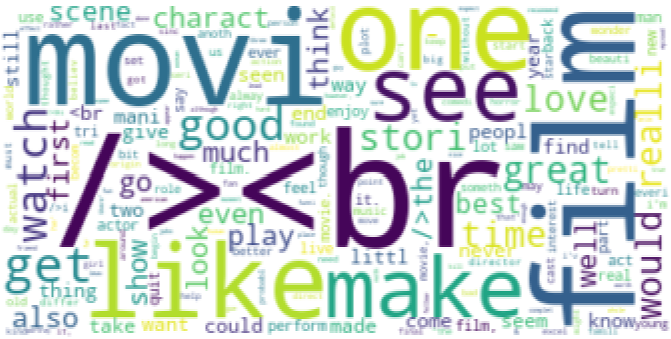
d_ii :

Removing Stop words, Stemming

Word cloud for modified_positive class



Word cloud for modified_negative class



d_iii :

Accuracy over Test set: 82.02 %

```
[[4409 2108]
 [ 591 7892]]
```

d_iv :

Accuracy over test set got increased form 80.73% to 82.02% after removing stopwords and stemming.This is because we have removed some unnecessary words which are trivial.

e_i :

Uni-grams ,stopwards removal ,stemming and Bi-grams :

Accuracy over Test set: 84.21 %

Confusion matrix

```
[[4528 1896]
[ 472 8104]]
```

Positive class

Precision = 0.941

Recall = 0.818

F1_score = 0.875

Negative class

Precision = 0.711

Recall = 0.899

F1_score = 0.794

e_ii :

New feature = Tri-grams

Uni-grams ,stopwards removal ,stemming and Tri-grams :

Accuracy over Test set: 82.4 %

Confusion matrix

```
[[4439 2069]
[ 561 7931]]
```

Positive class

Precision = 0.933

Recall = 0.793

F1_score = 0.857

Negative class

Precision = 0.682
Recall = 0.887
F1_score = 0.771

e_iii :

Overall accuracy dropped little by using tri-grams compared to bi-grams ,but increased compared to case-a.

f_i :

D-model is giving more accuracy

f_ii :

F1_score is a better metric here, because it takes into account both precision and recall and as the number of examples of each class are not equal then quantities involving only related to one category are not representative of results.

Binary Image Classification:

d = 8
So my classes of interest are 3 and 4

a) Using Linear Kernel:

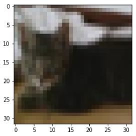
The number of support vectors I got in this case is: 2147
53.67% of the training examples constitute support vectors.

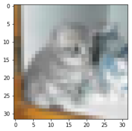
After calculating w, b and classifying the test data, accuracy of test data I got is 67.35%

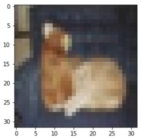
Intercept term is = -0.374

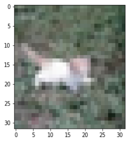
Training time = 66.05sec

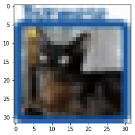
Plot for support vectors:



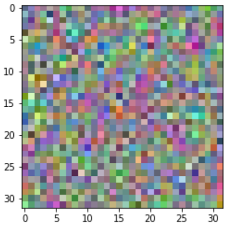








Plot for w:



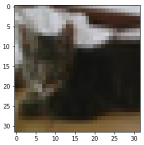
b) Using gaussian Kernel

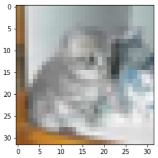
The number of support vectors I got in this case: 2636

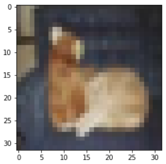
The number of support vectors matched with linear case: 1777

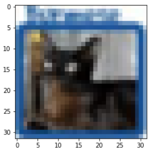
The test set accuracy after classification of the examples is 77.25%

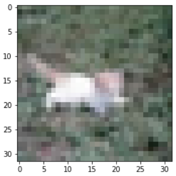
Plot for support vectors:











Test accuracy I got for the gaussian kernel is more than that of the linear case.

c)

No of support vectors for linear kernel: 2140 (almost same as 2147 in part a)

No of support vectors for gaussian kernel: 2631 (almost same as 2636 in part b)

Support vectors matched for linear kernel: 2140

Support vectors matched for gaussian kernel: 2631

The weight and bias obtained here (for linear kernel) are almost the same values as in the part a.

(Intercept here also is -0.388)

Test set accuracy for linear kernel is: 67.4% (>67.35%)

Test set accuracy for gaussian kernel is : 77.4% (>77.25%)

Time of implementation for linear kernel (cvxopt) is : 66.05 s

Time of implementation for gaussian kernel (cvxopt) is : 60.48 s

Time of implementation for linear kernel (sklearn) is: 38.5 s Time of implementation for gaussian kernel (sklearn) is: 26.3 s

The training time taken for linear case is more than the gaussian in each implementation. And for both gaussian and linear case cvxopt is taking more time than the sklearn.

Multi-Class Image Classification:

a)Using CVXOPT

For values of C=1 and gamma=0.001, after classifying test data

Test set accuracy = 59.04%

Training time: 679.86s

b) Using scikit-learn

For values of C=1 and gamma=0.001, after classifying test data,

Test set accuracy =59.3%

Training time: 222.28 s

The test set accuracy is almost the same value as in part a. While the training time considerably decreased using the scikit-learn function.

c)

Confusion matrix for part 3a:

```
[[738.  74.  84.  54.  50.]

 [118. 721.  42.  84.  35.]

 [150.  60. 412. 128. 250.]

 [ 96.  97. 123. 561. 123.]

 [ 99.  45. 217. 119. 520.]]
```

Confusion matrix for part 3b:

```
[[729.  83.  78.  61.  49.]

 [100. 731.  42.  92.  35.]

 [145.  61. 409. 133. 252.]

 [ 82.  97. 123. 572. 126.]

 [ 98.  48. 212. 118. 524.]
```

The values in both confusion matrices are almost the same.

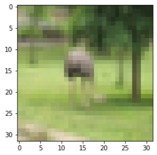
Note : In the confusion matrix, rows are for predicted values and columns are for true value

We could see the maximum values are in the diagonal which accounts to a good accuracy of the model. (Since the diagonal elements of matrices are the true positives)

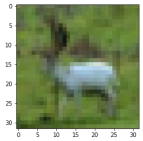
The top 10 misclassified objects are: (m-n denotes: predicted is m, true value is n)

4-2, 2-4, 0-2, 3-2, 4-3, 2-3, 3-4, 0-1, 3-1, 1-0

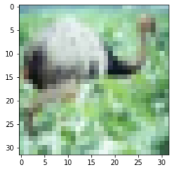
Predicted-4 & actual-2



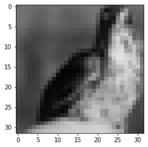
Predicted-2 & actual - 4



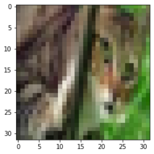
Predicted- 0 & actual - 2



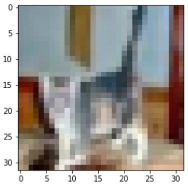
Predicted-3 & actual -2



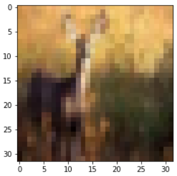
Predicted-4 & actual -3



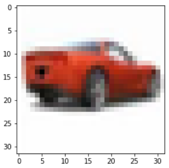
Predicted-2 & actual -3



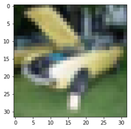
Predicted-3 & actual -4



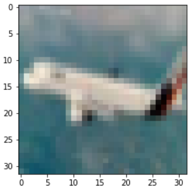
Predicted-0 & actual -1



Predicted-3 & actual -1



Predicted-1 & actual -0



Class-0 : airplane

Class-1 : cars

Class-2 : bird

Class-3 : cat

Class-4 : deer

Yes these results make sense,for eg in fig-8 the car is in white background and when seen unclearly it looks like an aeroplane in sky,hence it is justified.