

COL774 Assignment 2

Brian Sajeev Kattikat

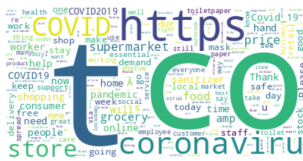
2021CS50609

October 2023

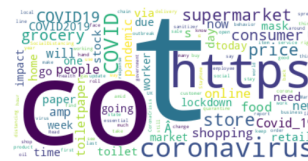
1 Naive

(a) Naive Bayes was implemented using Multinoulli model.

- i.
 - Accuracy in training set = 85.05%
 - Accuracy in validation set = 67.05%
- ii. Word clouds for each class:



Positive



Neutral



Negative

(b) For random guessing, accuracy obtained was 32.13 %, which is close to expected value of 33.33%. For always predicting positive, the accuracy is 43.85%, which would just simply show the percentage of positive tweets in the test data.

The Part a) of Naive Bayes gives 109% improvement over random guessing, and gives 52.91% gain over always predicting positive.

(c) • The confusion matrices for training set are as follows:

NaiveBayes			Random				
	AP	AN _u	AN		AP	AN _u	AN
PP	15711	2158	1078	PP	5582	2381	4702
PN _u	177	3574	171	PN _u	5464	2372	4727
PN	714	1364	12917	PN	5556	2343	4737

	All Positive		
	AP	AN _u	AN
PP	16602	7096	14166
PN _u	0	0	0
PN	0	0	0

- The confusion matrices for validation set are as follows:

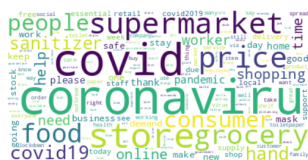
NaiveBayes				Random			
	AP	AN _u	AN		AP	AN _u	AN
PP	1197	338	305	PP	471	200	410
PN _u	21	101	17	PN _u	514	213	417
PN	226	178	910	PN	459	204	405

All Positive			
	AP	AN _u	AN
PP	1444	617	1232
PN _u	0	0	0
PN	0	0	0

We can show that the Naive Bayes classifier has much higher accuracy than random guessing or constant classification.

(d) The following transformations were done in order:

- Convert HTML references to unicode, i.e. convert "&" to &, "<" to etc.
- convert to lowercase.
- remove non-ascii characters.
- remove links and @tags.
- Tokenize to extract only alphanumeric and ' values.
- remove stopwords.
- Lemmatize.



Positive



Neutral



Negative

The output accuracy produced was:

- Naive Bayes prediction accuracy with stemming = 69.85%
- Naive Bayes prediction accuracy with lemmatizing = 70.91%

Observations:

- Accuracy increased because after cleaning there is lesser noise in the data.
- Lemmatizing seems to have better accuracy and performance over stemming.

(e) Bigrams has not been implemented:

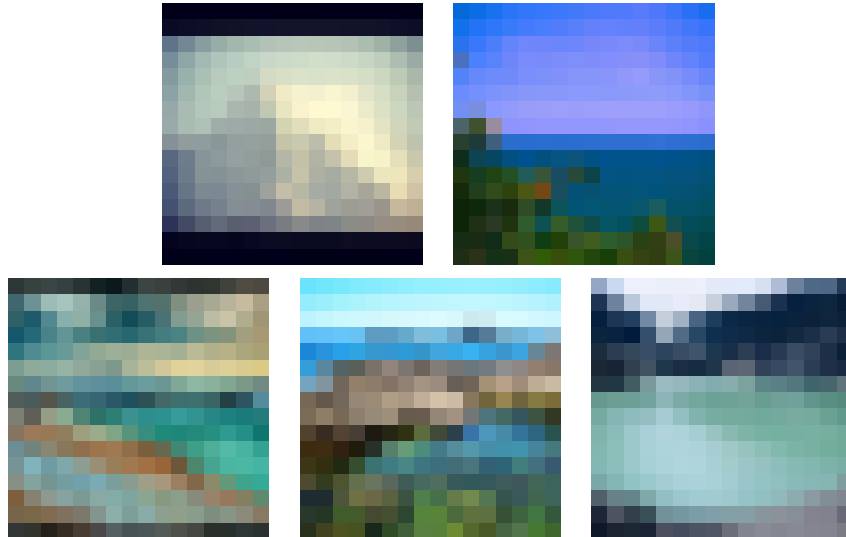
(f) Using Domain adaptation the following results were obtained:

- For size 1
- combined with corona tweets:
- Accuracy:0.43240556660039764
- Without corona tweets
- Accuracy:0.3465871438038436
- For size 2
- combined with corona tweets:
- Accuracy:0.43538767395626243
- Without corona tweets
- Accuracy:0.35155732273028495
- For size 5
- combined with corona tweets:
- Accuracy:0.4469847581179589
- Without corona tweets
- Accuracy:0.41583830351225975
- For size 10
- combined with corona tweets:
- Accuracy:0.46421471172962225
- Without corona tweets
- Accuracy:0.44532803180914515
- For size 25
- combined with corona tweets:
- Accuracy:0.48111332007952284
- Without corona tweets
- Accuracy:0.4486414844267727
- For size 50
- combined with corona tweets:
- Accuracy:0.4993373094764745
- Without corona tweets
- Accuracy:0.4854208084824387

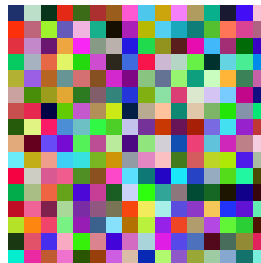
i. The obtained graph is as follows:

2 Binary SVM

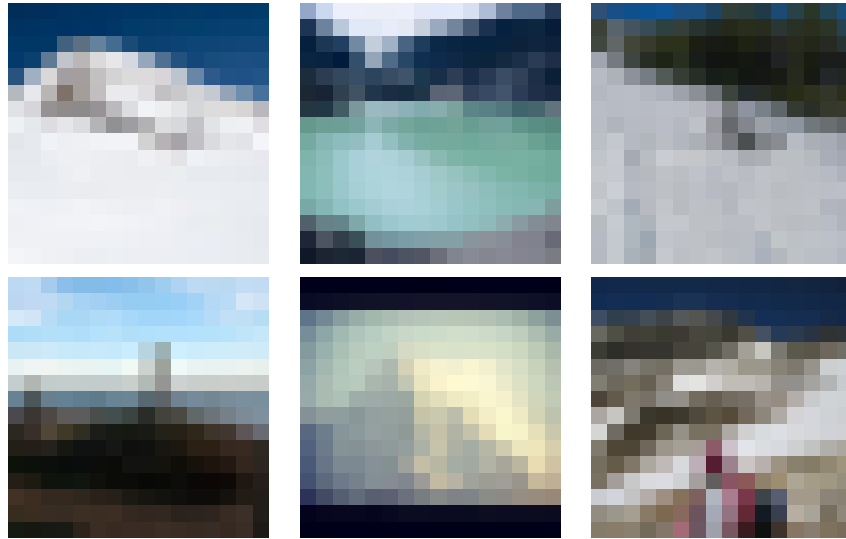
- (a) i. For linear kernel, datasets 3 and 4,
- No of support vectors: 3066 out of 4760
 - % of support vectors wrt training examples: 64.41%
- ii. The test accuracy we obtain is 71.00%
- iii. The top 6 support vectors are:



The weight image obtained is:



- (b) i. For gaussian kernel, datasets 3 and 4,
- No of support vectors: 3682 out of 4760
 - % of support vectors wrt training examples: 77.35%
 - No of common support vectors between linear and gaussian: 2895
- ii. The test accuracy we obtain is 76.25%
- iii. The top 6 support vectors are:



iv. We can observe that gaussian kernel can get slightly more accuracy than linear kernel.

(c) i. Number of Support Vectors obtained, using sklearn's SVC:

- Linear Kernel: 2942, and 2942 SVs in common with CVXOPT version.
- Gaussian Kernel: 3393, and 3393 SVs in common with CVXOPT version.
- between both linear and gaussian there are 2722 common SVs.

ii. Comparison of weight and bias in linear kernel:

- CVXOPT: $b = -0.7401936386523351$
- sklearn.svm: $b = -0.80421845$
- $\text{norm}(w_{cv} - w_{skl}) = 0.01106776175233239$

iii. Validation set accuracy is as follows:

- Linear Kernel: sklearn.svm obtains 71.50% accuracy over CVXOPT's 71.00% accuracy
- Gaussian Kernel: sklearn.svm obtains 76.75% accuracy over CVXOPT's 76.25% accuracy

iv. The training times are given below:

Scikit RBF	3.99s
Scikit linear	6.96s
CVXOPT RBF	51.67s
CVXOPT linear	61.28s