COL774 A2 report

cs1210107

September 2023

1 Naive Bayes Text Classification

1.1 Dataset

Training Dataset consists of 3 classes Positive(16602), Negative(14166),Neutral(7096) and a total of 37864 tweets. I've maintained a hashmap for storing the frequency of each occurring in each class.

1.2 1a

traning accuracy = 85.05% validation accuracy = 66.81%

1.3 Wordcloud

(a) Positive (b) Negative (c) Neutral

2 1b

if we assign randomly assign the class to each tweet then probability that it is assigned the correct class is

$$P(y = k|x) = 1/3$$

accuracy = 33.33% for both training and validation accuracy.
accuracy calculated = 32.78% which is consistent with our estimate(33%).

if we assign positive class to every tweet then
accuracy = P(y=Positive)=43.85%
accuracy calculated = 43.85% which same as our estimation

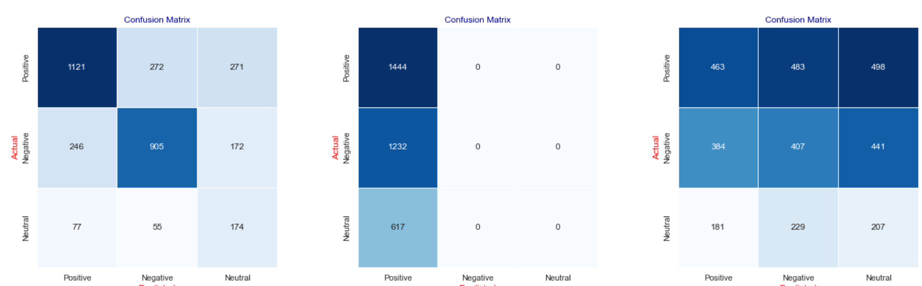
2.1 Observations

our naive bayes model gives an training accuracy and validation accuracy 85% and 66.81% respectively which is nearly double the accuracy given by random guessing and predicting positive .

2.2 Confusion Matrix

(a) naive bayes model (b) Positve all (c) random

Figure 2: Training set



(a) naive bayes model (b) Positive all (c) random

Figure 3: Validation set

for almost all cases positive class has highest diagonal entry because Positive is the dominant class in the dataset.

3 Stemming

After Stemming , training accuracy = 82.31% Validation accuracy = 70.21%
Observations:
After stemming Validation accuracy increases, training accuracy decreases.

(a) Positive

(b) Negative

(c) Neutral

4 Additional Features

4.1 Bigrams

training accuracy =94.75%
validation accuracy = 69.12%

4.2 Trigrams

training accuracy = 99.21%
validation accuracy = 64.96%

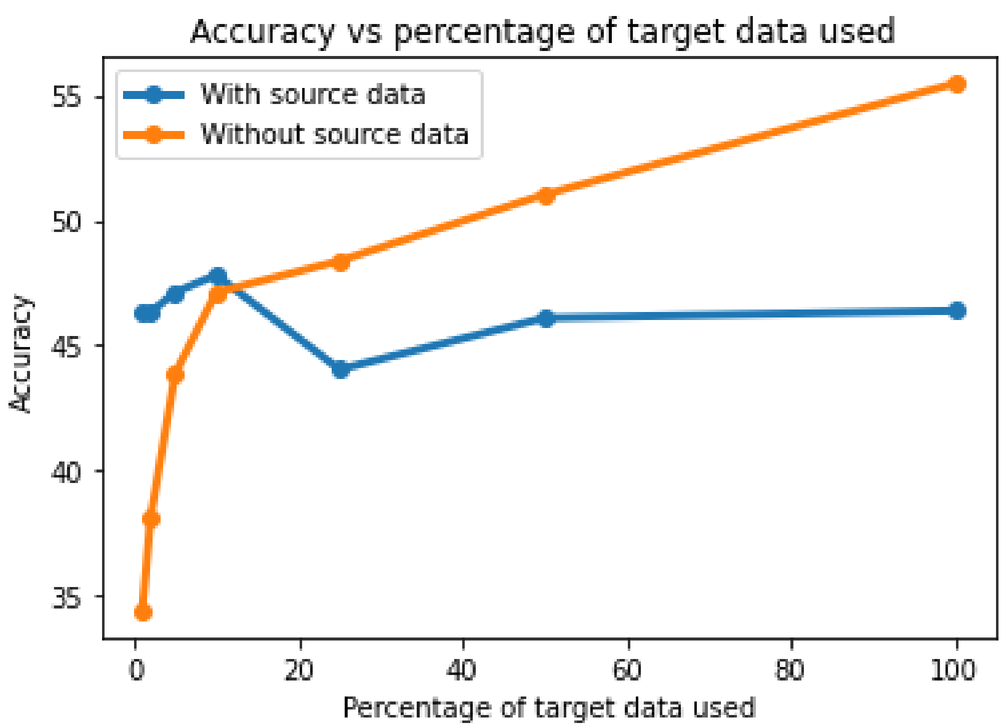
4.3 Observation

These bigram increases accuracy than that in part a($\geq 66.81\%$) but is less than in part d ($\leq 70.21\%$)
but trigram's accuracy is less than that in part a and part d.

5 Domain Adaptation

% Domain data used	With Source Domain	Without Source Domain
1	46.288%	34.393%
2	46.322%	38.104%
5	47.117%	43.870%
10	47.813%	47.117%
25	44.035%	48.376%
50	46.090%	51.027%
100	46.388%	55.467%

Table 1: Validation Set Accuracies



6 Image Classification

6.1 one v one classification

my entry no. is 2021CS10107 so i am choosing classes 7 mod 6 = 1 and 8 mod 6 = 2

Library	Kernel	Train Accuracy	Validation Accuracy	nSV
CVXOPT	Linear	97.37%	92%	670 (14.076%)
CVXOPT	Gaussian	94.96%	93.25%	1092 (22.94%)
LIBSVM	Linear	95.38%	94.0%	1038 (21.79%)
LIBSVM	Gaussian	95.13%	93.75%	1086 (22.84%)

Table 2: SVMs Performance Comparison

6.1.1 using CVXOPT

in this model, training time is high as nearly 2 minutes. using Gaussian Kernel as well as Linear Kernel

6.1.2 Linear Kernel

To train using the CVXOPT package, we need to first transform the dual problem into the form

$$\begin{aligned} &\alpha^T P\alpha + q^T\alpha + d \\ &G\alpha \preceq H \\ &A\alpha = b \end{aligned} \tag{1}$$

The dual in summation format is given as:

$$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y^i y^j (x^i)^T x^j - \sum_{i=1}^{m}\alpha_i \tag{2}$$

It is easy to see that P_{ij} is the coefficient of $\alpha_i\alpha_j$. Therefore, P_{ij} is given as:

$$\begin{aligned} &P_{ij} = y^i y^j (x^i)^T x^j \\ \implies &P = X_y \times X_y^T \\ &\text{where } X_y = \text{each row of } X \text{ multiplied by } Y \end{aligned} \tag{3}$$

Also, d is 0 and q is a vector with all -1:

$$\begin{pmatrix} -1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} \tag{4}$$

The condition on α_i is $0 \leq \alpha_i \leq c$. Therefore G and H are given as:

$$
G = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ -1 & 0 & \dots & 0 \\ 0 & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -1 \end{pmatrix}
$$

$$
H = \begin{pmatrix} c \\ c \\ \vdots \\ c \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{5}
$$

The equality condition is $\sum_{i=1}^{m} \alpha_i y^i = 0$. Therefore A and b are given as:

$$
\begin{aligned} A &= \begin{pmatrix} y_1 & y_2 & \dots & y_m \end{pmatrix} \\ b &= 0 \end{aligned} \tag{6}
$$

with linear kernel , b = -3.95075181

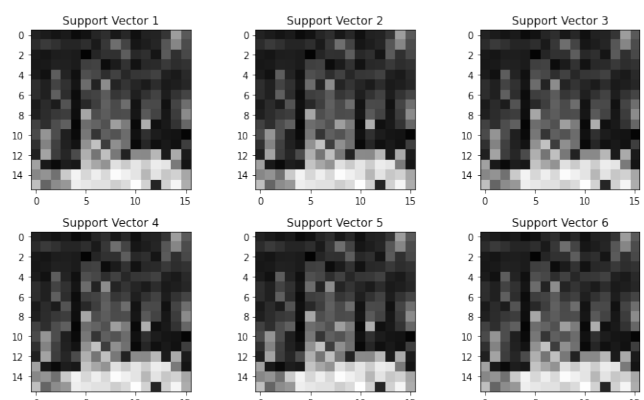


Figure 5: Top 6 support vectors using Linear Kernel

6.1.3 Gaussian Kernel

The dual SVM problem is given as:

$$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y^i y^j \phi(x^i)^T\phi(x^j) - \sum_{i=1}^{m}\alpha_i \tag{7}$$

The only difference will be in the value of P, and P is given as:

$$\begin{aligned} & P_{ij} = y^i y^j \phi(x^i)^T\phi(x)^j \\ \implies & P_{ij} = y^i y^j \exp(-\gamma||x^i - x^j||^2) \\ \implies & P_{ij} = y^i y^j \exp(-\gamma(||x^i||^2 + ||x^j||^2 - 2x^i x^j)) \end{aligned} \tag{8}$$

We generalise this equation for computing the *product* of any two vectors X, Z of sizes n, m respectively as:

$$\mathcal{P}(X,Z) = \begin{pmatrix} ||X_1||^2 & ||X_1||^2 & (m \text{ times})\dots & ||X_1||^2 \\ ||X_2||^2 & ||X_2||^2 & (m \text{ times})\dots & ||X_2||^2 \\ \vdots & \vdots & \ddots & \vdots \\ ||X_n||^2 & ||X_n||^2 & (m \text{ times})\dots & ||X_n||^2 \end{pmatrix} + \begin{pmatrix} ||Z_1||^2 & ||Z_2||^2 & \dots & ||Z_m||^2 \\ ||Z_1||^2 & ||Z_2||^2 & \dots & ||Z_m||^2 \\ \vdots & \vdots & & \vdots \\ n \text{ times} & n \text{ times} & \ddots & n \text{ times} \\ \vdots & \vdots & & \vdots \\ ||Z_1||^2 & ||Z_2||^2 & \dots & ||Z_m||^2 \end{pmatrix} - 2(X \otimes Z) \tag{9}$$

Here $\otimes$ is outer product. We can now compute P as:

$$P = (Y \otimes Y) \circ \exp(-\gamma\mathcal{P}(X,X)) \tag{10}$$

$\circ$ is Hadamard product. The values are then again passed to the CVXOPT quadratic problem solver. We make predictions as:

$$(\alpha \circ Y) \circ \exp(-\gamma\mathcal{P}(X_{SV}, X_{data})) + b \tag{11}$$

with gaussian kernel, b = -1.55535714

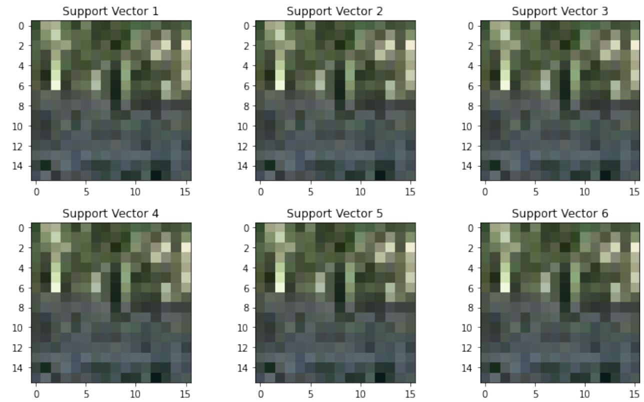


Figure 6: Top 6 support vectors using Gaussian Kernel

6.1.4 using LIBSVM

in this model, training time is low as 5 seconds. using Gaussian Kernel as well as Linear Kernel
with linear kernel , b = -2.7138097483069137
for linear kernel,

$$||W_{cvxopt} - W_{libsvm}|| = 404.4433202367066$$

with gaussian kernel, b = -2.597234343350003

using LIBSVM gives increases validation accuracies slightly and gives results faster.

6.2 resizing to 32x32x3

if images are resized to 32x32x3 then both training and test accuracies are same in linear case (95.37815126050421) and gaussian case (96.26050420168067)

7 Multi Class Image Classification

7.1 Using CVXOPT

in this model we are calculating 6C2 = 15 classifiers each taking 2-3 minutes to train and totally taking a training time of 40 minutes. validation accuracy = 53.91%

$$\begin{bmatrix} 76 & 15 & 22 & 29 & 21 & 37 \\ 10 & 145 & 1 & 5 & 7 & 32 \\ 12 & 2 & 121 & 29 & 23 & 13 \\ 31 & 5 & 24 & 124 & 9 & 7 \\ 31 & 12 & 55 & 39 & 52 & 11 \\ 23 & 22 & 10 & 10 & 6 & 129 \end{bmatrix}$$

7.2 using LIBSVM

this model gives faster results with training time 55 seconds only . it is comparatively 40 times faster than using CVXOPT validation accuracy = 55.91% which is slightly higher than the case above.

$$\begin{bmatrix} 92 & 15 & 16 & 20 & 22 & 35 \\ 10 & 148 & 1 & 3 & 11 & 27 \\ 14 & 2 & 133 & 25 & 18 & 8 \\ 16 & 8 & 23 & 129 & 21 & 3 \\ 23 & 16 & 46 & 24 & 87 & 4 \\ 24 & 14 & 10 & 6 & 5 & 141 \end{bmatrix}$$

7.3 Misclassified images

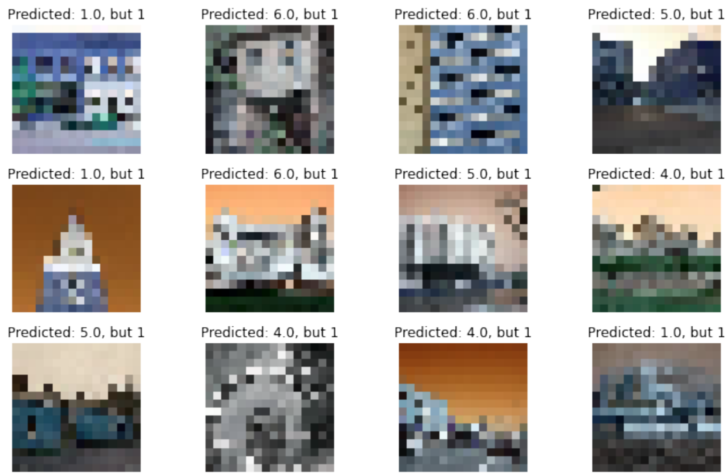


Figure 7: 12 Misclassified images

8 Kfold Cross Validation

Table 3: Validation and K-fold Accuracy

C Value	Validation Accuracy	K-fold Accuracy
1×10^{-5}	40.17	15.64
1×10^{-3}	40.17	16.64
1	55.92	49.71
5	59.25	58.58
10	60.83	63.87

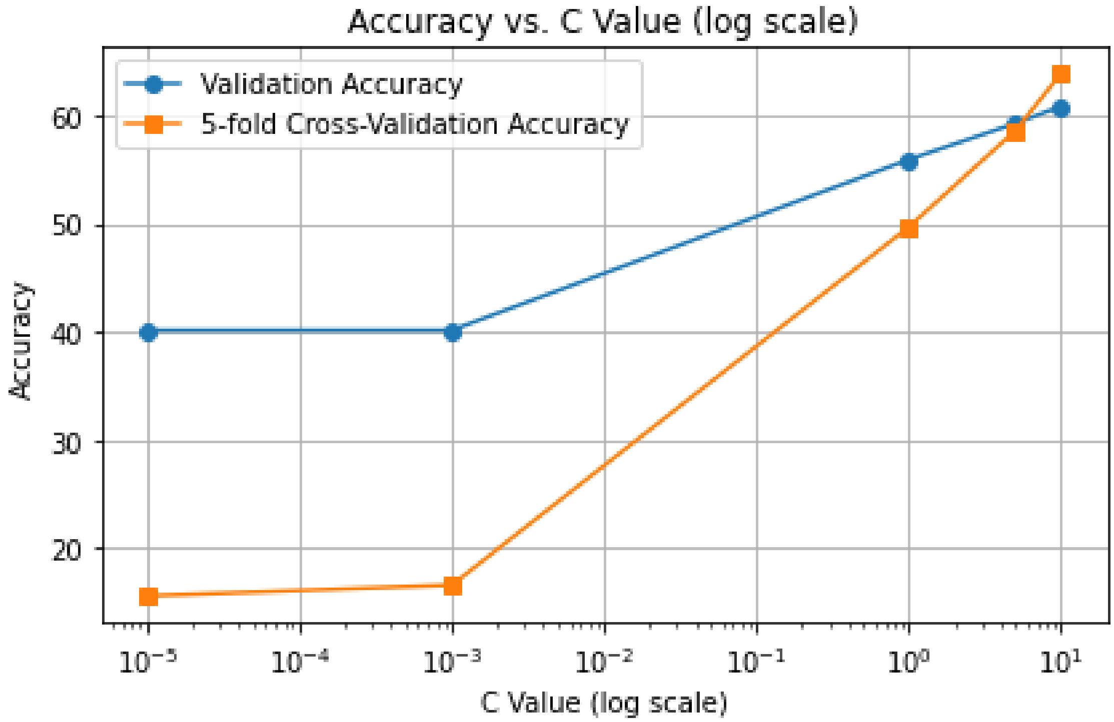


Figure 8: C vs accuracy

8.1 Observations

both K-fold accuracy and Validation Accuracy increases as C increases C = 10 gives best accuracy for 5-fold Cross Validation and validation accuracies.