

HATE SPEECH CLASSIFICATION

Preprocessing of Training Data:-

First carefully observed the Training Data set that we have given, It was not clean so I have considered the following factors,

- 1) There were hashtags and links which were needed to be replaced with Particular content.
- 2) There were emojis which have their own meaning and are important to consider because they show emotion. I have replaced them with their respective meanings in text.
- 3) There were CamelCase words which I split and converted all the letters in lowercase.
- 4) Rest removed digits etc.

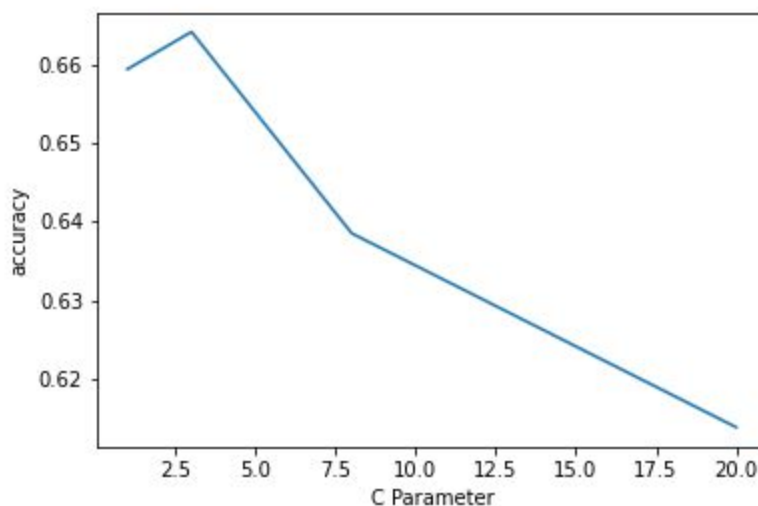
Choosing the different Models : -

Logistic Regression

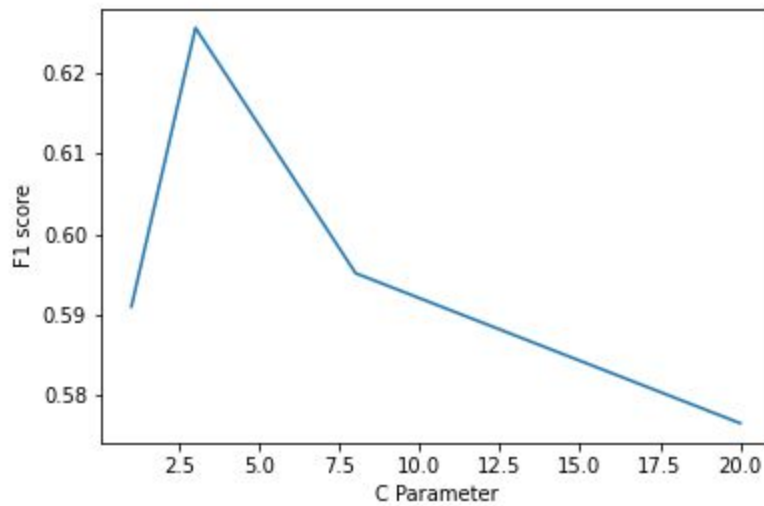
We have to classify the data into two categories 0/1 , so logistic regression is a well known classifier for binary classification.

HyperParameters used : Experimented with various values of parameters, and used GridSearchCV for tuning these parameters.

C vs Accuracy Score :



F1 score vs C:



So we can See from these observation, I am getting best F1 score and best Accuracy at the C between 2 and 3

Classification report using best C:

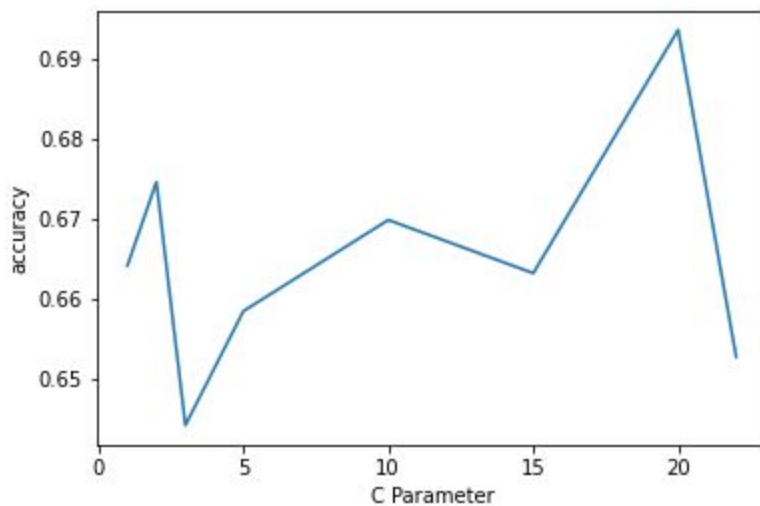
	precision	recall	f1-score	support
0	0.56	0.38	0.45	382
1	0.70	0.83	0.76	672
accuracy			0.67	1054
macro avg	0.63	0.61	0.61	1054
weighted avg	0.65	0.67	0.65	1054

SVM

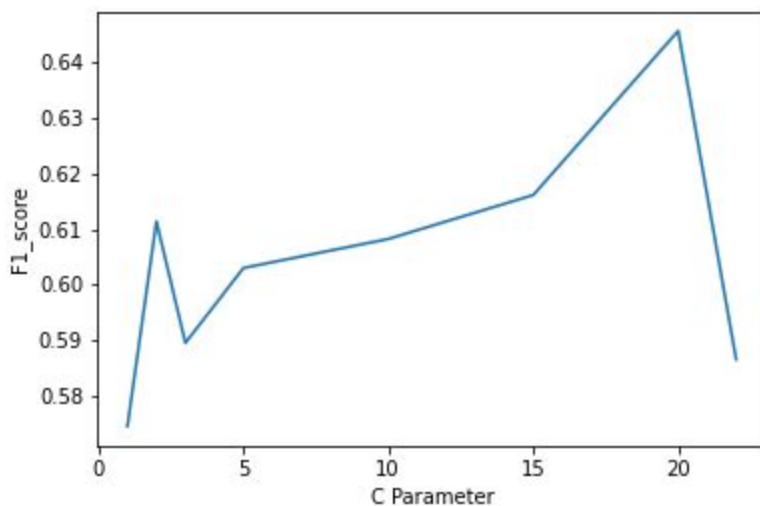
To find a better performance, I used SVM because it is also a Binary Classifier.

Observation during Setting Hyperparameters: Kernel - rbf

Accuracy score vs C: -



F1_score vs C :-



So as we can see that at $C = 20$ we are getting best accuracy and F1 score and using svm is giving us better performance than logistic regression here we are getting 69 percent accuracy where using logistic regression we were able to achieve only 65 percent.

Classification report using best C = 20 :-

```
[0 0 1 ... 0 0 0]
0.6935483870967742
0.6434755380457037

              precision    recall  f1-score   support

         0         0.64        0.43        0.51         395
         1         0.71        0.85        0.78         659


 accuracy          0.69         1054
 macro avg          0.67         1054
 weighted avg       0.68         1054
```

LSTM

Here along with LSTM I have used the embeddings of the glove.

Tried this classifier but wasn't giving the better performance along with it taking a long time to run.

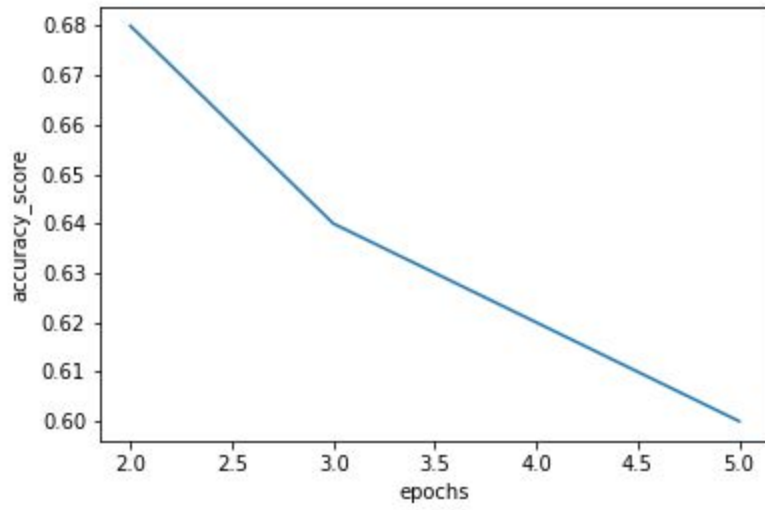
Accuracy_score : 63.34654

BERT WITH CNN

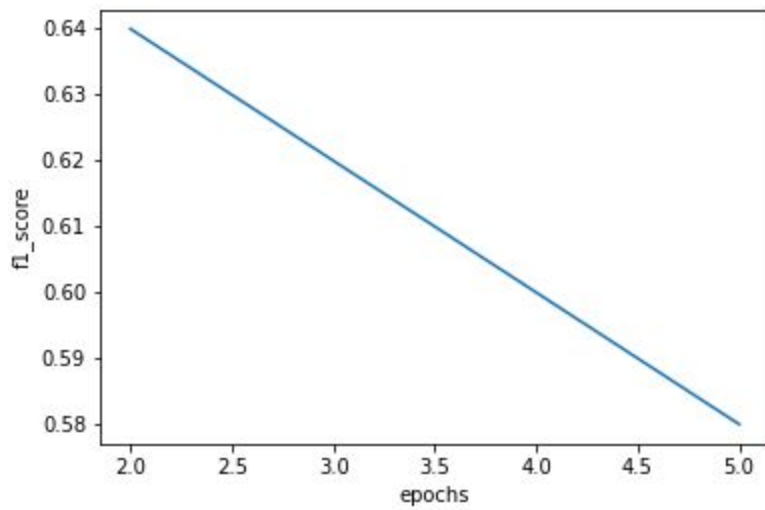
BERT is a language model that was created and published in 2018 by Jacob Devlin and Ming-Wei Chang from Google.

Tried with different number of epochs as well as different number of max_features used. This method generally overfits the data after 4 epochs so i avoided overfitting.

Epochs vs accuracy score: -



Epochs vs f1_score:-



Classification report:-

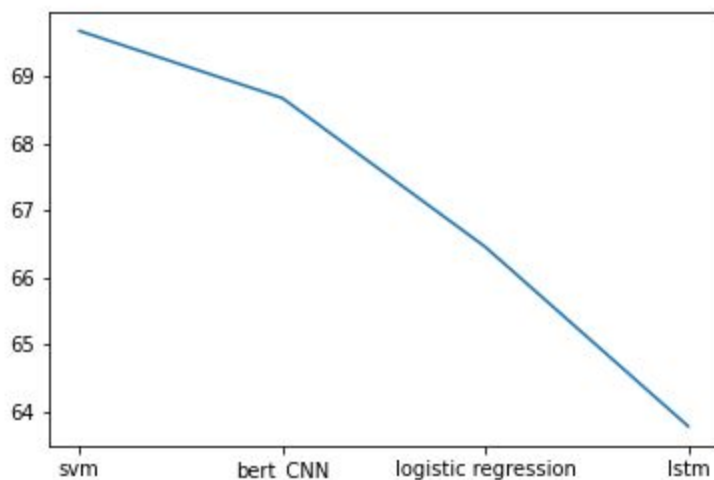
```
1 learner.validate(val_data=(x_test, y_test), class_names=[0,1])
```

	precision	recall	f1-score	support
0	0.58	0.52	0.55	398
1	0.73	0.78	0.75	656
accuracy			0.68	1054
macro avg	0.65	0.65	0.65	1054
weighted avg	0.67	0.68	0.67	1054


```
array([[205, 193],  
       [147, 509]])
```

Conclusion:-

We can say that there is only one classifier which is giving the best answer so far and that classifier is SVM with the accuracy of 69 percent and it is also classifying hate and Non- hate Speech in uniform manner. So this would be the best classifier.



How to run the svm code : - hardcoded the path of the train and test files just change that only.
