Assignment 2 Report

Shubham (2018CS10641)

October 10, 2021

NOTE: I was having some issue with shell file. So, just running "./run.sh" will run all the parts at once.

Text Classification

(a) Multinomial Naive Bayes

Train Accuracy = 0.47056
Test Accuracy = 0.6370714285714286

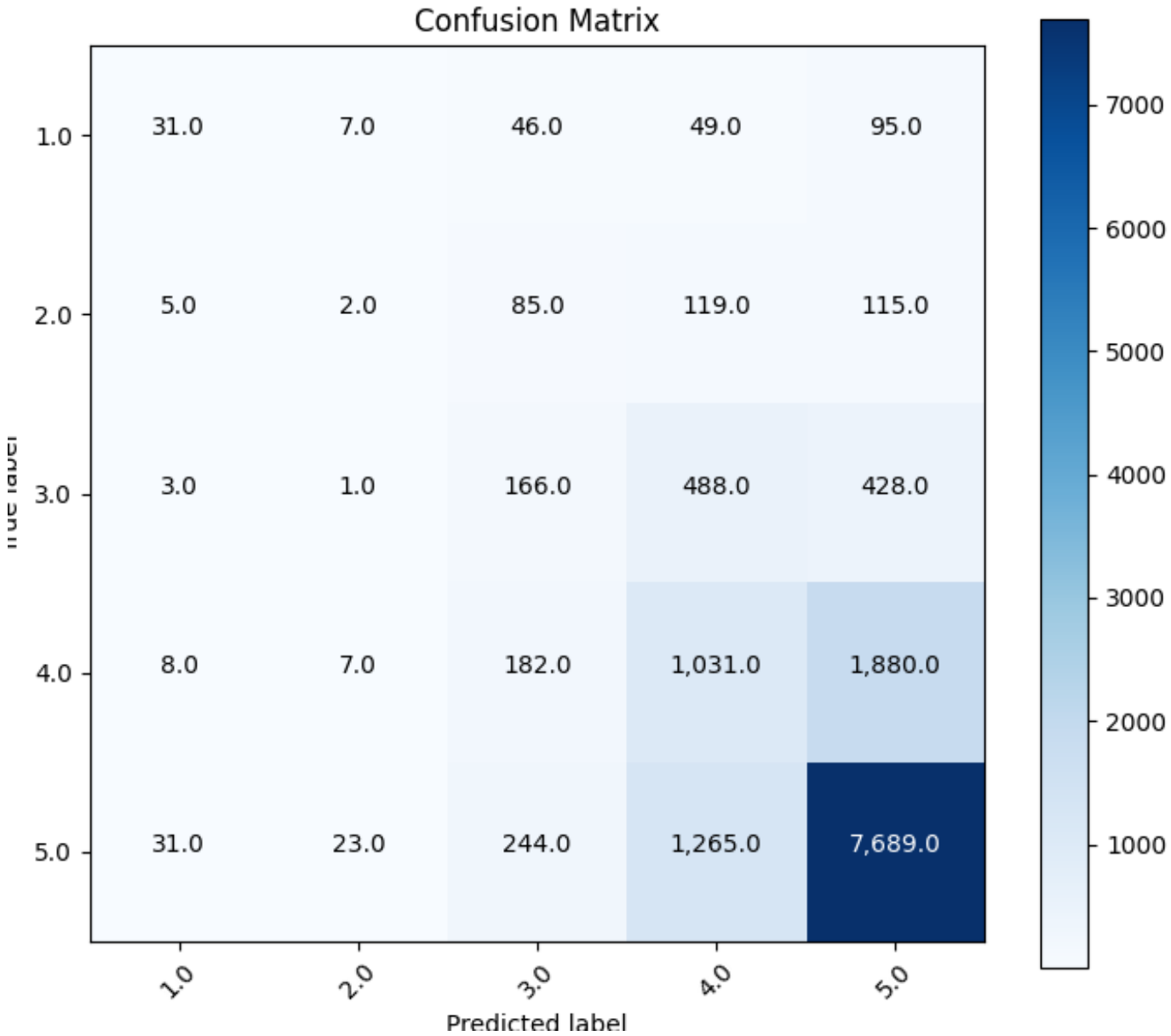
(b) Random Guessing

Test Accuracy = 0.1987857142857143
Improvement = -0.4382857142857143
So as expected, accuracy for random prediction is close to 0.2(as there are 5 target classes), which is way less than the accuracy by learned model on test data

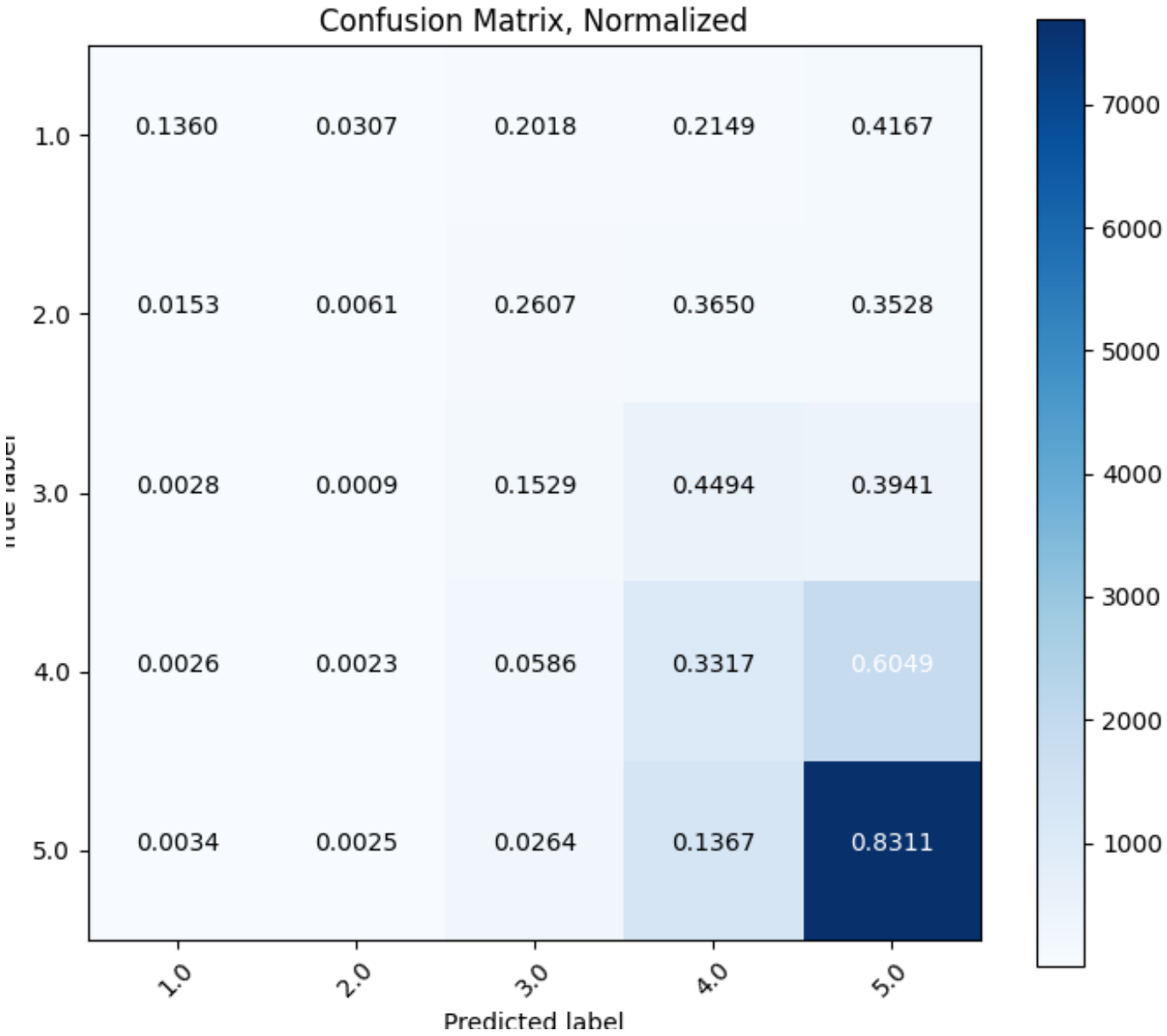
(b) Majority Class

Test Accuracy = 0.6608571428571428
Improvement = 0.0237857142857142

(c) Multinomial Naive Bayes - Confusion Matrix



As we can see:
1) The rating "5.0" has highest diagonal entry. Diagonal entries corresponds to the true positives (the sample correctly classified). It means that most of the correct classification were made for the rating "5.0"(the number of samples with "5.0" as the label was also high).
For drawing more details, let's look at the normalized confusion matrix:



- We can observe that most of the correct predictions were made for data with labels,"4.0" and "5.0". Then comes "3.0".- Least accurate predictions were made for the rating "2.0".
- The predictions accuracy for rating "5.0" was 83.1 percent, the highest. 33.17 percent of the"4.0" samples were correctly classified.
- 60 percent of the labels which were "4.0" were mis-classified to "5.0". A lot of texts predicted to have higher rating that it actually had, as we can see from the normalized matrix above.

(d) Multinomial Naive Bayes - With Stemming and Stopwords Removal

Train Accuracy = 0.42494
Test Accuracy = 0.6481428571428571
There is not much increase in accuracy as compared with the model with no stopword removal and stemming. For training data, accuracy got reduced slightly. This is because of class imbalance that we can observe in from the confusion matrix.

(e) Multinomial Naive Bayes - Feature Engineering

Feature 1: 1st few words are most popular most of the times in reviews. For ex, "I loved it", "Music was so soothing and my whole family loved it", "Excellent" etc. So, 1st 10 words are counted 10 times.
Test Accuracy = 0.635
No significant change in test accuracy.

Feature 2: The words like:
'not', 'never', 'bad', 'good', 'amazing', 'great', 'awesome', 'awful', 'worst', 'nice', 'loved', 'hate', 'clean', 'like', 'soothing', 'beautiful'
give a very fair idea about what the class could be. For ex., a review like, "The song was beautiful" should most likely be classified to a higher rating. Such words if present in the review are counted 5 times.
Test Accuracy = 0.6519285714285714
There is little improvement in test accuracy.

Feature Engineering Report

After trying many feature engineering techniques, there is no significant improvement over baseline Multinomial Naive Bayes model. Even with stopword removal and stemming, there is not much improvement over the baseline model. This is mainly due to the class imbalance in the training data. After adding more relevant features, there is a reduction in training accuracy. From this we can infer that our original baseline models were underfitting the training data as there was very high difference in train and test accuracies of baseline model.

(f) F1-score and macro-F1 score

F1-score for label 1.0: 0.06374501992031872
F1-score for label 2.0: 0.0
F1-score for label 3.0: 0.05098684210526316
F1-score for label 4.0: 0.27247138299868645
F1-score for label 5.0: 0.7959955932365761
Macro-F1 score: 0.2366397676521689
Class with highest diagonal entry = 5.0
From the low Micro F1 score, we can infer that there is class imbalance. Also from the Confusion Matrix we can observe that there are more predictions of class 5, this means that the model is influenced a lot by the class imbalance.So, Macro-F1 score seems more suited for this kind of dataset.

(g) Predictions with "Summary" Field

Accuracy on training data after Stemming and Stopwords removal with SUMMARY column: 0.54372
Accuracy on test data after Stemming and Stopwords removal with SUMMARY column: 0.64685714285714

MNIST Digit Classification

(1) Binary classification Part a

Time taken to train SVM model(CVXOPT) and Linear Kernel = 28.776sec
Linear Kernel Results for using CVXOPT...
The number of support vectors are = 158
The number of support vectors per class are = [66, 92]
The value of b obtained is = -1.221
The training accuracy of the model is = 100.000
The test accuracy of the model is = 99.031

(2) Binary classification Part b

Time taken to optimize SVM model (CVXOPT) and Gaussian Kernel = 35.389sec
Guassian Kernel Results for using CVXOPT...
The number of support vectors are = 845
The number of support vectors per class are = [173, 672]
The value of b obtained is = 0.881
The training accuracy of the model is = 99.975
The test accuracy of the model is = 99.585

(3) Binary classification Part c

————————LIBSVM on Linear Kernel————————-
optimization finished, iter = 4258
nu = 0.008999
obj = -18.274204, rho = 1.219513
nSV = 158, nBSV = 2
Total nSV = 158
Finished training Linear Kernel SVM in time: 0.8108329772949219 sec
Train Accuracy
Accuracy = 100 (4000/4000) (classification)
(100.0, 0.0, 1.0)
Test Accuracy
Accuracy = 99.0309 (2146/2167) (classification)
(99.03091832025842, 0.03876326718966313, 0.9616075346733688)

————————LIBSVM on Guassian Kernel————————-
optimization finished, iter = 384
nu = 0.114497
obj = -316.531025, rho = -2.043874
nSV = 477, nBSV = 435
Total nSV = 477
optimization finished, iter = 1288
nu = 0.040537
obj = -87.883228, rho = -0.890975
nSV = 848, nBSV = 40

Total nSV = 848
Finished training Guassian Kernel SVM in time: 3.7904200553894043 sec
Train Accuracy
Accuracy = 99.975 (3999/4000) (classification)
(99.97500000000001, 0.001, 0.999000499750125)
Test Accuracy
Accuracy = 99.5847 (2158/2167) (classification)
(99.58467928011075, 0.01661282879556991, 0.9834651532405393)