COMP 551 Mini Project 2: Classification of Textual Data

Dian Basit
260771254
dian.basit@mail.mcgill.ca

Hadi Zia
260775855
syed.h.rizvi@mail.mcgill.ca

Sagar Nandeshwar
260920948
sagar.nandeshwar@mail.mcgill.ca

***Abstract*—This project explores the performance of machine learning algorithms: naive Bayes(NB), Softmax regression (SR) and K-fold cross-validation. The Two datasets that have been used in training the models are; TwentyNewsGroup Dataset and Sentiment140 Dataset. The goal of this project was to explore linear classification and compare different features and models. We used k-fold validation to tune hyperparameter and compared the performance of naive Bayes and softmax regression on the two datasets using their best hyperparameters, and found that naive Bayes performs better overall.**

I. Introduction

In this project we implemented two machine learning models: naive Bayes and Softmax Regression (using Scikit Library) and evaluated the performance of these models. We performed hyper-parameter tuning and model selection using K-fold cross validation. Then, multiclass classification was conducted on the two datasets, and an evaluation between naive Bayes and softmax regression was compared. 5-fold cross validation was used to estimate performance in all of the experiments that were carried out.

A. Naive Bayes Classifier

Naive Bayes is a generative classifierer which assumes that the features are conditionally independent which reduces the number of parameters required. This main idea is tofind the probabilities of categories given a text document by using the joint probabilities of words and categories

B. Logistic Regression

Logistic regression is a classifier which uses a sigmoid function with a cross entropy cost to find a linear decision boundary.

II. Datasets

A. TwentyNewsGroup

The TwentyNewsGroup Dataset consists of close to 18000 newsgroups posts split on 20 different topics. It is divided into two subsets: one for training and the other one for testing. The split between the train and test set is based on whether a post was made before or after a specific date. Fig.1

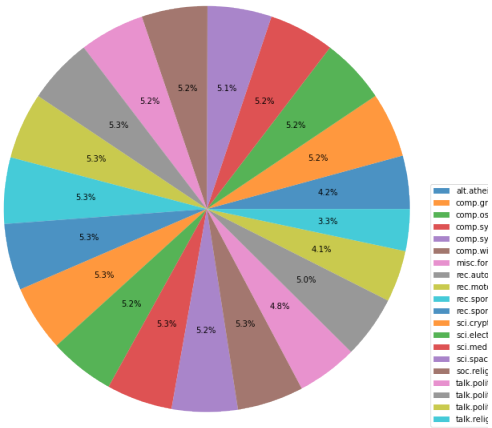


Fig. 1. TwentyNewsGroup

B. Sentiment140

The Sentiment140 dataset is a CSV that consists of various tweets with emoticons removed. There are 6 fields in the data; the polarity of the tweet (0 = negative, 2 = neutral, 4 = positive), the tweet id, the date the tweet was made, the query, the user that tweeted it, and the content of the tweet [2]. For our application, we only considered binary classification (0 = negative and 4 = positive) for the project.

III. Implementation

Naive Bayes, and K-fold cross-validation are implemented from scratch using Pandas and Numpy libraries. Softmax Regression was implemented using Logistic Regression class from scikit library. The implementation can be found in the attached jupyter notebook.

IV. Feature Engineering

A. Count Vectorizer

We start with the text data and convert text to feature vectors.

B. Cleaning dataset

1) Removing headers, footers and quotes: Use the default train subset (subset='train', and remove=(['headers', 'footers', 'quotes']) in sklearn.datasets) to train the models and report the final performance on the test subset

2) Pre-Processing: We removed html tags, punctuation, brackets, non-ASCII characters and digits from our data.

C. TF-IDF Vectorizer

This method uses word's frequency(TF) and the inverse document frequency(IDF). Each word isgiven a TF score and an IDF score and then the method weight of the word is the product of the scores. The term frequency is just the frequency of the word in the document. The IDF score is the log of the ratio of total number of documents and the number of documents which contain the word. Therefore, rarer words will have a higher score.

V. Results

Due to computing power constraints, for the TwentyNewsGroup dataset, we only considered 4 out of 20 classes whereas for Sentiment140, we considered 6% training samples out of a possible 1.6 million. For feature selection method for each datasets, we only went with countvectorizer. Again, due to time and resource constraints, the parameters we chose for each dataset min_df, max_df are as follows.

Parameters for datasets		
Dateset	min_df	max_df
TwentyNewsGroup	0.01	0.7
Sentiment140	0.005	0.8

TABLE I

Features were selected within these thresholds. Hyperparameter for norm was chosen as cost for Softmax regression. We used 5-fold cross-validation on both L1-norm and L2-norm, extracted the accuracies on the validation set and then chose the best one. L2-norm turns out to be the best norm with a percentage of 66.33% compared to L1's 66.31%.

```
Train Accuracy:  0.6656718749999999  Test Accuracy:  0.66309375
Train Accuracy:  0.6656953125  Test Accuracy:  0.66328125
Norm that yields highest accuracy on Sentiment140 test data is l2-norm with accuracy of 0.6633
```

Fig. 2. Train and test accuracies of L1 and L2 on sentiment140

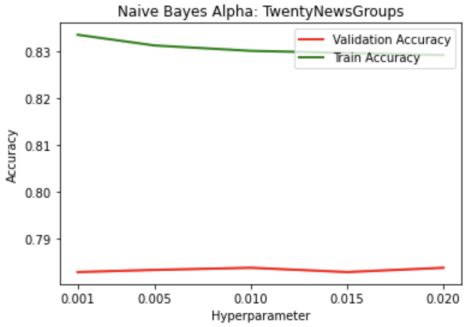
We then performed hyperparameter tuning. The hyperparameters selected for each of the dataset are shown below

Hyperparameters for datasets	
Dateset	Hyperparameters
TwentyNewsGroup	0.001, 0.005, 0.010, 0.015, 0.020
Sentiment140	0.01, 0.05, 0.1, 0.5, 1.00

TABLE II

We applied 5-fold validation on all hyperparameters for both models (NB and Softmax regression) on each dataset, and then selected the best hyperparameters for both models. The results are shown in the figures below.

```
Dataset:  TwentyNewsGroups
Train Accuracy:  0.8335997366783575  Test Accuracy:  0.7828627212226817
0.001  alpha completed
Train Accuracy:  0.8313093760470412  Test Accuracy:  0.783321436819012
0.005  alpha completed
Train Accuracy:  0.830164097379176  Test Accuracy:  0.7837791027229022
0.01  alpha completed
Train Accuracy:  0.8297060383665403  Test Accuracy:  0.7828616715302416
0.015  alpha completed
Train Accuracy:  0.8292479793539048  Test Accuracy:  0.7837780530304621
0.02  alpha completed
```

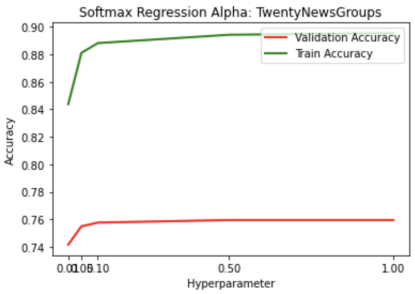



```
Best Hyperparameter for TwentyNewsGroups with Naive Bayes Model is: 0.010
```

Fig. 3. NB Alpha for TwentyNewsGroup

To summarize, for TwentyNewsGroup, Alpha for NB = 0.010 and Lambda (Regularization Coefficient) for SR = 0.5. For Sentiment140, Alpha for NB = 0.001 and lambda for SR = 1.00. These are the best hyperparameters for each dataset and model.

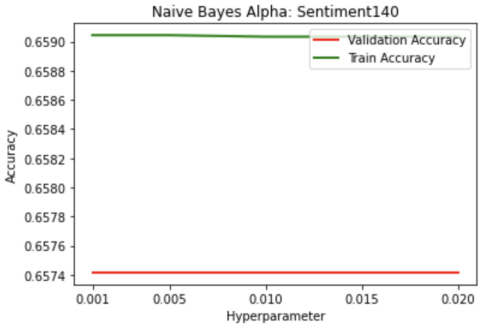
```
Train Accuracy:  0.8437926971519168  Test Accuracy:  0.7416255537127621
0.01  alpha completed
Train Accuracy:  0.8811267359984158  Test Accuracy:  0.7549094115424181
0.05  alpha completed
Train Accuracy:  0.8882270441030966  Test Accuracy:  0.7576596057355195
0.1  alpha completed
Train Accuracy:  0.8942970800541069  Test Accuracy:  0.7594934184284006
0.5  alpha completed
Train Accuracy:  0.8954423587219722  Test Accuracy:  0.7594934184284006
1.0  alpha completed
```




```
Best Hyperparameter for TwentyNewsGroups with Softmax Regression Model is: 0.500
```

Fig. 4. SR Alpha for TwentyNewsGroup

```
Dataset:  Sentiment140
Train Accuracy:  0.6590442708333334  Test Accuracy:  0.6574166666666665
0.001  alpha completed
Train Accuracy:  0.6590442708333334  Test Accuracy:  0.6574166666666665
0.005  alpha completed
Train Accuracy:  0.6590338541666667  Test Accuracy:  0.6574166666666665
0.01  alpha completed
Train Accuracy:  0.6590338541666667  Test Accuracy:  0.6574166666666665
0.015  alpha completed
Train Accuracy:  0.6590338541666667  Test Accuracy:  0.6574166666666665
0.02  alpha completed
```



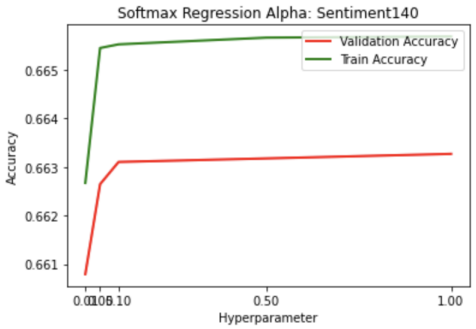

```
Best Hyperparameter for Sentiment140 with Naive Bayes Model is: 0.001
```

Fig. 5. NB Alpha for Sentiment140

We then applied the best hyperparameters for both models on test sets and selected the best model out of the two. The figure below shows the results.

In addition, to testing our test set with our best hyperparameters, we also varied the testing sample 20%, 40%, 60%, 80% using these percentages. We then trained the model and predicted the test results. The results for these are summarized below: We calculated the mean accuracy of both models under all fraction of testing sample and then selected the best performance model out of both for both datasets. The figure above shows that for TwentyNewsGroup and Sentiment140, Naive Bayes performed better with accuracy of 72.52% and 65.81%

```
Train Accuracy:  0.662671875  Test Accuracy:  0.6607916666666667
0.01  alpha completed
Train Accuracy:  0.665453125  Test Accuracy:  0.6626458333333333
0.05  alpha completed
Train Accuracy:  0.6655286458333334  Test Accuracy:  0.6631041666666666
0.1  alpha completed
Train Accuracy:  0.6656666666666666  Test Accuracy:  0.6631770833333334
0.5  alpha completed
Train Accuracy:  0.6656901041666667  Test Accuracy:  0.6632708333333334
1.0  alpha completed
```




```
Best Hyperparameter for Sentiment140 with Softmax Regression Model is: 1.000
```

Fig. 6. SR Alpha for Sentiment140

```
Dataset:  TwentyNewsGroups
Test Accuracy for TwentyNewsGroups Dataset using Naive Bayes Model is: 0.7385
Test Accuracy for TwentyNewsGroups Dataset using Softmax Regression Model is: 0.7233
Best Model for TwentyNewsGroups Dataset is Naive Bayes with accuracy of 0.7385

Dataset:  Sentiment140
Test Accuracy for Sentiment140 Dataset using Naive Bayes Model is: 0.6546
Test Accuracy for Sentiment140 Dataset using Softmax Regression Model is: 0.6045
Best Model for Sentiment140 Dataset is Naive Bayes with accuracy of 0.6546
```

Fig. 7. Best model for each dataset

respectively.

```
Dataset:  TwentyNewsGroups
20% of Training Data
Test Accuracy for TwentyNewsGroups Dataset using Naive Bayes Model is: 0.6979
Test Accuracy for TwentyNewsGroups Dataset using Softmax Regression Model is: 0.5953
Best Model for TwentyNewsGroups Dataset with 20% of training data is Naive Bayes with accuracy of 0.6979
40% of Training Data
Test Accuracy for TwentyNewsGroups Dataset using Naive Bayes Model is: 0.7309
Test Accuracy for TwentyNewsGroups Dataset using Softmax Regression Model is: 0.6621
Best Model for TwentyNewsGroups Dataset with 40% of training data is Naive Bayes with accuracy of 0.7309
60% of Training Data
Test Accuracy for TwentyNewsGroups Dataset using Naive Bayes Model is: 0.7378
Test Accuracy for TwentyNewsGroups Dataset using Softmax Regression Model is: 0.6903
Best Model for TwentyNewsGroups Dataset with 60% of training data is Naive Bayes with accuracy of 0.7378
80% of Training Data
Test Accuracy for TwentyNewsGroups Dataset using Naive Bayes Model is: 0.7343
Test Accuracy for TwentyNewsGroups Dataset using Softmax Regression Model is: 0.7096
Best Model for TwentyNewsGroups Dataset with 80% of training data is Naive Bayes with accuracy of 0.7343

Optimum Model for TwentyNewsGroups Dataset is Naive Bayes with mean accuracy of 0.7252
```

Fig. 8. TwentyNewsGroup Fractional Training Set Performance

```
Dataset:  Sentiment140
20% of Training Data
Test Accuracy for Sentiment140 Dataset using Naive Bayes Model is: 0.6657
Test Accuracy for Sentiment140 Dataset using Softmax Regression Model is: 0.6323
Best Model for Sentiment140 Dataset with 20% of training data is Naive Bayes with accuracy of 0.6657
40% of Training Data
Test Accuracy for Sentiment140 Dataset using Naive Bayes Model is: 0.6630
Test Accuracy for Sentiment140 Dataset using Softmax Regression Model is: 0.6072
Best Model for Sentiment140 Dataset with 40% of training data is Naive Bayes with accuracy of 0.6630
60% of Training Data
Test Accuracy for Sentiment140 Dataset using Naive Bayes Model is: 0.6574
Test Accuracy for Sentiment140 Dataset using Softmax Regression Model is: 0.5933
Best Model for Sentiment140 Dataset with 60% of training data is Naive Bayes with accuracy of 0.6574
80% of Training Data
Test Accuracy for Sentiment140 Dataset using Naive Bayes Model is: 0.6462
Test Accuracy for Sentiment140 Dataset using Softmax Regression Model is: 0.6100
Best Model for Sentiment140 Dataset with 80% of training data is Naive Bayes with accuracy of 0.6462

Optimum Model for Sentiment140 Dataset is Naive Bayes with mean accuracy of 0.6581
```

Fig. 9. Sentiment140 Fractional Training Set Performance

VI. Discussion and Conclusion

From the result, we can see that Naive Bayes performed well overall. However, a more accurate assessment and a better evaluation would have been made, if we would have tweaked the following parameters if we had not experienced limited computing resources:

- Compared efficient feature extraction using CountVectorizer, tf-idf matrix, CounterVectorizer bigram and chose the optimum feature extraction method.
- Used greater training model for Sentiment140 (6% taken) and more classes for TwentyNewsGroups (4 out of 20 were taken) datasets. However, with initial assessment, it was pointed out that if all classes for TwentyNewsGroups were taken then accuracy drops down to 50%. A more assessment was needed however, this consumes an enormous amount of time and running K-fold validation with large amount of classes or testing sample was inadequate.
- Confusion matrix for both models on dataset would have been plotted to understand the true/false positive by the model.
- More hyperparameters for softmax regression could have been explored like solver, number of epochs, fit intercept etc. We only norm, and regularization coefficient for this project.
- Increased the number of hyperparameters and number of K-folds for better hyperparameter tuning.

VII. Statement of Contribution

Dian - Implemented Softmax Regression. Tested and rectified any bugs in both Softmax Regression, and Naive Bayes model. Implemented hyperparameter tuning for both models and selected the best model.
Hadi - Primary duty for writing the report and summarise the findings.
Sagar - Implemented Multinomial Naive based. Assisted in data cleaning and implementing LR. Worked on writing report.

References

[1] ”7.2. Real world datasets”, scikit-learn, 2022. Available: https://scikit-learn.org/stable/datasets/real_world.html#newsgroups-dataset.

[2] ”For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool”, Help.sentiment140.com, 2022. Available: http://help.sentiment140.com/for-students.