Miguel Arabi

Text Classification for Yelp Reviews Dataset

March 21st, 2019

Introduction

Sentiment analysis uses computational tools to determine the emotional tone behind words. This classification task allows us to extract information in order to gain an understanding of the attitudes, opinions, and emotions recorded in text data.

At a high level, sentiment analysis involves Natural Language Processing and machine learning tasks by taking the text element, transforming it into a format that a machine can read, and using statistics to determine the actual sentiment by way of classification.
These techniques are applied to the reviews contained in the Yelp Academic Dataset to conduct sentiment classification.

Step 1: Obtaining and processing the data

The yelp_reviews.csv file is downloaded then separated into a positive and a negative file. Positive reviews are those with 4 or 5 stars in the reviews while negative reviews are those which received 1 star. Since the data contains over a million records combined, random samples of 20,000 each are generated to keep the tasks manageable. Below is an example of a negative reviews from the sample:

["*First time here... the place was empty, sat at the bar, had one drink and had to wait 25 minutes to for a s econd. The bartender was closing out his own shift and no one wanted to help us. I had to leave my seat a nd grab someone just to close out. I wasn't even able to order my food. Needless to say, I had such high h ope for this new local place but I won't be returning."]*

Step 2: NLTK Features

Bag of Words, Stopwords and punctuation filtering, bigram collocations, POS tagging are some of the methods used for feature set generation. Respective functions are written in python to extract features from the data.
Some words (e.g. no, not, more, most, below, over, too, very, etc.) have been removed from the standard stopwords available in NLTK; those words can have some sentiment impact in our review dataset.

NLTK Naïve Bayes (NB) classifier is used on the train a model using the features on the feature sets. One-third of the data is used as test feature set and the remaining 2/3 (two-third) is used as training feature set:

- train set = 75% of positive data + 75% of negative data
- test set = 25% of positive data + 25% of negative data

Classification accuracy is measured in terms of general **Accuracy**, **Precision**, **Recall**, and **F-measure**. The evaluation is done using **5-fold cross-validation.** In this process, positive and negative features are combined and then it is randomly shuffled. This is necessary to avoid negative or positive class bias in the test or train sets. n in the below code indicates the folds.

Step 3: Classification experiments:

1- Using all-words feature we obtain the following results for the NB classifier:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.7596
precision: 0.8332
recall   : 0.7596
f-measure: 0.7455


--------------------------
Beginning Cross-validation
--------------------------
Fold: 1 Acc       : 0.7606
Fold: 1 pos_prec  : 0.9907 neg_prec  : 0.6766
Fold: 1 pos_recall: 0.5281 neg_recall: 0.9950
Fold: 1 pos_fmeas : 0.6890 neg_fmeas : 0.8054
--
Fold: 2 Acc       : 0.7539
Fold: 2 pos_prec  : 0.9895 neg_prec  : 0.6702
Fold: 2 pos_recall: 0.5159 neg_recall: 0.9945
Fold: 2 pos_fmeas : 0.6782 neg_fmeas : 0.8007
--
Fold: 3 Acc       : 0.7610
Fold: 3 pos_prec  : 0.9869 neg_prec  : 0.6788
Fold: 3 pos_recall: 0.5278 neg_recall: 0.9930
Fold: 3 pos_fmeas : 0.6878 neg_fmeas : 0.8064
--
Fold: 4 Acc       : 0.7709
Fold: 4 pos_prec  : 0.9895 neg_prec  : 0.6885
Fold: 4 pos_recall: 0.5448 neg_recall: 0.9943
Fold: 4 pos_fmeas : 0.7027 neg_fmeas : 0.8136
--
Fold: 5 Acc       : 0.7518
Fold: 5 pos_prec  : 0.9860 neg_prec  : 0.6701
Fold: 5 pos_recall: 0.5103 neg_recall: 0.9928
Fold: 5 pos_fmeas : 0.6725 neg_fmeas : 0.8001
--
-----------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
-----------------------------------------------------------
accuracy : 0.7596
precision: 0.8327
recall   : 0.7596
f-measure: 0.7456
```

2- Using the stopwords filter features, the following results are obtained for the NB classifier:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.8327
precision: 0.8685
recall   : 0.8327
f-measure: 0.8285


-------------------------
Beginning Cross-validation
-------------------------
Fold: 1 Acc       : 0.8316
Fold: 1 pos_prec  : 0.9816 neg_prec  : 0.7542
Fold: 1 pos_recall: 0.6734 neg_recall: 0.9876
Fold: 1 pos_fmeas : 0.7988 neg_fmeas : 0.8552
--
Fold: 2 Acc       : 0.8364
Fold: 2 pos_prec  : 0.9847 neg_prec  : 0.7586
Fold: 2 pos_recall: 0.6814 neg_recall: 0.9896
Fold: 2 pos_fmeas : 0.8055 neg_fmeas : 0.8588
--
Fold: 3 Acc       : 0.8233
Fold: 3 pos_prec  : 0.9786 neg_prec  : 0.7417
Fold: 3 pos_recall: 0.6655 neg_recall: 0.9851
Fold: 3 pos_fmeas : 0.7922 neg_fmeas : 0.8462
--
Fold: 4 Acc       : 0.8303
Fold: 4 pos_prec  : 0.9834 neg_prec  : 0.7514
Fold: 4 pos_recall: 0.6707 neg_recall: 0.9888
Fold: 4 pos_fmeas : 0.7975 neg_fmeas : 0.8539
--
Fold: 5 Acc       : 0.8239
Fold: 5 pos_prec  : 0.9858 neg_prec  : 0.7422
Fold: 5 pos_recall: 0.6584 neg_recall: 0.9905
Fold: 5 pos_fmeas : 0.7895 neg_fmeas : 0.8486
--
-----------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
-----------------------------------------------------------
accuracy : 0.8291
precision: 0.8662
recall   : 0.8291
f-measure: 0.8246
```

Note: Accuracy increased from 76% to 83%

3- The next experiment uses bigram features with the following results:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.8260
precision: 0.8678
recall   : 0.8260
f-measure: 0.8209
```

```
--------------------------
Beginning Cross-validation
--------------------------
Fold: 1 Acc       : 0.8307
Fold: 1 pos_prec  : 0.9910 neg_prec  : 0.7504
Fold: 1 pos_recall: 0.6657 neg_recall: 0.9940
Fold: 1 pos_fmeas : 0.7964 neg_fmeas : 0.8552
--
Fold: 2 Acc       : 0.8353
Fold: 2 pos_prec  : 0.9913 neg_prec  : 0.7534
Fold: 2 pos_recall: 0.6783 neg_recall: 0.9940
Fold: 2 pos_fmeas : 0.8054 neg_fmeas : 0.8571
--
Fold: 3 Acc       : 0.8185
Fold: 3 pos_prec  : 0.9901 neg_prec  : 0.7387
Fold: 3 pos_recall: 0.6378 neg_recall: 0.9938
Fold: 3 pos_fmeas : 0.7759 neg_fmeas : 0.8475
--
Fold: 4 Acc       : 0.8156
Fold: 4 pos_prec  : 0.9897 neg_prec  : 0.7310
Fold: 4 pos_recall: 0.6415 neg_recall: 0.9932
Fold: 4 pos_fmeas : 0.7784 neg_fmeas : 0.8421
--
Fold: 5 Acc       : 0.8289
Fold: 5 pos_prec  : 0.9918 neg_prec  : 0.7460
Fold: 5 pos_recall: 0.6650 neg_recall: 0.9945
Fold: 5 pos_fmeas : 0.7962 neg_fmeas : 0.8525
--
------------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
------------------------------------------------------------
accuracy : 0.8258
precision: 0.8674
recall   : 0.8258
f-measure: 0.8207
```

Note: We still see improvements from baseline but no gain from experiment 2.

4- This experiment uses bigram features in combination with stopwords and punctuation filters:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.8855
precision: 0.9032
recall   : 0.8855
f-measure: 0.8842


--------------------------
Beginning Cross-validation
--------------------------
Fold: 1 Acc       : 0.8782
Fold: 1 pos_prec  : 0.9838 neg_prec  : 0.8095
Fold: 1 pos_recall: 0.7708 neg_recall: 0.9872
Fold: 1 pos_fmeas : 0.8644 neg_fmeas : 0.8895
--
Fold: 2 Acc       : 0.8824
Fold: 2 pos_prec  : 0.9868 neg_prec  : 0.8138
Fold: 2 pos_recall: 0.7769 neg_recall: 0.9894
Fold: 2 pos_fmeas : 0.8694 neg_fmeas : 0.8930
--
Fold: 3 Acc       : 0.8878
Fold: 3 pos_prec  : 0.9892 neg_prec  : 0.8216
Fold: 3 pos_recall: 0.7832 neg_recall: 0.9915
Fold: 3 pos_fmeas : 0.8743 neg_fmeas : 0.8986
--
Fold: 4 Acc       : 0.8840
Fold: 4 pos_prec  : 0.9911 neg_prec  : 0.8147
Fold: 4 pos_recall: 0.7758 neg_recall: 0.9930
Fold: 4 pos_fmeas : 0.8704 neg_fmeas : 0.8950
--
Fold: 5 Acc       : 0.8838
Fold: 5 pos_prec  : 0.9824 neg_prec  : 0.8206
Fold: 5 pos_recall: 0.7780 neg_recall: 0.9864
Fold: 5 pos_fmeas : 0.8683 neg_fmeas : 0.8959
```

```
------------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
------------------------------------------------------------
accuracy : 0.8832
precision: 0.9014
recall   : 0.8832
f-measure: 0.8819
```

Note: We are now at 88% accuracy for the NB classifier using 5-fold cross-validation. This is very promising.

5- Now for the bag of words (BOW) features and stopwords filter:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.8328
precision: 0.8687
recall   : 0.8328
f-measure: 0.8286


---------------------------
Beginning Cross-validation
---------------------------
Fold: 1 Acc       : 0.8271
Fold: 1 pos_prec  : 0.9836 neg_prec  : 0.7458
Fold: 1 pos_recall: 0.6680 neg_recall: 0.9887
Fold: 1 pos_fmeas : 0.7956 neg_fmeas : 0.8502
--
Fold: 2 Acc       : 0.8259
Fold: 2 pos_prec  : 0.9815 neg_prec  : 0.7464
Fold: 2 pos_recall: 0.6640 neg_recall: 0.9875
Fold: 2 pos_fmeas : 0.7921 neg_fmeas : 0.8502
--
Fold: 3 Acc       : 0.8286
Fold: 3 pos_prec  : 0.9816 neg_prec  : 0.7498
Fold: 3 pos_recall: 0.6690 neg_recall: 0.9875
Fold: 3 pos_fmeas : 0.7957 neg_fmeas : 0.8524
--
Fold: 4 Acc       : 0.8306
Fold: 4 pos_prec  : 0.9874 neg_prec  : 0.7506
Fold: 4 pos_recall: 0.6689 neg_recall: 0.9915
Fold: 4 pos_fmeas : 0.7975 neg_fmeas : 0.8544
--
Fold: 5 Acc       : 0.8325
Fold: 5 pos_prec  : 0.9804 neg_prec  : 0.7546
Fold: 5 pos_recall: 0.6779 neg_recall: 0.9865
Fold: 5 pos_fmeas : 0.8015 neg_fmeas : 0.8551
--
------------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
------------------------------------------------------------
accuracy : 0.8290
precision: 0.8662
recall   : 0.8290
f-measure: 0.8245
```

Note: We see improvement from our baseline but not from experiment 4.

6- Now for the POS features:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.5898
precision: 0.5915
recall   : 0.5898
f-measure: 0.5878
```

```
--------------------------
Beginning Cross-validation
--------------------------
Fold: 1 Acc       : 0.5994
Fold: 1 pos_prec  : 0.5888 neg_prec  : 0.6134
Fold: 1 pos_recall: 0.6690 neg_recall: 0.5292
Fold: 1 pos_fmeas : 0.6263 neg_fmeas : 0.5682
--
Fold: 2 Acc       : 0.5833
Fold: 2 pos_prec  : 0.5734 neg_prec  : 0.5966
Fold: 2 pos_recall: 0.6592 neg_recall: 0.5069
Fold: 2 pos_fmeas : 0.6133 neg_fmeas : 0.5481
--
Fold: 3 Acc       : 0.5904
Fold: 3 pos_prec  : 0.5804 neg_prec  : 0.6028
Fold: 3 pos_recall: 0.6463 neg_recall: 0.5347
Fold: 3 pos_fmeas : 0.6116 neg_fmeas : 0.5667
--
Fold: 4 Acc       : 0.5886
Fold: 4 pos_prec  : 0.5791 neg_prec  : 0.6012
Fold: 4 pos_recall: 0.6558 neg_recall: 0.5212
Fold: 4 pos_fmeas : 0.6150 neg_fmeas : 0.5583
--
Fold: 5 Acc       : 0.5901
Fold: 5 pos_prec  : 0.5765 neg_prec  : 0.6079
Fold: 5 pos_recall: 0.6579 neg_recall: 0.5232
Fold: 5 pos_fmeas : 0.6146 neg_fmeas : 0.5624
--
------------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
------------------------------------------------------------
accuracy : 0.5904
precision: 0.5920
recall   : 0.5903
f-measure: 0.5885
```

Note: Accuracy dropped to 60%: not a good model.

7- Trying **LIWC** (Linguistic Inquiry and Word Count) features:

```
-------------------------------------
Result for Single Fold(Naive Bayes)
-------------------------------------
accuracy : 0.7064
precision: 0.7071
recall   : 0.7064
f-measure: 0.7061

--------------------------
Beginning Cross-validation
--------------------------
Fold: 1 Acc       : 0.7051
Fold: 1 pos_prec  : 0.6973 neg_prec  : 0.7139
Fold: 1 pos_recall: 0.7321 neg_recall: 0.6777
Fold: 1 pos_fmeas : 0.7143 neg_fmeas : 0.6953
--
Fold: 2 Acc       : 0.7084
Fold: 2 pos_prec  : 0.6948 neg_prec  : 0.7238
Fold: 2 pos_recall: 0.7409 neg_recall: 0.6760
Fold: 2 pos_fmeas : 0.7171 neg_fmeas : 0.6991
--
Fold: 3 Acc       : 0.6966
Fold: 3 pos_prec  : 0.6866 neg_prec  : 0.7078
Fold: 3 pos_recall: 0.7242 neg_recall: 0.6690
Fold: 3 pos_fmeas : 0.7049 neg_fmeas : 0.6878
```

```
Fold: 4 Acc       : 0.7079
Fold: 4 pos_prec  : 0.6939 neg_prec  : 0.7235
Fold: 4 pos_recall: 0.7381 neg_recall: 0.6780
Fold: 4 pos_fmeas : 0.7153 neg_fmeas : 0.7000
--
Fold: 5 Acc       : 0.7130
Fold: 5 pos_prec  : 0.7001 neg_prec  : 0.7277
Fold: 5 pos_recall: 0.7452 neg_recall: 0.6807
Fold: 5 pos_fmeas : 0.7220 neg_fmeas : 0.7034
--
------------------------------------------------------------
5-Fold Cross Validation results for Naive Bayes Classifier
------------------------------------------------------------
accuracy : 0.7062
precision: 0.7069
recall   : 0.7062
f-measure: 0.7059
```

Note: No improvements from baseline.

8- As a more advanced experiment, a new Scikit-Learn classifier such as Liner Support Vector Machine (SVM) Classifier is used to compare against our baseline model. A special function in python was developed to handle multiple classifiers. Results are as follows:

```
-------------------------
Beginning Cross-validation
-------------------------
Fold: 1 Acc       : 0.9420
Fold: 1 pos_prec  : 0.9439 neg_prec  : 0.9401
Fold: 1 pos_recall: 0.9397 neg_recall: 0.9443
Fold: 1 pos_fmeas : 0.9418 neg_fmeas : 0.9422
--
Fold: 2 Acc       : 0.9457
Fold: 2 pos_prec  : 0.9448 neg_prec  : 0.9467
Fold: 2 pos_recall: 0.9462 neg_recall: 0.9453
Fold: 2 pos_fmeas : 0.9455 neg_fmeas : 0.9460
--
Fold: 3 Acc       : 0.9429
Fold: 3 pos_prec  : 0.9390 neg_prec  : 0.9467
Fold: 3 pos_recall: 0.9459 neg_recall: 0.9399
Fold: 3 pos_fmeas : 0.9424 neg_fmeas : 0.9433
--
Fold: 4 Acc       : 0.9426
Fold: 4 pos_prec  : 0.9442 neg_prec  : 0.9411
Fold: 4 pos_recall: 0.9411 neg_recall: 0.9441
Fold: 4 pos_fmeas : 0.9427 neg_fmeas : 0.9426
--
Fold: 5 Acc       : 0.9477
Fold: 5 pos_prec  : 0.9462 neg_prec  : 0.9494
Fold: 5 pos_recall: 0.9513 neg_recall: 0.9441
Fold: 5 pos_fmeas : 0.9487 neg_fmeas : 0.9467
--
------------------------------------------------------------
5-Fold Cross Validation results for SVM Classifier
------------------------------------------------------------
accuracy : 0.9442
precision: 0.9442
recall   : 0.9442
f-measure: 0.9442
```

Note: This is a high accuracy model—the best so far. Seems like this algorithm is suitable to handle this dataset using standard features. All metrics, including precision, recall and F-Measure are now at **94%.**

Conclusions

We have seen that it is possible to achieve high accuracy (94%) for sentiment classification using NLTK features and Scikit-learn support vector machines classifier applied to the Yelp reviews dataset. As a recommendation for future experimentation, we can increase the number of random sample data and see if the models can perform better with all different features used in this project. Since the Yelp's original file includes star scores from 1 through 5, another interesting variation of this experiment would be to attempt training a multi-label classification model which can be deploy in a production environment to predict sentiment on review data that has not been processed before.

References

1- Yelp Open Dataset: https://www.yelp.com/dataset
2- Text Classification for Sentiment Analysis https://streamhacker.com/2010/05/24/text-classification-sentiment-analysis-stopwords-collocations/
3- Working with text data: https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html
4- Learning to classify text: https://www.nltk.org/book/ch06.html