Project 4 - Report

Phil Bailey

November 23, 2018

Overview

The goal of this project was to use sentiment analysis to predict positive/negative movie reviews. The input was a dataset of 50k reviews that were pre-labeled either positive or negative. We used 3 different splits to seperate the data into 25k train and 25k test datasets. We then attempted to predict the 25k test reviews as either positive or negative. The output is a single text file titled mysubmission.txt containing the individual review id and probability of that review being positive. The 3 splits have already been run and are provided as Results_1.txt, Results_2.txt, and Results_3.txt.

Customized Vocabulary

A customized vocabulary was created using split3 and optimized to be used for all 3 splits. The idea of the vocabulary was to create a smaller list of terms (<=3k) to simplify our model, while still reaching an AUC >=.96. The vocabulary was created using a screening method - details of that screening method are referenced at the bottom of this report in the *Code References* section.

Technical Details

- First we used split3 to create a customized vocabulary.
  - Our vocabulary was built using <=2ngrams with a small set of stopwords.
  - Details of how the vocabulary was built using a screening method can be found in the *Code References* section at bottom of this report.
- Second we used the text2vec library to generate our test/train document term matrices (dtm)
- Once the necessary dtm's were created, we used a binary classifier to predict probablity of being a positive review.
  - Model used was a glmnet binary classifier using ridge regularization
  - Performance metric used was AUC
- In summary we were able to hit the >=.96 AUC requirement on all 3 splits with a vocabulary <=3k terms.

System Specs & Runtime

System Specs: **Windows 10, 2.7GHz, 8GB Ram**

Mean processing time to complete each split via glmnet binary ridge classifier: **85 seconds**

Programming Language: **R**

Model Results

Table 1: AUC

Model	Split1	Split2	Split3	Average
glmnet-ridge	0.9655	0.9659	0.9631	0.9648

Future mods that may improve accuracy include:

- Further customization of vocabulary
- Enhanced pre-processing to include additional stopwords, stemming, etc.
- Test classification accuracy of other algorithms such as xgboost

Code References

- text2vec sentiment analysis overview:
  https://cran.r-project.org/web/packages/text2vec/vignettes/text-vectorization.html

- Vocabulary screening method:
  https://piazza.com/class/jky28ddlhmu2r8?cid=663