Case Study 3: Email Spam

Matt Farrow

June 6, 2022

1 Introduction

In this case study, my goal is to build a email spam classifier. The data for this analysis comes from Apache’s Spam Assassin [dataset](https://spamassassin.apache.org/old/publiccorpus/), and is subset into spam messages and ham (a play on words to denote non-spam messages).

The objective of this case study is to use clustering and Naïve Bayes into order build a classifier than can separate the spam and non-spam messages. One of the challenges in this project is determining how to weight the classifier. Making it too aggressive may lead to non-spam messages being flagged as spam, too lax and the user may find themselves with unreasonable amounts of spam in their inbox.

2 Methods

## 2.1 Data Examination

The initial data set is comprised of email messages grouped into one of five folders: easy\_ham, easy\_ham\_2, hard\_ham, spam, and spam\_2. The first order of business was to read in the messages, identify which directory they came from, and where spam was found or not found.

Table

Description automatically generated

**Figure 1: Results of Initial Email Load**

Next, the data was normalized through a text cleanup process that removed special characters; converted text to lowercase; removed common, or stop, words; tokenized the document, and removed short tokens that might trip up the classifier.

Table

Description automatically generated

**Figure 2: Post-Text Normalization**

After normalizing the text, linear discriminant analysis (LDA) clustering was performed using Mallet, a Java-based package that is designed for working with language. In my case, I used it to perform clustering and topic modeling on the emails. 15 clusters were created with word weights showing the most important terms in each of the clusters.

|  |  |
| --- | --- |
| Word Weight | Term |
| 0.0607 | com |
| 0.0288 | net |
| 0.0239 | received |
| 0.0195 | localhost |
| 0.0125 | org |
| 0.0115 | jul |
| 0.0114 | esmtp |
| 0.0109 | netnoteinc |
| 0.0101 | content |
| 0.0101 | http |
| 0.0097 | zzzz |
| 0.009 | aug |
| 0.0085 | mail |
| 0.0076 | mon |
| 0.0075 | sep |

**Table 1: Cluster Example**

Each topic was then examined by its frequency within each of the original directories using a heatmap.

Shape, square

Description automatically generated

**Figure 3: Clustering Heatmap**

In addition to the LDA clustering that was undertaken with Mallet, an additional clustering was done using K-means clustering and an elbow plot was generated to examine the most appropriate cluster number to use.

Chart, line chart

Description automatically generated

**Figure 4: K-Means Clustering Elbow Plot**

## 2.2 Model Preparation & Execution

Two approaches were taken to model building. In both, the original data was run through a Multinomial Naïve Bayes classifier. Numeric values were scaled and missing values were imputed using the median. Categorical values were one-hot encoded with missing values being replaced by the most frequent result.

3 Results

## 3.1 Model Results

As seen in **Table 2**, the accuracy of this model was only about 79% with a recall of approximately 24%, indicating a poor performance when it comes to filtering spam out of user’s inboxes. That fairly poor result can be further seen in the ROC curve (**Figure 5**) as well as the model’s confusion matrix **(Figure 6)**.

|  |  |
| --- | --- |
| Metric | Value |
| Accuracy | 0.7857 |
| Recall | 0.2375 |
| Precision | 0.7651 |
| F1 | 0.3625 |

**Table 2: Model 1 Performance Metrics**

Chart, line chart

Description automatically generated

Figure 5: Model 1 ROC Curve

Chart

Description automatically generated

Figure 6: Model 1 Confusion Matrix

In the second model, the same pre-processing steps were taken as in the first, but with the addition of the clustering that was previously undertaken.

|  |  |
| --- | --- |
| Metric | Value |
| Accuracy | 0.8856 |
| Recall | 0.8979 |
| Precision | 0.7232 |
| F1 | 0.8011 |

Table 3: Model 2 Performance Metrics

Chart, line chart

Description automatically generated

Figure 7: Model 2 ROC Curve

Chart

Description automatically generated

Figure 8: Model 2 Confusion Matrix

4 Conclusion

The second model that was built on the clustering results showed significantly improved recall as well as improved precision and accuracy. In addition to increasing the ability of our classifier to sift spam out of email inboxes, the inclusion of clustering means that our model can continue to adapt as new data is fed in in the form of new emails as well as user behavior in flagging messages as spam or moving them out of spam into their inboxes.

# Appendix

## Code

Code begins on the following page.