SMS SPAM CLASSIFICATION

CPTS 315

INTRO TO DATA MINING

SUBMITTED BY:

ABHILASH AMBATI 11602294

INTRODUCTION:

In today's globalized world, SMS is a primary source of communication. This communication can vary from personal, business, and corporate to government. With the rapid increase in SMS usage, there has also been an increase in SPAM SMS. Apart from being annoying, spam SMS can also pose a security threat to people and their data. It is estimated that spam cost businesses on the order of $100 billion in 2007. In this project, we use text mining to perform automatic spam filtering to use SMS effectively. We try to identify patterns using Data-mining classification algorithms to enable us to classify the SMS as HAM or SPAM.

LEARNING DATA:

The data used for this project was taken from the Spam Assassin public corpus website. It consists of 5573 text messages. This data was divided into two data sets: train and test. Each dataset contains a randomly selected collection of SMSS in plain text format, which has been labeled as HAM or SPAM. The training data is used to build a model for classifying SMS into HAM and SPAM. The test data is used to check the accuracy of the model built with the training data. The training data set contains 5573 SMS with 774 spam and 4825 ham messages. 30% of the data was reserved for testing and the rest was used for training.

DATA PREPROCESSING:

The data was instead into a two-column data frame and converted the spam column to binary and converted every text to lower case and removed all the punctuations and shuffled the complete data.

BAG OF WORDS:

I used the training data to grab all the text that is spam and join them together making it a giant string, each string is separated using a space, all these strings are basically words that are used in a spam text and put into another string, this process was repeated for the non-spam texts. After this, we put together a string that consists of both words that appear in both spam and non-spam words.

I created a dictionary that consists of spam words and non-spam words, these bags contain the proportionate of words from all the spam texts.

PREDICTION OF TEST SET:

It tests the probability that the word came from the spam bag of words and if the probability is high, we're going to classify it as spam if it's very low we're going to classify it as valid.

“Urgent please call this number” is a spam text, I sed this method to check the probability of each word.

$$\log(\text{"urgent"})\,|SPAM) * P(\text{please})|SPAM) * \ldots\ldots * P(call)\ldots.$$

```
     word  spam_prob  non_spam_prob       ratio
0  urgent   0.003879       0.000021  188.634600
1    call   0.018929       0.003311    5.717620
2    this   0.005275       0.003537    1.491529
3  number   0.001629       0.001049    1.553461
Spam Score: -23.16448028206801
Non-Spam Score: -29.15569965721826

True
```

Since the spam score is higher than the non-spam score, the text is spam, by using this method, we predict if the text is spam or not spam.

Examples:

```
offer for unlimited money call now
```

```
       word  spam_prob  non_spam_prob      ratio
0      offer   0.001552       0.000144 10.779120
1        for   0.010939       0.007197   1.519856
2 unlimited   0.000698       0.000062 11.318076
3      money   0.000078       0.000802   0.096736
4       call   0.018929       0.003311   5.717620
5        now   0.010784       0.003989   2.703114
Spam Score: -38.192756947863074
Non-Spam Score: -41.98513617615185

True
```

```
are you at class yet?
```

```
  word  spam_prob  non_spam_prob     ratio
0    are   0.004422       0.006005 0.736450
1    you   0.016447       0.025992 0.632762
2     at   0.001396       0.005778 0.241667
3 class   0.000155       0.000658  0.235793
4    yet   0.000078       0.000576 0.134739
Spam Score: -36.318752270659445
Non-Spam Score: -28.85333457585457

False
```

TESTING DATA:

I used the same technique for the whole testing data set which gives the prediction for the whole data and programmed it to predict the correctly detected spam text, which came out to be 91.44% and this was repeated for the fraction of text sent to spam which came out to be 2.6%

CONCLUSION:

Given a set of words, we used feature selection to obtain words that allows us to distinguish between spam and ham spam. We also compared the accuracy of various classifiers in predicting

the class attribute. It is possible that the accuracy increase, as the training data increase the program learns more so the predictions gets even better and the accuracy increases.

Sources:

Data Set: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset