CSM 484

Information Retrieval
Project 2 Report

Ziyad Alghamdi – 444105687

Ibrahim Binnafisah - 444101505

Overview

- We implemented a Multinomial Naïve Bayesian classifier for email spam detection.
- Our goal here is to classify emails as either spam or ham (not spam).
- We are testing the classifier with four different configurations and compare their performance using F-Score.

Design Choices

- We have processed the text using NLTK library (just the tokenization).
- We represented the features using word frequency distribution/class.
- The probability calculation is: $\mathrm{P}(word|class) = \frac{Frequency\ of\ Word}{Total\ words\ in\ Class}$
- We have used the prediction rule to compare probabilities and classify the emails accordingly.
- We have used the time library to just to measure the time it takes for the model to train and how much time it needs to be tested.
- We have used the math library to directly implement the log function.

Experiment Setup

- The training was done on an email dataset from an energy company called Enron with more than 18,000 documents (Emails).
- After training the model, we tested it on 800 emails the dataset is also from the same company.
- Four different configurations of use_log and smoothing used.
- Model evaluated using F1-Score (Which balances precision & recall).
- Tested using Python on a PC running Windows 10, I7-7700K CPU, and 32GB of RAM.

Results

Log	Smoothing	F-Score (Rounded to 3dp)
True	**True**	0.970
True	**False**	0.200
False	**True**	0.535
False	**False**	0.198

- Additional Note: Average training time is about 29.267 seconds.
- We can see that by using logarithms and smoothing we achieve an excellent score of about 97% !

```
Training time: 28.30 seconds
Log&Smoothing
True&True F1-Score: 0.9697986577181208
True&False F1-Score: 0.1997503121098627
False&True F1-Score: 0.5352798053527981
False&False F1-Score: 0.19796954314720813
```

```
Training time: 30.55 seconds
Log&Smoothing
True&True F1-Score: 0.9697986577181208
True&False F1-Score: 0.1997503121098627
False&True F1-Score: 0.5352798053527981
False&False F1-Score: 0.19796954314720813
```

```
Training time: 28.95 seconds
Log&Smoothing
True&True F1-Score: 0.9697986577181208
True&False F1-Score: 0.1997503121098627
False&True F1-Score: 0.5352798053527981
False&False F1-Score: 0.19796954314720813
```

Final Notes

- As this was our first “Machine-Learning” project we used help from GeeksForGeeks, we focused on working on the logic itself instead of relying on a pre-made library that cuts it down.
- Without any doubt, using logarithms and smoothing directly impacts the score as we could have unseen words or rare words. Both of them are important to have a good F-score.
- We can improve on this project more by using word embeddings that we took later on in this course (Word2Vec).
- We have used ChatGPT as a debugger and helper as we are relatively new to python but all of the primary work has been done solely by us.