

CS5691 Pattern Recognition and Machine Learning

Assignment 3

Submitted by

CS22M025 Ayana Satheesh B

Indian Institute of Technology Madras



Contents

List of Figures	i
List of Tables	ii
1 Question 1:Spam email classification using Naive Bayes	1
1.1 Dataset	1
1.2 Preparing the dictionary using the dataset	1
1.3 Learning the model for classification	2
1.4 Saving the model learned	3
1.5 Testing the model on test data	3
1.6 Performance of the algorithm	3

List of Figures

1.1	Snapshot of a small part of the dictionary	2
1.2	Accuracy of Naive Bayes algorithm on test data	3

List of Tables

Chapter 1

Question 1:Spam email classification using Naive Bayes

1.1 Dataset

The dataset used have been formed from "enron dataset". The enron dataset contains nearly 33,000 emails (including both ham and spam) which was collected from real world (i.e. from employees). The dataset used here is a subset of enron dataset and this dataset contains around 10,000 emails(including both ham and spam). This dataset was chosen as it formed from real emails and so it would help in classification of emails.

1.2 Preparing the dictionary using the dataset

The dataset was divided in 8:2 ratio. The bigger part was used for training and making the dictionary. The dictionary of words is prepared by iterating through each of the emails (which have either ham or spam in their filename) and filtering the words. The preproceesing includes the following:

- i)Removing the punctuation marks
 - ii)Removing the stopwords since the stopwords (such as 'in', 'at', 'the', etc) does not add any extra information of the email being spam or ham.
 - iii)Removing any words of length less than or equal to two.
 - iv)The digits present were also removed since these does not help in classification.
 - v)All the words after these were converted to lower case . This means that the word "Hello" and the word "hello" will be treated as the same word.
- After pre-processing the words that were present is made into a dictionary and these words were sorted and they represent the features of the model. A

small part of the dictionary is shown in figure1.1



```
'airborne',  
'aircraft',  
'airdancer1',  
'airdrop',  
'aiready',  
'aired',  
'airedale',  
'aires',  
'airfare',  
'airflow',  
'airfoil',  
'airframe',  
'airing',  
'airlift',  
'airline',  
'airlines',  
'airlock',  
'airmail',  
'airman',  
'airmass',  
'airmen',  
'airpanel',  
'airpark',  
'airplane',  
'airplus',  
'airport',  
'airspace',  
'airspeed',
```

Figure 1.1: Snapshot of a small part of the dictionary

1.3 Learning the model for classification

Each email in the dataset was taken and split into words after removing the punctuation marks from it. All the words were converted to lower case. For each of the words present in the email, no. of times it appears in that email was counted. An array was made which contains the count of each word present in the email . In this way for every feature we get the count of the words present in that email.

After obtaining the count of each features in each email, Naive Bayes parameters were calculated. This includes calculating the probability of each word given it occurs in the spam . The same was calculated for non spam also.

The probability of spam and non spam emails are also calculated from the training dataset.

Laplace smoothing was done by adding a vector of ones after learning the parameters so that when any feature does not occur in any of the spam or non spam, the total probability does not goes to zero.

1.4 Saving the model learned

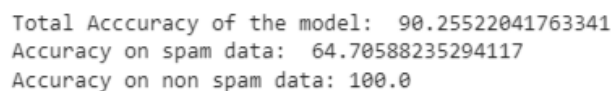
The model learned was saved into seperate files, so that once learned it can be used to test any no. of times. This includes all the probabilities of the words, the probability of spam and non spam emails, and the dictionary.

1.5 Testing the model on test data

The learned probabilities were used to classify the test emails into spam and non spam. The emails in the test dataset were read and converted to words by removing punctuations and making it into lower case . For each of the email in test data the probabilities were calculated of the email being in spam and non spam. The email was classified as spam or non spam based on which category had the highest probability.

1.6 Performance of the algorithm

The accuracy of the classification using Naive Bayes algorithm was calculated and is shown in figure1.2. The total accuracy of the algorithm on the test data is 90.25%. The algorithm performs well on ham emails whereas the accuracy on spam emails is found to be less.



```
Total Accuracy of the model: 90.25522041763341
Accuracy on spam data: 64.70588235294117
Accuracy on non spam data: 100.0
```

Figure 1.2: Accuracy of Naive Bayes algorithm on test data