

# Project- Support Vector machines (SVM) to build a Spam Classifier

B. Shadrack Jabes

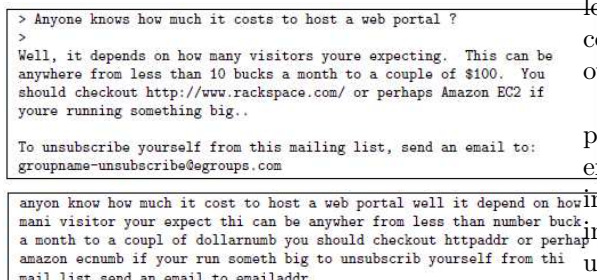
June 6, 2020

## 1 Introduction

Most of the email services today provide spam filters which classifies whether an email is spam or not?. The aim of this project is to build a spam filter using SVMs.

## 2 Dataset

The dataset is an email that contains an URL, email address in the text, some numbers, symbols of money (dollars) and amount. Because the URLs or amount may be different and there is no unique way of representing them, I preprocess them and replace it with some kind of unique strings. This improves the performance of the spam classifier.



> Anyone knows how much it costs to host a web portal ?  
>  
Well, it depends on how many visitors youre expecting. This can be anywhere from less than 10 bucks a month to a couple of \$100. You should checkout <http://www.rackspace.com/> or perhaps Amazon EC2 if youre running something big..  
  
To unsubscribe yourself from this mailing list, send an email to: [groupname-unsubscribe@egroups.com](mailto:groupname-unsubscribe@egroups.com)

anyon know how much it cost to host a web portal well it depend on how mani visitor your expect thi can be anyvher from less than number buck. a month to a coupl of dollarnumb you should checkout httpaddr or perhap amazon ecnumb if your run someth big to unsubscrib yourself from thi mail list send an email to emailaddr

Figure 1: Figure displays a sample email before and after preprocessing

The preprocessing steps include the following simple steps:

- Lower casing: Ignore capitalization of words
- stripping HTML tags: I remove the HTML tags leaving the contents
- replacing URLs: Now I replace the URLs with some string
- replacing amount: Numbers are replaced with text values
- replacing symbols of amount: All symbols are replaced with the text value
- replacing Email address: replacing email adress with text email address.

- word stemming: words that share similar word stems are replaced with just the stem keyword. For instance, 'discount', 'discounts', 'discounted', 'discounting' are all replaced with 'discount'.
- removing punctuations: remove non-words (like comma, semicolon, etc), punctuations.

The final processed form yields better SVM performance in extracting features and in building spam classifiers. A sample email before and after preprocessing is shown in figure 1.

## 3 Vocabulary list

I have a list of words which is given as a part of the project. The list of words are collected by looking at the occurance of the words in spam corpus. For example all words that occur at least over 100 times in the spam corpus is stored.

Now I applied the vocabulary list to the preprocessed email text. So a particular word in the email is now mapped on to an index mentioned in the vocabulary list. If the word is not indexed in the vocabulary list then the word can be left unchanged.

## 4 Feature extraction

In this step each mail is converted into a vector with n dimensions. The feature vector  $x$  is set to ones or zeros depending upon whether or not the  $i^{th}$  word in the vocabulary present in email. The final array of  $x$  will consist of an array with values either ones or zeros.

## 5 Training SVM for spam classification

The training set includes 4000 emails both spam and nonspam ones. The test set contains 1000 emails. Each of these emails are preprocessed and the features are extracted by following the recepie mentioned above. The code trains the SVM classifier to classify the emails and flags it spam ( $y = 1$ ) or non spam ( $y = 0$ )

## 6 Contribution

I implemented the vectorised SVM classifier to identify spam mails.