Make Spam classification Model work in Real Time (gmail inbox)

Raj Kishore Sahni

Information Technology, Maharaja Agrasen Institute of Technology, New Delhi, India.

Abstract: The upsurge in the volume of unwanted emails called spam has created an intense need for the development of more dependable and robust antispam filters. Machine learning methods of recent are being used to successfully detect and filter spam emails. We present a systematic review of some of the popular machine learning based email spam filtering approaches. Our review covers survey of the important concepts, attempts, efficiency, and the research trend in spam filtering. The preliminary discussion in the study background examines the applications of machine learning techniques to the email spam filtering process of the leading internet service providers (ISPs) like Gmail, Yahoo and Outlook emails spam filters. Discussion on general email spam filtering process, and the various efforts by different researchers in combating spam through the use machine learning techniques was done. Our review compares the strengths and drawbacks of existing machine learning approaches and the open research problems in spam filtering. We recommended deep learning and deep adversarial learning as the future techniques that can effectively handle the menace of spam emails

After developing mmodel we can make our model work in realtimeusing google apps scripts. For that we have to write a script in javascript which will take incoming email's body.

KEYWORDS: *How To Make you spam model work on gmail*

DOI of the Article: https://doi.org/10.46501/IJMTST0706040

Available online at: http://www.ijmtst.com/vol7issue06.html

As per UGC guidelines an electronic bar code is provided to seure your paper

To Cite this Article: Raj Kishore Sahni. Make Spam classification Model work in Real Time (gmail inbox). *International Journal for Modern Trends in Science and Technology* 2021, *7*, 0705079, pp. 234-237. https://doi.org/10.46501/IJMTST0706040

Article Info. Received: 16 May 2021; Accepted: 11 June 2021; Published: 17 June 2021

INTRODUCTION

Nowadays, e-mail provides many ways to send millions of advertisement at no cost to sender. As a result, many unsolicited bulk e-mail, also known as spam e-mail spread widely and become serious threat to not only the Internet but also to society. For example, when user received large amount of email spam, the chance of the user forgot to read a non-spam message increase. As a result, many e-mail readers have to spend their time removing unwanted messages. E-mail spam also may cost money to users with dial-up connections, waste bandwidth, and may expose minors to unsuitable content. Over the past many years, many approaches have been provided to block e-mail spam

For filtering, some email spam are not being labelled as spam because the e-mail filtering does not detect that email as spam. Some existing problems are regarding accuracy for email spam filtering that might introduce some error. Several machine learning algorithms have been used in spam e-mail filtering, but Na¨ıve Bayes algorithm is particularly popular in commercial and open-source spam filters . This is because of its simplicity, which make them easy to implement and just need short training time or fast evaluation to filter email spam. The filter requires training that can be provided by a previous set of spam and non-spam messages. It keeps track of each word that occurs only in spam, in non-spam messages, and in both. Naıve Bayes can be used in different datasets where each of them has different features and attribute.

The research objectives are: (i) to implement the Na¨ıve Bayes algorithm for e-mail spam filtering on two datasets, (ii) to evaluate the performance of Na¨ıve Bayes algorithm for e-mail spam filtering on the chosen dataset.

The rest of the paper is organized as follows: Section II describes the related work on Na¨ıve Bayes algorithm for e-mail spam filtering. Section III presents the methodology process of e-mail spam Section IV presents the experimental setup. Section V shows the result and analysis on two datasets. Finally, Section VI concludes the work and highlights the direction for future research. .

OBJECTIVES

The objective of this project is to build a spam model and we will be using this model to classify the spam and non spam emails. This model will help us to classify emails in real time incoming emails in gmails and if will spam it will put that email in a gmail spam box after running the script The efficiency of the model depends on the your model. What parameters it is taking to classify email is spam or not

RELATED WORK

Spammers are now able to launch large scale spam campaigns, malware and botnets helped spammers to spread spam widely. Upon receiving and opening a spam email, Internet users is exposed to security issues as spams are normally broadcast for bad intention. One of the common email spam example received by users are an email requesting for IDs and passwords(Refer to

There is a rapid increase in the interest being shown by the global research community on email spam filtering. In this section, we present similar reviews that have been presented in the literature in this domain. This method is followed so as to articulate the issues that are yet to be addressed and to highlight the differences with our current review. Lueg presented a brief survey to explore the gaps in whether information filtering and information retrieval technology can be applied to postulate Email spam detection in a logical, theoretically grounded manner, in order to facilitate the introduction of spam filtering technique that could be operational in an efficient way. However, the survey did not present the details of the Machine learning algorithms

METHODOLOGY

This section describes the methodology that is used for the research. The methodology that is used for the filtering method is machine learning techniques that divide by three phases.The methodology is used for the process of e-mail spam filtering based on Na¨ıve Bayes algorithm.

3.1. Na¨ıve Bayes classifier

The Na¨ıve Bayes algorithm is a simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combination of values in a given dataset [4]. In this research, Na¨ıve Bayes classifier use bag of words features to identify spam e-mail and a text is representing as the bag of its word. The bag of words is always used in methods of document classification, where the frequency of

occurrence of each word is used as a feature for training classifier. This bag of words features are included in the chosen datasets.

Na¨ıve Bayes technique used Bayes theorem to determine that probabilities spam e-mail. Some words have particular probabilities of occurring in spam e-mail or non-spam e-mail. Example, suppose that we know exactly, that the word Free could never occur in a non-spam e-mail. Then, when we saw a message containing this word, we could tell for sure that were spam email. Bayesian spam filters have learned a very high spam probability for the words such as Free and Viagra, but a very low spam probability for words seen in non-spam e-mail, such as the names of friend and family member. So, to calculate the probability that e-mail is spam or non-spam Na¨ıve Bayes technique used Bayes theorem as shown in formula below.

$$P(spam \mid word) = \frac{\mathrm{P(spam).\,P(word|spam)}}{\mathrm{P(spam).\,P(word|spam) + P(non-spam).\,P(wor}}$$

Where:

(i) Bayes Classifier

The following sections will explain the activities that involve in each phases in order to develop this project. Figure 2 shows the process for e-mail spam filtering based on Na¨ıve Bayes algorithm.

3.2. Pre-processing

Today, most of the data in the real world are incomplete containing aggregate, noisy and missing values [9]. Pre-processing of e-mails in next step of training filter, some words like conjunction words, articles are removed from email body because those words are not useful in classification.

(Refer to Figure 3 for sample of data).

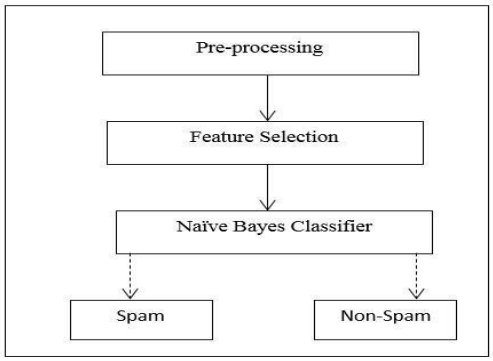

Figure 2. Process of E-mail spam filtering based on Na¨ıve Bayes Algorithm

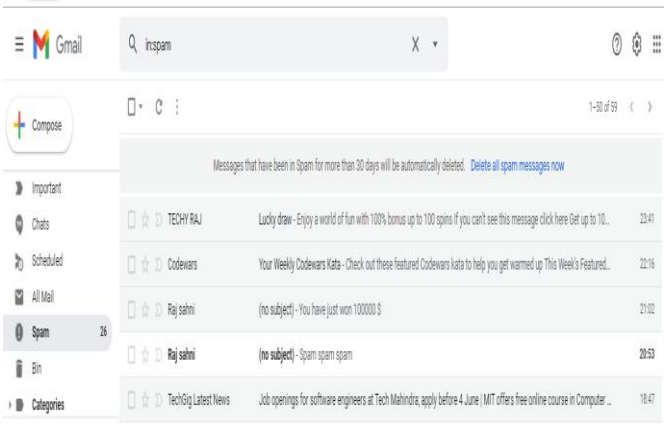

Figure 3. Sample of spam datafragment

A full list of the attributes in this data set appears in the ”Attributes” frame as shown in Figure 4. Random selection of attribute are performed for the further process.

Attributes *capital run length –average,capital run length longest* and *capital run length total* are removed from the list by checking the box to their left and hitting the Remove button.

3.3. Feature Selection

After the pre-processing step, we apply the feature selection algorithm, the algorithm which deploy here is Best First Feature Selection algorithm

MAKE IT WORK IN TREAL TIME

To make it work in realtime first you have to create the web api of the machine learning model using flask .

Then we need procfileand requirements .txt file .requirements .txt file contains all the libraries which we have used in the project. It will be imported during hosting on cloud

Upload the web api ,procfile and requirements.txt file on github using repositories.

After uploading in repo we need to host our model on some cloud hosting platform . In my project I have used heroku to host my model after hosting your model and get the herroku web app link

1. Create Google App Script

Google Apps script is a scripting platform developed by Google for light – weight application development in the google workspace platform

After hosting your model create app script for using gmail emails on your model put that email in a spam box

After creating app script we need to create trigger and have to set timer which will fire the script after the specified amount of time .

When the script runs it basically take the incoming emails body from the inbox and take emails body using stop word it removed the unwanted words and using countVectorizer it will matrix of words . if the value of the spam>0. It will put that email into Gmail spam box

Here below is the app script which I have used in the google app script

```
function classify() {
 // Wrap the entire function in a try / catch, in case there is an error, log it.
 try {
  // Looks for threads in the inbox you can make changes according to your requirements
   var threads = GmailApp.search('in:inbox', 0, 50);
   // If there are threads
   if (threads.length> 0) {
    // For each thread
    for (var t = 0; t <threads.length; t++) {
     // Get the current thread we are iterating over
     var thread = threads[t];

     // Get the first message in the thread
     var message = thread.getMessages()[0];

     var from = message.getFrom();
     var body = message.getPlainBody();
     var email = from.match(/<(.*?)\>/)[1];
     //if(email === 'your test email address goes here for testing purposes'){
     var url =
      'https://spamflask.herokuapp.com/classify?msg='
+
encodeURIComponent(body);
     //Logger.log(url);
     var response = UrlFetchApp.fetch(url, { muteHttpExceptions: true });
     out = JSON.parse(response);
     if (out['spam'] > 0) {
thread.moveToSpam();
     }
     //}
    }
   }
 } catch (e) {
  Logger.log(e.toString());
 }
}
```

Here is the screenshot which is classified by the model on incoming emails first this email was in inbox but after running the script it is in spam box

CONCLUSION

We have used naïve bayes classifier in the model. Which is a very good classifier.But the accuracy of the model depends on the things you have taken in consideration while creating your model.In my model I have just taken 2 column that is text column and other spam column .If you consider more things like ip address ext your model will be more precised

In this for testing I have used real time incoming emails for testing.it has given good result it perfectly classifying the email whether it is spam or not according to my model trained in a way.

You can make your own spam classification model and can set it in your gmail.it will work as your own classifier and you can also adjust which type of mail you want to receive

REFERENCES

1. Rushdi, S. and Robet, M, "Classification spam emails using text and readability features", *IEEE 13th International Conference on Data Mining*, 2013.
2. Androutsopoulos, I., Paliouras, G., and Michelakis, "E. Learning to filter unsolicited commercial e-mail", *Technical report NCSR Demokritos*, 2011.
3. Rathi, M. and Pareek, V. "Spam Mail Detection through Data Mining A Comparative Performance Analysis", *I.J. Modern Education and Computer Science*, 2013, 12, 31-39.
4. Patil, T. and Sherekar, S. "Performance Analysis of Na¨ıve Bayes and Classification Algorithm for Data Classification", *International Journal Of Computer Science And Applications*, 2013.
5. www.researchgate.net
6. https://towardsdatascience.com/email-spam- detection