Email Monitoring System

Jannatul Supi Jenifa

A Project Submitted to

GRAND VALLEY STATE UNIVERSITY

In

Partial Fulfillment of the Requirements

For the Degree of

Master of Science in Cybersecurity

School of Computing and Information Systems

December 2023

Index

Abstract

Email communication is one of the most preferred and most used platforms in the professional sector. As much as the usability of this platform grows, so do the security concerns. Because the possibility of using this platform as an attack vector for a cyberattack has also increased. So, we need to be more concerned about security, and this is how I connected with this project.

This project has two folded parts. One part is a spambot that is able to send spam emails to the recipient's email account. Actually, this part is a simulation of how this spambot can become an attack vector to address a DoS attack. Another part is a spam email detector that uses an ML model to detect whether an email is spam or not. In this section, we conducted a comparative analysis of our model's performance evaluation with other ML algorithms. Our model's accuracy, which was 98.691%, was the highest of the rest.

1. Introduction

Email is currently one of the most popular modes of communication in both the business and academic worlds. This sector's danger rises in proportion to its increased usage. And it turns into one of the most widely used openings for attacks. Email usage has increased in recent years for both personal and professional purposes. Regrettably, spam email is growing far more quickly[1].This increases my awareness and curiosity to work on this topic.

Spam is a popular type of email assault [4] in which a hacker sends unsolicited messages to a big number of recipients. Usually, the emails are promotional in nature.Email security measures like anti-spam and anti-phishing are useless [6] against email bomb attacks because the "emails flood" coming from these sources are authentic, possibly even ones the victim has previously corresponded with.Domain validation defenses therefore identify email bomb communications as authentic. Filtering email bomb traffic is not a good use for rate-limiting defenses or defenses against Denial of Service (DoS) assaults, for example[7], since the attackers can use them to block legitimate services.

On the other hand, machine learning technology has become a game changer in the tech field as well as the cybersecurity sector. In the present time, machine learning is widely used to detect malware, sophisticated attacks, phishing activities, etc.Machine Learning is an engineering approach that enables computational instruments to act without explicit programming, making it a valuable tool for detecting and addressing spam issues [8]. Large datasets of emails, both spam and non-spam, are used to train machine learning models to find patterns and characteristics that mark spam emails apart from legitimate ones. Examples of spam and non-spam emails are given to these models, with labels designating whether

each is spam or ham. They pick up on typical traits from spam emails, like text patterns, keywords, and phrases. The model can accurately classify new emails as spam or ham by comparing them to the patterns and attributes it has acquired after being trained on a sizable dataset. Because of machine learning efficiency, I used this technology to detect spam emails by monitoring an email and also solve the class imbalance problem which is the most common problem of spam dataset .

The objectives of this project:

a. Develop a model for spam email detection.
b. Implement a spambot to spamming receipt email to address the attack vector for Dos attack.
c. To increase awareness of the threats of email accounts and look forward to increasing the security of email accounts.

2. Problem Statement

For a long period of time, email has been a highly used platform in the professional sector, and it’s evolved over time. So, the attack vector of this sector is also increased, but security modules are not as effective as required. And for protection and prevention, the first and most important step is the identification of risk factors and the detection of malicious activities, where machine learning is widely used. Class imbalance is a prevalent issue in machine learning spam email detection, which can significantly impact the model's accuracy.The primary contribution of this project is the spam detection model, which tackles the problem of unequal class distribution.Another problem here is awareness about the possible most significant impact of this spam attack. Most of the time, we know the problem and the instant outcome but don’t realize where it can go. In

this project, implement a spambot as an experiment to show it can be an attack vector to address DOS attacks. This is a folded part of the project.

3. Background Study

Spam email is the most common term in email accounts, and it's the most common way to attack an email account. But there are also different types of attacks on email accounts, like email relaying, spam emails, impersonation, spoofing, DOS attacks, etc. Dos attack is a very common attack and very powerful, and via spambot it's easier to do.Denial of Service (DoS) attacks pose a threat to the availability of Internet and its services[5]. The relative ease and low costs of launching such attacks have made them one of the top threats to the stability of the Internet[9]. And here in this project, the spambot is a simulation to illustrate how easy and convenient it is to use the spambot as an attack vector to address DOS attacks.

In the case of spam detection, there is a lot of comparative research on ML models in this sector. where used some supervised algorithms in detection like random forest, SVM, KNN, Naïve Bayes, etc. The algorithms are used with different modes, for instance, multinomial Naïve Bayes and Gaussian Naïve Bayes[1]. In [1], the Multinomial naive bayes algorithm's performance was the highest compared to others algorithms and in this study comparing SVM, Random Forest, Logistic Regression, Gaussian naive bayes and multinomial naive bayes. In [2] comparative study, SVM algorithm's performance was the highest among other algorithms such as KNN, Decision Tree, Linear Regression, Neural networks etc.

According to my findings and experiments, the multinomial Naïve Bayes is the best performer for this particular dataset, which is elaborately discussed below.

4. Spambot

This spambot has the ability to continuously overwhelm a victim's email account with emails. This spambot can send a significant number of emails in a matter of seconds.As the email account is flooded with numerous emails, the load on the system increases. The system continuously consumes resources, and the load on the account increases continuously. After receiving too many emails in a short amount of time, when the load on the account reached out of its capacity and the email server could not take the load, it hung up and crashed. As a result, the victim cannot access the account resource, and this is how the DOS attack happened.This is actually a simulation of how to counter a denial-of-service attack by using a spambot as an attack vector.

The goal of creating the spambot is to demonstrate the risk factors to spam someone's email account and respond to a denial-of-service attack on the account.And this is the initial version of the spambot.

Spambot is developed with python and uses the “smtplib” library to implement this.

As an experiment, I sent 1000 emails to the recipient account in just a few seconds by spambot.Here, I have created an email account for the purpose of using it as a victim account for spamming. For this experiment, use a gmail account and SMTP server. However, this spambot is capable of sending more emails than the mentioned number. The following figure shows the email account flooded with spam emails that were sent by

the demo version of the spambot and which shows an initial impact of spamming receipt's email account:

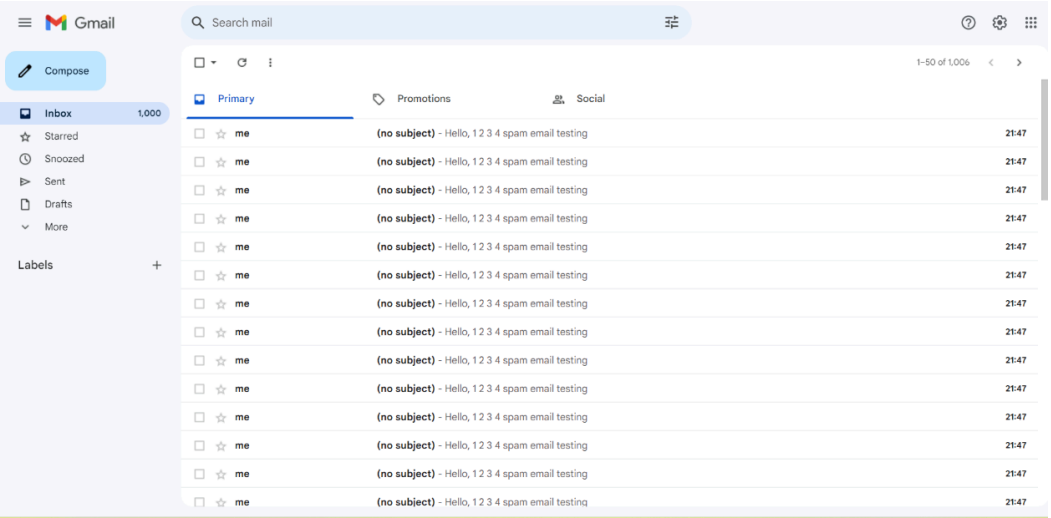

Figure 1: Email account flooded with spam emails.

5. Spam Email Detector

The main contribution of the project is created in this section, where we build a machine learning model for spam email identification. In this instance, after receiving the email as input, the spam detection model assesses it to decide if it is spam or not. In this detector model, the learning algorithm is the Naive Bayes algorithm, which performs the best when compared to other machine learning approaches that are applied in this project, which are discussed in more depth below.

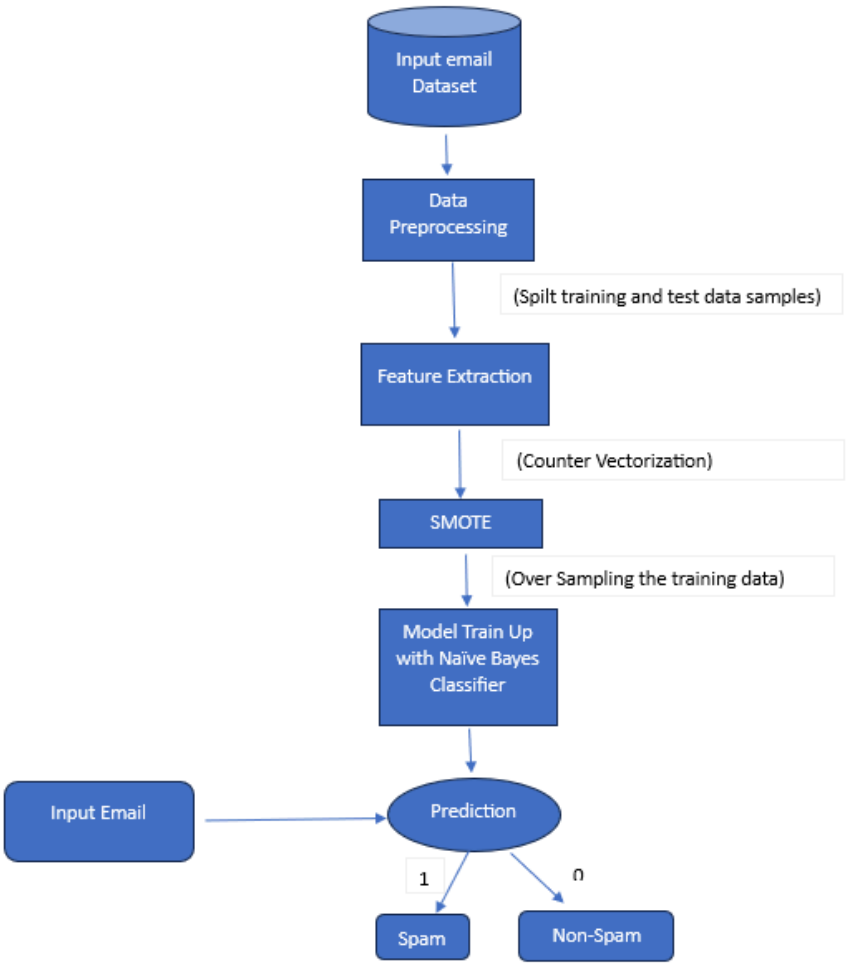

Figure 2: Spam email detector ML model.

This figure illustrates the whole process of spam detection. Every steps of the detection model is descriptively discussed below:

Dataset Description

The following dataset is prepared for spam email detection by observing the email body text. It is collected from the "Kaggle.com" site. This dataset has 4360 legitimate data and 1368 spam data.The dataset contains 5728 rows and 2 columns where the last column

carries the data label of an individual row. And the first column carries the email body text. The data in the first column is in text format, and the data in the last column is in numerical form. There have two predictor classes in the dataset:

– Legitimate: where 0 stands for legitimate.

– Spam: where 1 stands for spam.

Data Preprocessing

After collecting the dataset from the “Kaggle.com” site, there were some preprocessing steps for using the dataset in the project. The steps are given below:

1. Fit the dataset into **pandas** data frame.
2. Check duplicate and null data of the dataset.
3. Then split the data frame into train and test samples using sklearn “**train_test_spilt**” module.

Feature Extraction

Utilize Python's "CountVectorizer()" function to extract features. Text is tokenized using CountVectorizer by randomly assigning a number to each word. The frequency of each word is then determined and saved to a variable known as CV. Here, the email text is transformed into a vector of term or token counts using CountVectorizer. This is how this technique is used to extract the main feature of spam detection, which is email text.

SMOTE

The dataset has a class imbalance issue. I used SMOTE to find a solution to this issue. SMOTE is a data augmentation method that addresses the issue of class imbalance in machine learning and data mining. Using a random selection from the minority class, SMOTE determines a point's k-nearest neighbors. The synthetic points of the specified point are added to those of its neighbors. This is how SMOTE functions.

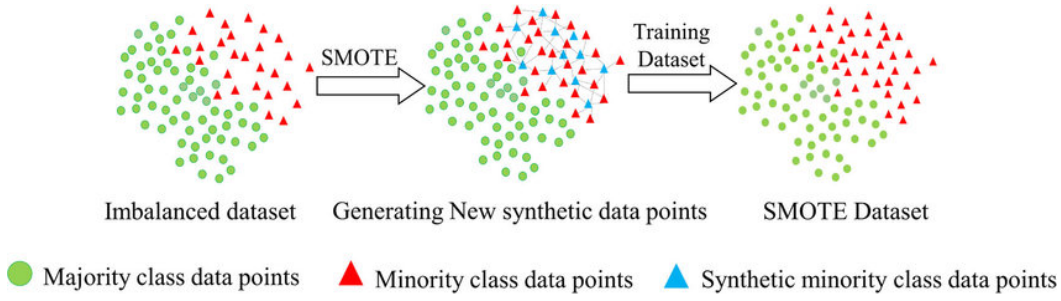

Figure 3: SMOTE Technique on training dataset.

ML Classifier

In this project use naïve bayes ML classifier. It is a classification method built on the Bayes Theorem with the premise of predictor independence. The simple form of the calculation for Bayes Theorem is as follows:

$P(A|B) = P(B|A) * P(A) / P(B)$

A Naive Bayes classifier, to put it simply, believes that the existence of one feature in a class has nothing to do with the presence of any other feature.

A fruit might be categorized as an apple, for instance, if it is red, rounded, and about 3 inches in circumference. Even though some of these characteristics are contingent upon

others or on the presence of other characteristics, each of these traits independently increases the likelihood that this fruit is an apple, which is why it is referred to as "Naive". An NB model is simple to construct and is especially beneficial for very sizable data sets. Along with being straightforward, Naive Bayes is known to perform better than even the most sophisticated categorization techniques.

The Naïve Bayes classifier has different modes. In this case multinomial naïve bayes. Multinomial naïve bayes are basically used for text data classification.

Performance Evaluation

The following performance evaluation matrices used for performance evaluation:

Accuracy: The percentage of accurate predictions is expressed using the metric of accuracy in classification problems. It can be calculated by dividing the number of correct predictions by the total number of predictions.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ Of\ Predictions}$$

$$=> Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

Where,

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

Precision: The ratio of true positives to all expected positives is known as precision.

$$Precision = \frac{TP}{TP+FP}$$

Recall: Recall is the proportion of real positives to all the positives in the ground truth.

$$Recall = \frac{TP}{TP+FN}$$

F1-score: The F1 Score represents the Harmonic Mean between Recall and Precision.

Applied different ML classifiers to compare the performance of the model with others.

ML Classifier	Accuracy	Precision	Recall	F1-score
Random Forest	95.28795	1.00	0.80	0.89
Adaboost Classifier	88.6561	0.68	0.95	0.79
k nearest Neighbors	91.2739	0.96	0.64	0.77
Support Vector Machine	95.7242	0.96	0.64	0.77
Naive bayes	98.6910	0.97	0.98	0.97

Table 1: Performance evaluation of different ML models.

This table illustrates the comparison of different machine learning models based on the performance evaluation of the spam email dataset. The Naïve Bayes classifiers outperformed the others overall. Nevertheless, the random forest classifier has the best precision matrix performance. But in this case, the F1-score is more important than the precision matrix.

6. Conclusions and Expansion

To conclude the project with the developed spam detector model, which can detect spam email by monitoring an email body, and the experimental implementation of a spambot, which can be used as a weapon for DoS attacks.

Attacks alert notification and minimize the attack vectors for email accounts can be the best scope for expansion of this project. We have to be more conscious of the security of emails because it is becoming a time requirement.

Bibliography

1. Cota, Rodica Paula, and Daniel Zinca. "Comparative results of spam email detection using machine learning algorithms." *2022 14th International Conference on Communications (COMM)*. IEEE, 2022.
2. Thakur, Prazwal, et al. "Detection of Email Spam using Machine Learning Algorithms: A Comparative Study." *2022 8th International Conference on Signal Processing and Communication (ICSC)*. IEEE, 2022.

3. Lanka, Sai Charan, et al. "Spam based Email Identification and Detection using Machine Learning Techniques." *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*. IEEE, 2023.

4. Khan, Wazir Zada, et al. "A comprehensive study of email spam botnet detection." *IEEE Communications Surveys & Tutorials* 17.4 (2015): 2271-2295.

5. Schneider, Markus, Haya Shulman, and Michael Waidner. "Blocking email bombs with email glass." *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 2020.

6. Herzberg, Amir. "DNS-based email sender authentication mechanisms: A critical review." *Computers & security* 28.8 (2009): 731-742.

7. Herzberg, Amir, and Haya Shulman. "DNS authentication as a service: preventing amplification attacks." *Proceedings of the 30th Annual Computer Security Applications Conference*. 2014.

8. Mansoor, R. A. Z. A., Nathali Dilshani Jayasinghe, and Muhana Magboul Ali Muslam. "A comprehensive review on email spam classification using machine learning algorithms." *2021 International Conference on Information Networking (ICOIN)*. IEEE, 2021.

9. Sen, Jaydip. "A robust mechanism for defending distributed denial of service attacks on web servers." *arXiv preprint arXiv:1103.3333* (2011).

10. “Spam Email Dataset.” *Www.kaggle.com*, www.kaggle.com/datasets/jackksoncsie/spam-email-dataset. Accessed 8 Dec. 2023.