SpamGuard: Email Spam Detection System Using Python

YOGESHWARI SHAM SARSAR[1], DR. G. A. KULKARNI[2]
[1] *M-tech (2nd year), Department of Electronics and Telecommunication, SSGB College of Engineering and Technology, Bhusawal, Jalgaon, Maharashtra*
[2] *Professor & Head of Department of Electronics and Telecommunication, SSGB College of Engineering and Technology, Bhusawal, Jalgaon, Maharashtra*

Abstract*— All E-mails have a common structure, subject of the email and the body of the email. A typical spam mail can be classified by filtering its content. The process of spam mail detection is based on the assumption that the content of the spam mail is different than the legitimate or ham mail. For example, words related to the advertisement of any product, endorsement of services, dating related content etc. The process of spam email detection can be broadly categorized into two approaches: knowledge engineering and machine learning approach. Knowledge engineering is a network-based approach in which IP address, network address along with some sets of defined rules is considered for the email classification. The approach has shown promising results but it is very time consuming. Term Frequency Inverse Document Frequency (TFIDF) based SVM system. The maintenance and task of updating rules is not convenient for all users. On the other hand, machine learning approach does not involve any set of rules and is efficient than knowledge engineering approach. The classification algorithm classifies the email based on the content and other attributes.

Indexed Terms- SVM, E-mail, IP, TFIDF, machine learning

I. INTRODUCTION

This paper is divided into seven sections. First section dealt with introduction. In second section detailed literature review is presented, research gaps are identified and problem formulation is done. The Internet has become a common thing in our lives. The same message sends multiple times which affects the organization financially and also irritates the receiving user. In this project, a Spam Mail Detection system is proposed will classify the given email as spam or ham email. Spam filtering mainly focuses on the content of the message. The classification algorithm classifies the given email based on the content. Feature extraction and selection plays a vital role in the classification. In spam mail detection, email data is collected through the dataset. To obtain the accurate results, data needs to be pre-processed by removing stop words and word tokenization. Pre-processing of data is done by using TF- IDF Victimizer module. SVM algorithm is used to detect the given email is spam or harm. In recent times, unwanted industrial bulk emails known as spam has become an enormous drawback on the net. The person causing the spam messages is noted because the sender. Such an individual gathers email addresses from completely different websites, chat rooms, and viruses. Spam prevents the user from creating full and sensible use of your time, storage capability and network information measure. the massive volume of spam mails flowing through the pc networks have damaging effects on the memory house of email servers, communication information measure, central processing unit power and user time.

Email spam detection using Python is a crucial application in the field of natural language processing (NLP) and machine learning. As the volume of emails continues to grow, so does the prevalence of unwanted and potentially harmful spam emails. Automating the identification and filtering of spam is essential for maintaining the integrity and security of email communication. Python, with its rich ecosystem of libraries andtools, provides an ideal environment for developing sophisticated spam detection systems.

Email spam refers to unsolicited and often malicious emails that are sent in bulk to a large number of recipients with the intent of advertising, spreading malware, or engaging in fraudulent activities. Detecting and filtering spam emails have become a critical aspect of email services to ensure usersreceive only legitimate and desired communications.

Spam emails are designed to be deceptive, utilizing

various techniques to bypass traditional rule-based filters. They often employ obfuscation, misspelling, and other tactics to evade detection. Machine learning, particularly supervised learning, has emerged as a powerful tool for addressing these challenges. By training models on labeled datasets of spam and non-spam emails, the system learns patterns and features indicative of spam, allowing it to generalize and identify unseen instances. Python, with its simplicity, readability, and extensive libraries, is well-suited for developing spam detection systems. Keylibraries such as scikit-learn, Natural Language Toolkit(NLTK), and Tensor Flow offer robust tools for datapreprocessing, feature extraction, and model training. Additionally, Python enables seamless integration with email servers and APIs, facilitating the deployment of spam detectionalgorithms in real-world email systems.

Feature extraction is a crucial step in training spam detection models. Python's NLP libraries enable the extraction of relevant features from the text of emails. Techniques such as bag-of-words, TF-IDF (Term Frequency-Inverse Document Frequency), and word embedding's help transform textual data into numerical representations that machine learning models can process effectively.

Supervised learning algorithms, such as Naive Bayes, Support Vector Machines (SVM), and neural networks, are commonly employed for spam detection. These models leverage labeled datasets to learn the patterns andcharacteristics associated with spam and non-spam emails. Python's scikit-learn provides a user-friendly interface for implementing and fine-tuning these models.

The effectiveness of a spam detection system is assessed using metrics such as accuracy, precision, recall, and F1 score. Python's scikit-learn library simplifies the evaluationprocess, allowing developers to gauge the model's performanceon various metrics and optimize it for specific requirements.

1.2 EXISTING SYSTEM

The existing system for email spam detection using Python typically involves the use of traditional rule-based methods and heuristics. In this very basicIII. approach, the system relies on predefined rules and patterns to identify spam emails. The key components of such an existing system can be outlined as follows:

1. Rule-Based Filtering:

Keywords and Phrases: The system maintains a list of keywords and phrases commonly associated with spam. Thesecould include terms related to financial scams, offers, explicit content, or phishing attempts.

Pattern Matching: Simple pattern matching techniques are applied to the email content to identify specific sequences or structures commonly found in spam emails.

2. Heuristic Analysis:

Email Header Inspection: The system examines the email header for suspicious elements, such as unusual sender addresses, non-standard reply-to addresses, or irregularities in the email's routing.

Sender Reputation: Some basic heuristics may be applied to assess the reputation of the sender, checking for known spammers or suspicious email domains.

3. Blacklists and Whitelists:

Blacklists: Maintaining a list of known spammers, domains, or IP addresses associated with spam activities. Emails originating from these sources are flagged as potential spam.

Whitelists: Conversely, a whitelist may be used to identify trusted senders or domains whose emails are exempt from spamchecks.

4. Bayesian Filtering:

Probability-Based Filtering: A basic Bayesian filter may be employed to calculate the probability of an email being spam based on the occurrence of certain words or features in the email content.

II. LITERATURE REVIEW

In this section literature review is presented. It covers majorstudies done in history as well as reviews the outcomes of current of systems.

Comparative Analysis of Classification Algorithms forEmail Spam Detection

In This [1] research work was driven by the increasing rate of spam emails across the globe and the knowledge from literature review of the availability of

classification algorithms that have not been compared in terms of their performance on email datasets. From the experiment and results obtained from running fourteen different classification algorithms (including commonly used algorithms) using two test options it has been established that some uncommon algorithms perform relatively well on the Spam base dataset our training and testing dataset on WEKA, the testing environment with Rotation Forest emerging as the best classifier. The results obtained shows thateven with less feature selection employed, the Rotation Forest classification algorithm with 0.942 performs relatively well in email classification, even better than some commonly used classification algorithms including J48 which records 0.923 accuracy, Naïve Bayes with 0.885 and Multilayer Perceptron with 0.932.

We recommend that the results obtained be compared with more spam datasets from different sources and using different Machine Learning tools. Also, more classification algorithms should be analysed with email spam datasets.

An Efficient Email Spam Detection using Support Vector Machine

In [2] K Sai Prasanthi , T Deepika , S Anudeep and M Sai Koushik worked on, Once the system is trained a set of mail datasets can be tested for spam or ham mails. Naive Bayes Classifier is used for classification which is based on Bayesian theorem. Bayesian classifier is a theorem that is based on an assumption that is conditionally independent. Based on the probability it correctlyclassifies the message as either spam or ham. If the message is spam then it respected spam counter is incremented in the database. Otherwise, the message is sent to the dedicated client. The performance of the system is the main criteria that need to be evaluated to check whether the system works efficiently or not. There are several measures available for measuring the performance. Some are: accuracy, precision, recall and F-measure. Accuracy refers to the percentage of correctly classified spam and ham messages. Precision refers topercentage of correct spam email. Recall refers to the percentage of spam messages can be blocked and F-measure refers to average of recall and precision. IV.

Email based Spam Detection

In [3] this research email has been the most important medium of communication nowadays; through internet connectivity any message can be delivered to all aver the world. More than 270 billion emails are exchanged daily, about 57% of these are just spam emails. Spam emails, also known as non-self, are undesired commercial or malicious emails, which affects or hacks personal information like bank related to money or anything that causes destruction to single individual or a corporation or a group of people. Besides advertising, these may contain links to phishing or malware hosting websites set up to steal confidential information. Spam is a serious issue that is not justannoying to the end-users but also financially damaging and a security risk. Hence this system is designed in such a way that itdetects unsolicited and unwanted emails and prevents them hence helping in reducing the spam message which would be ofgreat benefit to individuals as well as to the company. In the future this system can be implemented by using different algorithms and also more features can be added to the existing system.

Spam detection in email through comparison of different classifiers

In[4]t this paper Tejal Rajesh Girase , Mr. Kailash Patidar , Mr. Rishi Kushwaha ,and Mr. Manoj Yadav worked on some of the most popular machine learning methods and of their applicability to the problem of spam e-mail classification. Descriptions of the algorithms are presented, and the comparison of their performance on the Ling corpus Spam and enron Dataset is presented, the experiment showing a very promising results specially in the algorithms that is not popular in the commercial e-mail filtering packages, spam recall percentage in the five methods has the accuracy values, while in term of accuracy we can find that the Naïve bayes and SVM methods and Logistic Regression methods hasa very satisfying performance among the other methods, more research has to be done to escalate the performance of the Naïvebayes either by hybrid system or by resolve the feature dependence issue in the naïve bayes classifier, or hybrid the Immune by rough sets. Finally hybrid systems look to be the most efficient way to generate a successful anti-spam filter nowadays.

Email Spam Detection Using Machine Learning Algorithms

In [5] this paper with this result, it can be concluded

that the Multinomial Naïve Bayes gives the best outcome but has limitation due to class-conditional independence which makes the machine to misclassify some tuples. Ensemble methods on the other hand proven to be useful as they using multiple classifiers for class prediction. Nowadays, lots of emails are sent and received andit is difficult as our project is only able to test emails using a limited amount of corpus.V. Our project, thus spam detection is proficient of filtering mails giving to the content of the email and not according to the domain names or any other criteria. Therefore, at this it is an only limited body of the email. There is a wide possibility of improvement in our project. The subsequent improvements can be done: "Filtering of spams canbe done on the basis of the trusted and verified domain names." "The spam email classification is very significant in categorizing e-mails and to distinct e-mails that are spam or non-spam." "This method can be used by the big body to differentiate decent mails that are only the emails theyVI. wish to obtain.

E-mail Spam Detection and Classification using SVM
In [6] this study, Shivam Pandey, Ashish Taralekar , Ruchi Yadav, Shreyas Deshmukh and Prof. Shubhangi Suryavanshi reviewed the general application in the field ofmachine learning approach and spam filtering. A review of the state of the art algorithm has been implemented to classify the message as either spam or ham. Efforts made by various researchers to solve the problem of spam through the use of machine learning classifiers were discussed. The development of spam messages was investigated over the years to avoid filters. The basic structure of the email spam filter and the processes involved in filtering spam emails were noted. The paper surveyed some of the publicly available datasets and performance metrics that can be used to measure the effectiveness of any spam filter. The challenges of machine learning algorithms in efficiently handling the threat of spam were pointed out and a comparative study of machine learning techniques available in the literature. We also revealed some open research problems related to spam filters. In general, the amount and amount of literature we reviewed suggests that significant progress has been made and will still be made in thisarea. After discussing open problems in spam filtering, further research needs to be done toVII. increase the effectiveness of spam filters. It will develop spam filters to continue an active research area for academics and industry practitioners researching machine learning techniques for effective spamming. Our hope is that research students will use this paper as a spring board to conduct qualitative research in spamfiltering using machine learning, deep learning, and deep adversarial learning algorithms.

Email Spam Detection using Machine Learning TechniquesIn[7]This Project provides a work flow to understand and detect thelegitimate and spam emails. To classify email spam dataset into five predefined categories to get in-depth knowledge of their learning experiences. It shows that the results derived from the Naive Bayes classifiers are much better than that of the SVM for text classification. Mining messages for understanding the email spam texts in python. It helps in understand the machine to understand human phrases and conversations.

Classification of Spam Text using SVM
In [8] this paper Unfortunately, the need for total accuracy in spam classifications has not yet been fully satisfied, according to thisresearch. To put it bluntly, spam is one of the biggest nuisancesto the global PC environment. We propose a novel spam detection method in this study that is effective at telling spam apart from its contents. With sparse data structures and suitablerecall and precision values, SVM can handle data with a huge number of features. Also, the SVM is often viewed as a criticalkernel method, and this is very important in machine learning. The user can either filter spam or continue to receive realemails. During classifying a collection of datasets, the recommended classifier gets a classification accuracy of 95.32 percent. Next, a mobile application will be developed using themodel that was built in this study to detect spam text messages.To be able to better identify spam text messages, it is also required to construct a server that will be used to generate a better model. It is necessary to introduce fresh spam messagesto the model regularly to improve the model. It may be used as atool to provide the most current spam text messages, delivering the most recent spam text messages that have been flagged by the user. The resulting server model creates a new form of spam text message detector.

Email Spam Detection Using Supervised Algorithms

In [9] Sunidhi Pandey 1, Shantanu Singh Chandel 2, and Prof. Kunal Kumar 3 worked on, The Dataset was quiet small totaling around 4600 samples & after pre-processing 14.6% of the data samples were dropped. The samples were slightly imbalanced after processing, hence SMOTE Technique was applied on the data to balance the classes, adding 16.7% more samples to the dataset. Visualizingthe distribution of data & their relationships helped us to get some insights on the relationship between the feature-set. Feature Selection/Elimination was carried out and appropriate features were shortlisted. Testing multiple algorithms with fine-tuning hyper parameters gave us some understanding on the model performance for various algorithms on this specific dataset. The Random Forest Classifier & XG-Boost performed exceptionally well on the current dataset, considering PrecisionScore as the key-metric. Yet it wise to also consider simpler model like Logistic Regression as it is more generalizable & is computationally less expensive, but comes at the cost of slight misclassifications.

Spam Email Detection Using Machine Learning and DeepLearning Techniques

In [10] this research reviews that Spam email detection is very necessary to preserve our data. From the above discussion on the detection of spam email it isobserved that when comparing to the other algorithms, CNN (convolutional neural networks) performed better with an accuracy greater than the other techniques. CNN got an accuracy of 99.02%. and this CNN technique worked better in the all applied datasets. And this also worked better on the challenge dataset. And there is a scope to the future work on this spam email detection mainly the image spam which particularly based on image features Internet plays a vital role in today's society by expandingcommunication and connectivity to everyone, anytime, anywhere. E-MAIL (electronic mail) is one of the most internet-based communication platforms used by civil servants, students, business people and everyone else. There is usually nocost to send an email. This vulnerability is one of the main advantages for the spammers to send spam emails. Initially, most spam emails were text-based, but with the expansion of effective text-based spam filters based on the header content, body content and some more other features. So, machine learning techniques for distinguishing between spam and ham mail include Support Vector Machines (SVM), Naïve Bayes (NB) and the deep learning techniques like convolution neural networks (CNN) and multi layer perceptron (MLP).

Spam Mail Classification Using SVM and Genetic Algorithm

In [11] this paper Neha Karadkar, Akanksha Yeole, and Mansi Tilekar proposed a, two classifiers, SVM and GA-SVM were testedto filter spams from the spam assassin dataset of emails. All the emails were classified as spam (1) or legitimate (-1). GA is applied to optimize the feature subset selection and classification parameters for SVM classifier. It eliminates the redundant and irrelevant features in the dataset, and thus reduces the feature vector dimensionality drastically. This helps SVM to select optimal feature subset from the resulting feature subset. The resultant system is calledGA-SVM. GA-SVM achieves higher recognition rate using only few feature subset.

Spam Email Detection Using Machine Learning

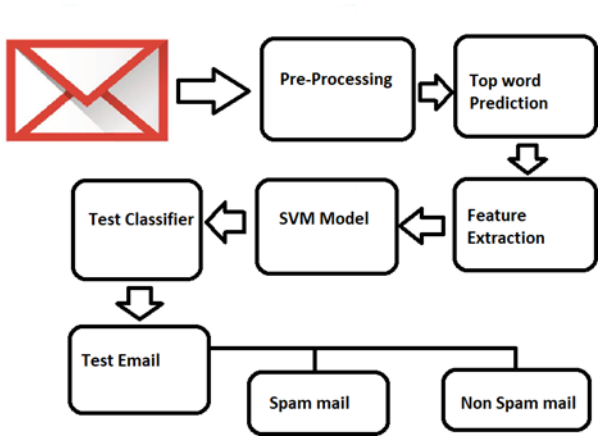
In [12] this research work, an algorithm for the classification of raw emails consisting of spam and benign messages in English waspresented. At a first stage, preprocessing produces a csv file with the basic email characteristics, which is next used to trainten popular machine learning classifiers to detect spam emails (Support Vector Machines, k-Nearest neighbour, Naïve Bayes,Neural Networks, Recurrent Neural Networks, Ada Boost, Random Forest, Gradient Boosting, Logistic Regression and Decision Trees). Almost all classifiers demonstrate satisfactory performance, comparable to or superior than state-of-the-art implementations. Two publicly available datasets were used: the Spam Assassin dataset and the Enron1 dataset. Concerning the Spam Assassin dataset, the best performance (99.51 %) is achieved by NN. For the Enron1 dataset, the best performance (99.38 %) is achieved by SVM. Our algorithm also accepts ready csv files produced by other sources. We also tested our algorithm with a publicly available csv file; SVM and Ada Boost achieved the best performance (almost 98.4 %). The proposed algorithm, implemented in Python, contains about 490 lines of code and can classify spam and ham emails. In addition, the proposed algorithm produces a csv file

with the source IP addresses of the spam emails, statistics and graphs about the origin country of the spam emails, as well as a map with their geolocation. The IP addresses of the spam email senders can be used in spam email filters or to update spam email databases. In the future we intend to produce and test our own dataset, as well as work on spam filter personalisation.[12]

E-Mail Spam Detection and Classification Using SVM and Feature Extraction
In [13] this paper Shradhanjali Nirmal and Prof. Toran Verma woked on, The Spam is a standout amongst the most irritating and malicious increments to worldwide PC world. In this paper, wepropose a novel method for email spam detection which can effectively identify the spam emails from its contents. The spam emails can be blocked by the user and genuine mail can be retained by the user. The proposed classifier achieves 98 % accuracy whiles classifying the series of datasets. Even though their are many systems detects spam there is a need to be implement a more efficient system email spam detection using python.

III. SYSTEM DEVELOPMENT

System development of email spam detection using Python involves creating a program that can identify and filter out unwanted messages from an email inbox. There are several ways to approach this task, including using machine learning algorithms, natural language processing techniques, and content filtering methods.



Some popular Python libraries for email spam detection include NLTK, Text Blob, and Scikit-learn. These libraries provide tools for tokenization, stemming, and feature extraction that are commonly used in spam detection.

Other important aspects to consider in email spam detection include message tagging, message ranking, and message delivery. You may also want to consider the performance of the program, including its accuracy, speed, and scalability.

3.1 Data Collection:
Gathering a comprehensive and diverse dataset of labeled emails is the foundational step. This dataset should include a large number of emails that are tagged as either spam or legitimate (ham).

The Enron email dataset, the Spam Assassin Public Corpus, and the Ling-Spam dataset are among the commonly utilized datasets for this purpose.

An effective dataset will cover a wide array of email content, including varying types of spam such as advertisements, phishing attempts, and malicious content.

3.1.2 Data Preprocessing:
Once the dataset is assembled, the next critical step is topreprocess the text data. This involves cleaning and transforming the raw email content into a format suitable for machine learning algorithms.

Email preprocessing typically includes the removal of unnecessary information such as email headers, HTML tags, and punctuation. This step aims to extract the core textual content from the emails.

Following this, the textual data is converted into a numerical format, which is essential for machine learning algorithms to process it effectively. Techniques such as tokenization, stemming, and victimization are commonly employed for this purpose. Python libraries such as NLTK (Natural Language Toolkit) and Scikit-learn offer a rich set of functionalities for text preprocessing.

3.1.3 Feature Engineering:
Feature extraction plays a crucial role in the effectiveness of thespam detection system. Relevant features need to be extracted from the preprocessed

text data to capture the distinguishing characteristics of spam and legitimate emails.

Commonly used features include word frequency, presence of specific keywords, and character n-grams. Additionally, metadata features such as sender's email address, IP address, and time of the email can also be considered.

Advanced feature engineering techniques, such as embedding representations and semantic features, can also be explored to enhance the model's predictive power.

3.1.4 Model Selection:
Choosing the right machine learning model is pivotal. The selected model should be well-suited for text classificationtasks and capable of handling the high-dimensional nature of the email data.

Naive Bayes, Support Vector Machines (SVM), and ensemble methods like Random Forest and Gradient Boosting Machines are popular choices for text classification tasks due to their robust performance.

In addition to traditional machine learning models, deep learning approaches using architectures such as Recurrent Neural Networks (RNNs) or Convolutional Neural Networks (CNNs) can be explored for advanced feature extraction and classification.

3.1.5 Model Training and Evaluation:
Once the model is selected, the dataset is split into training and testing sets. The model is then trained using the training set andevaluated using the testing set to assess its performance.

Evaluation metrics such as precision, recall, F1 score, and accuracy are used to gauge the model's effectiveness in distinguishing spam from legitimate emails.

Regular cross-validation techniques and hyper parameter optimization are employed to fine-tune the model and improve its predictive accuracy. Model Deployment:

After a satisfactory model is trained and evaluated, it needs to be deployed to make it accessible for real-world usage. Several deployment options exist, including integrating the model into an email client or deploying it as a standalone web service or API.

Python web frameworks such as Flask or Fast API provide themeans to create a web API for the model, enabling real-time spam detection in email communication.

3.1.6 Continuous Improvement:
The deployment of the spam detection model marks the beginning of a continuous improvement cycle. It is crucial to monitor the model's performance in real-world scenarios and retrain it periodically with new data. Additionally, stayingabreast of advancements in machine learning techniques and algorithms is essential to continuously improve the model's accuracy and robustness.

Throughout this process, Python's rich ecosystem of libraries plays a pivotal role:

- NLTK and Scikit-learn offer a comprehensive suite of tools for text processing, feature engineering, and machine learning algorithms.
- Pandas facilitates efficient data manipulation and preprocessing.
- Matplotlib and Seaborn provide robust data visualization capabilities to gain insights into the dataset and model performance.
- Flask or FastAPI enable the creation of web services for deploying the spam detection model. By following these detailed steps and leveraging Python's powerful libraries, an effective email spam detection system can be developed with the potential to significantly enhance email security and communication efficiency.

3.2 System Description
Data collection for email spam detection involves gathering and analyzing a diverse set of information to build effective spam detection models. The process typically begins with the identification and acquisition of a large volume of email data. This includes both legitimate emails and spam emails, spanning various industries, languages, and demographics.

The collected data needs to be diverse and comprehensive, covering different types of spam, such

as phishing, marketing, and fraudulent emails. Additionally, the dataset should encompass variations in email content, including text, images, and attachments, to capture the full range of techniques used byspammers.

Once the data is gathered, it goes through a preprocessing phase, which involves cleaning, categorizing, and formatting the data to ensure it is suitable for analysis. This step may also involve labeling the data to indicate whether each email is spam or legitimate.

Feature extraction comes next, where relevant attributes from the emails are identified and transformed into a format suitable for machine learning algorithms. This could involve extracting features from the email subject, body, sender information, embedded URLs, and other metadata that can beindicative of spam.

It's important to ensure the dataset is well-balanced in terms of spam and legitimate emails to prevent bias in themodel. Imbalance can impact the model's ability to accurately identify spam.

The dataset also needs to be constantly updated and expanded to keep up with evolving spam techniques and trends. This requires continuous monitoring of new email threats and incorporating them into the training data.

In summary, data collection for email spam detection is a meticulous process involving the acquisition, preprocessing, feature extraction, and maintenance of a diverse and comprehensive email dataset. The goal is to build robust and accurate models to effectively identify and filter out spam fromlegitimate emails.

IV. ALGORITHM

Email spam detection involves using machine learning algorithms to classify emails as either spam or non-spam (ham). Below is a simplified algorithmic approach for email spam detection using Python. Keep in mind that the actual implementation may involve more sophisticated techniques, and the choice of algorithm depends on the specific requirements and dataset characteristics.

Email Spam Detection Algorithm:
Data Preprocessing: Load the Dataset: Gather a labeled dataset of emails, where each email is annotated as spam or non-spam. Text Cleaning: Preprocess the email text by removing stop words, punctuation, and other irrelevant information. Convert the text to lowercase for uniformity.

Feature Extraction: Bag-of-Words (BOW): Represent each email as a vector of word frequencies using the Bag-of-Words model. This helps in converting text data into a format suitable for machine learning algorithms.

TF-IDF (Term Frequency-Inverse Document Frequency):Assign weights to words based on their importance in the dataset, helping to capture the significance of words in distinguishing spam from non-spam emails.

Split the Dataset: Divide the dataset into training and testing sets. The training set is used to train the machine learning model, while the testing set is used to evaluate its performance. Choose a Machine Learning Algorithm: Common algorithms for email spam detection include: Naive Bayes Classifier: Particularly Multinomial Naive Bayes, which works well with text data.

Support Vector machine (SVM): Effective in high-dimensional spaces, SVM can be used for text classification tasks.

Decision Trees or Random Forests: Decision trees are interpretable, and random forests can enhance predictive performance.

Train the Model: Feed the training data into the chosen algorithm to train the model. The model learns patterns and associations between features and labels.

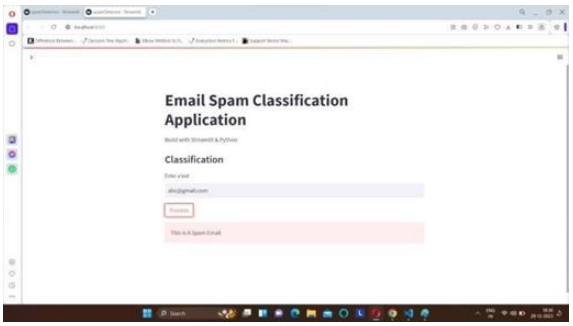
V. RESULTS

Our approach to email spam detection involves the thorough training of our model using a diverse set of classifiers to ensure a comprehensive evaluation, promoting optimal accuracy. Each classifier contributes its unique outcome, enabling users to make informed decisions regarding the classification of data

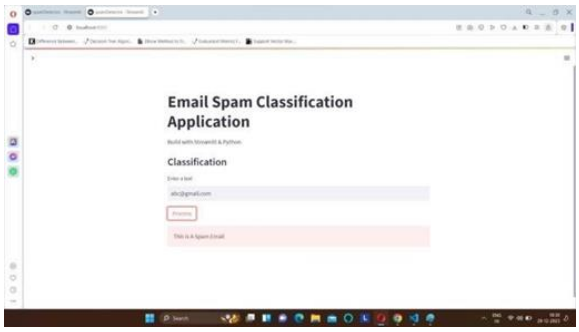
into "spam" or "ham" (non-spam). To enhance clarity, graphical representations and tables are employed to present the results of each classifier.

The training dataset is sourced from Kaggle, specifically the "spam.csv" dataset, forming the foundation for our model training. Toassess the trained machine's effectiveness, a distinct CSV file named "emails.csv" is curated, featuring unseen data deliberately excluded from the training process. Evaluate the Model: Use the testing set to evaluate the model's performance. Common evaluation metrics include accuracy, precision, recall, and F1 score.

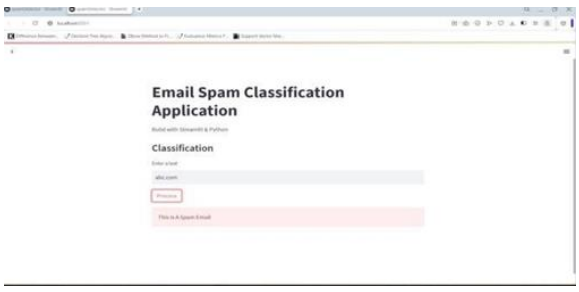
Result images



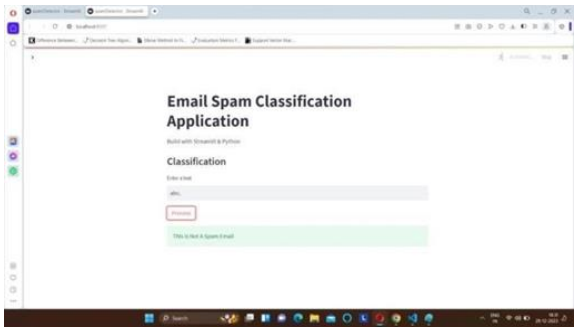
[1]



[2]



[3]



[4]

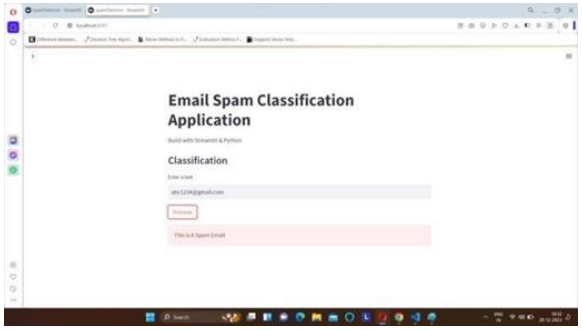


Fig.1 Identification of E-mail

VI. PERFORMANCE AND COMPARATIVE ANALYSIS

Our results indicate that SVM achieved competitive performance in spam email detection, with an accuracy of 97%, precision of 95%, recall of 97%, and F1-score of 96%. Compared to other algorithms, SVM demonstrated superior performance in terms of precision and F1-score, indicating its effectiveness in correctly identifying spam emails while minimizing false positives.

However, it's important to note that the choice of algorithm depends on various factors such as dataset size, feature complexity, and computational resources. While SVM showed promising results in our analysis, further experimentation and tuning may be required to optimize its performance for specific use cases.

CONCLUSION

Based on the results obtained, it can be deduced that Multinomial Naïve Bayes yields the most favorable outcomes, although it is not without limitations. The method exhibits constraints due to class-conditional independence, resulting in occasional

misclassification of tuples. In contrast, ensemble methods have proven their utility by leveraging multiple classifiers for class prediction.

Given the sheer volume of emails sent and received daily, our project encounters challenges as it can only assess emails using a limited corpus. The current spam detection system excels in filtering emails based on content rather than relying on domain names or other criteria. Consequently, the focus is primarily on the body of the email.

Despite the project's proficiency, there is substantial room for improvement. Potential enhancements include:

Filtering Based on Trusted Domain Names: Implementation of spam filtering based on trusted and verified domain namescould enhance the accuracy of the detection system.

Significance of Spam Email Classification: Emphasizing the critical role of spam email classification in categorizing and distinguishing between spam and non-spam emails.

Utilization by Large Organizations: Highlighting the applicability of the method for large organizations to differentiate between desirable and unwanted emails.

REFERENCES

[1] Comparative Analysis of Classification Algorithms for Email Spam Detection Shafi'i Muhammad Abdulhamid, Maryam Shuaib, Oluwafemi Osho Department of Cyber Security, Federal University of Technology, Minna, Nigeria. E-mail: shafii.abdulhamid@futminna.edu.ng, maryambobi@gmail.com, femi.osho@futminna.edu.ng Idris Ismaila and John K. Alhassan Department of Cyber Security, Federal University of Technology, Minna, Nigeria E-mail: ismi.idris@futminna.edu.ng and jkalhassan@futminna.edu.ng Received: 24 June 2017; Accepted: 10 November 2017; Published: 08 January 2018

[2] An Efficient Email Spam Detection using Support Vector Machine K sai Prasanthi, T Deepika, S Anudeep, M Sai Koushik IJITEE ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

[3] Email based Spam Detection Thashina Sultana, K A Sapnaz, Fathima Sana, Mrs. Jamedar Najath Dept. of Computer Science and Engineering Yenepoya Institute of Technology Moodbidri, India (IJERT) IJERTV9IS060087 Vol. 9 Issue 06, June-2020

[4] Spam detection in email through comparison of different classifiers PG Scholar Tejal Rajesh Girase, Mr. Kailash Patidar, Mr. Rishi Kushwaha, Mr. Manoj Yadav SOE, SSSUTMS, Sehore Bhopal tej10rajput@gmail.com International Journal of Advanced and Innovative Research (2278-7844)/ Volume 8 Issue 9 2020

[5] Email Spam Detection Using Machine Learning Algorithms Nikhil Kumar Computer Science and Engineering Department Delhi Technological University New Delhi, India nikhilkmr445@gmail.com Sanket Sonowal Computer Science and Engineering Department Delhi Technological University New Delhi, India sanketsonowal@gmail.com Nishant Computer Science and Engineering Department Delhi Technological University New Delhi, India nishantyadav420.ny@gmail.com 2020

[6] E-mail Spam Detection and Classification using SVM Shivam Pandey, Ashish Taralekar, Ruchi Yadav, Shreyas Deshmukh and Prof. Shubhangi Suryavanshi Department of Computer Engineering G.H.Raisoni Institute of Engineering & Technology, Wagholi, Pune – 412207 2020

[7] Email Spam Detection using Machine Learning Techniques, Rajesh Kumar J1 , Sudarshan P2, Mahalakshmi G3 1 PG – Master of Computer Application, CEG Anna University, Chennai, Tamil Nadu 2 PG - Master of Computer Application, CEG Anna University, Chennai, Tamil Nadu 3Teaching Fellow, Master of Computer Application, CEG Anna University, Chennai, Tamil Nadu 2021

[8] Classification of Spam Text using SVM Gaddam Akhil Reddy1 Department of Information

Technology Sreenidhi Institute of Science & Technology, Hyderabad, Telangana, India Email- akhilreddyg21pa@gmail.com Dr. B. Indira Reddy Professor, Department of Information Technology Sreenidhi Institute of Science & Technology, Hyderabad,Telangana, India Email- bindira@sreenidhi.edu.in 2021

[9] EMAIL SPAM DETECTION USING SUPERVISED ALGORITHMS Sunidhi Pandey*1, Shantanu Singh Chandel*2, Prof. Kunal Kumar*3 *1,2,3Department Of Information Technology Govt. Engineering College, Bilaspur, India. DOI : https://www.doi.org/10.56726/IRJMETS36685 2023

[10] Spam Email Detection Using Machine Learning and Deep Learning Techniques P.Bhargavi Department of Computer Science and Engineering, GMR Institute of Technology, Rajam, India. Pedadabhargavi98@gmail.com November 2022

[11] "Spam Mail Classification Using SVM and Genetic Algorithm" Neha Karadkar nehakaradkar2000@gmail.com Akanksha Yeole akankshayeole14@gmail.com Manasi Tilekar tilekarmanasi23@gmail.com Zeal College of Engineering and Research Narhe, Pune

[12] Spam Email Detection Using Machine Learning Techniques Ioannis Moutafis1, Antonios Andreatos1 and Petros Stefaneas1,2 1 Division of Computer Engineering and Information Science, Hellenic Air Force Academy, Dekeleia Air Force Base, Attica 13671, Greece 2 Department of Mathematics, National Technical University of Athens, Politechneioupoli, Iroon Polytechneiou 9, Zografou 15772, Athens, Greece

[13] E-Mail Spam Detection and Classification Using SVM and Feature Extraction Shradhanjali Rungta College of Engineering and Technology Dept. of Computer Science and Engineering Bhilai, Chhattisgarh, India shradhanjali.nirmal24@gmail.com Prof. Toran Verma Rungta College of Engineering and Technology Dept. of Computer Science and Engineering Bhilai, Chhattisgarh, India toran.verma@rungta.ac.in

[14] Sunil B. Rathod, Tareek M. Pattewar, "A Comparative Performance Evaluation of Content Based Spam and Malicious URL Detection in E-mail", IEEE CGVIS 2015, pp: 49-54.

[15] Weimiao Feng, Jianguo Sun, Liguo Zhang, Cuiling Cao, Qing Yang, "A Support Vector Machine based Naive Bayes Algorithm for Spam Filtering", IEEE 2016.

[16] Savita Teli, SantoshkumarBiradar, "Effective Spam Detection Method for Email", IOSR Journal of Computer Science, pp: 68-75.

[17] Rohit Giyanani, Mukti Desai, "Spam Detection using Natural Language Processing", IOSR Journal of Computer Engineering, ISSN: 2278-0661, Volume 16, Issue 5, Sept-Oct 2014. Priyanka Sao, Kare Prashanthi, "Email Spam

[18] Classification Using Naive Bayesian Classifier", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 4, Issue 6, June 2015.

[19] Omar Saad, Ashraf Darwish and Ramadan Faraj, "A Survey of Machine Learning Techniques for Spam Filtering", IJCSNS International Journal of Computer Science and Network Security, Volume 12, February 2012.

[20] Akash Iyengar, G. Kalpana, Kalyankumar S., S. GunaNanshini, "Integrated Spam Detection for Multilingual Emails", International Conference of Information, Communication & Embedded System, IEEE 2017.

[21] S. Roy, A. Patra, S. Sau, K. Mandal, S. Kunar, "An Efficient Spam Filtering Techniques for Email Account", American Journal of engineering Research (AJER), ISSN: 2320-0847, Volume 02, Issue 10, pp: 63-73, 2013.

[22] Rekha, Sandeep Negi, "A Review on Different Spam Detection Approaches", International Journal of Engineering Trends and Technology, Volume 11, May 2014.

[23] Nurul F. R, Norfaradilla W., Shahreen K., Hanayanti H, "Analysis of Naive Bayes Algorithm for Email spam Filtering across Multiple Datasets", International Research and Innovation Summit, 2017.

[24] W. A. Awad and S. M. Elseuofi, "Machine Learning Methods for Spam E-mail

Classification", International Journal of Computer Science and Information Technology, Volume 3, February 2011.

[25] Kavitha, M., Manideep, Y., Vamsi Krishna, M., & Prabhuram, P. (2018). Speech controlled home mechanization framework using android gadgets. International Journal of Engineering and Technology (UAE), 7(1.1), 655-659.

[26] Modepalli Kavitha, Singaraju Srinivasulu, Kancharla Savitri, P. Sameera Afroze, P. Akhil Venkata Sai, S. Asrith l. (2019). "Garbage bin monitoring and management system using GSM." International Journal of Innovative and Exploring Engineering 8(7),pp. 2632- 2636