**EMAIL SPAM CLASSIFICATION**

A Report submitted in fulfilment of two-week internship program

By

DASARI RAMYA SREE

21955A0416

Chief mentor: Padmaja

Designation: Associate Professor

Co-Ordinator: Indu

Designation:



CAREER DEVELOPMENT CENTRE

INSTITUTE OF AERONAUTICAL ENGINEERING

Dundigal, Hyderabad, Telangana 50004

MAY 2023

**CERTIFICATE**

This is to certify that the project report entitled EMAIL SPAM CLASSIFICATION submitted by DASARI RAMYA SREE to the Institute of aeronautical engineering , Dundigal Telangana, in partial fulfilment for the two week internship programme is a bonafide record of project work carried out by her under our supervision.

**DECLARATION**

I declare that this project report titled EMAIL SPAM CLASSIFICATION submitted in partial fulfilment of the two week internship is a record of original work carried out by me under the supervision of

**ACKNOWLEDGEMENT**

**ABSTRACT**

Email is the most used source of official communication method for business purposes. The usage of the email continuously increases despite of other methods of communications. Automated management of emails is important in the today’s context as the volume of emails grows day by day. Out of the total emails, more than 55 percent is identified as spam. This shows that these spams consume email user time and resources generating no useful output. The spammers use developed and creative methods in order to fulfil their criminal activities using spam emails, Therefore, it is vital to understand different spam email classification techniques and their mechanism. This paper mainly focuses on the spam classification approached using machine learning algorithms. Furthermore, this study provides a comprehensive analysis and review of research done on different machine learning techniques and email features used in different Machine Learning approaches. Also provides future research directions and the challenges in the spam classification field that can be useful for future researchers. Index Terms—Spam Detection, Spam Classification, Spam Filter, E-mail, Supervised Learning, Machine Learning Algorithms, Email Classification, Spam Email Detection, Email Categorization, Email Feature Set Analysis, Spam Detection Using Machine Learning Algorithms.

**A.Brief Introduction**

Email is the easiest way to communicate worldwide today. To get e-mail junk mail, the preceding set of rules compares every e-mail message with junk mail records earlier than generating receivers. In the project, an electronic mail acquisition machine primarily based totally on SVM development has been advocated and more NB is used in the proposed system. With word spreads, the length of the word meaning, the proportion of word stops can also be seen. To highlight the electronic mail junk mail trend, a singular version that enhances the arbitrary production of the detector of SVM and NB algorithmusing both spam and non-spam spaces. Theater analysis and test results show that the performance detection of the enhanced SVM and NB is higher.

**B.Existing system**

The existing systems for email spam classification typically employ a combination of rule-based approaches, machine learning techniques, and heuristics to identify and filter out spam emails. Some common techniques used in the existing systems are:

1.Rule-based Filtering: These systems use predefined rules and patterns to identify spam emails. Rules may include checking for specific keywords, phrases, or patterns commonly found in spam emails. While rule-based filters can be effective in catching known spam patterns, they may struggle with new or evolving spam techniques.

2.Bayesian Filtering: Bayesian spam filters use probabilistic models to classify emails as spam or legitimate. These filters calculate the probability of an email being spam based on the occurrence of certain words or patterns. Bayesian filters learn from labeled examples and adjust their model based on the presence or absence of specific words in spam and legitimate emails.

3.Machine Learning Approaches: Supervised machine learning algorithms, such as Naive Bayes, Support Vector Machines (SVM), or Random Forests, are used to classify emails based on labeled training data. These algorithms learn from features extracted from emails (e.g., word frequencies, presence of specific patterns) and their corresponding labels to build a predictive model. The model is then used to classify new emails as spam or legitimate.

4.Collaborative Filtering: Collaborative filtering techniques leverage the collective knowledge of a community to classify emails. These systems analyze user feedback (e.g., marking emails as spam or not spam) to identify common patterns and characteristics of spam emails. The identified patterns are then used to classify emails for other users in the community.

5.Blacklisting and Whitelisting: Blacklisting involves maintaining a list of known spam sources or email addresses associated with spamming activities. Emails from these sources are automatically classified as spam. Whitelisting, on the other hand, includes maintaining a list of trusted email sources or addresses, ensuring that emails from these sources are always considered legitimate.

**C.Proposed system**

The proposed system for email spam classification aims to leverage machine learning techniques and advanced features to develop an accurate and efficient email spam classifier. The system will overcome limitations of existing systems by incorporating the following components and techniques:

1.Advanced Feature Extraction: The proposed system will employ advanced feature extraction techniques to capture more nuanced characteristics of spam emails. This can include the use of natural language processing (NLP) techniques to analyze the semantic meaning of email content, sentiment analysis to identify manipulative language or persuasive tactics commonly used in spam, and feature engineering to identify patterns beyond simple keyword matching.

2.Ensemble Learning: Ensemble learning techniques, such as combining multiple classifiers or models, will be utilized to improve the overall performance and robustness of the system. This can include methods like stacking, boosting, or bagging, which aggregate predictions from multiple models to make final classifications. Ensemble methods can help mitigate the impact of individual model weaknesses and provide more accurate and reliable spam classification.

3.Deep Learning Approaches: The proposed system will explore the use of deep learning algorithms, such as recurrent neural networks (RNNs) or convolutional neural networks (CNNs), to capture complex patterns and dependencies in email data. Deep learning models have demonstrated strong performance in various natural language processing tasks and may offer improved spam classification accuracy.

4.Active Learning and Feedback Loop: The system will incorporate active learning techniques to continuously improve the classifier's performance. Active learning enables the system to select the most informative or uncertain instances for manual labeling, allowing the model to learn from the newly labeled data and adapt to evolving spamming techniques. Additionally, user feedback loops can be incorporated to incorporate user-specific preferences and adjust the classifier's decisions accordingly.

5.Real-Time Updates: The proposed system will be designed to receive regular updates to adapt to new spamming techniques and emerging threats. This can involve regular model retraining using updated datasets, integration with external threat intelligence sources, and continuous monitoring of email traffic to identify new patterns or trends in spam emails.

6.Evaluation and Performance Metrics: The system will be rigorously evaluated using appropriate performance metrics such as accuracy, precision, recall, and F1-score. The evaluation will be conducted on labeled test datasets, comparing the system's performance against existing state-of-the-art solutions. This will ensure the effectiveness, efficiency, and reliability of the proposed system.

**TABLE OF CONTENTS**

**ACRONYMS**

**1.Introduction:**

Email spam classification is an essential task in today's digital world, where spam emails continue to inundate our inboxes, causing inconvenience, waste of time, and potential security risks. The need to accurately classify emails as spam or legitimate is crucial for efficient email management, protecting users from phishing attempts, and ensuring a seamless communication experience. In this report, we will delve into the motivation behind email spam classification, define the problem statement, and outline the organization of the report.

**1.1Motivation:**

The ever-increasing volume of spam emails poses a significant challenge to email service providers and individual users. The motivation behind email spam classification lies in the desire to mitigate the negative effects of spam, such as wasted time, decreased productivity, and potential exposure to malicious content. By accurately identifying and filtering spam emails, users can focus on legitimate communications, maintain a clean inbox, and reduce the risk of falling victim to scams or phishing attacks.

**1.2Problem Statement**:

The problem statement of email spam classification revolves around developing effective techniques and algorithms to automatically differentiate between spam and legitimate emails. The main challenge lies in distinguishing malicious or unwanted messages from genuine ones, considering the constantly evolving tactics employed by spammers. The classification process involves analyzing various features of an email, such as sender information, subject line, content, attachments, and embedded links, to make an informed decision about its classification.

**1.3Report Organization**:

This report is structured to provide a comprehensive understanding of email spam classification, including its motivation, problem statement, and key aspects related to its implementation. The organization of the report is as follows:

1. Introduction: This section provides an overview of the report, outlining the motivation, problem statement, and report organization.
2. Background: Here, we delve into the background information on email spam, discussing its prevalence, impact, and the need for effective classification techniques.
3. Literature Review: This section explores the existing research and techniques employed in email spam classification. We review various approaches, such as rule-based methods, machine learning algorithms, and hybrid models, highlighting their strengths and limitations.
4. Data Collection and Preprocessing: In this part, we discuss the process of gathering and preprocessing email data for classification tasks. We explore techniques for feature extraction, data cleaning, and sampling strategies.
5. Feature Selection and Extraction: Here, we focus on identifying relevant features from email data that contribute to effective spam classification. We discuss various techniques, such as content-based analysis, header analysis, and sender reputation scoring.
6. Classification Algorithms: This section examines different classification algorithms commonly used for email spam classification, including Naive Bayes, Support Vector Machines (SVM), Decision Trees, and Neural Networks. We evaluate their performance, scalability, and suitability for real-time classification.
7. Evaluation Metrics: We discuss the evaluation metrics used to measure the performance of email spam classification models, such as accuracy, precision, recall, and F1 score. We also address the challenges of imbalanced datasets and cross-validation techniques.
8. Experimental Results: This part presents the results and analysis of experiments conducted on various email spam classification models. We compare the performance of different algorithms and discuss their strengths and weaknesses.
9. Conclusion and Future Work: Finally, we conclude the report by summarizing the key findings, highlighting the importance of email spam classification, and suggesting potential areas for future research and improvement.

Top of Form

Bottom of Form

**2.Related Work:**

In the section on related work, we explore the existing methods and approaches employed in email spam classification. We review relevant research papers, studies, and industry practices to understand the advancements in this field. The related work is divided into two subsections: existing methods and proposed methods.

1. Existing Methods: This subsection focuses on the traditional approaches and techniques used for email spam classification. It includes rule-based methods, keyword-based filtering, blacklisting, and heuristic-based algorithms. We discuss their advantages, limitations, and the effectiveness of these methods in handling different types of spam.
2. Proposed Methods: In this subsection, we explore the more advanced and modern approaches for email spam classification. These may include machine learning algorithms, such as Naive Bayes, Support Vector Machines (SVM), Random Forest, or deep learning models like Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). We analyze the strengths and limitations of these methods and their potential for improving spam classification accuracy.

**System Design:**

The system design section outlines the technical aspects of implementing an email spam classification system. It includes the hardware and software components required for building such a system. The system design can be subdivided into two subsections: hardware and software.

1. Hardware: This subsection focuses on the hardware requirements for implementing an email spam classification system. It includes the necessary computational resources, such as servers, processors, memory, and storage, to handle the incoming email traffic and perform the classification tasks efficiently. Additionally, hardware considerations related to scalability, load balancing, and high availability are discussed.
2. Software: This subsection explores the software components and architecture needed for email spam classification. It includes the selection of programming languages, frameworks, and libraries for developing the classification algorithms. The software design may involve components such as data preprocessing, feature extraction, model training and evaluation, and real-time classification. Additionally, considerations regarding scalability, performance optimization, and integration with existing email systems are discussed.

By addressing the related work and system design aspects, this report provides a comprehensive overview of the existing methods employed in email spam classification and proposes potential advancements in the field. It also highlights the hardware and software requirements necessary for implementing an efficient and effective email spam classification system.

**2.1Existing Methods:**

This subsection discusses the traditional approaches used for email spam classification, such as rule-based filtering. Rule-based methods involve defining a set of predefined rules to identify spam emails based on specific keywords, patterns, or heuristics. We discuss the advantages and disadvantages of such methods, including their simplicity and potential limitations in adapting to evolving spam techniques.

1. Machine Learning Approaches: Here, we delve into the machine learning techniques applied to email spam classification. We explore popular algorithms like Naive Bayes, Support Vector Machines (SVM), Random Forest, and K-nearest neighbors (KNN). We discuss how these algorithms can be trained using labeled datasets and feature extraction techniques. Additionally, we highlight the challenges of feature selection, dimensionality reduction, and model evaluation in machine learning-based approaches.
2. Deep Learning Techniques: This subsection focuses on the application of deep learning methods in email spam classification. We explore the use of convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory (LSTM) networks for feature extraction and classification. We discuss the advantages of deep learning, such as automatic feature learning, and address the challenges of large-scale training and potential overfitting.

**2.2Proposed Methods:**

In the proposed methods section, we introduce novel approaches and techniques that can enhance the accuracy and effectiveness of email spam classification. These methods could be based on combining existing techniques or introducing innovative approaches.

1. Hybrid Models: This subsection explores the combination of multiple techniques to improve spam classification. For example, integrating rule-based filtering with machine learning or deep learning algorithms to leverage the strengths of both approaches. We discuss how hybrid models can enhance classification accuracy and adaptability to new spam patterns.
2. Ensemble Methods: Here, we discuss ensemble learning techniques for email spam classification. Ensemble methods involve combining predictions from multiple classifiers to make a final decision. We explore approaches like bagging, boosting, and stacking, and discuss their benefits in terms of reducing bias, improving generalization, and handling imbalanced datasets.
3. Feature Engineering: In this subsection, we address the importance of feature engineering in email spam classification. We explore techniques for extracting relevant features from email content, headers, and metadata. Additionally, we discuss the use of advanced features, such as text embeddings, semantic analysis, and email network analysis, to capture more nuanced spam patterns.

**2.3Hardware/Software:**

In the hardware/software section, we discuss the requirements and considerations for implementing email spam classification systems.

1. Hardware Requirements: Here, we outline the hardware resources necessary for running email spam classification algorithms effectively. We discuss considerations such as processing power, memory, and storage requirements, taking into account the scalability of the system as the volume of emails increases.
2. Software Frameworks: In this subsection, we explore the software frameworks and libraries commonly used for email spam classification. We discuss programming languages (Python, Java, etc.), machine learning frameworks (Scikit-learn, TensorFlow, PyTorch, etc.), and natural language processing tools (NLTK, spaCy, etc.) that facilitate the development and deployment of spam classification systems.
3. Scalability and Real-Time Processing: Here, we address the challenges of scalability and real-time processing in email spam classification. We discuss techniques such as distributed computing, parallel processing, and streaming data processing frameworks that enable efficient classification of large volumes of emails in real-time.

3.Methodology/Implementation:

In this section, we discuss the methodology and implementation steps involved in email spam classification. We outline the key stages of the process, from data preparation to model training and evaluation.

1. Data Collection and Pre-processing:
   * Obtain a label dataset consisting of spam and legitimate emails. This dataset can be obtained from public repositories or by manually label a representative sample of emails.
   * Pre-process the email data by removing irrelevant information, such as HTML tags, headers, and signatures. Clean the text by removing stop words, punctuation, and converting to lowercase.
   * Split the dataset into training and testing sets, ensuring a balanced distribution of spam and legitimate emails in each set.
2. Feature Extraction:
   * Extract relevant features from the pre-processed email data. This can include textual features such as word frequency, n-grams, and TF-IDF scores.
   * Additionally, extract other features such as sender information, email headers, and metadata (e.g., attachment types, embedded URLs) that can provide valuable information for classification.
3. Model Selection:
   * Choose an appropriate classification algorithm based on the dataset characteristics and problem requirements. Popular choices include Naive Bayes, SVM, Decision Trees, Random Forest, and neural networks.
   * Consider the strengths and limitations of each algorithm, such as their ability to handle imbalanced data, scalability, and interpretability.
4. Model Training:
   * Train the chosen model using the label training dataset and the extracted features.
   * Fine-tune the model hyperparameters through techniques like cross-validation or grid search to optimize performance.
   * Utilize techniques such as oversampling or under sampling to address class imbalance if necessary.
5. Model Evaluation:
   * Evaluate the trained model using the label testing dataset.
   * Calculate standard evaluation metrics such as accuracy, precision, recall, and F1 score to measure the model's performance.
   * Consider additional metrics like ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) for assessing the model's ability to handle false positives and false negatives.
6. Iterative Improvement:
   * Analyze the model's performance and identify areas of improvement. This could include exploring alternative feature extraction methods, experimenting with different classification algorithms, or adjusting the model's hyperparameters.
   * Iterate the training and evaluation process to refine the model until satisfactory performance is achieved.
7. Deployment and Integration:
   * Once a satisfactory model is obtained, integrate it into an email processing pipeline or email client.
   * Ensure the system can handle real-time classification of incoming emails.
   * Continuously monitor and update the model to adapt to new spam patterns and evolving email threats.

**4.RESULTS &DISCUSSIONS**

In this section, we present the results obtained from the email spam classification experiments and discuss their implications. We analyze the performance of the classification models, evaluate their effectiveness in distinguishing between spam and legitimate emails, and address any notable findings or challenges encountered during the process.

* + .

1.Comparative Analysis:

* + Compare the performance of the implemented models with existing approaches or state-of-the-art methods.
  + Evaluate the strengths and weaknesses of the proposed models in terms of accuracy, efficiency, and robustness.
  + Discuss any significant differences in performance between the algorithms used and provide insights into their suitability for email spam classification.

1. Impact of Feature Extraction:
   * Analyze the contribution of different features extracted from the email data in the classification process.
   * Discuss the effectiveness of textual features, metadata, sender information, and other extracted attributes in differentiating spam and legitimate emails.
   * Identify the most informative features and their relative importance in the classification models.
2. Handling False Positives and False Negatives:
   * Investigate the occurrence of false positives (legitimate emails classified as spam) and false negatives (spam emails classified as legitimate) in the classification results.
   * Discuss the implications of these errors and potential strategies to reduce their occurrence.
   * Address the trade-off between minimizing false positives and false negatives, considering the impact on user experience and security.

4.Scalability and Real-Time Processing:

* + Evaluate the scalability of the implemented models, particularly in handling large email volumes in real-time.
  + Discuss any performance bottlenecks encountered and propose strategies for improving scalability and efficiency .

5.Robustness and Generalization:

* + Assess the robustness of the classification models by evaluating their performance on unseen or new email datasets.
  + Discuss the generalization capabilities of the models and their ability to adapt to evolving spam patterns.
  + Address any challenges encountered in maintaining model performance over time.

6.Limitations and Future Directions:

* + Identify any limitations or challenges faced during the email spam classification process.
  + Discuss potential areas for improvement or future research, such as incorporating more advanced feature extraction techniques, exploring ensemble methods, or leveraging deep learning approaches.
  + Consider the integration of real-time feedback mechanisms to continuously update and improve the classification models based on user feedback and evolving email threats.

By presenting the results and engaging in discussions around the performance, limitations, and future directions of the email spam classification models, this section provides insights into the effectiveness of the implemented approaches and sets the stage for further advancements in combating email spam.

Top of Form

**5.CONCLUSION**

After the comprehensive analysis on the selected research studies, we have identified several research findings and observation. These have been detailed discussed in the prior sections with adequate explanations. In this section, we will be more focused on main findings and the conclusions of the study. High adoption rate for supervised machine learning approach can be seen throughout the review. This approach is used mainly because it generates higher accuracy results with less variation giving high consistency for this approach. Apart from that, we have found out that certain algorithms such as Na¨ıve Based and SVM have high demand compared to other Machine Learning Algorithms. The multi algorithm used systems are more common in use to cater better outcome rather than using single algorithm. Researchers have more focused on email features such as BoW and Body text creating future research opportunities to develop systems to detect spam on other email features.

**6.REFERENCES**

[1] “Global spam volume as percentage of total e-mail traffic from January 2014 to September 2019, by month.” https://www.statista.com/statistics/420391/spam-email-traffic-share/.

[2] T. Ouyang, S. Ray, M. Allman, and M. Rabinovich, “A large-scale empirical analysis of email spam detection through network characteristics in a stand-alone enterprise,” Elsevier, vol. 2015, pp. 101–102.

[3] O. Saad, A. Darwish, and R. Faraj, “A survey of machine learning techniques for Spam filtering,” IJCSNS Int. J. Comput. Sci. Netw. Secur.

[4] K. Asif, A. Sami, S. Bharindhan, and K. Krishan, “A Comprehensive Survey for Intelligent Spam Email Detection,” IEEEXplore, 2019.

[5] “Number of e-mail users worldwide from 2017 to 2024.” [Online]. Available: https://www.statista.com/statistics/255080/number-of-e-mailusers-worldwide/.

[6] M. Guntrip, “https://www.proofpoint.com/us/corporateblog/post/fbi-reports-125-billion-global-financial-lossesdue-business-email-compromise.” [Online]. Available: https://www.proofpoint.com/us/corporate-blog/post/fbi-reports-125- billion-global-financial-losses-due-business-email-compromise.

[7] “Australian Competition and consumer Commission,” Scam Stat., [Online]. Available: https://www.scamwatch.gov.au/scamstatistics?scamid=all & date=2018.

[8] K. Jackowski, B. Krawczyk, and M. Wozniak, “Application of adaptive ´ splitting and selection classifier to the spam filtering problem,” Cybern. Syst. An Int. J.

[9] Sathya and A. Abraham, “Comparison of supervised and unsupervised learning algorithms for pattern classification,” ResearchGate.

[10] F. Qian, Y. C. H. Abhinav Pathak, Z. M. Mao, and Y. Xie, “A case for unsupervised-learning-based spam filtering,” Univ. Minnesota J., 2010.

[11] Y. Alamlahi and A. Muthana, An Email Modelling Approach for Neural Network Spam Filtering to Improve Score-based Anti-spam Systems. Modern Education and Computer Science Press, 2018.

[12] L. Melian and A. Nursikuwagus, “Prediction student eligibility in vocation school with Na¨ıve-Byes decision algorithm,” 2018.

[13] A. S. Aski and N. K. Sourati, “Proposed efficient algorithm to filter spam using machine learning techniques,” Elsevier, vol. 2016, pp. 145–149.

[14] K. Pawar and M. Patil, “Pattern classification under attack on spam filtering,” IEEExplore, 2015.

[15] A. K. Rajan, V, and A K. “V, V., & Rajan, “An Improved Spam Detection Method With Weighted Support Vector Machine,” IEEE Explor. .” IEEExplore.

[16] H. Kaur and A. Sharma, “Improved Email Spam Classification Method Using Integrated Particle Swarm Optimization and Decision Tree,” IEEE Xplore, vol. 2016, pp. 516–521.