Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/NA11.20260414
Published: June 9, 2026

Cybersecurity Threat Analysis and Detection of Spam Emails

Jannatul Supi Jenifa
School of Computing
Grand Valley State University
Allendale, MI, USA
jenifaj@mail.gvsu.edu

Baqer Aljabr
Mechanical, Industrial, and Manufacturing Department
The University of Toledo
Toledo, OH, USA
Baqer.Aljabr@rockets.utoledo.edu

Muteb Aljasem
School of Engineering
Bowling Green State University
Bowling Green, OH, USA
aljasem@bgsu.edu

Yara Mohammed
Information Science
University of North Texas
Denton, TX, USA
Yara.Mohammed@unt.edu

Abstract

As a widely adopted official communications medium across the world, email is often a target of cybersecurity attacks. Spam e-mail is a growing security concern and is frequently used in phishing attacks, data stealing, and denial-of-service (DoS) attacks. Spam has increased these threats due to its fast expansion and proliferation. Most of the current research is concentrated on spam email detection as a classification problem, and very little is discussed about the implications or broadening of spam email detection in the aspect of cyber security. In this paper, this is done by analyzing spam emails from a security perspective, applying threat modeling, and using a spambot-based simulator to demonstrate how the spam emails can be used as an origin for a DoS attack. We employed a count-based vectorization approach for feature extraction to train a Naïve Bayes classifier for spam email detection. Moreover, a comparative analysis was performed by employing different machine learning classifiers, such as Adaboost, Random Forest, Support Vector Machine, and K-Nearest Neighbors, to compare the effectiveness of Naive Bayes for this task. We evaluated the performance of our method on a publicly available spam email dataset of 5728 samples, including 4360 legitimate and 1368 spam data entries. Synthetic Minority Over-Sampling Technique (SMOTE) was applied during model training to counter class imbalance. The accuracy of 98.517% indicates the effectiveness of the proposed method for spam email detection.

Keywords
Cybersecurity, Denial of Service (DoS), Machine Learning, Spambot, Spam Email Detector.

1. Introduction
Electronic mail (email) is a formal communication platform in the modern-day corporate and academic domains, but it is also vulnerable to a variety of attacks, including phishing, spoofing, and malware. In recent years, email usage has grown substantially, with nearly 4.8 billion users sending and receiving emails daily, accompanied by a rapid rise in spam emails (Cota et al. 2022). Spam email represents a significant security threat that demands serious attention and effective mitigation strategies. Spam email attack (Khan et al. 2015) involves attackers sending unwanted emails to people, with the primary intent of committing financial fraud. Attackers can also send spam automatically to flood email systems. The normal email protections, such as anti-spam and anti-phishing systems, may fail against floods coming from real or trusted senders (Herzberg et al. 2009). Even strong domain validation systems can be fooled because the emails look real. Similarly, standard denial-of-service (DoS) prevention techniques like rate control can fail against email floods and may block legitimate emails by mistake (Herzberg et al. 2014).

Machine learning (ML) algorithms are being widely used as a potent tool in various domains, including the information security industry, to detect malware, phishing, and other advanced attacks. Machine learning algorithms are trained on massive data, including many spam and legitimate emails, for identifying and solving spam problems (Mansoor et al. 2021). Emails are categorized based on text patterns, keywords, and phrases, and new emails can be correctly labeled as spam or legitimate.

Most existing research has focused on improving classification accuracy, with limited attention given to analyzing spam emails as coordinated cyberattacks through the application of threat modeling approaches. In light of these problems in filtering spam and its increasing potential as a cybersecurity issue, this paper focuses on developing a robust spam email detection system. In this study, a threat analysis was carried out that employs threat modeling and an example based on spambots, showing how automated spam generation can potentially be used as a denial-of-service vector against email infrastructures. We proposed an ML-based spam email detector employing a count-based vectorization feature extraction approach with a Naïve Bayes classifier to reliably spot the spam emails. To overcome the class imbalance, this study applied the Synthetic Minority Over-sampling Technique (SMOTE). Additionally, a comparative analysis involving various conventional ML classifiers was also performed to determine the most effective ML algorithm for spam email detection. The contributions of this research could help raise awareness of spam-based email threats and support the development of models for spam email detection.

2. Threat Model
We analyzed various threats caused by spam emails from a security perspective. Spam emails often redirect users to malicious websites and employ social engineering practices like lottery scams and fake jobs to lure users to provide sensitive data, which can lead to unauthorized access and eventually confidential data stealing and financial fraud. Additionally, spammers also launch spambots to flood users' mailing accounts with many emails at once, leading to a denial of service state. This demonstrates how spam emails can be exploited as an attack vector for DoS attacks. This threat analysis is abstract and, therefore, aimed at identifying typical attack vectors and how they may affect users and email infrastructures.

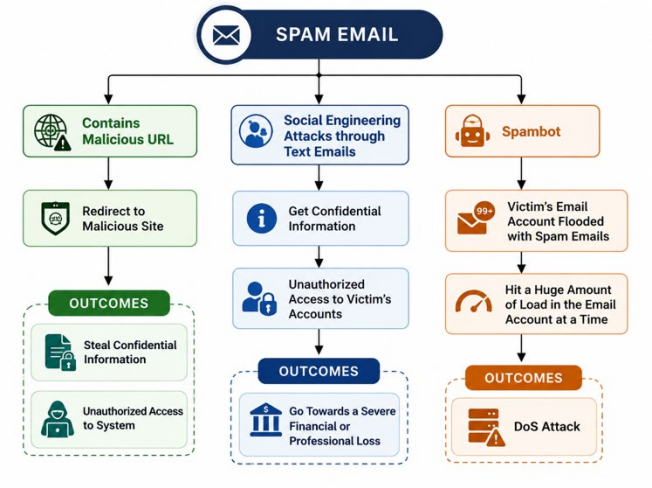

Figure 1: Potential threats of spam emails.

3. Related Work

Email spam is a problem that many individuals and organizations deal with every day, creating serious security risks. Additionally, it creates a bulk of emails in the user's inbox, resulting in lower productivity and opens a channel for further threats, thus existing methods focus on both detection and mitigation (Vijayakumar et al. 2025)

3.1 Threat Modeling of Spam Email as a Denial of Service Attack Vector

One of the most serious threats to organizations is email-based attacks, which can trick users into participating in more malicious attacks and are used by attackers to a great degree to gain access to the organization through their socially engineered messages (Wassermann et al. 2023). Spear phishing and business email compromise are low-volume, high-stakes threats, but definitions of them in the literature vary, making it difficult to define how to best mitigate against them, while their rarity compared to bulk phishing also presents significant class-imbalance issues that favour context-aware detection approaches (Wassermann et al. 2023). In addition to targeted fraud, other types of attacks affect the availability of email systems, such as mailbombing, where the inbox is inundated with email messages to reach the mailbox limit, overload the mailbox, and prevent legitimate messages from being received (Islam et al. 2025). More recent studies also indicate that today's mailbomb scripts are more difficult to detect as they randomize content, headers, incorporate multi-threading and proxy routing, and generate AI-based text to bypass machine learning spam filters, highlighting the need for enhanced controls like rate limiting, reputation screening, and volume anomaly detection (Islam et al. 2025). Moreover, even when the message is a legitimate security alert, it may not be noticed by the recipient or the recipient may mistake it for spam and delete it, since the recipients' inboxes are already filling up with a high volume of spam e-mails, which makes it difficult for important messages to be noticed (Hennig et al. 2025). This shows that notification visibility problems can persist even when messages are delivered successfully.

Botnets contribute to email abuse because infected machines can be used to generate spam and support other disruptive activity. SpotBotML presents a network-based approach that represents Zeek metadata as structured text and uses supervised large language models, including fine-tuned LoRA LLaMA variants, to classify traffic as legitimate or malicious while aiming to preserve privacy and remain effective against evolving and previously unseen botnets (Matlock et al. 2025). In addition to behavior-based network detection, Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM), and Domain-based Botnets contribute to email abuse because infected machines can be used to generate spam and support other disruptive activity. SpotBotML presents a network-based approach that represents Zeek metadata as structured text and uses supervised large language models, including fine-tuned LoRA LLaMA variants, to classify traffic as legitimate or malicious while aiming to preserve privacy and remain effective against evolving and previously unseen botnets (Matlock et al. 2025). In addition to behavior-based network detection, SPF, DKIM, and Domain-based Message Authentication, Reporting, and Conformance (DMARC) are mostly employed for sender verification, yet they can fail in case of inconsistent email metadata across different clients (Ragheb et al. 2023). A reliable approach that can improve phishing detection is by cross-checking headers, relay paths, and metadata consistency across platforms to flag anomalies that traditional filters may miss (Kalamata et al. 2025).

3.2 Machine Learning Spam Detection

Older prevention methods, including blocking specific domains or looking for certain keywords, are still often used by many spam filters. The issue is that spammers are always evolving their language and scams, so these tricks do not stay useful for long. For that reason, (Alurkar et al. 2017) suggested applications of machine learning for identifying repeated phrase patterns from spam emails. They seek to enable spam detection that is smarter, more flexible, and even allows users to classify types of spam. They also link this to the idea of designing a PWA that enables a smoother filtering experience. In (Mansoor et al. 2021), it is reported that most of the spam detection research has focused on supervised learning rather than unsupervised learning due to the better performance of supervised methods. The work also discusses possible improvements for the future, like detecting spam in real time, updating filters more easily as spam evolves, improving processing speed, and including more information beyond email text. They also encourage exploring semi-supervised and unsupervised learning methods. Few works have also performed a comparative analysis to find the best model. In (Lanka et al. 2023), several algorithms, Naïve Bayes, SVM, Decision Trees, Random Forest, Bagging, and AdaBoost were compared. Naïve Bayes is the most accurate with a small number of false positives. However, not all results are the same. For example, (Thakur et al. 2022) demonstrated that the SVM model yields better results than logistic regression, Decision Trees, Naïve Bayes, and KNN on the same dataset. Another more comprehensive study (Yadav et al. 2023) focused on the entire process of spam detection from tokenization to feature extraction to techniques like keyword detection and models such as Naïve Bayes and Decision Trees. Five classifiers (SVM, Random Forest, logistic regression, multinomial Naïve Bayes, and Gaussian Naïve Bayes) were evaluated on two datasets in (Cota et al. 2022). Despite Naïve Bayes showing promising results on one dataset, the Random Forest was better overall, with a mean accuracy of 86.3% and 83.7% for different splits of the data.

Abdullahi et al. (2021) studied spam filtering from various perspectives and compared header-based, content-based, and OCR-based systems (for spam concealed in images). They experimented with Naive Bayes, Decision Trees, Random Forest, SVM, and clustering models. Random Forest was again the high-performing algorithm with a high accuracy and low error rate. In the mean time, (Taloba et al. 2019) presented a hybrid approach named GADT, which was able to achieve better performance than those of the common text classifiers and was more practical for real-world filtering. Furthermore, (Zaid et al. 2016) highlighted that feature selection is important and is often overlooked. Naïve Bayes, SVM, and Multilayer Perceptron were evaluated in an academic email setting, and better features resulted in higher accuracy and efficiency. To sum up in a few words, higher quality information and any powerful algorithm will do a much better job. Security classification tasks often involve these difficulties, such as having a minority class (e.g., attacks or spam) that is not adequately represented and models that may misclassify examples of that class (Yusuf et al. 2025; Arfaoui et al. 2026). The near-perfect performance of 99.93% accuracy yielded by the best combination of mutual information-based feature selection and class balancing with SMOTE on BPNN in the experiments conducted on CICIDS2017 proves that this combination improves the detection of rare attack samples and reduces irrelevant features (Yusuf et al. 2025). A complementary study shows that the accuracy on imbalanced spam data could be significantly affected by the choice of the class-balancing strategy, and that among the different feature vectorizers, resampling techniques, and classifiers employed in a unified pipeline, the SMOTEENN hybrid resampling method combined with LinearSVC using TF-IDF features yielded the best results in terms of accuracy on the SpamAssassin corpus (Arfaoui et al. 2026). Alongside classical ML pipelines, transformer-based language models are increasingly used for email text classification because they capture richer context than sparse TF–IDF features, while still requiring careful handling of imbalance in real email datasets (Uddin et al. 2026). Robustly Optimized BERT Pretraining Approach (RoBERTa) can strengthen email text classification by capturing richer context than many traditional machine-learning features; after cleaning the data and addressing class imbalance, a fine-tuned RoBERTa model reports 98.45% accuracy and adds interpretability through LITA, which combines LIME with transformer attribution methods to make predictions easier to understand (Uddin et al. 2026).

4. Methodology

This study aims to raise awareness of spam email threats and show the usefulness of machine learning for spam detection. More specifically, the research goal was to study the security risks of spam emails using threat modeling and spambot simulation, and to build a machine learning-based detection model. We also employed the Synthetic Minority Over-Sampling Technique (SMOTE) to solve the class imbalance. A spambot was used to simulate the risk of spam emails being used for DoS attacks.

4.1 Spambot

DoS attacks are common and powerful cyberattacks that can be launched using spambots. In this research, a spambot was used as a simulation to demonstrate how easily it can flood an inbox with many emails in a short time, causing the account to become overloaded and the email server to fail. The simulation shows that the automated creation of spam might be used to effectively overload email services, and this can be used as an example of proof-of-concept of denial-of-service threats. In this study, Python and the 'smtplib' library were used to build the spambot. For this experiment, a Gmail account was used along with an SMTP server

4.2 Spam Email Detector

In this critical phase of the study, we developed a machine learning model for the purpose of detecting spam emails. We also used various learning algorithms and analyzed the performance of these algorithms on the same dataset.

1) **Data Collection and Preprocessing:** We used the publicly available dataset for spam email detection from Kaggle, which includes the email body text. It consisted of 4360 legitimate data and 1368 spam data entries, totaling 5728 entries. Two predictor classes were present: Legitimate (0) and Spam (1). After collecting the data, we performed preprocessing steps. These steps included checking for duplicate and null data within the dataset, followed by splitting the dataframe into train and test samples using the "train test split" module from the sklearn library in Python.
2)

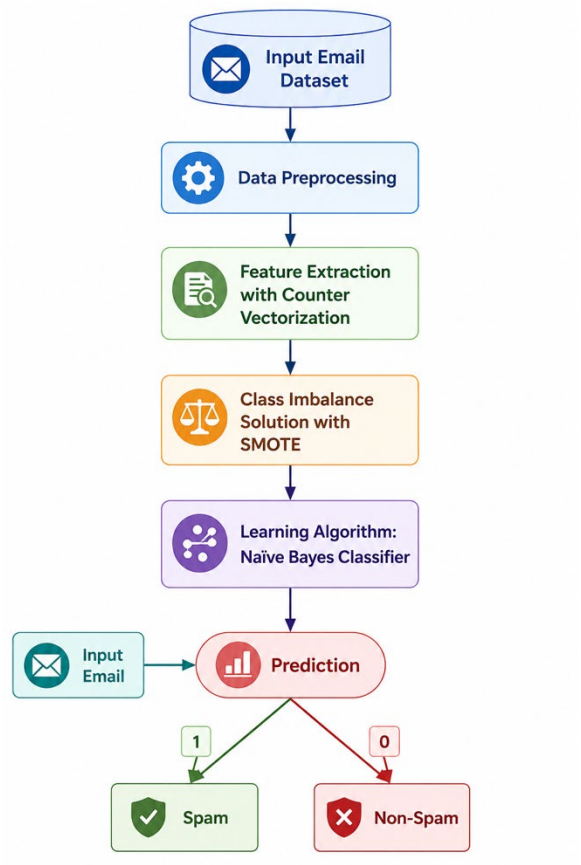

Figure 2: Spam email detection model.

3) **Feature Extraction:** To extract features, the text of each email was turned into a number based on how often certain phrases or tokens appear. Each email was shown as a vector, with each element showing how many times a certain word appears in the text. Count-based vectorization was used in feature extraction, in which every email was converted to a numerical feature vector in terms of word frequency. This strategy transforms the text information that is not structured into a structured format understandable by a machine learning classifier.

4) **SMOTE:** The dataset had a class imbalance issue, so we 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. The application of SMOTE was limited to the training data only to prevent data leakage and achieve fair evaluation of the model. Fig. 3 shows how SMOTE functions.

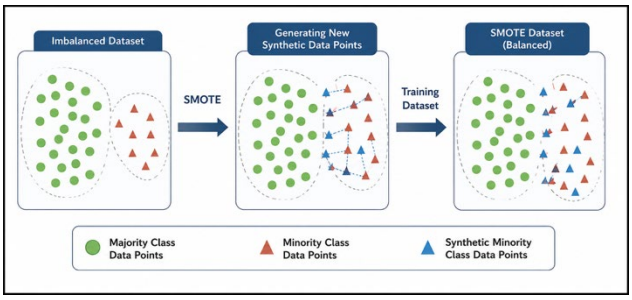

Figure 3: SMOTE Technique on training dataset.

5) **Classification**
After feature extraction using a count-based vectorization approach and SMOTE, we employed a Naïve Bayes model to classify the spam and non-spam emails. The probabilistic approach of Naive Bayes ensures efficient handling of text data with minimum computational cost and interpretable results, thus making it a significant classifier for spam email detection.

5. Experiments and Results

This section provides the details of evaluation metrics, experiments, and results discussion for assessing the performance of our method.

5.1 Performance Evaluation Metrics

We employed accuracy, precision, recall, and F1-score to evaluate our method.

1) **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.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{Total Samples}} \quad (1)$$

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

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

3) **Recall:** The proportion of real positives to all the positives in the ground truth.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FN}} \quad (3)$$

4) **F1-Score:** The Harmonic Mean between Recall and Precision.

$$\text{F1} - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (4)$$

Where TP = True Positive and TN = True Negative, FP = False Positive, and FN = False Negative.

5.2 Evaluation of Proposed Method

This experiment was performed to assess the significance of our method for spam email detection. For this purpose, we extracted the features using a count-based vectorization approach and employed SMOTE on these features, and later trained the Naïve Bayes using these processed features to classify between the spam and genuine emails. We attained the accuracy, precision, recall, and F1-score of 98.517%, 98%, 97.5%, and 98%, respectively. These results illustrate the efficacy of our method for reliable detection of spam emails.

5.3 Comparative Analysis

We conducted an experiment to compare the performance of five machine learning algorithms in identifying spam emails. For a fair comparison, five supervised learning algorithms, namely Random Forest, AdaBoost, SVM, KNN,

and Naïve Bayes, were trained and evaluated on the same dataset. The details of comparative ML models are presented below.

1) **Random Forest:** Random Forest is an ensemble learning approach that combines many decision trees to create more robust and accurate predictions. It delivers a diverse collection of decision trees by infusing randomization throughout the tree-building process. Random Forest is less prone to overfitting and provides higher generalization compared to individual decision trees (Belgiu et al. 2016).
2) **Adaboost:** AdaBoost, short for Adaptive Boosting, is an ensemble machine learning method that may be utilized in a wide variety of classification and regression applications. It combines numerous classifiers to boost the accuracy of the classifiers. AdaBoost is an iterative ensemble approach. The AdaBoost classifier develops a strong classifier by merging numerous poorly performing classifiers to generate a high-accuracy strong classifier (Wang et al. 2021).
3) **K-Nearest Neighbors (KNN):** KNN is a fundamental and straightforward classification algorithm that classifies a new data point based on the majority class of its k-nearest neighbors in the feature space. The choice of **k** determines the sensitivity of the model. KNN is straightforward to create and can be successful in cases when the decision boundary is non-linear (Liu et al. 2017).
4) **Support Vector Machine (SVM):** SVM is a sophisticated supervised learning approach used for classification and regression applications. It seeks to discover the best hyperplane that optimally splits data points belonging to distinct classes. With the kernel approach, SVM is good at managing high-dimensional data and can manage both linear and nonlinear boundaries (Awad et al. 2015).
5) **Naïve Bayes:** Naïve Bayes is a probabilistic classification technique based on Bayes' theorem. It assumes that every attribute is independent of another, which is why it is regarded as "Naïve." Despite its origin, Naïve Bayes works effectively in many text classification and document categorization challenges. It is very beneficial when dealing with high-dimensional data, such as natural language processing work. The Naïve Bayes classifier has several modes. In this scenario, we employed a multinomial Naïve Bayes classifier, which is generally used for text data categorization (Yang et al. 2018).

The results of the experiments are summarized in Table 1. From the results, we can observe that Naïve Bayes attained the best results, Random Forest was the second best, whereas SVM and KNN attained the lowest results. This comparative analysis reveals the importance of Naïve Bayes classifier for reliable spam email detection. These results show that probabilistic learning approaches are highly effective for detecting spam emails. KNN and SVM were unable to achieve good performance, indicating their inability to handle unseen data.

Table 1: Comparative Analysis of different ML classifiers.

ML Models	**Accuracy**	**Precision**	**Recall**	**F1-score**
Random Forest	95.812	94%	94%	94%
AdaBoost	85.777	81.5%	89.5%	83%
SVM	54.625	67%	70.5%	54%
KNN	54.625	67%	70.5%	54%
Naïve Bayes	98.517	98%	97.5%	98%

To better understand model performance, we also presented the class-wise results in Table 2 for all the comparative models. Since spam detection is important for security, these measures help identify the risk of misclassification. These measures help spot the risk of misclassifying spam emails. The Naïve Bayes and Adaboost models attained balanced results, but Naive Bayes outperformed the Random Forest in terms of all metrics. Whereas SVM, KNN, and Adaboost attained imbalanced results, struggling to identify either spam or non-spam emails. This class-wise performance comparison also highlights the potency of Naïve Bayes among all the comparative ML models for spam email detection.

Table 2. Classwise performance comparison.

ML Models	Precision (0)	Recall (0)	F1-Score (0)	Precision (1)	Recall (1)	F1-Score (1)
Random Forest	97	97	97	91	91	91
AdaBoost	99	82	90	62	97	76

SVM	100	41	58	33	100	50
KNN	100	41	58	33	100	50
Naïve Bayes	99	99	99	97	96	97

6. Conclusion and Future Work

This work has presented a reliable spam email detection system. Spambot was used in a simulation to illustrate its potency for DoS attacks. The goal was to create an effective detection model for spam emails, increase awareness of email account dangers, and improve security. Performance was evaluated on a publicly available spam email dataset. Experimental study, including a comparison with different ML models, indicates the superiority of the Naïve Bayes model, with an accuracy of 98.517%, among all comparative models. Future research can focus on real-time spam detection and integrating adaptive security mechanisms to further strengthen email system defenses.

References

Abdullahi, Muhammad, et al. "A Review on Machine Learning Techniques for Image Based Spam Emails Detection." 2020 IEEE 2nd International Conference on Cyberspace (CYBER NIGERIA). IEEE, 2021.

Alurkar, Aakash Atul, et al. "A proposed data science approach for email spam classification using machine learning techniques." 2017 Internet of Things Business Models, Users, and Networks. IEEE, 2017.

Arfaoui, N. (2026). Enhancing Machine Learning Algorithms for Imbalanced Data. A Case Study: Spam Detection. IEEE Access.

Awad, Mariette, et al. "Support vector machines for classification." Efficient learning machines: Theories, concepts, and applications for engineers and system designers (2015): 39-66.

Belgiu, Mariana, and Lucian Drăguţ. "Random forest in remote sensing: A review of applications and future directions." ISPRS journal of photogrammetry and remote sensing 114 (2016): 24-31.

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.

Hennig, A., Veit, M., Schmidt-Enke, L., Neusser, F., Herrmann, D., & Mayer, P. (2025). "I believe it's incredibly difficult to fight against this flood of spam": Towards enhancing strategies for creating effective vulnerability notifications. Computers & Security, 104682.

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

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

Islam, N. (2025). Modern Mailbomb Attacks: From Attack Mechanisms to Mitigation Techniques.

Kalamata, R. (2025). Data-Driven Phishing Email Detection by Analyzing Metadata Across Platforms for Enhanced Security. Available at SSRN.

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

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.

Liu, Jie, et al. "An improved KNN text classification algorithm based on Simhash." 2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC). IEEE, 2017.

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.

Matlock, P. (2025). SpotBotML (LLM: IPv4+ IPv6+ tunnel): LLMs for Botnet detection via Network Behavior [using Zeek metadata; supervised generic & 'fine-tuned'Large Language Models (x2)]. Accessed: Apr, 6.

Ragheb, M. S., Elmedany, W., & Sharif, M. S. (2023, August). The Effectiveness of DKIM and SPF in Strengthening Email Security. In FiCloud (pp. 422-426).

Taloba, Ahmed I., and Safaa Ismail. "An intelligent hybrid technique of decision tree and genetic algorithm for e-mail spam detection." 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS). IEEE, 2019.

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.

Uddin, M. A., Mahiuddin, M., & Sarker, I. H. (2026). An explainable transformer-based model for phishing email detection: A large language model approach. Computer Networks, 112061.

Vijayakumar, B., & Thomas, C. (2025). The ethics of envisioning spam free email inboxes. AI and Ethics, 5(3), 2125-2148.

Wang, Wenyang, and Dongchu Sun. "The improved AdaBoost algorithms for imbalanced data classification." Information Sciences 563 (2021): 358-374.

Wassermann, S., Meyer, M., Goutal, S., & Riquet, D. (2023). Targeted attacks: Redefining spear phishing and business email compromise. arXiv preprint arXiv:2309.14166.

Yadav, Deepak, and Er Bhupinder Kaur. "Machine Learning Models for Email Spam Detection: A Review." 2023 Second International Conference on Smart Technologies for Smart Nation (SmartTechCon). IEEE, 2023.

Yang, Feng-Jen." An implementation of Naïve Bayes classifier." 2018 International conference on computational science and computational intelligence (CSCI). IEEE, 2018.

Yusuf-Asaju, Y. O., Gbolagade, K. A., Abdulsalam, S. O., Asaju-Gbolagade, A. W., & Babatunde, A. N. (2025). Development of an Enhanced Intrusion Detection System Using Mutual Information Feature Selection and Data Classification Algorithms. Journal of Institutional Research, Big Data Analytics and Innovation, 1(3), 208-217.

Zaid, Aisha, Ja'far Alqatawna, and Ammar Huneiti. "A proposed model for malicious spam detection in email systems of educational institutes." 2016 Cybersecurity and Cyberforensics Conference (CCC). IEEE, 2016.