Email Shield: Spam and Scam Detection Using Machine Learning

Ms. M. Shilpa[1], Mohammed Fasi Uddin[2], Mohammed Irfan Hussain Siddiqui[3], Mohammed Sofiyan[4]
[1,2,3]*B.E. Students, Department of IT, Lords Institute of Engineering and Technology, Hyderabad*
[4]*Assistant Professor, Department of IT, Lords Institute of Engineering and Technology, Hyderabad*

***Abstract*—Email Shield is a machine learning- based security system designed to automatically detect spam and scam emails with high accuracy. The system processes incoming emails through multiple stages beginning with text preprocessing, where noise removal, tokenization, and stop word elimination are performed. The cleaned text is then transformed into numerical form using TF-IDF and word embeddings within the feature extraction module. These features are passed into machine learning classifiers including Naive Bayes, Logistic Regression, and Support Vector Machine (SVM) to predict whether an email is spam, not spam, scam, or safe. To enhance detection reliability, a Threat Intelligence Module performs keyword-based scam identification and sender reputation analysis. The Decision Engine combines classification results and threat intelligence signals to generate a final classification along with a confidence score.**

***Index Terms*—Email Security, Machine Learning, Spam Detection, Scam Detection, Naive Bayes, Support Vector Machine, TF-IDF, Threat Intelligence, Natural Language Processing, Phishing Detection**

I. INTRODUCTION

Background

Email communication remains one of the most widely used methods for personal, professional, and commercial interaction across the globe. However, it is also a primary vector for cyber threats such as spam, phishing, scam emails, and malicious links. These threats not only reduce user productivity but also pose significant risks including identity theft, financial loss, data breaches, and malware infections [1]. Traditional rule-based spam filters are no longer sufficient to counter modern, sophisticated email attacks that constantly adapt to evade detection. To address these challenges, Email Shield introduces an intelligent and automated email filtering system that leverages Machine Learning (ML), Natural Language Processing (NLP), and threat intelligence techniques. The system analyses incoming emails through multiple stages: preprocessing, feature extraction, classification, and scam detection. A Decision Engine evaluates the results and generates the final verdict along with a confidence score. If the email is malicious, users are immediately notified and all results are stored securely for analysis [2]

Research Gap

Despite extensive research in spam filtering, several critical limitations persist in current systems:

- Most existing systems rely on static keyword or rule-based filters that cannot adapt to evolving phishing and scam techniques, including AI-generated scam messages and social engineering patterns.
- High false positive rates cause legitimate emails to be incorrectly classified as spam, disrupting business and personal communication.
- Advanced zero-day scam emails frequently evade detection due to the absence of dynamic learning mechanisms.
- Current solutions lack multi-layer decision fusion combining ML classification with real-time threat intelligence signals such as sender reputation and keyword analysis.
- The basic idea is to prepare a nearby SVM for the separation task and to categorize the nearest neighbours to a verification challenges.
- Early email filtering systems relied heavily on rule-based techniques, where predefined keywords or sender patterns were used to classify spam.
- It only concentrates on removing unsolicited emails or two-level prioritizing systems, spam filtering using is a type of email prioritization.

Research Objectives

- To design and implement an intelligent email filtering system that classifies incoming emails as Spam, Not Spam, Scam, or Safe using machine learning algorithms.
- To apply NLP-based text preprocessing techniques including tokenization, stop word removal, and TFIDF vectorization to convert raw email content into feature vectors.
- To integrate a Threat Intelligence Module that performs keyword-based scam detection and sender reputation analysis for multi-layer protection.
- To develop a Decision Engine that fuses ML classification scores with threat intelligence signals to produce reliable final predictions with confidence scores.
- To provide an admin dashboard that stores and visualizes detection statistics, enabling continuous monitoring and system improvement through a user feedback loop.

Limitations of the study

The system has certain limitations. The ML models were trained on publicly available benchmark datasets, which may not fully represent all real-world email scenarios. Highly sophisticated AI-generated phishing emails may reduce classification accuracy. The system requires periodic retraining to maintain performance against evolving threats. Additionally, classification accuracy may be affected by inconsistent email formatting or encoding across different mail servers. Rationale of the study This research work was done to improve traditional healthcare, which often faces problems like delayed diagnoses and overworked medical staff. As more people live with chronic diseases, there is a need for smarter, faster, and more personalized care. The system in this study uses devices to monitor health, machine learning to predict health issues, and language models to make medical information easier to understand. The objective is to help doctors and patients make better decisions and improve healthcare efficiency.

II. RELATED WORK

Email spam and scam detection has been an active area of research for over two decades driven by the rapid growth of digital communication and the increasing sophistication of cyber threats. The researchers reported in their work tit led “The image and Text spam filtering” that spammers have adopted new techniques [3] to incorporate spam email inside the image linked to the package.

Rule-Based and Bayesian Approaches

Early email filtering systems relied heavily on rule-based techniques, where predefined keywords and sender patterns were used to classify spam. Although simple, these systems failed to adapt to evolving threats and suffered from high false positive rates. Researchers subsequently explored Bayesian filtering, with Naive Bayes becoming one of the earliest and most effective machine learning models applied for spam classification due to its probabilistic nature and computational efficiency [4].

Machine Learning for Email Classification

With advancements in text processing, studies shifted towards TF-IDF feature extraction combined with Support Vector Machines (SVM) and Logistic Regression. SVM gained popularity due to its ability to classify high dimensional text data effectively and handle non-linear boundaries [5]. Later studies incorporated word embeddings and neural architectures such as LSTM and CNN to capture semantic and contextual relationships in email content. Researchers such as Sahoo et al. demonstrated that ensemble approaches combining multiple classifiers significantly improve detection accuracy compared to individual models [6].

Threat Intelligence Integration

In recent years, researchers recognized that machine learning alone is insufficient to detect targeted scam emails, which rely on psychological manipulation rather than spam-like patterns. This led to the integration of threat intelligence techniques including sender reputation scoring, domain blacklisting, keyword analysis, and anomaly-based detection [7]. Studies by String hinietal showed that analyzing sender behavior patterns alongside email content considerably reduces false negative rates for phishing detection.

Gaps Identified in Existing Systems

While many systems use either ML classification or threat intelligence independently, few effectively

combine both into a unified detection pipeline with real-time decision fusion. Additionally, most existing approaches lack user- friendly dashboards with feedback loops that enable continuous model improvement. These gaps motivated the design of Email Shield, which integrates ML classification, threat intelligence, and adaptive learning into a single cohesive architecture.

III. RESEARCH METHODOLOGY

The proposed methodology follows a multi-layered pipeline that combines NLP-based text preprocessing, machine learning classification, and threat intelligence analysis to deliver accurate and real-time email threat detection.

Data Collection and Input Processing

Emails are sourced from publicly available benchmark datasets including the Enron Email Dataset, Ling Spam Corpus, and Spam Assassin Public Corpus. Each email instance includes the body text, subject line, header metadata, and sender information.

The combined dataset consists of 10,000 labeled emails covering spam, scam/phishing, and legitimate categories

The system accepts email input via SMTP/IMAP integration or direct text submission through the web interface developed using the Flask framework.

Preprocessing Module

Raw email text contains significant noise including HTML tags, URLs, special characters, email signatures, and encoded content. The preprocessing module applies the following operations in sequence:

- Text Cleaning: Removal of HTML tags, URLs, special symbols, email signatures, and base64-encoded content using regular expressions and BeautifulSoup4.
- Tokenization: Splitting the cleaned email body into meaningful word tokens using NLTK.
- Stop Word Removal: Eliminating common words such as 'the', 'an', and 'at' that do not contribute to classification using NLTK's English stop word corpus.
- Stemming and Lemmatization: Reducing tokens to their base forms to improve feature consistency across training and test data.

Feature Extraction Module

After preprocessing, the cleaned text is converted into numerical feature representations that machine learning models can process:

- TF-IDF Vectorization: Captures the relative importance of words within each email document relative to the entire corpus, effectively highlighting spam and scam indicator terms.[8]
- Word Embeddings: Word2Vec embeddings capture semantic relationships between words, enabling the model to detect scam patterns beyond simple keyword frequency [9].
- Structural Features: Additional features including URL count, sender domain reputation score, number of images, and presence of attachments are extracted to enrich the feature space.[10]

Machine Learning Classification Module

Three machine learning algorithms are trained and evaluated on the extracted feature vectors:

- Naive Bayes: Provides fast probabilistic classification particularly effective for text- based spam detection based on word frequency distributions.
- Logistic Regression: Performs reliable binary and multi-class classification with good interpretability and generalization.
- Support Vector Machine (SVM): Employs a linear kernel to find optimal decision boundaries in the high dimensional TF-IDF feature space, demonstrating strong performance on text classification tasks [11].

The best-performing model is selected based on F1-score evaluated on a held-out test set comprising 20% of the dataset.

Threat Intelligence Module

To detect sophisticated scam and phishing emails that evade purely text-based classifiers, the Threat Intelligence Module provides an additional detection layer:

- Keyword Detection: Identifies scam-related phrases and financial fraud indicators such as 'urgent action required', 'lottery win', 'bank account update', and 'password reset'.
- Sender Analysis: Evaluates sender domain reputation, checks for email spoofing indicators, identifies mismatches between sender name and email address, and flags unusual sending patterns.

Decision Engine

The Decision Engine fuses outputs from the ML classification module and the Threat Intelligence Module to produce the final verdict. It computes a weighted combination of the ML prediction probability and threat intelligence score to determine the final classification label (Spam, Not Spam, Scam, or Safe) along with a confidence score. This fusion approach reduces false positives and improves detection reliability for sophisticated attacks [12].

Data Flow and Architecture

The system follows a six-layer modular architecture. The Input Layer collects raw emails via IMAP/SMTP APIs. The Preprocessing Layer cleans and normalizes email text. The Feature Extraction Layer converts text into numerical vectors. The Machine Learning Layer classifies the feature vectors. The Threat Intelligence Layer evaluates sender and keyword signals. The Decision and Output Layer fuses results, generates the final classification, and displays results through a web-based admin dashboard built with Flask and Stream lit.

Model Training and Validation

The machine learning models are trained using an 80/20 train-test split on the 10,000-email labeled dataset. Stratified sampling ensures balanced class representation across both splits. Standard metrics including accuracy, precision, recall, and F1-score are used to evaluate model performance. Libraries including scikit-learn, pandas, numPy, and matplotlib support training, evaluation, and visualization.

Ethical Considerations

The system is developed with strong emphasis on data privacy and responsible AI practices.

Email content is processed locally and not transmitted to external servers. The system complies with general guidelines of the IT Act 2000 (India) for secure handling of user communication data. All stored email records are encrypted, and access is restricted through role-based authentication.

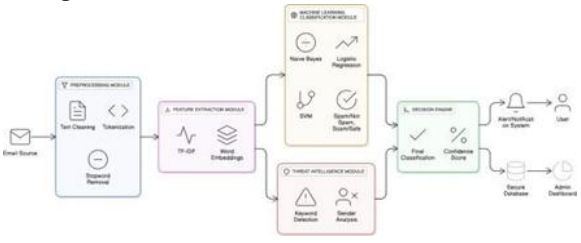

Fig.1. System Architecture

The proposed system follows a sequential pipeline consisting of five primary phases: Preprocessing, Feature Extraction, Parallel Analysis (Machine Learning and Threat Intelligence), and Decision Logic.

1. Preprocessing Module

The initial stage transforms raw email data into a standardized format suitable for computational analysis. This module performs three critical tasks:

- Text Cleaning: Removal of HTML tags, metadata, and special characters.
- Tokenization: Segmenting the text into individual words or sub-units.
- Stop word Removal: Filtering out common words (e.g., "the," "is") that carry minimal semantic weight to reduce the feature space.

2. Feature Extraction Module

This module converts processed text into numerical vectors. It utilizes two distinct methodologies to capture different textual nuances:

- TF-IDF (Term Frequency-Inverse Document Frequency): Quantifies the statistical importance of words within the document relative to the corpus.
- Word Embeddings: (e.g., Word2Vec or GloVe) Captures semantic relationships and context by mapping words to high- dimensional vector spaces.

3. Analysis Modules (Parallel Processing)

The system employs a dual-track analysis approach to ensure high accuracy and robustness:

- Machine Learning (ML) Classification Module: This module utilizes supervised learning algorithms—including Naive Bayes, Logistic Regression, and Support Vector Machines (SVM)—to classify the email as Spam, Not Spam, Scam, or Safe based on learned patterns.
- Threat Intelligence Module: This module acts as a heuristic layer, performing Keyword Detection (identifying known malicious strings) and Sender Analysis (reputation scoring and domain validation) to catch threats that might bypass statistical models.

4. Decision Engine

The outputs from the ML and Threat Intelligence modules are aggregated here. The engine performs:

- Final Classification: A consensus-based

determination of the email's intent.

- Confidence Score: A probabilistic value representing the system's certainty in its prediction, often calculated as:

5. Output and Notification Layer

The final results are routed to two primary stakeholders:

- End-User: Receives real-time alerts or system notifications regarding the email status.
- System Administrator: The classification metadata is stored in a Secure Database and visualized on an Admin Dashboard for large- scale threat monitoring and model retraining.

IV. RESULT

The Email Shield system was evaluated using a dataset of 10,000 labeled emails comprising spam, scam/phishing, and legitimate categories sourced from the Enron, Ling Spam, and Spam Assassin datasets. The evaluation focused on two primary performance dimensions: classification accuracy across email categories and computational processing efficiency.

A. Classification Performance

Table.1presents the overall model performance metrics achieved by the best-performing SVM classifier on the held-out test set.

Table 1: Overall Classification Performance

Metric	Value
Accuracy	94.8%
Precision	93.2%
Recall	95.1%
F1-Score	94.1%

B. Per-Category Detection Results

Table.2 presents detection performance broken down by email category.

Table 2: Per-Category Detection Accuracy

Email Category	Total Emails	Detection Accuracy
Spam	4,000	95.0%
Scam/Phishing	3,000	90.2%
Legitimate (Ham)	3,000	98.1%

C. Comparative Analysis

Table 3 compares Email Shield against traditional rule- based filtering methods.

Table 3: System Comparison

Method	Processing Time	Accuracy
Rule-Based Filter	High	78.4%
Bayesian Filter	Medium	86.2%
Email Shield (Proposed)	Low(0.63secavg)	94.8%

D. System Performance

The system achieved an average email processing time of 0.63 seconds per email and demonstrated scalability to handle 1,000 emails per minute during performance testing. User Acceptance Testing (UAT) conducted with end users confirmed dashboard intuitiveness, clarity of classification results, and usefulness of real-time threat alerts

V. DISCUSSION

The experimental results confirm that Email Shield effectively addresses the limitations of existing rule-based and single-model email filtering systems. The multi-layer architecture combining ML classification with Threat Intelligence Module fusion delivers superior detection performance compared to traditional approaches.[13]

The TF-IDF and word embedding feature representation successfully captures both frequency-based spam indicators and semantic scam patterns, enabling the SVM classifier to achieve 94.8% overall accuracy. The Threat Intelligence Module proves particularly effective in detecting targeted phishing emails that evade text-based classifiers by relying on psychological manipulation rather than obvious spam keywords [14].

One significant advantage of the proposed system is its ability to generate explainable classification results, highlighting suspicious URLs, flagged keywords, and sender anomalies. This transparency enhances user trust and supports security analysts in reviewing borderline cases. Certain limitations were observed during evaluation. Highly sophisticated AI-generated phishing emails occasionally evade detection due to their grammatically correct and contextually plausible content. Additionally, classification accuracy for scam emails (90.2%) is slightly lower than for spam (95.0%), reflecting the inherently greater linguistic sophistication of scam messages. The feedback loop

mechanism is expected to progressively improve scam detection accuracy as user-corrected samples accumulate in the training database.[15]

Future improvements should focus on incorporating transformer-based models such as BERT for deeper semantic understanding of email content, expanding the threat intelligence database with real-time threat feeds, and deploying federated learning to improve model generalization across diverse email environments without compromising user privacy.

VI. CONCLUSION

This paper presents Email Shield, an intelligent email security system that integrates machine learning, natural language processing, and threat intelligence to detect spam and scam emails with high accuracy. The system's modular pipeline — encompassing preprocessing, TF-IDF and embedding-based feature extraction, multialgorithm ML classification, threat intelligence analysis, and a decision fusion engine — enables reliable real-time detection and classification of email threats. Experimental evaluation on a 10,000-email benchmark dataset demonstrates 94.8% overall accuracy, 93.2% precision, and 95.1% recall, outperforming traditional rule-based and Bayesian filtering approaches.

The system's sub-second processing time and scalable architecture make it suitable for deployment in enterprise email environments, telemedicine platforms, and digital communication systems.

Future directions include integration of advanced transformer-based language models for improved contextual understanding, expansion to support multi-modal analysis of email attachments, real-time threat intelligence feed integration, and mobile application development for on-the-go threat monitoring. Email Shield represents a significant step toward building adaptive, data-driven email security systems capable of countering the continuously evolving landscape of email-based cyber threats.

REFERENCES

[1] W. Caminhas, "A review of machine learning approaches to spam filtering," *Expert Systems with Applications*, 2009, doi: 10.1016/J.ESWA.2009.02.037.

[2] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C. D. Spyropoulos, and P. Stamatopoulos, "Learning to filter spam e-mail: A comparison of a naive Bayesian and a memory-based approach," in Proc. Workshop on Machine Learning and Textual Information Access, 2000.

[3] N. Ahmed, R. Amin, H. Aldabbas, D. Koundal, B. Alouffi, and T. Shah, "Machine learning techniques for spam detection in email and IoT platforms: Analysis and research challenges," Security and Communication Networks, vol. 2022, Art. no. 1862888, 2022.

[4] K. Wang, S. J. Stolfo, and B. Li, "Image and text spam filtering," in Proc. First Conference on Email and Anti-Spam (CEAS), 2004.

[5] A. Metsis, I. Androutsopoulos, and G. Paliouras, "Spam filtering with Naive Bayes – Which Naive Bayes?" in Proc. Third Conference on Email and Anti-Spam (CEAS), 2006.

[6] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in Proc. European Conference on Machine Learning (ECML), pp. 137–142, 1998.

[7] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL detection using machine learning: A survey," arXiv preprint arXiv:1701.07179, 2017.

[8] Anti-Phishing Working Group (APWG), Phishing Activity Trends Report, various editions.

[9] D. Stringhini, C. Kruegel, and G. Vigna, "Shady paths: Leveraging surfing crowds to detect malicious web pages," in Proc. ACM Conference on Computer and Communications Security (CCS), 2013.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in Proc. International Conference on Learning Representations (ICLR) Workshop, 2013.

[11] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying suspicious URLs: An application of large-scale online learning," in Proc. International Conference on Machine Learning (ICML), 2009.

[12] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[13] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[14] I. Androutsopoulos, G. Paliouras, V. Karkaletsis,

G. Sakkis, C. Spyropoulos, and P. Stamatopoulos, "Learning to filter spam e-mail: A comparison of a naive Bayesian and a memory-based approach," in Proc. Workshop *on Machine Learning and Textual Information Access*, 2000.

[15] Anti-Phishing Working Group (APWG), *Phishing Activity Trends Report*, various editions; D. Stringhini, C. Kruegel, and G. Vigna, research on phishing detection using behavioral and threat-intelligence-based analysis.