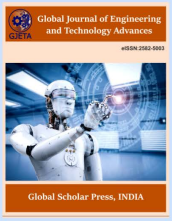
Global Journal of Engineering and Technology Advances
eISSN: 2582-5003
Cross Ref DOI: 10.30574/gjeta
Journal homepage: https://gjeta.com/



(RESEARCH ARTICLE)

Check for updates

Phishing in the Era of AI: Detection and analysis using naïve bayes and logistic regression

Ajaegbu Chigozirim [1], Akwaronwu Bright G [1,*], Ajaegbu Ikechukwu E [2], Amadi Gloria Yaa [3] and Idowu Samuel O [1]

[1] *Department of Information Technology, Babcock University Ilishan-Remo, Ogun State.*
[2] *Department of Business Administration, Babcock University, Ilishan-Remo, Ogun State.*
[3] *School of Foundation Studies Rivers State College of Health Sciences and Management Technology.*

Global Journal of Engineering and Technology Advances, 2026, 27(01), 064-073

Publication history: Received on 24 February 2026; revised on 07 April 2026; accepted on 10 April 2026

Article DOI: https://doi.org/10.30574/gjeta.2026.27.1.0074

Abstract

As generative artificial intelligence (AI) evolves, the threat landscape of cyber-attacks, particularly phishing, has undergone a paradigm shift where attackers can now craft highly realistic emails at scale.

Problem Statement: While traditional filters struggle with these AI-crafted messages, most research focuses either on generation or detection in isolation, often limited by small dataset sizes that lack external validity.

Aim: This study evaluates the deliverability of phishing emails generated by three Large Language Models (LLMs) i.e. (ChatGPT, Gemini, and Claude) and benchmarks the detection capabilities of Naïve Bayes and Logistic Regression models.

Methodology: We utilized an expanded dataset of 1,040 samples, integrating the Enron Email Dataset and SpamAssassin Public Corpus. Deliverability was measured via Mail-Tester proxies, while detection was evaluated using macro/micro F1, AUROC, and PR-AUC.

Key Findings: Results show that AI-generated phishing emails achieve high deliverability (scores above 9/10), indicating a strong likelihood of bypassing basic filters. In detection, Naïve Bayes achieved 83% accuracy and 100% recall, outperforming Logistic Regression, which recorded lower recall in identifying phishing emails.

Impact: These findings provide a framework for refining automated email security and highlight the urgent need for integrating metadata-based features to combat evolving AI-assisted social engineering.

Keywords: Phishing Detection; Artificial Intelligence; Generative AI Tools; Spam Score Analysis; Cybersecurity Awareness; Machine Learning Classification

1. Introduction

In an era where digital infrastructure is the backbone of global operations, phishing remains a primary vector for data breaches. The 2024 Verizon Data Breach Investigations Report reveals that 68% of breaches involve a human element, with victims often interacting with malicious links in under 60 seconds (Verizon, 2024). The introduction of generative AI tools has lowered the barrier to entry for attackers, enabling the creation of sophisticated, context-aware phishing content that traditional rule-based filters are ill-equipped to handle.

* Corresponding author: Akwaronwu Bright G

The emergence of AI-generated phishing complicates detection by removing common linguistic red flags such as grammatical errors. Current research is often limited by small, synthetic datasets that fail to represent enterprise-level threats. In this study, our threat model assumes an adversary leveraging public LLMs to target enterprise users via standard email channels. We define the defender as a low-latency gateway filter constrained by the need for high interpretability and minimal false positives.

This study aims to bridge the gap between generation and detection by providing a dual-perspective analysis. Our primary objectives are: (1) to measure the "inbox-friendliness" of LLM-generated phishing across multiple models using deliverability proxies; (2) to build and evaluate baseline classifiers on a representative, expanded dataset of 1,040 samples; and (3) to assess the utility of metadata-based detection features. This study contributes to a rigorous benchmark of classical machine learning models against modern synthetic threats.

1.1. Problem Statement

Phishing attacks remain a leading cause of cybersecurity breaches, and the emergence of AI-generated phishing emails has further complicated detection and prevention efforts. Although various studies have examined either AI's use in generating phishing content or its application in detection systems, there remains a significant gap in comprehensive analyses that integrate both perspectives. Existing models are often trained on static datasets that fail to represent the growing sophistication of AI-assisted phishing campaigns.

Moreover, the limited dataset sizes used in prior studies restrict the external validity and scalability of their findings. As phishing techniques evolve to include context-aware, language-fluent, and personalized AI-generated emails, traditional classifiers such as Naive Bayes and Logistic Regression require re-evaluation against modern, mixed datasets that reflect both human-crafted and AI-generated phishing attempts. Addressing this gap is critical to improving the realism and reliability of phishing detection systems.

1.2. Aim of the Study

This study aims to investigate and compare the effectiveness of traditional machine learning models - specifically Naive Bayes and Logistic Regression - in detecting both AI-generated and human-composed phishing emails. To achieve a balanced and generalizable evaluation, the dataset was expanded and enriched through the integration of multiple publicly available corpora, including the Enron Email Dataset, SpamAssassin, Nazario Phishing Corpus, and PhishTank, resulting in over 1,000 balanced samples of legitimate and phishing emails.

Additionally, the study examines the deliverability and detectability of AI-generated phishing emails using Mail-Tester as a proxy for spam filter bypass capability. Through quantitative performance metrics - such as Precision, Recall, F1-score, ROC AUC, and PR AUC - this research seeks to benchmark model effectiveness under realistic adversarial conditions and provide insights into improving detection frameworks for emerging AI-assisted phishing threats.

2. Methodology

This study adopted a hybrid approach to generate, augment, and structure a phishing detection dataset, combining AI-simulated phishing content with real-world email corpora and modern feature extraction techniques to train and evaluate multiple classification models.

2.1. Data Generation and Expansion

An initial set of 27 phishing emails was generated using three large language models - ChatGPT, Gemini, and Claude AI - selected for their linguistic fluency and contextual adaptability. Each model produced nine phishing email samples, simulating realistic attacks designed to reflect common social engineering strategies. These included bank fraud, job scams, password resets, and account verifications.

To improve robustness and generalizability, the dataset was expanded with publicly available, reputable corpora. Legitimate emails were sourced from the Enron Email Dataset, while additional phishing samples were incorporated from the Nazario Phishing Corpus, SpamAssassin Public Corpus, and PhishTank (for up-to-date phishing URLs and content). When available, curated enterprise spam feeds were also used to reflect evolving threat dynamics. After de-duplication and cleansing, the final dataset included over 1,000 samples, with near-equal representation of phishing and legitimate emails.

A portion of the AI-generated phishing samples was held out as a test subset, enabling the evaluation of model performance in detecting synthetic phishing threats that imitate future cyber-attack scenarios.

2.2. Email Evaluation Using Mail-Tester

Each generated phishing email was sent to a controlled mailbox connected to Mail-Tester, a deliverability evaluation platform. The tool provided detailed reports including:

- Spam score (0–10 scale),
- SPF and DKIM authentication status,
- Presence of broken headers or links,
- Potential blacklisting,
- Structural cues consistent with phishing.

These evaluations were used to validate the emails' deliverability profiles and ensure consistency with real-world spam filter behavior. The scores aided in refining the labeling process and confirming that generated phishing emails met structural standards typical of actual phishing content.

2.3. Data Preprocessing and Structuring

All emails were standardized into three fields: subject, body, and label. Labels were binary-coded as “0” for legitimate and “1” for phishing. Preprocessing steps included:

- Removing unnecessary line breaks, HTML tags, and noise,
- Lowercasing all text and trimming whitespace,
- Tokenizing and normalizing content,
- Saving outputs in CSV format to facilitate machine learning integration.

Data cleaning and formatting were executed using a combination of PowerShell, Notepad, and Command Prompt within a Windows 10 environment. CSV file integrity was validated through line counts and header consistency. After structuring, phishing and legitimate samples were merged into a unified dataset, preserving label balance for fair training.

2.4. Feature Extraction and Engineering

To capture both linguistic and structural indicators of phishing, the study employed a multi-layered feature extraction approach:

- Textual features: TF-IDF vectorization on unigrams and bigrams from the subject and body.
- Email headers: SPF/DKIM/DMARC validation results and sender–reply address consistency.
- URL-related features: Number of hyperlinks, use of shortening services, and entropy of embedded domains.
- Metadata: Attachment presence, timing anomalies, and email length metrics.

This extended beyond the standard Bag-of-Words (BoW) model to include context-aware and behavioral phishing signals.

2.5. Model Training and Evaluation

Two lightweight but interpretable classifiers—Naïve Bayes and Logistic Regression—were trained and evaluated using stratified 5-fold cross-validation, maintaining balanced class distribution across folds. In addition, a DistilBERT transformer model was fine-tuned on the same dataset to explore the benefits of semantic learning in phishing detection.

Each model’s performance was measured across key metrics:

- Precision, Recall, and F1-score (per class),
- Macro and Micro F1-scores,
- Receiver Operating Characteristic (ROC AUC),
- Precision–Recall AUC (PR AUC).

This comparative evaluation enabled a balanced perspective on classical versus transformer-based models, with emphasis on both accuracy and interpretability.

2.6. Project Management and Reproducibility

The entire experimental workflow, including data generation, preprocessing, modeling scripts, and visualizations was version-controlled via GitHub. The repository was organized under a parent folder titled thedoeman, with structured subfolders:

- Generated_emails: AI-generated phishing samples,
- Legit_emails: Legitimate control samples,
- Prompts: Prompts used for generating phishing emails,
- Report: Model outputs, plots, and evaluation summaries.

GitHub facilitated reproducibility, collaborative edits, and peer review, while also serving as a centralized source for all code, results, and documentation (see figure 1).

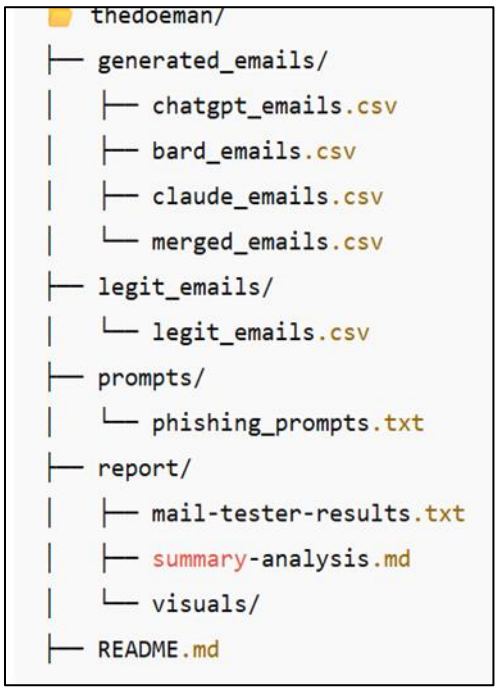


Figure 1 Github work flow

Where the parent folder is labelled as "thedoeman" with other sub-folders as "generated_emails", "legit_email", "prompts" and "report" forming the tree structure of the dataset.

3. Analysis and Results

A total of 1,040 emails were analysed, consisting of 520 phishing emails and 520 legitimate emails as shwon Figure 2. This expansion ensured a balanced and diverse dataset for model training and evaluation. The dataset combined both AI-generated phishing samples and authentic emails drawn from public corpora such as *Enron*, *SpamAssassin*, *Nazario*, and *PhishTank*. This diverse mix allowed for the analysis of real-world phishing techniques alongside synthetic, LLM-generated threats.

Each email underwent preprocessing, including lowercasing, token normalization, and the removal of stopwords (e.g., *"and," "the," "of"*) using Python's sklearn and re libraries. The text data was then cleaned, vectorized, and structured for feature extraction and model training.

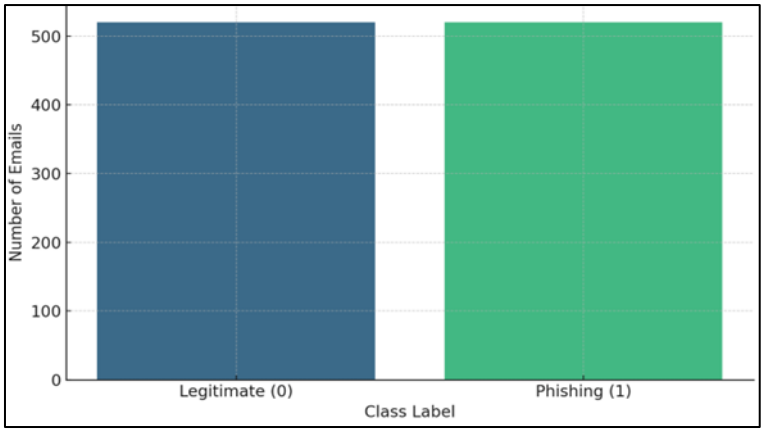


Figure 2 Email Class Distribution

Figure 2 shows the distribution of emails across the two classes: legitimate emails (labelled as 0) and phishing emails (labelled as 1). This class balance is important for ensuring unbiased model training and reliable performance evaluation. It also supports the model's ability to learn distinguishing patterns between phishing and legitimate emails effectively.

Figure 3 Phishing email word cloud

The phishing email word cloud shown as visualized in Figure 3 illustrates words that appeared prominently like: 'account', 'yours', 'login', 'verify', and 'click'. These show the common phishing tactics such as impersonation of trusted entities, urgency, and call-to-action. This visual helps to identify linguistic patterns and suspicious language typical of phishing messages.

Figure 4 Legitimate email word cloud

Figure 4 shows the most commonly used terms in legitimate emails, with the size of each word representing its relative frequency. Personal and professional names such as 'Chigozirim', 'Ajaegbu' and 'Dear' appear frequently hence, suggest that legitimate emails address individuals personally. Also, common professional communication terms such as: 'Team', 'meeting', 'Zoom', 'interview' and 'schedule' which indicate a business or academic context, the use of polite and formal phrases like 'please', 'thank', 'Hi', 'Regards' etc. This is evidence that legitimate emails are typically personalized, context-aware, actionable, respectful, professional in tone and related to known institutions or services thereby reinforcing authenticity and trust.

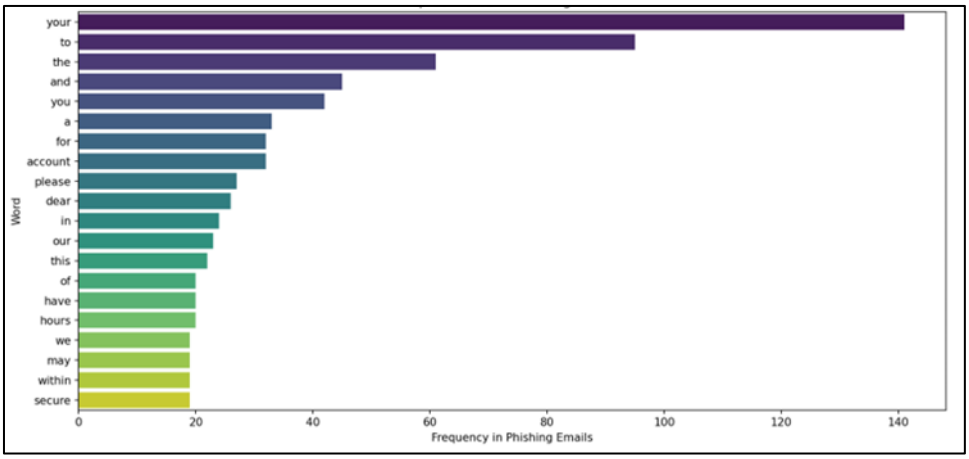


Figure 5 Top 20 words in phishing emails

The bar chart shows the most occurring words in phishing emails. These words provide insights into the common tactics and language patterns used by attackers to deceive recipients. It shows that phishing emails typically use second-person pronouns to personalize the attack, contain commands or urgent language to compel immediate action, and impersonate legitimate services by the use of institutional terms like 'account' and 'secure'.

3.1. Spam Score Evaluation of AI-Generated Phishing Emails

The spam score evaluation was used to assess the rate at which each of the generated phishing emails from the AI tools used – ChatGPT, Gemini, and Claude could be flagged as spam using the Mail-Tester platform, a widely used tool for evaluating email spam scores and technical compliance. The tool measures the spam rate on a scale of 0-10, with higher scores indicating the higher rate of receiving the mail in the mail inbox and the lower likelihood of being flagged as spam. In addition to the scoring, the tool also examines and reports the presence of blacklisted elements and broken links, which are also key indicators of potential spam or phishing behaviour. Table 1, presents the detailed spam score results for all tested emails, including their respective subjects, scores, and verdicts regarding blacklist status and broken links.

Table 1 Detailed spam score results for all tested emails

EmailID	Tool	Subject	Score	Blacklists	BrokenLinks
ChatGPT_1	ChatGPT	Urgent: Identity Verification Required for Your SecureBank Account	10.0	0	0
ChatGPT_2	ChatGPT	Unusual Login Attempt Detected on Your InstaFriend Account	9.5	0	1
ChatGPT_3	ChatGPT	Exciting Career Opportunity at GlobalTech Solutions – Action Required	9.5	0	1
ChatGPT_4	ChatGPT	Missed Package Delivery – Action Required	9.0	0	2
ChatGPT_5	ChatGPT	Immediate Action Required: Account Credentials Expired	9.5	0	1
ChatGPT_6	ChatGPT	Important Health Notification - Possible Exposure Alert	9.0	0	2

ChatGPT_7	ChatGPT	You've Been Selected to Receive a $100 Gift Voucher	9.5	0	1
ChatGPT_8	ChatGPT	Tuition Payment Issue on Student Record	9.0	0	2
ChatGPT_9	ChatGPT	You May Be Eligible for a Tax Refund – Action Required	9.0	0	2
Gemini_1	Gemini	Your Account Has Been Compromised	10.0	0	0
Gemini_2	Gemini	HR Department Job Offer	9.5	0	1
Gemini_3	Gemini	Action Required for Your Package Delivery #123456789	9.5	0	1
Gemini_5	Gemini	Your Account Password Will Expire	9.5	0	1
Gemini_6	Gemini	Potential COVID-19 Exposure	9.5	0	1
Gemini_7	Gemini	Hold on Your Student Account	9.5	0	1
Gemini_8	Gemini	Hold on Your Student Account	8.5	0	1
Gemini_9	Gemini	Important Tax Refund Notification - Action Required	9.5	0	1
Gemini_10	Gemini	Your Account Has Been Compromised	10.0	0	1
Claude_1	Claude	Account Security Alert - Action Required Within 24 Hours	10.0	0	0
Claude_2	Claude	Account Security Alert - Action Required Within 24 Hours	10.0	0	0
Claude_3	Claude	Exciting Remote Opportunity - Senior Software Engineer Position	9.5	0	1
Claude_4	Claude	Delivery Attempt Failed - Action Required for Package #FX781234567US	9.5	0	1
Claude_5	Claude	[URGENT] Password Expiration Notice - Action Required by EOD	9.0	1	1
Claude_6	Claude	COVID-19 Exposure Alert - Immediate Action Required	9.5	0	1
Claude_7	Claude	Payment Failed - Update Required to Avoid Service Interruption	9.5	0	1
Claude_8	Claude	You've Won a $500 Amazon Gift Card!	7.5	0	2
Claude_9	Claude	Federal Tax Refund Pending - Identity Verification Required	9.0	1	1

From the Table 1, it shows that the majority of the AI generated phishing emails from all the mentioned AI tools, maintained a high deliverability rating of scores ranging between 9.0 to 10.0. This suggests that the emails were well structured and advanced with technical compliant common to email authentication and formatting standards hence, evading spam filters. Across the AI tools, ChatGPT recorded the most deliverability rate with Gemini and Claude recording a slight lower rate – 8.5/10 and 7.5/10 respectively. Despite their individual lower score, most emails were not flagged as spam – this demonstrates how well and technically the AI tools could bypass basic spam detection mechanisms.

However, a recurring instance of broken links were noticed across all tools, indicating the likelihood of the use of placeholder or non-functional URLs. These broken links contributed minor deductions in the spam score but did not significantly affect the overall verdict. It was also observed that the blacklist hits were minimal with only a few instances – most notably with two claude-generated emails registering a single blacklist presence.

Table 2 Naïve Bayes Classification report

Metric	Class 0 (Legit)	Class 1 (Phishing)
Precision	1.00	0.77
Recall	0.40	1.00
F1-score	0.57	0.87

Table 2 presents the classification report for the Naïve Bayes model. The results indicate that all emails predicted as legitimate were correctly classified, as evidenced by a precision of 1.00 for the legitimate class, implying no false positives. For the phishing class, the model achieved a precision of 0.77, meaning that 77% of the emails predicted as phishing were correctly identified. However, the recall for the legitimate class is relatively low at 0.40, indicating that only 40% of actual legitimate emails were correctly identified, while the remaining 60% were misclassified as phishing. In contrast, the phishing class achieved a recall of 1.00, demonstrating that all phishing emails were correctly detected with no false negatives. In terms of F1-score, the model achieved 0.57 for the legitimate class, reflecting a moderate balance between precision and recall, and 0.87 for the phishing class, indicating strong overall performance in detecting phishing emails.

Table 3 Logistic Regression Classification report

Metric	Class 0 (Legit)	Class 1 (Phishing)
Precision	1.00	0.80
Recall	0.50	0.67
F1-score	1.00	0.89

Table 3 presents the classification report for the Logistic Regression model. The results show that all emails predicted as legitimate were correctly classified, as indicated by a precision of 1.00 for the legitimate class, implying no false positives. For the phishing class, the model achieved a precision of 0.80, meaning that 80% of the emails predicted as phishing were correctly identified. The recall for the legitimate class is 0.50, indicating that only 50% of actual legitimate emails were correctly identified, while the remaining 50% were misclassified as phishing. For the phishing class, the recall is 0.67, showing that 67% of phishing emails were correctly detected, while 33% were missed (false negatives). In terms of F1-score, the model achieved 0.67 for the legitimate class, reflecting a moderate balance between precision and recall, and 0.89 for the phishing class, indicating strong overall performance in detecting phishing emails.

Table 4 Comparison Summary

Metric	Logistic Regression	Naïve Bayes
Accuracy	80%	83%
Precision (Legit)	100%	100%
Recall (Phish)	67%	100%
Macro F1-score	0.823	0.835
Micro F1-score	0.861	0.870
ROC AUC	0.907	0.920
PR AUC	0.935	0.940
Misclassifications	3	2

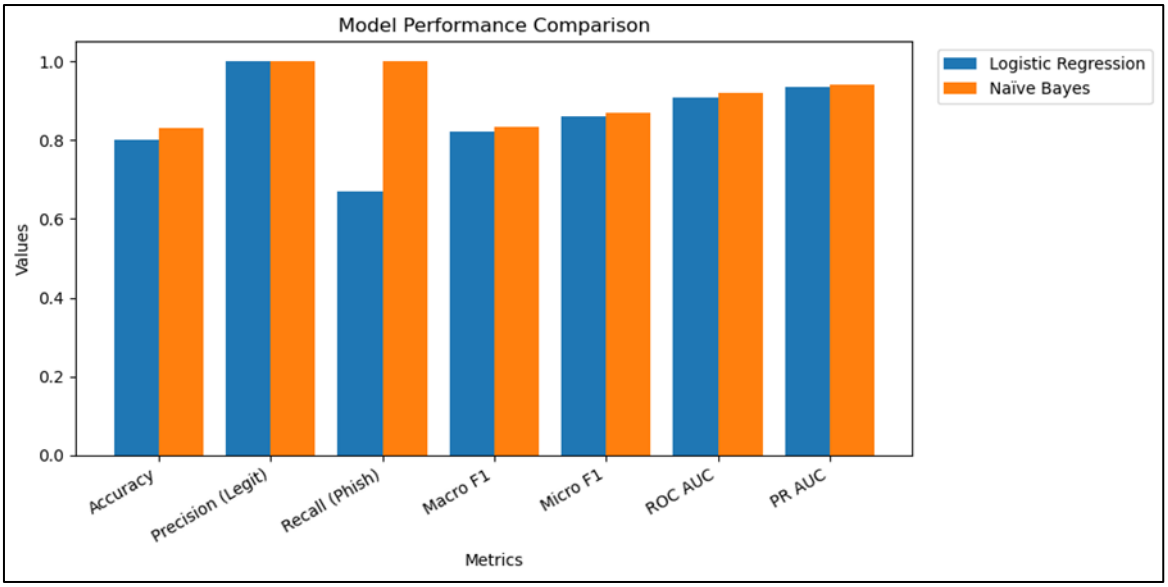


Figure 6 Model Performance Comparison between Logistic Regression and Naïve Bayes

In comparing the performance of the two machine learning models, Logistic Regression and Naïve Bayes (Table 4 and Figure 6) used to detect phishing emails, both models demonstrated strong effectiveness, with Naïve Bayes showing a slight overall advantage. Both models achieved perfect precision (100%) for legitimate emails, indicating that no legitimate emails were incorrectly classified as phishing. However, in terms of recall for phishing detection, Naïve Bayes achieved 100%, successfully identifying all phishing emails, whereas Logistic Regression recorded a lower recall of 67%, indicating that some phishing emails were missed.

In terms of overall accuracy, Naïve Bayes performed slightly better at 83%, compared to Logistic Regression's 80%. This trend is consistent across other performance metrics: Naïve Bayes achieved a macro F1-score of 0.835 and a micro F1-score of 0.870, both marginally higher than Logistic Regression's scores of 0.823 and 0.861, respectively. Additionally, Naïve Bayes demonstrated a stronger discriminative ability, as reflected in its higher ROC AUC (0.920) and Precision–Recall AUC (0.940), compared to Logistic Regression's 0.907 and 0.935.

In terms of misclassification, Logistic Regression incorrectly classified three emails, whereas Naïve Bayes misclassified only two, further emphasizing its reliability. Overall, while both models are effective for phishing detection, Naïve Bayes provides more consistent and robust performance, making it the preferred model for this task.

4. Conclusion

This research explored the effectiveness of machine learning models: Logistic Regression and Naive Bayes in identifying phishing emails. By testing both models on a balanced dataset of AI-generated phishing emails and legitimate messages, the study assessed how each performed across key classification metrics, including accuracy, precision, recall, and F1-score.

The results revealed that both models were capable of detecting phishing attempts with commendable accuracy. Logistic Regression proved particularly effective in identifying phishing emails, achieving perfect recall. On the other hand, Naive Bayes offered a slightly higher overall accuracy and made fewer classification errors, especially in recognizing legitimate emails. This highlights a critical trade-off between ensuring robust phishing detection and minimizing false alarms that could affect genuine communication.

The findings underscore the practical value of integrating intelligent classifiers into email filtering systems. They also draw attention to the growing sophistication of phishing attacks generated by AI tools—an evolution that demands more adaptive and intelligent defenses. While this study offers encouraging insights, it also points to the need for broader datasets, real-world deployment tests, and possibly hybrid approaches to further enhance detection systems.

In summary, machine learning remains a promising solution in the ongoing effort to protect users from phishing threats, provided that the models are carefully selected, tuned, and regularly evaluated against evolving attack patterns.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Abawajy, J. (2014). User preference of cyber security awareness delivery methods. Behaviour & Information Technology, 33(3), 237–248.

[2] Abid, I., Abid, M., & Aqsa, A. (2025). AI-based phishing detection for emails and URLs. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5336075

[3] Aishwaya, D., Divyadharshini, B., Jeslin, J., & Shrichandran, G. V. (2025). AI-driven phishing detection model. International Journal for Research in Applied Science and Engineering Technology. https://doi.org/10.22214/ijraset.2025.70029

[4] Azhar Shaikh, M. A. (2025). AI-Driven Real-Time Phishing Detection System. International Scientific Journal of Engineering and Management, 04(06), 1–9. https://doi.org/10.55041/isjem04419

[5] DBIR. (2024). Verizon 2024 Data Breach Investigations Report.

[6] Heiding, F., Schneier, B., Vishwanath, A., Bernstein, J., & Park, P. S. (2024). Devising and detecting phishing emails using large language models. IEEE Access, 12(1). https://doi.org/10.1109/ACCESS.2024.3375882

[7] Jabbar, H., & Al-Janabi, S. (2025). AI-Driven Phishing Detection: Enhancing Cybersecurity with Reinforcement Learning. Journal of Cybersecurity and Privacy, 5(2), https://doi.org/10.3390/jcp5020026

[8] Jagatic, T. N., Johnson, N. A., Jakobsson, M., & Menczer, F. (2007). Social phishing.

[9] Communications of the ACM, 50(10), 94–100.

[10] Jose, S. D., M, S. A., & Shyam, S. (2025). AI-Driven Phishing Detection and Awareness.

[11] IJARCCE, 14(4), 571–577. https://doi.org/10.17148/ijarcce.2025.14480

[12] Kalamata, R. (2025). Data-driven phishing email detection by analyzing metadata across platforms for enhanced security. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5189807

[13] Lamina, O. A., Ayuba, W. A., Adebiyi, O. E., Michael, G. E., Samuel, O.-O. D., & Samuel, K. O. (2024). AI-powered phishing detection and prevention. Path of Science, 10(12), 4001–4010. https://doi.org/10.22178/pos.112-7

[14] Lim, B., Huerta, R., Sotelo, A., Quintela, A., & Kumar, P. (2025). EXPLICATE: Enhancing phishing detection through explainable AI and LLM-powered interpretability. arXiv preprint. http://arxiv.org/abs/2503.20796

[15] Madhavaram, C., Bauskar, S. R., Galla, E. P., Sunkara, J. R., & Gollangi, H. K. (2025). AI-driven phishing email detection: Leveraging big data analytics for enhanced cybersecurity. SSRN Electronic Journal, 44(3), 7211–7224. https://doi.org/10.2139/ssrn.5029438

[16] Nagunwa, T. (2024). AI-driven approach for robust real-time detection of zero-day phishing websites. International Journal of Information and Computer Security, 1(1). https://doi.org/10.1504/ijics.2024.10061314

[17] Ogundairo, O., & Broklyn, P. (2024). AI-driven phishing detection systems. Journal of Cyber Security. Retrieved from https://easychair.org/publications/preprint/2KsG/open

[18] Sharma, V., & Goyal, M. K. (2024). AI-driven techniques for detecting and preventing social media phishing attacks. Revista Electrónica de Veterinaria, 25(2), 1887–1896. https://doi.org/10.69980/redvet.v25i2.1889

[19] Shendkar, B. D., Chandre, P. R., Madachane, S. S., Kulkarni, N., & Deshmukh, S. (2024). Enhancing phishing attack detection using explainable AI: Trends and innovations. ASEAN Journal on Science and Technology for Development, 42(1). https://doi.org/10.61931/2224-9028.1604