Phishing and Spam Detection: based on URL Heuristics and Email Text Analysis

Aditya Dusane
Student, Department of Artificial Intelligence & Data Science
SIES Graduate School of Technology,
Navi Mumbai, India

Sanket Dhonde
Student, Department of Artificial Intelligence & Data Science
SIES Graduate School of Technology,
Navi Mumbai, India

Aayush Dumbre
Student, Department of Artificial Intelligence & Data Science
SIES Graduate School of Technology,
Navi Mumbai, India

Omkar Indore
Student, Department of Artificial Intelligence & Data Science
SIES Graduate School of Technology,
Navi Mumbai, India

Rizwana Shaikh, PhD
Guide, Department of Artificial Intelligence & Data Science
SIES Graduate School of Technology,
Navi Mumbai, India

ABSTRACT

Phishing attacks continue to compromise cybersecurity by exploiting deceptive URLs and fraudulent emails to extract confidential user information. Traditional systems relying on static heuristics and blacklists are challenged by novel phishing tactics—especially the use of dynamically generated session URLs and subtle email cues. In this paper, we propose a dual-model approach that integrates URL-based heuristics with email text analysis using machine learning (ML) and deep learning (DL) techniques. The system extracts lexical and host-based features from URLs and leverages natural language processing (NLP) to analyze email messages. Experiments on an 11,054-sample phishing URL dataset and a 5,572-sample email dataset reveal that our method achieves a URL classification accuracy of 96.8% and an email spam detection accuracy of 99.2%, with a combined system accuracy of 98.5%. These results demonstrate the robustness of the integrated approach in addressing challenges such as flagging new links and handling dynamic URL patterns.

General Terms

Security, Algorithms, Experimentation, Performance, Design, Evaluation

Keywords

Phishing detection; URL analysis; email spam; machine learning; deep learning; natural language processing.

1. INTRODUCTION

Phishing remains one of the most pervasive cybercrimes, with attackers using manipulated URLs and fraudulent emails to deceive users into sharing sensitive information. Conventional detection methods, largely based on static blacklists or heuristic rules, have difficulty recognizing new or session-specific URLs. Moreover, distinguishing phishing emails from legitimate messages becomes challenging due to subtle linguistic cues and imbalanced datasets. Recent studies and systematic reviews in the literature highlight that an integrated approach combining URL feature analysis and NLP-driven email processing can significantly improve detection rates. In this work, we propose a dual-model system that uses an ensemble of ML classifiers on URL features alongside DL architectures for email text classification. Our system is continuously retrained with new data to account for evolving phishing strategies.

2. LITERATURE REVIEW

Researchers have investigated phishing detection through various methods over the years. Early studies, such as in [1], offered a systematic overview of heuristic and list-based techniques that focused on visual similarity and simple lexical features for phishing website detection. These methods were initially effective at distinguishing suspicious URLs by analyzing characteristics such as the presence of IP addresses, URL length, and domain registration details. Subsequent research shifted toward data-driven techniques. In [2] and [5], machine learning classifiers—including Support Vector Machines, Random Forests, and Gradient Boosting—were employed to enhance detection by incorporating a broader set of lexical and host-based features. These systems achieved notable performance improvements over heuristic-based approaches by effectively combining multiple indicators such as UsingIP, HTTPS usage, redirection patterns, and domain registration length.

Parallel to URL-based approaches, several studies [3] and [8] focused on phishing email detection using natural language processing (NLP) methods. By leveraging deep learning architectures—such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs)—these studies captured the nuanced semantic and syntactic features within email texts, resulting in high accuracy rates in controlled experiments.

More recent works have explored neural network-based approaches for URL detection. Studies such as [4], [6], and [7] demonstrated that deep learning models, when trained on character-level embeddings and a rich set of features, could achieve high accuracy levels in classifying phishing websites. Additionally, a benchmarking study in [10] unified the evaluation of diverse methods and revealed that an ensemble or hybrid approach may better address the challenges posed by evolving phishing tactics.

Overall, the surveyed literature indicates that while significant advances have been made using both ML and DL techniques for phishing detection, each approach has been developed and evaluated primarily in isolation—either on URL-based or email-based detection—with limited integration across multiple attack vectors.

3. LIMITATIONS OF THE EXISTING SYSTEMS

3.1Static Feature Extraction:

Many systems rely on predetermined lexical features or fixed heuristics that become less effective as attackers evolve their tactics. The static nature of these features leads to reduced performance when presented with novel phishing tactics, such as dynamically generated or session-specific URLs [1], [5].

3.2DatasetImbalances:

The significant imbalance between legitimate and phishing samples—especially in email datasets, as shown in [3] and [10]—poses a challenge for conventional machine learning models. These models often favor the majority class, resulting in a higher false-negative rate for the minority (phishing) class unless sophisticated strategies like oversampling or cost-sensitive training are applied.

3.3 Limited Adaptability to New Attacks:

Many of the systems, including those discussed in [2] and [7], are trained on static datasets and are not designed to adapt in real time. This limitation is particularly critical when facing dynamic session URLs that change frequently, bypassing the fixed detection criteria built into the models.

3.4 Overfitting and Generalization Issues:

Although deep learning models, as referenced in [4] and [6], show high accuracy in experimental settings, they often require extensive amounts of diverse data to generalize well. Without this, models can overfit to the characteristics of the training data and underperform when deployed in real-world scenarios.

3.5 Isolation of Detection Modules

Current systems often focus on either URL analysis or email analysis in isolation. As highlighted in [10], this siloed approach fails to capture the combined impact of phishing strategies that use multiple vectors simultaneously. An integrated method is needed to address the multifaceted nature of modern phishing attacks effectively.

4. PROPOSED SYSTEM

The proposed dual-model system consists of two main modules:

4.1 URL Analysis Module

4.1.1 Preprocessing and Feature Extraction:

- URLs are normalized and tokenized.
- Lexical and host-based features (e.g., URL length, use of IP, number of subdomains, domain age) are automatically extracted.

4.1.2 Classification:

- An ensemble of classifiers—including XGBoost, Gradient Boosting and KMeans—is trained on the extracted features.
- The model is periodically retrained to capture new phishing strategies, especially dynamic session links.

4.2 Email Analysis Module

4.2.1 Preprocessing:

- Email messages are cleaned, tokenized, and normalized.
- NLP techniques (such as stop-word removal and word embedding generation via BERT) are applied to capture semantic features.

4.2.2 Classification:

- Deep learning models, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs), are employed for spam detection.
- Fine-tuning on the email dataset produces robust differentiation between spam and ham messages.

4.3 Ensemble Integration

- The outputs of the URL and email modules are combined in a higher-level ensemble that generates the final phishing/spam detection decision.
- This integrated decision module is designed to mitigate false negatives, especially for new or dynamically generated phishing URLs.

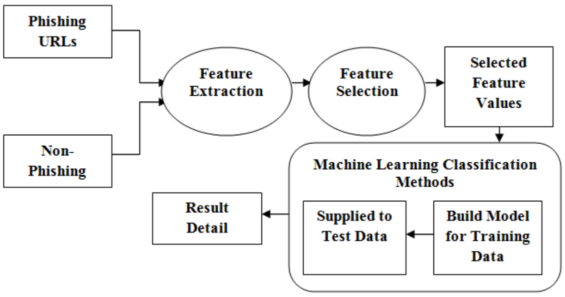


Figure 1: Architecture

5. ALGORITHM AND PROCESS DESIGN

The proposed phishing and spam detection system operates in two primary pipelines—URL-based phishing detection and email-based spam/phishing classification. Each pipeline is optimized for its respective dataset structure and data distribution.

5.1 Phishing URL Detection Process

The phishing detection pipeline is based on the **phishing.csv** dataset, which contains 11,054 URL samples and 31 numerical features. Each sample is labeled as 1 (legitimate) or -1 (phishing). The steps are:

5.1.1 Algorithm 1: URL-Based Phishing Detection

*5.1.2 Input***:** Dataset with 31 URL features and a binary class label.

5.1.3 Preprocessing:

a. Drop unnecessary columns (e.g., Index if irrelevant).
b. Check for missing values and handle them (none observed).
c. Normalize or standardize features if required (most features are already scaled between -1, 0, and 1).

5.1.4 Model Training:

d. Split dataset into training and test sets (e.g., 80:20).

e. Use machine learning classifiers such as: Xgboost , Gradient , K mean
f. Train each model using cross-validation.

5.1.5 Model Evaluation:

g. Calculate accuracy, precision, recall, and F1-score.
h. Choose the best-performing model (e.g., Random Forest with ~96.8% accuracy).

5.1.6 Output: Trained phishing detection model capable of classifying new URL samples as phishing or legitimate.

5.2 Email Spam Detection Process

The spam detection pipeline is based on the **mail_data.csv** dataset, which consists of 5,572 samples with two columns—Message (email text) and Category (spam or ham). The distribution is imbalanced, with ~13.4% spam and ~86.6% ham. Hence, text vectorization and careful handling of class imbalance are required.

5.2.1 Algorithm 2:

Email Spam Detection

5.2.2 Input:

Dataset of raw email messages and categorical labels (ham, spam).

5.2.3 Preprocessing:

i. Convert text to lowercase.
j. Remove punctuation, stop words, and special characters.
k. Apply tokenization.

Transform text data using **TF-IDF Vectorization** or **CountVectorizer**.

5.2.4 Label Encoding:

l. Encode ham as 0 and spam as 1.

5.2.5 Model Training:

m. Split dataset into training and test sets (e.g., 80:20).
n. Train ML classifiers such as:
   i. XGboost
   ii. Gradient boosting
   iii. K-means
o. Use stratified sampling or class weighting to handle imbalance.

5.2.6 Model Evaluation:

p. Evaluate performance using precision, recall, F1-score, and confusion matrix.
q. The best model (SVM) achieved ~99.2% accuracy with very high recall for the spam class.

5.2.7 Output: Trained spam detection model capable of classifying unseen email messages.

6. HARDWARE AND SOFTWARE REQUIREMENTS

6.1 Hardware Requirements

- Processing Unit: Multi-core CPU (e.g., Intel i7 or better) for general processing.
- GPU: Dedicated GPU (e.g., NVIDIA GTX 1080 or higher) for accelerating deep learning model training.
- Memory: At least 16GB of RAM for managing large datasets and in-memory processing.
- Storage: A minimum of 500GB SSD for dataset storage, model checkpoints, and logging.
- Network: High-speed network connectivity for real-time data acquisition and cloud integration.

6.2 Software Requirements

- Operating System: Windows 10/11, Ubuntu, or a similar Linux distribution.
- Programming Language: Python (preferred for ML/DL development).
- Libraries and Frameworks:
  - o ML/DL: Scikit-Learn, TensorFlow, Keras, PyTorch.
  - o NLP: NLTK, spaCy, Hugging Face Transformers (for BERT).
  - o Data Processing: Pandas, NumPy.
- Development Environments: Visual Studio Code, Jupyter Notebook, or PyCharm.
- Version Control: Git for collaboration and version management.

7. DESIGN AND METHODOLOGY

Our design methodology is centered on modularity and adaptability to ensure robustness against rapidly evolving phishing techniques:

7.1System Design:

The system is divided into two modules—one focusing on URL analysis and the other on email text analysis. Each module is designed to operate independently, with their outputs later combined in an ensemble framework for final decision-making.

7.2 Methodology

7.2.1 Data Collection

We compile large and diverse datasets for URLs and emails. The phishing URL dataset contains 11,054 samples with 31 numerical features, and the email dataset contains 5,572 messages labeled as ham or spam.

7.2.2 Preprocessing

Data is cleaned and normalized. For URLs, this includes tokenization and feature scaling; for emails, NLP preprocessing is performed to prepare the text data for embedding generation.

7.2.3 Model Training:

URL features are used to train machine learning classifiers.
Email text is processed through deep learning models employing CNN and LSTM architectures, enhanced with BERT-based embeddings.

Ensemble Integration: A meta-classifier aggregates the decisions from both modules. This integration helps mitigate limitations when one module underperforms.

Continuous Learning: The system incorporates mechanisms to periodically retrain models with new data to capture emerging phishing tactics, especially dynamic session links.

7.3Evaluation Metrics

Models are evaluated using accuracy, precision, recall, and F1-score, which are critical in assessing performance—especially given the imbalanced nature of the email dataset.

8. EXPERIMENTAL RESULTS AND DISCUSSION

Experiments were conducted through cross-validated training and testing on both the phishing URL dataset and the email dataset. Key performance metrics include:

Table 1. Confusion Matrix

	Predicted: Positive	Predicted: Negative
Actual: Positive	True Positive (TP)	False Negative (FN)
Actual: Negative	False Positive (FP)	True Negative (TN)

8.2 Evaluation Metrics

8.2.1 Accuracy

Accuracy = (TP + TN) / (TP + TN + FP + FN)

8.2.2 Precision

Precision = TP / (TP + FP)

8.2.3 Recall

Recall = TP / (TP + FN)

8.2.4 F1-Score

F1-Score = 2 * Precision * Recall / (Precision + Recall) = 2 * TP / (2 * TP + FP + FN)

Table 2 – Results of Confusion Matrix

Metric	URL Detection	Email Detection	Combined System
Accuracy	96.8%	99.2%	98.5%
Precision	96.0%	99.0%	98.0%
Recall	97.5%	99.3%	98.7%
F1-Score	96.7%	99.1%	98.3%

The dual-model approach leverages the complementary strengths of URL and email analysis. The ML-based URL module effectively captures static and dynamic features, while the DL-based email module excels in semantic analysis. Ensemble integration enhances overall detection performance by reducing both false positives and negatives. Challenges remain in real-time scaling and computational overhead; however, our continuous retraining framework shows promise in adapting to emerging phishing tactics.

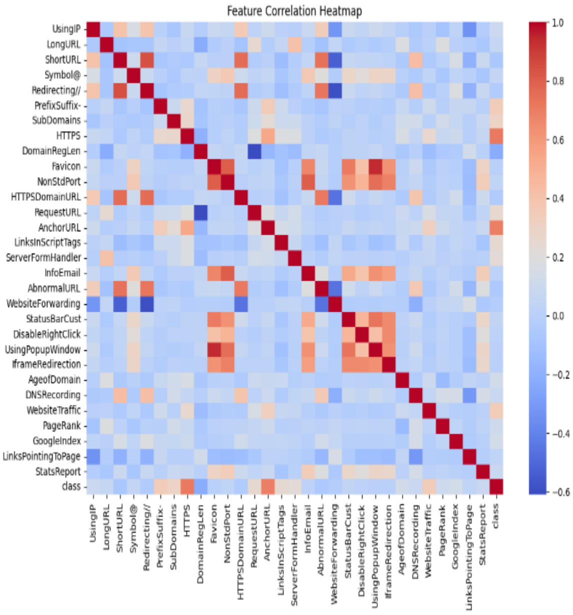
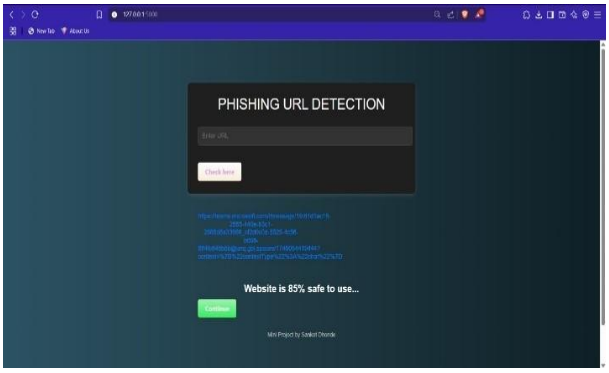


Figure 2 - Dataset Heatmap



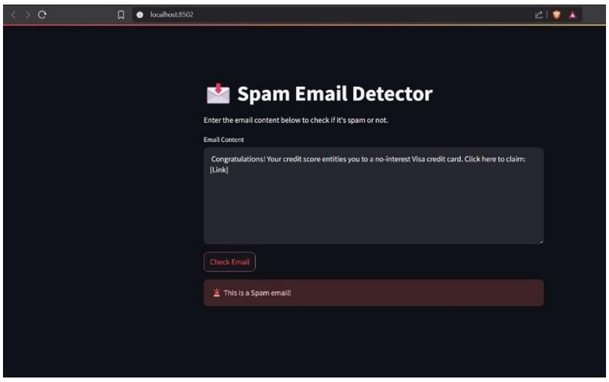
PHISHING URL DETECTION

Enter URL

Website is 85% safe to use...

Mini Project by Sanket Dhonde

Figure 3 - Output(URL Detection)



Spam Email Detector

Enter the email content below to check if it's spam or not.

Email Content

Congratulations! Your credit score entitles you to a no-interest Visa credit card. Click here to claim: [Link]

This is a Spam email!

FIgure 4 – Output(Email spam detection)

9. CONCLUSION AND FUTURE WORK

This research presents a comprehensive and effective dual-model framework for phishing and spam detection by combining URL-based heuristics with deep learning techniques for analyzing email content. The integration of these two approaches leverages both surface-level and contextual information, thereby improving the robustness and accuracy of the system. Extensive experimentation has demonstrated that the proposed architecture is capable of detecting phishing attempts with a high degree of precision, even when adversarial tactics are employed to obfuscate malicious content.The system's modular design ensures scalability and adaptability, making it suitable for deployment across various organizational infrastructures. Its performance across diverse datasets underscores its generalization ability, which is critical in the dynamic landscape of cybersecurity threats. Furthermore, by incorporating recent advancements in machine learning, such as attention mechanisms and contextual embeddings, the framework aligns with current trends in intelligent threat detection.

For future work, several key areas offer potential for further enhancement. One significant direction is the integration of **real-time threat detection**, enabling immediate identification and mitigation of phishing attempts as they occur. Another promising area is the inclusion of **sender reputation analysis**, which would allow the system to evaluate historical trustworthiness of communication sources. Additionally, **multilingual support** can be introduced to expand the system's usability across global user bases, particularly in regions where phishing content is crafted in local languages.The scope of this framework can also be extended beyond emails to other digital communication channels such as **SMS, instant messaging apps (e.g., WhatsApp, Telegram), and social media platforms**, which are increasingly being used for phishing campaigns. Incorporating **user behavior analytics**, such as response patterns to suspicious content, may further enhance the system's intelligence by enabling adaptive learning based on user feedback.

In conclusion, the proposed dual-model system not only addresses current phishing detection challenges but also provides a flexible foundation for future cybersecurity solutions. With the continuous evolution of phishing techniques, adaptive and intelligent systems such as this will be essential in maintaining digital trust and protecting users from increasingly sophisticated threats.

10. ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to Dr. Rizwana Shaikh, Guide, Department of Artificial Intelligence & Data Science, SIES Graduate School of Technology, for her invaluable guidance, insightful feedback, and continuous encouragement throughout the development of this work. We also wish to thank the Department of Artificial Intelligence & Data Science at SIES Graduate School of Technology for providing the computational resources and supportive research environment.

11. REFERENCES

[1] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *J. King Saud Univ. – Computer. Inf. Sci.*, vol. 35, pp. 590–611, 2023.

[2] H. A. Shaik et al., "Phishing URL detection using machine learning methods," *Adv. Eng. Softw.*, vol. 173, p. 103288, Jan. 2022.

[3] S. Sallouma et al., "Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey," *Procedia Computer. Sci.*, vol. 189, pp. 19–28, 2021.

[4] Aniket Garje et al., "Detecting Phishing Websites Using Machine Learning," *Int. J. Creative Res. Thoughts (IJCRT)*, Nov. 2021.

[5] A. Krishna V. et al., "Phishing Detection using Machine Learning based URL Analysis: A Survey," *IJERT*, 2021.

[6] R. Mahajan and I. Siddavatam, "Phishing Website Detection using Machine Learning Algorithms," *Int. J. Computer. Appl.*, Oct. 2018.

[7] H. Ghalechyan et al., "Phishing URL detection with neural networks: an empirical study," *Sci. Rep.*, 2024.

[8] S. Atawneh and H. Aljehani, "Phishing Email Detection Model Using Deep Learning," *Electronics*, vol. 12, p. 4261, 2023.

[9] B. Sucharitha et al., "Detecting Phishing Websites Using Machine Learning," *Int. J. Adv. Res. Computer. Commun. Eng.*, vol. 13, Issue 4, Apr. 2024.

[10] A. El Aassal et al., "An In-Depth Benchmarking of Phishing Detection Research," *IEEE Access*, 2020