RESEARCH ARTICLE OPEN ACCESS

Intelligent Phishing Website Detection System Using Multimodal AI Techniques

Mr.I.AnbuMuthu *,Mr.E. Dineshkumar**,
MS. A. Indhureka**,Mr.V. Iyandurai**, Ms.M. Jeevitha**
*Assistant Professor Computer Science and Engineering ,R P Sarathy Institute of Technology, Salem,
**UG Students Computer Science and Engineering, R P Sarathy Institute of Technology, Salem,
India

Abstract: Phishing attacks are becoming more complex. They exploit misleading URL patterns and harmful webpage behavior to evade traditional blacklist detection systems. These standard methods do not work well against newly created and zero-day phishing sites. To tackle this issue, this study suggests a phishing detection framework with three analytical parts: URL intelligence, HTML structural analysis, and behavioral analysis. The URL module identifies features like length, special characters, and suspicious keywords. The HTML module inspects structural elements such as forms, password fields, external links, and redirection scripts. The behavioral module assesses runtime characteristics like redirect chains and unusual response patterns. These features are combined and sorted using a supervised machine learning model to tell phishing sites apart from legitimate ones. Tests with a curated dataset show that combining URL, structural, and behavioral features enhances detection accuracy and lowers false positives. The proposed framework is efficient and suitable for real-time phishing detection applications.

1. Introduction

The rapid expansion of internet services has made web platforms essential for activities like banking, communication, and information sharing, but it has also increased cybersecurity threats. Among these, phishing remains a major concern, as attackers create fake websites that imitate legitimate platforms to steal sensitive user information through techniques such as domain spoofing, deceptive URLs, and hidden redirections.

Traditional detection methods, including blacklist and rule-based systems, rely on known malicious URLs and are therefore ineffective against zero-day phishing attacks. Additionally, approaches that focus on a single feature type often lack adaptability and may result in higher false positives.

To overcome these challenges, this research proposes a phishing detection framework that combines URL intelligence, HTML structural analysis, and behavioral analysis. By analyzing both static and dynamic webpage characteristics, the system aims to improve detection accuracy, reduce false alarms, and support efficient real-time deployment.

2. Existing System:

Traditional phishing detection systems mainly rely on filtering blacklists and analyzing URLs with rules. Blacklist approaches compare URLs to known malicious websites. Rule-based systems use manually defined guidelines, like URL length, special characters, or suspicious keywords. While these methods are efficient and simple to implement, they struggle against newly created or zero-day phishing websites not yet in databases. Some machine learning methods aim to improve detection by examining URL or HTML features separately. However, many of these systems focus on just one feature category. This leads to limited adaptability and higher rates of false positives or false negatives when attackers change their strategies.

3. PROPOSED SYSTEM:

The proposed system presents a well-organized phishing detection framework that combines three important analytical components: URL intelligence, HTML structural analysis, and behavioral analysis. Instead of relying on a single indicator, the system brings together multiple perspectives to improve detection accuracy and reliability.

The **URL intelligence module** focuses on identifying suspicious patterns within website addresses. It extracts both lexical and syntactic characteristics such as URL length, unusual subdomain usage, presence of special characters, and keywords commonly associated with phishing attempts. These indicators help in detecting malicious intent even before the webpage is fully loaded.

The **HTML structural analysis module** examines the internal structure of webpages to uncover elements that may facilitate credential theft. This includes analyzing form tags, password input fields, external resource links, hidden components, and embedded redirection scripts. Such structural irregularities often reveal attempts to mimic legitimate websites while secretly capturing user information.

The **behavioral analysis module** observes how a webpage behaves during runtime. It evaluates dynamic activities such as redirect chains, unexpected navigation flows, and abnormal server response patterns that may indicate deceptive operations.

Features extracted from all three modules are then combined into a unified feature set and processed using a supervised machine learning algorithm. This classification approach enables the system to effectively differentiate phishing websites from legitimate ones. By integrating multiple feature categories, the proposed framework improves detection robustness while remaining computationally applications.

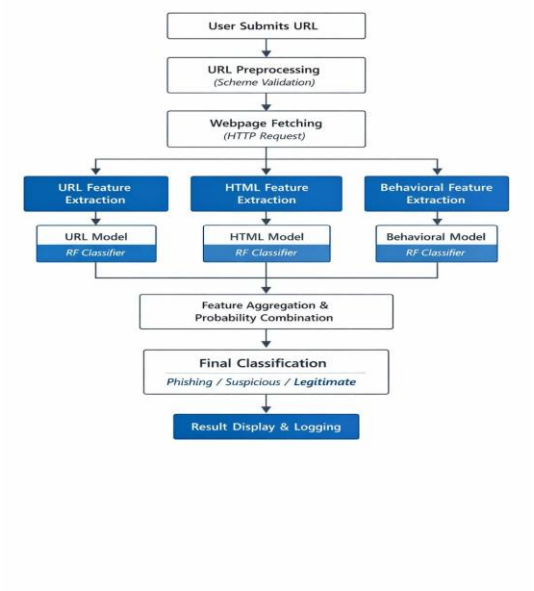

Figure3.1: Workflow diagram

3.1 URL Intelligence-Based Phishing Detection Model

3.1.1 Model Overview

The URL Intelligence Model detects phishing websites by analyzing lexical and structural characteristics of the URL string. Phishing attackers often manipulate URLs using excessive subdomains, suspicious keywords, special characters, or IP-based addressing to mimic legitimate domains. Since this model performs static analysis without fetching webpage content, it is computationally efficient and suitable for early-stage detection.

3.1.2 Feature Extraction

For each URL, the following features are extracted:

- URL length
- Number of dots and hyphens
- Presence of "@" symbol
- HTTPS usage
- Presence of phishing-related keywords (login, verify, secure, update, account)
- Subdomain count
- Presence of IP address

These features form a numerical feature vector used for classification.

3.1.3 Algorithm Used

The model uses the Random Forest classifier, an ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting. It performs well with mixed numerical and binary features and requires minimal parameter tuning.

The final prediction is obtained using majority voting:

$$y\hat{}=mode\{T1(x),T2(x),...,Tn(x)\}$$

Where T_i(x) represents the prediction made by the i-th decision tree.

3.1.4 Dataset and Implementation

The model was trained on a labeled dataset containing phishing and legitimate URLs using an 80:20 train-test split. Labels were encoded as binary values (1 = Phishing, 0 = Legitimate). Implementation was carried out in Python using Scikit-learn for classification, Pandas for data handling, and Regex for pattern detection.

3.1.5 MODEL OUTPUT:

Figure 3.1.1: URL MODEL OUTPUT

3.2 HTML Structural-Based Phishing Detection Model

3.2.1 Model Overview

The HTML Structural Model detects phishing websites by analyzing the structural components of a webpage. Phishing pages often contain abnormal HTML elements such as multiple login forms, hidden fields, external redirections, and suspicious scripts designed to capture user

credentials. By examining these structural irregularities, the model identifies malicious intent beyond simple URL patterns.

3.2.2 Feature Extraction

After fetching and parsing the webpage using BeautifulSoup, the following structural features are extracted:

- Number of <form> tags
- Count of password input fields
- Number of <iframe> elements
- Count of external links
- Presence of suspicious keywords (login, verify, update, etc.)
- Total HTML length

These features are converted into a structured numerical feature vector.

3.2.3 Algorithm Used:

A Random Forest classifier is used for classification due to its robustness and ability to handle nonlinear relationships between structural features.

The prediction is obtained using majority voting across multiple decision trees:

$$y\hat{}=mode\{T1(x),T2(x),...,Tn(x)\}$$

Where x represents the extracted HTML feature vector.

3.2.4 Dataset and Implementation

The model was trained on webpages corresponding to labeled phishing and legitimate URLs. HTML content was fetched dynamically, and rows with failed retrieval were removed. An 80:20 train-test split was used for evaluation.

The implementation was carried out in Python using BeautifulSoup for HTML parsing and Scikit-learn for Random Forest classification.

3.2.5 HTML MODEL OUTPUT

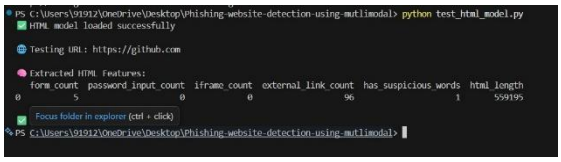

Figure 3.2.1 HTML MODEL OUTPUT

3.3 Behavioral-Based Phishing Detection Model

3.3.1 Model Overview

The Behavioral Model identifies phishing websites by analyzing runtime and interaction-based characteristics of webpages. Unlike purely structural analysis, this model focuses on how a webpage behaves during loading and user interaction. Phishing websites often exhibit behaviors such as forced redirection, hidden input fields, external form submissions, and the use of urgent or manipulative language to pressure users.

3.3.2 Feature Extraction

After parsing the webpage, the following behavioral indicators are extracted:

- Presence of login forms
- Number of password input fields
- Presence of submit buttons
- Count of hidden input fields
- Presence of urgent or phishing-related words
- Meta refresh (automatic redirection)
- External form action submission

These features are converted into a structured numerical feature vector for classification.

3.3.3 Algorithm Used

The Behavioral Model also uses a Random Forest classifier, selected for its ability to capture nonlinear feature interactions and reduce overfitting.

The final prediction is obtained using majority voting:

$$y\hat{}=mode\{T1(x),T2(x),...,Tn(x)\}$$

Where x represents the behavioral feature vector.

3.3.4 Dataset and Implementation

The model was trained using labeled phishing and legitimate webpages. Behavioral features were extracted from the HTML content and evaluated using an 80:20 train-test split.

The implementation was carried out in Python using BeautifulSoup for parsing and Scikit-learn for training the Random Forest classifier.

3.3.5 BEHAVIORAL MODEL OUTPUT

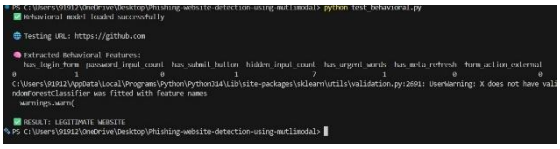

Figure 3.3.1 BEHAVIORAL MODEL OUTPUT

4.Model Fusion Strategy:

The final decision is obtained using a weighted late-fusion mechanism implemented in the application layer. The URL, HTML, and Behavioral models independently generate phishing probability scores using their respective Random Forest classifiers. These probabilities are combined using weighted averaging to compute a unified phishing confidence score. The final probability is calculated as:

$$P_{final} = \frac{w_u P_u + w_h P_h + w_b P_b}{w_u + w_h + w_b}$$

Where P_u, P_h, and P_b represent the phishing probabilities predicted by the URL, HTML, and Behavioral models, respectively, and w_u, w_h, and w_b are their assigned

weights. The final classification is determined using predefined thresholds (High, Medium, Low risk). This approach enhances robustness by balancing lexical, structural, and behavioral intelligence.

5.Conclusion:

This study presented a multi-layered phishing detection framework integrating URL intelligence, HTML structural analysis, and behavioral analysis to enhance detection reliability. Unlike traditional single-feature approaches, the proposed system combines complementary lexical, structural, and runtime signals using a weighted late-fusion strategy to generate a unified phishing probability score. Experimental evaluation demonstrated that individual modules perform effectively in isolation, while their integration significantly improves robustness and reduces false positives. The modular architecture ensures scalability and computational efficiency, making the system suitable for real-time deployment in web security applications. Future work may focus on adaptive weight optimization and large-scale real-world validation to further strengthen detection performance.

References

[1] A. K. Jain and B. B. Gupta, “Phishing detection: Analysis of visual similarity based approaches,” Security and Communication Networks, vol. 10, no. 8, pp. 1319–1333, 2017.

[2] M. Aburrous, M. A. Hossain, F. Thabtah, and K. Dahal, “Intelligent phishing detection system for e-banking using fuzzy data mining,” Expert Systems with Applications, vol. 37, no. 12, pp. 7913–7921, 2010.

[3] S. Marchal, J. François, R. State, and T. Engel, “PhishStorm: Detecting phishing with streaming analytics,” IEEE Transactions on Network and Service Management, vol. 11, no. 4, pp. 458–471, 2014.

[4] R. Verma and A. Das, “What’s in a URL: Fast feature extraction and malicious URL detection,” in Proceedings of the 3rd ACM on International Workshop on Security and Privacy Analytics (IWSPA), 2017, pp. 55–63.