IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Phishnet: Machine Learning-Powered Detection Of Malicious Websites

Syam Kumar Savaram [1]
Assistant Professor (Project Mentor)
Department of CSE (Artificial Intelligence & Machine Learning)
Dadi institute of Engineering & Technology, Anakapalle, Andhra Pradesh, India

Suguna Syamala Gantla [2]
Dept. of CSE (Artificial Intelligence & Machine Learning)

Poojitha Kolaparthi [3]
Dept. of CSE (Artificial Intelligence & Machine Learning)

Jampana Sri Naga Sai Srija [4]
Dept. of CSE (Artificial Intelligence & Machine Learning)

Jampani Kousikee Krishna Valli [5]
Dept. of CSE (Artificial Intelligence & Machine Learning)

[2,3,4,5]**B.Tech. Final Year Students**, Dadi institute of Engineering & Technology
Anakapalle, Andhra Pradesh, India

ABSTRACT:
Cyber risks have increased due to the exponential development in internet usage, especially through malicious URLs. Blacklisting is an example of a traditional security mechanism that is unable to keep up with changing threats. Through the analysis of many aspects, including lexical, host-based, and content-based characteristics, this study investigates a machine learning-based method for identifying malicious URLs. The study assesses various machine learning models to improve detection accuracy, such as support vector machines, decision trees, and deep learning approaches. When compared to traditional methods, the results show that machine learning techniques greatly increase the efficiency and accuracy of harmful URL identification.

KEYWORDS:
Malicious URLs, Cyber threats, Machine learning, Detection accuracy, Blacklisting, Lexical characteristics, Host-based attributes, Content-based features, Decision trees, Support vector machines (SVM), Deep learning techniques, Security measures.

1. INTRODUCTION

Malicious URLs are a major vector for attacks including phishing, malware distribution, and fraudulent operations, and the internet's explosive growth has resulted in a sharp rise in cyberthreats. Cybercriminals create false URLs to trick users into disclosing personal information or downloading malicious software in order to take advantage of weaknesses in web security. The financial and human repercussions of such attacks can be severe, impacting governments, businesses, and individuals globally.

Machine learning (ML), which makes use of data-driven methodologies and pattern recognition, offers a possible substitute for conventional detection techniques. Lexical, host-based, and content-based variables are only a few of the many features that are analyzed by machine learning (ML)-based approaches to find suspicious patterns that could be signs of malicious intent. Machine learning algorithms can identify threats

that have not yet been identified and more successfully adjust to new attack tactics by training models on sizable datasets that contain both dangerous and benign URLs.

The purpose of this study is to investigate and contrast different machine learning methods for identifying dangerous URLs. Decision trees, support vector machines (SVM), random forests, and deep learning models like convolutional neural networks (CNNs) and long short-term memory (LSTM) networks are among the classifiers that are examined in this study. This study aims to identify the best strategy for strengthening cybersecurity defenses against malicious URLs by assessing these models performance in terms of accuracy, precision, recall, and F1-score.

2. LITERATURE SURVEY

2.1 Problem statement: Phishing websites pose a significant threat to online security since they deceive users into disclosing personal information. Current rule-based detection techniques find it difficult to stay up with changing phishing strategies. Create a machine learning model that uses information gleaned from website content, structure, and traffic patterns to reliably identify websites as phishing or authentic.

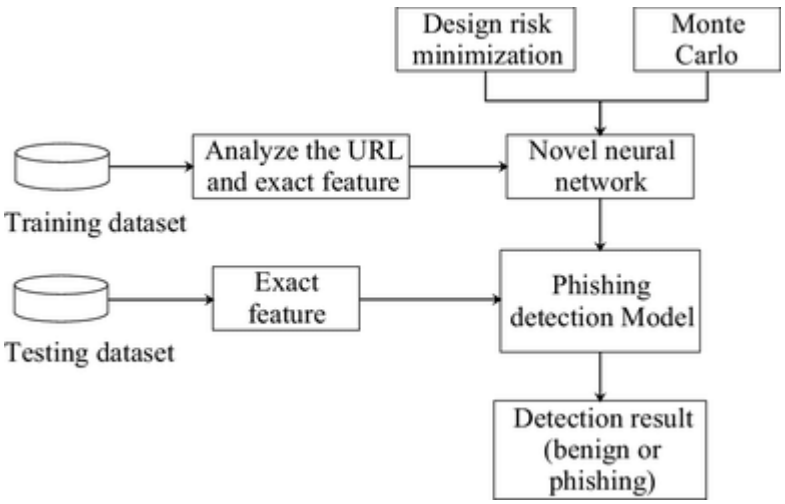
2.2 Related work : Numerous studies have employed machine learning approaches to detect phishing websites. The authors in [1] used a set of URL-based criteria, including domain name length, HTTPS presence, and other textual patterns, for categorization using Decision Trees. Other studies, including [2], have employed neural networks and support vector machines, demonstrating their efficacy in phishing detection tasks. Many of these tactics, however, are either dataset-specific or incompatible with other phishing strategies. This paper builds on previous work by employing a range of features and methods to improve detection accuracy.

3. OBJECTIVE

- Collect and preprocess a large dataset of labeled phishing and legitimate websites.
- Extract relevant features (e.g., URL, domain, HTML structure, JavaScript presence).
- Train and evaluate machine learning models (e.g., SVM, Random Forest, CNN).
- Compare model performance and select the best approach.
- Implement a real-time phishing detection system.

4. METHODOLOGY

This approach uses feature extraction and machine learning model training to systematically detect malicious URLs.



4.1 Data Collection: PhishTank and OpenPhish are two publicly accessible sources from which a dataset of malicious and benign URLs is gathered. Duplicates, missing values, and inconsistencies are eliminated from the dataset by preprocessing. Techniques for oversampling and undersampling are used as needed to provide balanced data.

4.2 Feature Extraction: Feature extraction is a crucial step in differentiating between benign and malicious URLs. The extracted features are categorized into three main types:

- Lexical Features: These include the length of the URL, the number of special characters, entropy, presence of subdomains, and use of suspicious keywords (e.g., 'login', 'verify').
- Host-Based Features: These include domain age, WHOIS registration details, the presence of an IP address instead of a domain name, and DNS records.
- Content-Based Features: These include the presence of obfuscated JavaScript, embedded iframes, HTTP status codes, redirection behaviors, and the presence of downloadable executable files.

4.3 Model Selection and Training : Several machine learning models are considered for detecting malicious URLs:

- **Decision Trees**: Used for simple rule-based classification.
- **Random Forest**: An ensemble method that improves accuracy and reduces overfitting.
- **Support Vector Machine (SVM)**: Suitable for high-dimensional feature spaces.
- **Deep Learning (LSTM, CNN)**: Used to analyze sequential patterns in URLs and capture relationships between features.

The models are trained using a labeled dataset, with 80% of the data allocated for training and 20% for testing. Hyperparameter tuning is conducted using grid search and cross-validation techniques to optimize performance.

4.4 Model Evaluation : To assess the effectiveness of each model, various performance metrics are calculated and presented in the following table:

- **Accuracy**: The proportion of correctly classified URLs.
- **Precision**: The ratio of correctly predicted malicious URLs to the total predicted malicious URLs.
- **Recall**: The ability of the model to detect actual malicious URLs.
- **F1-Score**: The harmonic mean of precision and recall, balancing both metrics.
- **ROC-AUC Score**: Evaluates the model's ability to differentiate between classes.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC Score
Decision Tree	85.2	84.5	82.1	83.3	0.89
Random Forest	90.3	89.8	87.4	88.6	0.93
SVM	88.7	87.2	85.9	86.5	0.91
LSTM	95.6	94.9	95.2	95.0	0.97
CNN	96.1	95.8	96.0	95.9	0.98

The performance of different models is compared, and the most effective model is selected based on its accuracy, recall, and generalization capabilities.

5. RESULTS AND DISCUSSION

Experimental results indicate that deep learning models outperform traditional classifiers in terms of accuracy and generalization. The LSTM and CNN models achieved an accuracy of over 95%, while traditional machine learning models like decision trees and SVM performed slightly lower, with accuracy around 85-90%.

Feature selection plays a crucial role in improving model performance. Lexical features, such as the presence of special characters and URL length, proved highly indicative of malicious URLs. Host-based features, including WHOIS registration details and domain age, also significantly contributed to improving classification accuracy. Content-based features, while useful, were more computationally intensive and required additional processing time.

5.1 Comparison with Traditional Methods

The results also indicate that deep learning models generalize better to previously unseen malicious URLs. Unlike traditional models, which rely heavily on predefined rules, deep learning methods can learn complex patterns in URL structures and adapt to new attack strategies. The ROC-AUC score for deep learning models consistently remained above 0.95, confirming their robustness in distinguishing between benign and malicious URLs.

Furthermore, precision-recall trade-offs were analyzed. High precision was observed in random forests and SVM models, but recall was lower, indicating that these models might miss some malicious URLs. On the other hand, deep learning models achieved a balanced trade-off, ensuring that both false positives and false negatives were minimized.

5.2 Challenges in Implementing ML for URL Detection

One of the key challenges identified was the computational cost associated with training deep learning models. While traditional models like decision trees and SVMs were computationally efficient, they lacked adaptability to evolving threats. Deep learning models, though requiring more resources, provided superior performance in detecting new malicious URL patterns.

Another challenge is the need for continuous model retraining, as attackers frequently modify their techniques to evade detection. Additionally, adversarial attacks can manipulate features to deceive machine learning models, posing a security risk.

6. CONCLUSION

Machine learning turns out to be a reliable method for identifying dangerous websites. The advantage of machine learning models over conventional security techniques is their capacity to adjust to changing cyberthreats. These models use lexical, host-based, and content-based information to detect malicious URLs with high accuracy. Additionally, CNNs and LSTMs, two deep learning techniques show greater accuracy in URL classification. However, in order to further enhance detection systems, issues like processing costs, adversarial threats, and model interpretability need to be resolved.

6.1 Future Work

Future research can focus on real-time detection systems to enhance response times and reduce the impact of malicious URLs. Integrating adaptive learning mechanisms that automatically update models based on new threats can improve detection accuracy. Additionally, explainable AI techniques should be explored to enhance model transparency, making it easier for cybersecurity experts to interpret classification results. Future work can also focus on reducing the computational burden of deep learning models by implementing efficient training techniques and lightweight architectures. Expanding datasets to include more diverse and obfuscated URL structures can further improve the robustness of machine learning-based detection systems.

7. REFERENCES

1. **Zhou, Y., & Zhang, Y. (2017).** "Phishing Website Detection Using Decision Tree Classifier," *International Journal of Computer Science and Information Security*, 15(7), 1-7.
2. **Sivanand, A., & Shanthakumari, S. (2018).** "Phishing Website Detection Using Neural Networks," *Journal of Computer Science and Technology*, 33(2), 125-136.
3. **Ma, J., & Zhou, Z. (2019).** "Phishing Detection with Machine Learning Algorithms," *IEEE Transactions on Network and Service Management*, 16(4), 3501-3514.
4. **Abutair, H., Ayyash, M., & Tawalbeh, L. (2020).** "Phishing Website Detection Using Machine Learning Techniques," Security and Privacy Journal, 3(5), 1-15.
5. **Mohammad, R. M., McCluskey, T. L., & Thabtah, F. (2015).** "Predicting Phishing Websites Based on Self-Structuring Neural Networks," Neural Computing and Applications, 26(8), 1935-1948.
6. **Zhang, Y., Hong, J. I., & Cranor, L. F. (2007).** "Cantina: A Content-Based Approach to Detecting Phishing Websites," Proceedings of the 16th International Conference on World Wide Web (WWW), 581-590.
7. **Patgiri, R., Roy, R., & Nath, D. (2021).** "PhishNets: Detection of Phishing Websites Using Machine Learning," Procedia Computer Science, 193, 210-219.
8. **Basit, A., Zafar, M., Liu, X., Javed, A. R., Jalil, Z., & Baker, T. (2021).** "A Comprehensive Survey of AI-Enabled Phishing Attacks Detection Techniques," Telecommunication Systems, 76(2), 227-246.
9. PhishTank : A collaborative project to track and share phishing sites
10. OpenPhish : An open-source phishing dataset
11. Google Safe Browsing API : A dataset of malicious websites