International Journal of Information Technology (IJIT) – Volume 10 Issue 6, Nov – Dec 2024

RESEARCH ARTICLE OPEN ACCESS

Adversarial Robustness of ML Models for Malicious URL Detection Using Lexical Features

Roopesh Kumar B N [1], Rekha B Venkatapur [2], Suman B S [3], Gagan Shivanna [4]
[1] Department of Computer Science & Engineering, K S Institute of Technology, Karnataka, India
[2] Department of Computer Science & Engineering, K S Institute of Technology, Karnataka, India
[3] Department of Computer Science & Engineering, K S Institute of Technology, Karnataka, India
[4] Department of Computer Science & Engineering, K S Institute of Technology, Karnataka, India

ABSTRACT
Malicious URL detection is vital in cybersecurity, proactively identifying threats before they reach users. Traditional methods often struggle to keep pace with evolving threats, whereas machine learning (ML) offers a more adaptable approach through its ability to learn and adjust to changing scenarios. Ensemble algorithms such as Random Forests, XGBoost, AdaBoost, and LightGBM have proven highly effective for detecting malicious URLs. Among these, Random Forests (RF), a bagging model, achieved 89.78% accuracy, showcasing strong performance and robustness.

Keywords — Malicious URL detection, Machine learning (ML), ensemble algorithms, Random Forests (RF).

I. INTRODUCTION

Traditional URL detection methods typically rely on signature-based or rule-based systems, which are limited in their ability to keep pace with the rapidly evolving nature of online threats. Such methods often require manual updates and struggle to adapt to new attack patterns, resulting in delayed or ineffective responses. In contrast, machine learning (ML) has emerged as a powerful alternative due to its ability to learn from data, recognize patterns, and generalize well to detect previously unseen malicious URLs.

Machine learning techniques use a variety of features extracted from URLs, such as lexical, host-based, and content-based features, to distinguish between malicious and benign URLs. In this study, we focus on the lexical features of URLs from the dataset, as they are more vulnerable and frequently exploited than other features. These ML models are trained on large datasets to identify complex patterns that would be challenging for traditional methods to capture. Ensemble algorithms, in particular, have proven highly effective for this task, as they combine the strengths of multiple models, improving overall accuracy and robustness.

The primary objective of this research is to develop a robust and scalable malicious URL detection framework leveraging machine learning techniques to address the limitations of traditional methods. To achieve this, the research focuses on three key objectives. First, it emphasizes lexical feature-based detection by utilizing structural characteristics extracted from URLs to distinguish between malicious and benign URLs without relying on network behaviour or content inspection.

II. LITERATURE SURVEY

The detection of malicious URLs has been an area of intense research, particularly with the advent of machine learning techniques. Previous studies have extensively explored various feature extraction methods and classification algorithms to enhance detection accuracy. Lexical-based features, which analyse the structural components of URLs, have gained significant attention due to their independence from network behaviour and content-based inspection, as highlighted by Ma et al. [1]. Their work demonstrated the effectiveness of combining lexical and host-based features for classifying URLs using machine learning techniques like Support Vector Machines (SVMs).

Subsequent advancements in the field have seen the emergence of ensemble learning techniques for malicious URL detection. For instance, Sahoo et al. [2] demonstrated the superiority of ensemble methods like Random Forest and Gradient Boosting Machines (GBMs) over traditional classifiers due to their robustness and ability to capture non-linear patterns. Similarly, Al-Daeef et al. [3] showcased the efficacy of boosting algorithms, such as AdaBoost and XGBoost, in achieving high detection accuracy, particularly when dealing with imbalanced datasets.

In recent years, researchers have also begun addressing the adversarial robustness of these machine learning models. Goodfellow et al. [4] introduced the Fast Gradient Sign Method (FGSM), which revealed vulnerabilities in neural networks and other machine learning models against adversarial attacks. Building on this, Papernot et al. [5] proposed adversarial training as a defense mechanism, which involves incorporating adversarial examples into the training process to enhance model robustness. These techniques have been explored in the domain of malware and intrusion detection but are still nascent in the context of malicious URL detection.

Furthermore, studies like those by Rao et al. [6] have emphasized the importance of combining multiple feature extraction techniques—lexical, host-based, and behavioural—to improve detection accuracy and robustness. However, many of these approaches remain vulnerable to adversarial perturbations, as noted by Huang et al. [7], who emphasized the need for models that balance accuracy and robustness.

III. MALICIOUS URL DETECTION USING MACHINE LEARNING

A. METHODOLOGY

The proposed methodology for malicious URL detection using machine learning is designed to follow a systematic approach to ensure accurate classification of malicious URLs. The workflow consists of several key stages, each contributing to the overall performance of the model. Below is a step-by-step explanation of the methodology:

1) Labelled Data Collection:

For this study, a labelled dataset comprising 450,176 URLs was utilized, sourced from Kaggle. The dataset is divided into two primary categories: Benign URLs (Class 0), representing legitimate and safe websites, and Malicious URLs (Class 1), consisting of harmful websites designed for phishing, malware distribution, or other malicious activities. The dataset was curated to ensure a balanced representation of real-world scenarios, providing a robust foundation for developing an accurate and effective malicious URL detection model.

2) Feature Extraction:

Once the labelled dataset is prepared, lexical feature extraction is performed to obtain meaningful attributes from the URLs. It focuses on the structure and composition of URLs, such as URL length, number of dots, special characters, presence of suspicious words, and entropy. These features are the primary focus in this study, as they provide insights into the lexical characteristics of the URLs.

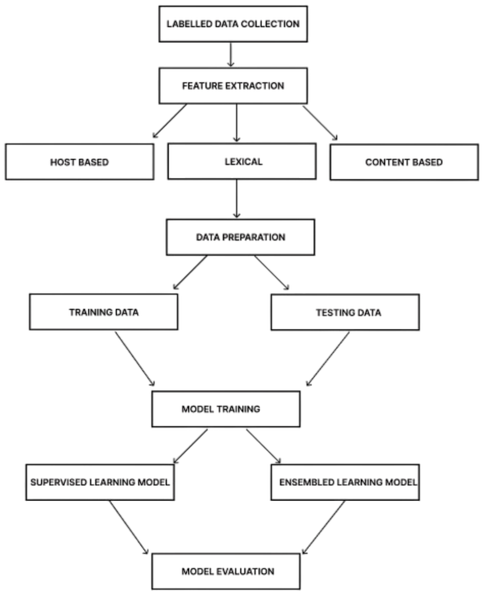

Figure 1: Methodology

3) Data Preparation:

The dataset is divided into two subsets:

Training Data is used to train the machine learning models, enabling them to learn from the labelled examples.

Test Data is set aside for evaluating the performance of the trained models, ensuring they can effectively generalize to unseen data.

4) Model Training:

The training phase involves constructing models using the training data, employing various machine learning algorithms to ensure a comprehensive evaluation. The workflow encompasses two categories of models: Supervised Learning Models and Ensemble Learning Models.

The supervised learning models include Logistic Regression, Support Vector Machine (SVM), and K-Nearest Neighbour (KNN) algorithms, which are foundational techniques for classification tasks. On the other hand, the ensemble learning models, such as Random Forests, AdaBoost, XGBoost, and LightGBM, leverage the power of combining multiple learning models to enhance accuracy and robustness. During the training process, hyperparameter tuning is conducted to optimize the models, ensuring improved accuracy and better generalization to unseen data.

B. URL feature extraction and selection

1) LIST OF LEXICAL FEATURES:

Feature	Description
URL Length	Total number of characters

	in the URL.
Number of Dots	Count of periods (.) in the URL.
Number of Hyphens	Count of hyphens (-) in the URL
Presence of IP Address	Indicates whether the URL contains an IP address instead of a domain name.
Number of Subdomains	Count of subdomains in the URL.
Suspicious Words	Count of specific words that indicate potential malicious intent (e.g., "login", "secure", "update").
Number of Parameters	Count of query parameters in the URL.
Presence of HTTPS	Boolean value indicating whether the URL starts with HTTPS.
Query String Length	Length of the query string in the URL.

Table 1: List of lexical features

The focus on lexical features for malicious URL detection is driven by their practicality, scalability, and independence from external data. Unlike content-based or host-based features, lexical features are extracted directly from URL strings, enabling real-time detection while avoiding resource-intensive processes and exposure to malware risks.

Key features such as URL length, number of dots, hyphens, subdomains, suspicious words (e.g., "login," "secure"), and HTTPS presence were chosen for their ability to capture structural and semantic patterns distinguishing malicious from benign URLs. This streamlined approach ensures high detection accuracy, efficiency, and scalability without the complexities of integrating external data sources.

C. MACHINE LEARNING ALGORITHM SELECTION

Selecting the right machine learning algorithms is crucial for building effective malicious URL detection models. This study evaluates both supervised models like Logistic Regression (LR), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) as baselines, and ensemble methods like Random Forest, AdaBoost, XGBoost, and LightGBM for improved performance. Ensemble techniques, by combining multiple models, demonstrate better resilience to noise, capture complex patterns, and handle imbalanced datasets more effectively.

D. DATASET

The dataset comprises of 450,176 URLs, each labelled as either benign (class 0) or malicious (class 1). It provides a comprehensive foundation for analysing patterns and features that differentiate safe URLs from potentially harmful ones. The dataset consists of two columns: "URL," which contains the website address, and "Class," which indicates whether the URL is benign or malicious. Benign URLs account for 76.8% of the data (345,738 entries), while malicious URLs make up 23.2% (104,438 entries).

E. RESULTS

Sl No.	**Model**	**Accuracy (in %)**
1	Logistic Regression	65.29
2	SVM	65.19
3	K-NN	81.88
4	Random Forest	89.89
5	AdaBoost	74.88
6	XGBoost	82.09
7	LightGBM	82.11

Table 2: Result of Machine Learning algorithms

F. EVALUATION METRICS

The evaluation of the models is conducted using metrics derived from the confusion matrix, which provides insights into their performance in malicious URL detection.

Based on the results shown in the accuracy comparison table, there is a clear distinction in the performance of supervised and ensemble machine learning algorithms for the task of malicious URL detection.

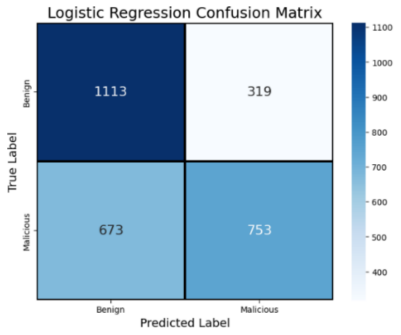

Fig 2: Logistic Regression

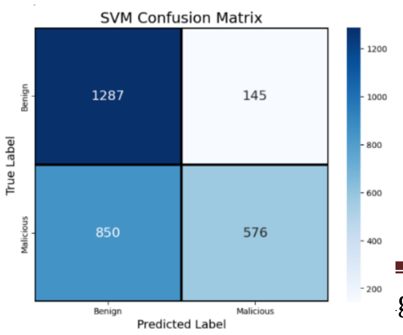

Fig 3: SVM

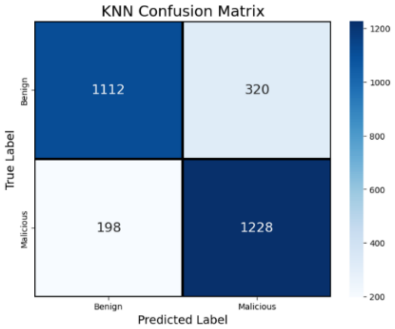

Fig 4: KNN

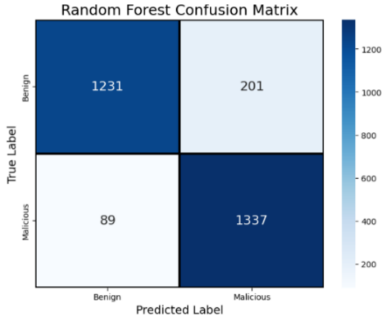

Fig 5: Random Forest

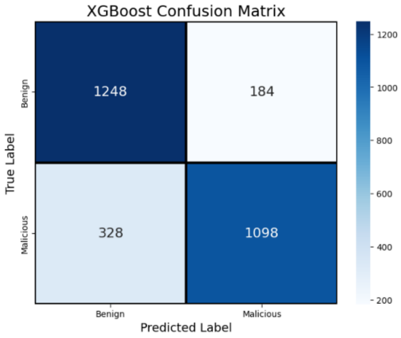

Fig 6: XGBoost

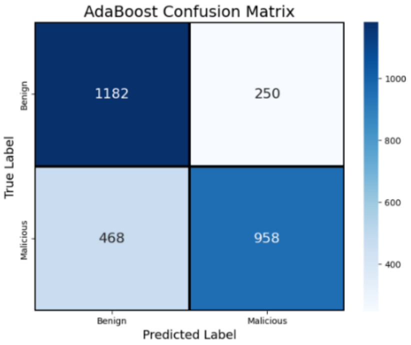

Fig 7: AdaBoost

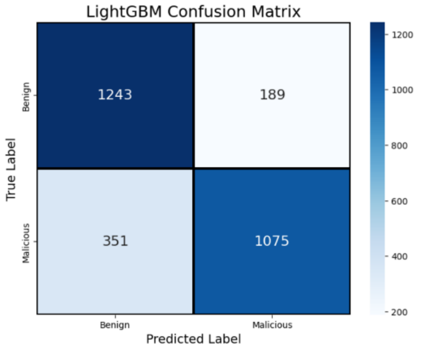

Fig 8: LightGBM

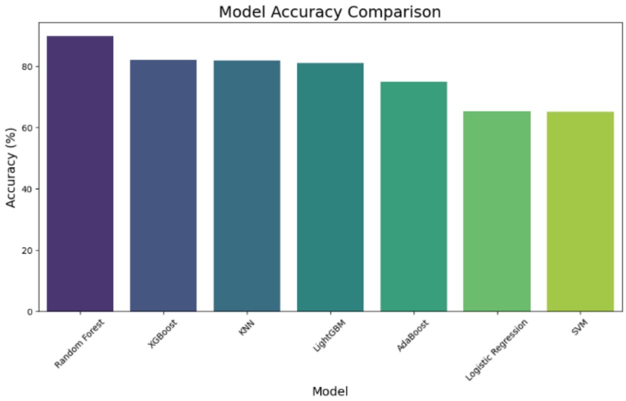

Fig 9: Accuracy Comparison Graph

Supervised learning models such as Logistic Regression (65.29%) and SVM (65.18%) demonstrate lower accuracies, likely due to their reliance on individual decision boundaries and sensitivity to class imbalance and feature quality. Ensemble methods like Random Forest (89.85%), XGBoost (82.08%), and LightGBM (81.10%) outperform these models by leveraging multiple learners to generalize better and handle complex patterns more effectively. K-Nearest Neighbors (81.87%), while not an ensemble method, also shows strong performance, indicating the potential of locality-based decision-making when the feature space is appropriately scaled.

Overall, ensemble algorithms outperform traditional supervised models by leveraging multiple hypotheses and decision-making strategies, making them better suited for tasks involving complex and imbalanced data distributions. This emphasizes the importance of ensemble techniques in achieving robust and reliable performance in malicious URL detection.

G. DISCUSSION

The results of our model accuracy comparison provide valuable insights into the effectiveness of various machine learning algorithms for malicious URL detection using lexical features. Among the evaluated models, Random Forest demonstrated the highest accuracy, exceeding 80%, showcasing its strong ability to identify patterns and relationships in the dataset. This highlights the suitability of Random Forest for this task compared to other models.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to K.S. Institute of Technology for their continuous support and guidance throughout this research. We extend our thanks to the contributors of the Kaggle dataset for providing valuable data, which formed the backbone of this study. We also appreciate the resources and tools offered by open-source communities that facilitated the implementation of our machine learning model.

REFERENCES

[1] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying malicious URLs: An application of large-scale online learning,” *Journal of Machine Learning Research*, vol. 10, pp. 1931–1957, 2009.

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

[3] M. M. Al-Daeef, O. Basir, and S. Ibrahim, “An enhanced phishing detection framework based on boosted random forest,” *Computers & Security*, vol. 85, pp. 195–212, 2019, doi: 10.1016/j.cose.2019.04.006.

[4] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint*, arXiv:1412.6572, 2014.

[5] N. Papernot, P. McDaniel, I. Goodfellow, et al., “Practical black-box attacks against machine learning,” *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519, 2016.

[6] J. Garera, N. Provos, M. Chew, and A. D. Rubin, “A framework for detection and measurement of phishing attacks,” *Proceedings of the 2007 ACM Workshop on Recurring Malcode*, pp. 1–8, 2007, doi: 10.1145/1314389.1314390.

[7] Q. Xu, R. Zheng, and Z. Xu, “Phishing detection using multi-model ensemble learning,” *IEEE Access*, vol. 8, pp. 43591–43601, 2020, doi: 10.1109/ACCESS.2020.2977364.

[8] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, “EXPOSURE: Finding malicious domains using passive DNS analysis,” *Proceedings of the 18th Annual Network and Distributed System Security Symposium (NDSS)*, 2011.

[9] A. C. Bahnsen, E. C. Bohorquez, S. Villegas, et al., “Classifying phishing URLs using recurrent neural networks,” *Proceedings of the APWG Symposium on Electronic Crime Research (eCrime)*, pp. 1–8, 2017, doi: 10.1109/ECRIME.2017.7945048.

[10] X. Yuan, P. He, Q. Zhu, and X. Li, “Adversarial examples: Attacks and defenses for deep learning,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2805–2824, 2019, doi: 10.1109/TNNLS.2018.2876010.

[11] H. Waheed, N. Zafar, W. Akram, A. Manzoor, A. Gani, and S. ul Islam, “Deep Learning-Based Disease, Pest Pattern, and Nutritional Deficiency Detection System for 'Zingiberaceae' Crop,” *MDPI*, 2023.