International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Machine Learning Approaches For Phishing Detection: A Comparative Analysis

Purva Kulkarni, Siddhi Jadhav, Tanya Gupta, Dr. Sangeeta Mishra

Department of Electronics and Telecommunication Engineering,
Thakur College of Engineering and Technology,
Mumbai, Maharshtra, India

Abstract - In this study, the effectiveness of four machine learning models in detecting phishing websites is evaluated. Utilizing a diverse dataset, the analysis reveals that Random Forest emerges as the top performer, achieving a test accuracy of 91.49%. Notably, Random Forest exhibits robustness in distinguishing between legitimate websites and malicious ones. While Decision Tree, K-Nearest Neighbors, and Naive Bayes also demonstrate promise, they encounter difficulties in accurately classifying phishing URLs, especially within certain categories. The findings underscore the pivotal role of machine learning in cybersecurity defence against phishing attacks. The study suggests avenues for future research, such as enhanced feature engineering and exploration of advanced ensemble techniques and deep learning approaches for improved phishing detection. This research contributes to the ongoing endeavours to develop more resilient anti-phishing tools and bolster digital security against evolving cyber threats.

Key Words: Phishing, Machine Learning, Cybersecurity, Random Forest, Decision Tree, K-Nearest Neighbors, Naive Bayes, Website Detection, Feature Engineering, Ensemble Techniques, Model Comparison, Fine-Tuning.

I. INTRODUCTION

Phishing remains a pervasive cyber threat, employing deceptive tactics to extract sensitive information from individuals, posing serious risks to personal and financial security [1]. With the increasing sophistication of phishing attacks and the potential consequences of falling victim, the demand for robust detection mechanisms has become paramount. Despite efforts to combat phishing, traditional detection methods have shown limitations in keeping pace with the evolving tactics employed by malicious actors [2] [3].

The landscape of cybersecurity is continually challenged by the dynamic nature of phishing attacks, which adapt and evolve to exploit vulnerabilities in human cognition and technological systems [4]. Existing detection methods, primarily reliant on rule-based heuristics and signature-based approaches, struggle to effectively identify new and previously unseen phishing attempts [5] [6]. Moreover, the sheer volume and diversity of phishing URLs make manual detection efforts impractical and ineffective [7].

In response to these challenges, the research community has turned to machine learning as a promising approach to enhance phishing detection capabilities [8] [9]. Machine learning algorithms, particularly those based on supervised learning techniques, offer the potential to automate the detection process and adapt to emerging threats in real time [10]. By leveraging vast datasets of labelled phishing and legitimate URLs, machine learning models can learn to distinguish between benign and malicious web addresses based on learned patterns and features [11] [12].

However, despite the growing interest in machine learning-based approaches, significant challenges remain in effectively identifying phishing URLs amidst the vast sea of legitimate web traffic [13]. The inherent variability and obfuscation techniques employed by attackers demand sophisticated algorithms capable of discerning subtle cues indicative of malicious intent [14]. Moreover, the rapid evolution of phishing tactics necessitates continuous refinement and adaptation of detection algorithms to remain effective [15].

In light of these considerations, this research endeavours to contribute to the ongoing efforts to combat phishing through an in-depth exploration of machine learning methodologies. By conducting a comparative analysis of prominent machine learning algorithms, including Decision Tree, Random Forest, K-Nearest Neighbors (KNN), and Naive Bayes, it is aimed to identify the most effective approach for phishing website detection. Through rigorous testing and evaluation, it is sought to shed light on the performance variations among these algorithms and provide insights into the challenges and opportunities in phishing URL classification. By unravelling the potential of machine learning, this research aims to fortify our digital defences against the ever-present danger of phishing attacks.

By filling the gap in the literature and providing insights into the performance variations among these algorithms, this research seeks to guide the development of more resilient and adaptive anti-phishing tools. In summary, this paper addresses the imperative need to combat the rising sophistication of phishing attacks through an in-depth exploration of machine learning methodologies, contributing valuable insights to the field of cybersecurity.

II. THEORY

In the pursuit of an effective phishing detection system, this research employs four distinct machine learning algorithms: Decision Tree, Random Forest, K-Nearest Neighbors (KNN), and Naive Bayes. Each algorithm brings unique characteristics to the task of classifying phishing websites.

A. Decision Tree –

Decision Tree is a straightforward algorithm that makes decisions by splitting data into subsets based on features. It builds a tree-like structure, where each node represents a decision based on a specific feature [3]. This method is known for its simplicity and interpretability in visualizing decision-making processes.

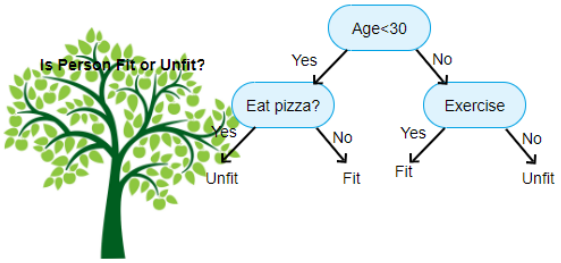

Fig - 1: Illustration of Working of Decision Tree

B. Random Forest –

Random Forest is an ensemble learning method that combines multiple decision trees to improve accuracy and robustness. Each tree in the forest independently classifies the data, and the final decision is determined by a majority vote [1]. Random Forest excels in handling noisy data and minimizing overfitting.

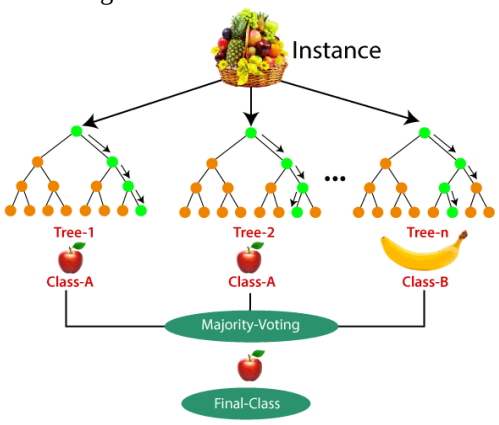

Fig - 2: Illustration of Working of Random Forest

C. K-Nearest Neighbors (KNN) –

K-Nearest Neighbors is a simple yet effective algorithm that classifies data points based on the majority class of their k-nearest neighbours. It relies on the assumption that similar data points belong to the same class [6]. KNN is particularly useful when dealing with non-linear decision boundaries.

Fig - 3: Illustration of Working of K-Nearest Neighbors

D. Naïve Bayes –

Naive Bayes is a probabilistic algorithm based on Bayes' theorem, assuming independence between features. It calculates the probability of a data point belonging to a particular class and selects the class with the highest probability [11]. Naive Bayes is computationally efficient and performs well with high-dimensional data.

By comprehending the principles behind these algorithms, we aim to evaluate their efficacy in the context of phishing website detection. The Decision Tree's simplicity, Random Forest's ensemble power, KNN's proximity-based classification, and Naive Bayes' probabilistic approach collectively contribute to the diversity of methodologies examined in this research [1] [3] [6] [11].

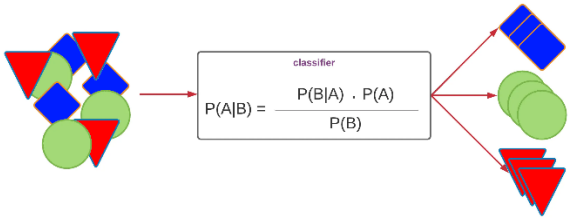

Fig - 4: Illustration of Working of Naive Bayes

III. METHODOLOGY

A. Dataset –

For this research, the dataset used is sourced from Kaggle, specifically the "Malicious URLs Dataset" available at https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset. This dataset encompasses a diverse collection of URLs, enabling a comprehensive evaluation of phishing website detection.

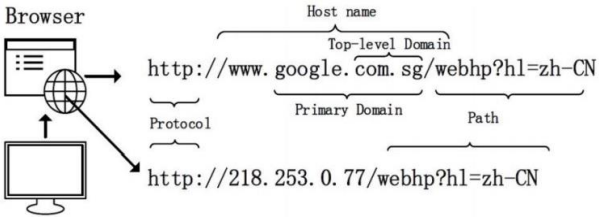

Fig - 5: Composition of a URL

B. Preprocessing –

Before model training, the dataset underwent several preprocessing steps to prepare it for machine learning algorithms:

1. Data Cleaning: Any missing values or inconsistencies in the dataset were addressed through imputation or removal, ensuring data integrity.
2. Feature Engineering: Relevant features such as URL length, domain age, and presence of special characters were extracted from the URLs to capture meaningful information for model training.
3. Normalization/Scaling: Numerical features were scaled to a standardized range to prevent biases during model training.

C. Programming Environment –

Google Colab, a cloud-based platform, was chosen for the implementation of machine learning models. Leveraging the collaborative nature and access to computing resources on Google Colab facilitated seamless coding and model evaluation.

Fig - 6: Google Colab Environment

D. Model Implementation –

The following machine-learning algorithms were implemented and evaluated for phishing website detection:

1. Decision Tree – Parameter Settings:
   The maximum depth of the decision tree and the minimum number of samples required to split a node were tuned using cross-validation to prevent overfitting.
2. Random Forest – Parameter Settings:
   The number of trees in the forest, maximum depth of individual trees, and minimum number of samples required to split a node were optimized using grid search cross-validation to maximize performance
3. K-Nearest Neighbors (KNN) – Parameter Settings:
   Parameter Settings: The number of neighbours (k) was tuned using cross-validation to achieve optimal classification performance.
4. Naïve Bayes – Parameter Settings:
   Since Naive Bayes is relatively simple and has fewer hyperparameters to tune, no extensive parameter optimization was performed. However, Laplace smoothing was applied to handle zero probabilities.

E. Evaluation Metrics –

The performance of each model was evaluated using standard classification metrics such as accuracy, precision, recall, and F1-score. Additionally, confusion matrices were generated to visualize the performance of the models across different classes of URLs.

F. Cross-Validation –

To ensure the robustness of the models and mitigate issues of overfitting, k-fold cross-validation was employed during the model training process. The dataset was randomly partitioned into k subsets, with each subset used as a testing set while the remaining subsets were used for training.

G. Flowchart –

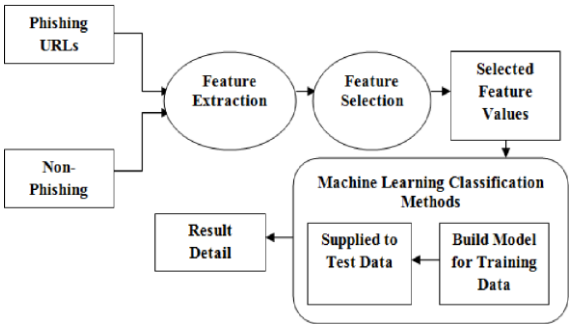

Fig - 7: Flowchart of the Methodology Used

IV. RESULT & DISCUSSION

The comparative analysis of the Decision Tree, Random Forest, K-Nearest Neighbors (KNN), and Naive Bayes models for phishing website detection yielded insightful findings, as presented in the following tables and discussion.

A. Performance Analysis –

1. Decision Tree: The Decision Tree model demonstrated commendable test accuracy of 90.96%, particularly excelling in classifying benign URLs. However, it faced challenges in accurately classifying phishing URLs, especially those with complex characteristics. The simplicity of decision trees makes them susceptible to overfitting, especially when dealing with noisy or imbalanced data, which may have impacted their performance in detecting phishing URLs.
2. Random Forest: Random Forest, as an ensemble of Decision Trees, outperformed its counterpart, showcasing improved test accuracy and robustness of 91.49%. The ensemble nature of Random Forest helps mitigate overfitting and captures complex decision boundaries, contributing to its superior performance in distinguishing between phishing and benign URLs.
3. K-Nearest Neighbors (KNN): The KNN model exhibited strong test accuracy of 88.96%, particularly in identifying benign URLs. However, challenges were observed in accurately classifying phishing URLs, especially those with subtle differences from legitimate websites. KNN's reliance on local similarities in feature space makes it sensitive to outliers and noise, potentially affecting its performance in detecting phishing URLs with atypical characteristics.
4. Naïve Bayes: The Naive Bayes model presented a lower overall test accuracy of 78.95%, with significant challenges in classifying certain categories of phishing URLs. Naive Bayes assumes feature independence, which may not hold for all phishing URLs, leading to suboptimal performance, especially in scenarios where features are correlated.

Table - 1: Comparison Between Algorithms

Model	Test Accuracy	Precision (Class 2)	Recall (Class 2)	F1-Score (Class 2)
Decision Tree	90.96 %	0.80	0.57	0.66
Random Forest	91.49 %	0.83	0.58	0.68
K-Nearest Neighbor	88.96 %	0.73	0.53	0.62
Naïve Bayes	78.95 %	0.60	0.02	0.04

B. Challenges in Phishing URL Detection –

The complexity of distinguishing phishing URLs from benign ones stems from the evolving tactics employed by malicious actors to mimic legitimate websites and evade detection. Phishing URLs often exhibit subtle variations in domain names, URL structure, and content, making them challenging to identify solely based on static features.

Furthermore, the dynamic nature of phishing attacks, including the rapid creation of new phishing URLs and the use of obfuscation techniques, poses additional challenges for traditional machine learning algorithms. Models may struggle to generalize effectively to unseen phishing URLs or adapt to changing attack patterns.

To address these challenges, future research efforts should focus on:

1. Enhanced feature engineering to capture nuanced characteristics of phishing URLs.
2. Exploration of advanced ensemble techniques and hybrid models to improve model robustness.
3. Investigation into dynamic and real-time detection approaches to adapt to evolving phishing threats.
4. Integration of deep learning approaches, such as neural networks, to extract and learn complex patterns from phishing URLs.

By systematically addressing these aspects, future research can significantly contribute to developing more robust and efficient phishing website detection systems, advancing cybersecurity practices.

V. CONCLUSION & FUTURE SCOPE

In conclusion, the comparative analysis of Decision Tree, Random Forest, K-Nearest Neighbors (KNN), and Naive Bayes models for phishing website detection underscores the significance of machine learning in cybersecurity. Each model demonstrated strengths and challenges, emphasizing the need for a balanced approach in choosing the right algorithm based on specific use cases.

While Random Forest emerged as the top performer with a test accuracy of 91.49%, all models faced difficulties in accurately classifying phishing URLs, particularly in Class 2. The study highlights the inherent complexity in distinguishing certain phishing characteristics and underscores the importance of ongoing research to enhance model capabilities

In the field of phishing website detection, future research holds great promise for improvement. Priorities include enhancing feature engineering to deepen the models' understanding of phishing characteristics. Exploring advanced ensemble techniques or hybrid models is crucial to overcome individual model limitations. Fine-tuning model parameters is critical for optimal performance, especially in addressing challenges specific to phishing URL classification. Prioritizing real-time detection capabilities aligns models with proactive cybersecurity needs. Exploring deep learning approaches, particularly neural networks, provides an exciting opportunity to gain additional insights and overcome challenges in traditional machine learning methods.

VI. REFERENCES

[1] Subasi, A., Molah, E., Almkallawi, F., & Chaudhery, T. J. (2017). Intelligent phishing website detection using random forest classifier. 2017 International Conference on Electrical and Computing Technologies and Application (ICECTA).

[2] Alnajim, A., & Munro, M. (2009). An approach to the implementation of the anti-phishing tool for phishing websites detection. 2009 International Conference on Intelligent Networking and Collaborative Systems.

[3] Peng, T., Harris, I., & Sawa, Y. (2018). Detecting Phishing Attacks Using Natural Language Processing and Machine Learning. Proceedings - 12th IEEE International Conference on Semantic Computing (ICSC).

[4] Aburrous, M., Hossain, M. A., Thabatah, F., & Dahal, K. (2008). Intelligent phishing website detection system using fuzzy techniques. 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications.

[5] Purwanto, R. W., Pal, A., Blair, A., & Jha, S. (2022). PhishSim: Aiding Phishing Website Detection With a Feature-Free Tool. IEEE Transactions on Information Forensics and Security.

[6] Parekh, S., Parikh, D., Kotak, S., & Sankhe, S. (2018). A new method for detection of phishing websites: URL detection. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT).

[7] Sharma, H., Meenakshi, E., & Bhatia, S. K. (2017). A comparative analysis and awareness survey of phishing detection tools. 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT).

[8] Abdulrahaman, M. D., Alhassan, J. K., Adebayo, O. S., Ojeniyi, J. A., & Olalere, M. (2019). Phishing attack detection based on random forest with wrapper feature selection method. International Journal of Information Processing and Communication, 7(2), 209-224.

[9] Wei, B., et al. (2019). A deep-learning-driven light-weight phishing detection sensor. Sensors, 19(19), 4258.

[10] Singh, C., & Meenu, S. (2020). Phishing website detection based on machine learning: a survey. In 6th International Conference on Advanced Computing & Communication Systems (ICACCS).

[11] Jain, A. K., & Gupta, B. B. (2018). PHISH-SAFE: URL Features-Based Phishing Detection System Using Machine Learning. Cyber Security. Advances in Intelligent Systems and Computing, 729.

[12] Gandotra, E., & Gupta, D. (2021). An Efficient Approach for Phishing Detection using Machine Learning. Algorithms for Intelligent Systems. doi:10.1007/978-981-15-8711-5_12.

[13] Kumar, J., Santhanavijayan, A., Janet, B., Rajendran, B., & Bindhumadhava, B. S. (2020). Phishing Website Classification and Detection Using Machine Learning. 2020 International Conference on Computer Communication and Informatics (ICCCI).

[14] Rao, R. S., & Pais, A. R. (2019). Jail-Phish: An improved search engine based phishing detection system. Computers & Security, 83, 246-267.

[15] Zamir, A., Khan, H. U., Iqbal, T., Yousaf, N., Aslam, F., et al. (2019). Phishing web site detection using diverse machine learning algorithms. The Electronic Library, 38(1), 65-80.

[16] Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357.

[17] Odeh, A., Keshta, I., & Abdelfattah, E. (2021). Machine Learning Techniques for Detection of Website Phishing: A Review for Promises and Challenges. 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC).

[18] Odeh, A., Alarbi, A., Keshta, I., & AbdelFattah, E. (2020). Efficient prediction of phishing website. Journal of Theoretical and Applied Information Technology, 98(16).

[19] Vilas, M. M., Ghansham, K. P., Jaypralash, S. P., & Shetty, P. (2019). Detection of Phishing Website Using Machine Learning Approach. 2019 4th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT).

[20] Patil, V., Thakkar, P., Shah, C., Bhat, T., & Godse, S. P. (1892). Detection and Prevention of Phishing Websites

using Machine Learning Approach. Oxford: Clarendon, 2, 68-73.

[21] Vazhayil, A., Kumar, R. V., & Soman, K. P. (2018). Comparative study of the detection of malicious URLs using shallow and deep networks. In 9th ICCCNT2018.

[22] Thakar, M., Parikh, M., & Shetty, P. (2018). Detecting Phishing Websites Using Data Mining. Proceeding of the Second International Conference On Electronics Communication and Aerospace Technology (ICECA).