Two-Phase Phishing Websites Detection Model by Cascading Light-weight and Advanced Classifiers

May Myat Moe Pwint Khine
Computere Science Department
Assumption University
Bangkok, Thailand
g6719600@au.edu

Kaung Htet Kyaw
Computere Science Department
Assumption University
Bangkok, Thailand
g6729566@au.edu

Khine Khine Myat Noe
Computere Science Department
Assumption University
Bangkok, Thailand
g6729567@au.edu

***Abstract*—Phishing websites pose a serious cybersecurity threat for Internet users, leading to extensive research on machine learning based phishing detection model. These existing research often face a trade-off between accuracy and computational efficiency. This study addresses this trade-off challenge by proposing Two-Phase Model by cascading Logistic Regression, a light-weight classifier, with XGBoost, a more advanced model with a reliable performance. The phase I classifier acts as a triage that filters out obvious phishing and legitimate sites and then the phase II classifier is applied only on the resulting intermediate suspicious ones, thus improving both efficiency and performance. We gather a total of 301,716 URLs, comprising 51,716 phishing and 250,000 legitimate websites and perform feature engineering for both phases. While phase I takes the lexical features of the urls, phase II utilizes additional host-based features. By merging the result of our Two-Phased Model, we obtained an overall accuracy of 90.47%, outperforming the baseline models.**

***Index Terms*—phishing website detection, classification, data mining, logistic regression, XGBoost**

I. INTRODUCTION

Phishing is a form of cybercrime that involves deceiving individuals and organizations to disclose sensitive information such as usernames, passwords and credit card details. As a result, victims suffer from serious consequences such as financial and/or intellectual property loss, identity theft and reputational damage. While there are many variations of phishing attacks including website spoofing, spear phishing, email phishing, vishing and pharming, this paper focuses on detecting phishing websites that pretend to be from legitimate sources, such as banks, or e-commerce. The number of phishing websites has risen steadily over the last decade, and Anti-phishing Working Group (APWG) has reported the detection of over 1M of unique phishing websites in the second quarter of 2025. [1]

With the rise of phishing attacks, many researchers have worked on various machine learning enabled heuristic-based detection models that rely on characteristics and behavioral patterns of the websites, in addition to traditional signature-based methods, to keep up with evolving phishing tactics. However, it comes with certain limitations: their characteristics-based nature often misclassifies legitimate websites with unconventional characteristics as phishing (false positives). [2] On the other hand, implementing advanced ML models within high accuracy imposes further complexity, demanding significant computational resources, making them less feasible for real-time detection on resource-constrained devices. [2]

The significance of this work is to bridge the gap between output accuracy and resource efficiency of existing phishing website detection techniques by cascading a light-weight classifier and an advanced classifier. The proposed two-phase model aims to quickly filter out obvious legitimate and phishing websites using a light-weight classifier (Logistic Regression), making it suitable for real-time deployment, and then analyze the remaining ambiguous cases using an advanced classifier (XGBoost), minimizing false positives, and achieving high overall accuracy.

II. RELATED WORK

The ongoing danger of phishing has prompted the creation of a number of detection methods, with an increasing emphasis on machine learning based strategies to get beyond the drawbacks of conventional blacklists.

In a thorough investigation, Joshi, et.al [3] suggest an ensemble classification model based on machine learning that uses static lexical data from URL strings to differentiate between harmful and benign websites. Based on the basic premise that malicious and benign URLs have different feature distributions, this technique is regarded as secure and quick because it does not need execution. Although this method works well, they noted that models that just use lexical characteristics might be computationally demanding and susceptible to advanced obfuscation strategies. This study emphasizes the necessity of a detection system that can respond to increasingly complex, dynamic threats in addition to handling static aspects.

Other academics have looked into using dynamic, host-based metadata to overcome the drawbacks of static analysis. Mulder [4] utilized WHOIS and SSL certificate analysis to improve early phishing detection. Because characteristics that rely on user-provided SSL fields are frequently left empty or are readily fabricated by attackers, the study suggests that features generated from WHOIS data considerably improve detection accuracy. His study emphasizes how crucial it is to include these dynamic aspects in order to build a detection system that is more reliable. His research also shows that several aspects of URLs, such the quantity of dots and dashes

and the existence of questionable patterns, are still reliable predictors of a domain's purpose. Because they support the inclusion of WHOIS data in the second, more in-depth analysis phase, the findings from this article are essential to our suggested two-phased paradigm.

Recent research has demonstrated that sophisticated ensemble boosting techniques, such as XGBoost and CatBoost, are more effective in detecting phishing attempts than traditional classifiers. In terms of accuracy, [5] shows that these tree-based classifiers perform remarkably well, with XGBoost exhibiting a modest advantage. The study's examination of two datasets demonstrates that XGBoost is a reliable option for identifying complicated phishing attempts due to its iterative nature and capacity to handle complex data. Combining XGBoost with optimization techniques significantly increases its efficacy.

For adaptive hyperparameter tuning, Birthiriya, et.al [6] provide a hybrid strategy that incorporates the Bat Algorithm and XGBoost. In order to strike a balance between finding new solutions and improving the existing ones, this bio-inspired system adjusts crucial parameters like learning rate and maximum tree depth. By preventing overfitting and greatly enhancing classification performance, this procedure makes sure the model performs well when applied to fresh, untested data. The experimental results of the study, which demonstrate a notable improvement in accuracy, offer a solid foundation for optimizing our selected classifier using metaheuristic optimization.

Similarly, Jovanović et al. [7] present a two-tier system that performs feature selection and hyperparameter tweaking for an XGBoost model using an enhanced firefly algorithm. It has been demonstrated that this all-encompassing strategy outperforms alternative techniques, demonstrating the promise of metaheuristics in machine learning model optimization for online security applications. This study confirms our decision to employ XGBoost as the central component of our advanced classification phase and supports our strategy to improve its performance through the use of a feature selection method.

III. Scope of Work

Although other researchers have used a variety of algorithms for phishing detection, we suggest a two-stage method that cascade an advanced learning model with a lightweight triage classifier. By utilizing two-phased strategy, this study seeks to close the gap between the output accuracy and resource efficiency of current phishing detection approaches. Our experiment also deal with up-to-date phishing sites that are currently online to reflect the evolving tactics. A visual representation of proposed two-phase model architecture is shown in Figure 1.

A. Phase I: Lexical Feature-Based Classification

The first usage functions as a simple triage mechanism. It starts by obtaining current lexical characteristics from both genuine and phishing website URLs. Logistic Regression is then used by the model as a rapid and effective triage classifier. For every URL, this classifier generates a likelihood score (from 0 to 1). High-confidence forecasts (such as those with scores near 0 or 1) are immediately categorized as phishing or authentic, respectively. Instead of being rejected, URLs with intermediate scores- which indicate uncertainty- are sent to the second stage for a more thorough examination.

B. Phase II: Enhanced Feature-Based Classification

The most challenging ambiguous URLs from step 1 are intended to be handled in this step. To provide a better, more complete dataset, it adds more elements such as host metadata taken from WHOIS. Then, to highlight the most important characteristics and improve model performance, the system applies a suitable feature selection technique. After that, XGBoost, a strong and effective classifier that can handle the enhanced feature set, processes chosen features to provide the final classification.

IV. Experiment and discussion

In this work, only URLS of phishing and legitimate website are obtained while all the features are self-crafted from the URL itself. Phishing URLs are acquired from PhishTank [8], a reputable platform that detect and keeps track of current phishing websites, and legitimate URLs are obtained from the Keggle dataset [9]. The phishing dataset contains 51,716 URLs that were verified by PhishTank and still online at the time the dataset was downloaded. The total legitimate URLs from Keggle dataset are 427,028 but only 250,000 URLs are randomly sampled to keep the class distributions approximately 1:5. The final dataset contains 301,716 URLs with 51,716 phishing, denoting Class 1 and 250,000 legitimate URLs, denoting Class 0. The dataset overview is shown in Table I.

TABLE I
Dataset Overview

0: Legitimate	1: Phishing	Total
250,000	51,716	301,716

A. Phase I: Lexical Feature-Based Classification

Phase I of the project focuses on a lexical feature-based classification approach. By analyzing the components of URLs, we carefully extracted 17 lexical features as shown in Table II. To use Logistic Regression as a Phase I Classifier, the features are standardized first and then applied 10-fold stratified cross-validation to evaluate the model performance. The model could filter out 223.875 samples out of 301,716 and achieved overall accuracy of **90.14**, leaving 77,844 intermediate suspicious samples, denoting as label 2.

B. Phase II: Enhanced Feature-Based Classification

The phase II of the project incorporates with Google Index info and dynamic host-based metadata from WHOIS. However, fetching WHOIS api has very limited request rate for the free tier, thus, infeasisble for this phase II dataset with 77,844 suspicious samples. Therefore, we disclaim that for

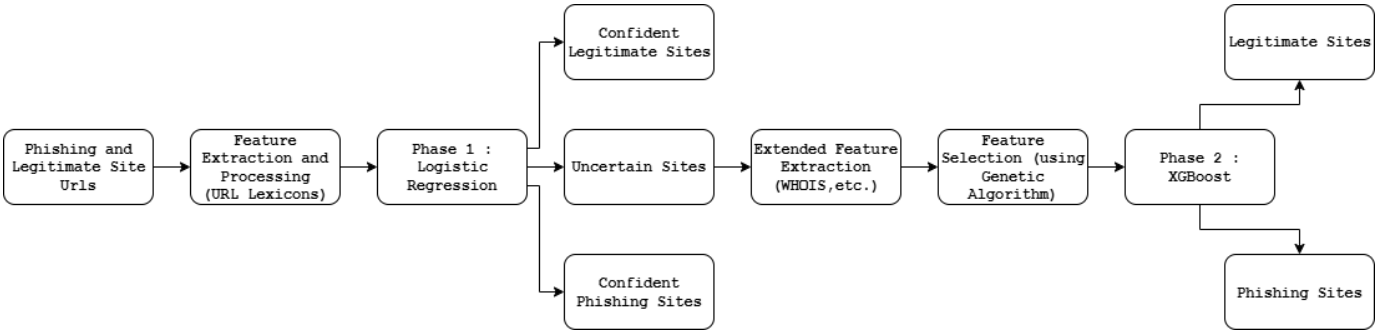

Fig. 1. Proposed Model Architecture

TABLE II
LEXICAL FEATURES

Feature	Description
url_length	Total length of the entire URL string.
domain_length	Length of the domain part.
query_length	Length of the query string.
num_dots	Count of '.' characters in the URL.
num_subdomains	Number of subdomains in the URL.
num_hyphens	Count of '-' characters in the URL.
num_ditis	Count of digit characters in the URL.
num_specials	Count of special characters (e.g., '@', '#')
digit_ratio	Ratio of digits to total characters
entropy_domain	Shannon entropy of the domain.
has_ip_address	1 if the domain is an IP address, else 0.
cert_flag	1 if HTTPS is used, else 0.
is_uncommon_tld	1 if TLD is uncommon, else 0.
is_long_url	1 if domain length ¿ 99th percentile, else 0
num_homoglyph_subs	Counts the number of characters in the domain that are commonly used as visual substitutions
has_homoglyph_subs	1 if the domain contains any homoglyph-like substitutions, else 0
keyboard_typo_score	Fraction of consecutive alphabetic characters in the domain that are not adjacent on a QWERTY keyboard

the purpose of proceeding to phase II classification with host-baseed metadata, we used label aware seeding function to generate synthetic feature that mimics WHOIS information. Table III shows the additional features extracted to be used in phase II. Among them, the categorical features are one-hot encoded, thus, forming a total of 54 features. Genetic Algorithm (GA) is used as a feature selection method to select the most relevant features, and then applied to XGBoost Classifier with 10-fold stratified cross-validation. The final model achieved overall accuracy of **91.41**%.

C. Evaluation

The proposed two-phase model is evaluated using standard classification metrics: accuracy, precision, recall, and F1-score. The confusion matrix of each phase is merge to obtain the performance metrics for the entire dataset and it is compared against baseline models such as Decision Tree, Random Forest, and Naive Bayes, which are trained and evaluated on the same dataset. Ablation studies are conducted to assess the

TABLE III
EXTENDED FEATURES FOR PHASE II CLASSIFICATION

Feature	Description
google_index	1 if domain or URL appears in Google Search Result, else 0
registrar	Domain registrar company
domian_age	today - creation_date
is_new_domain	1 if domain_age_days $<$ 90 else 0.
reg_period_days	expiration_date - creation_date
emails_domain	Domain of registrant contact emails
org_exist	if Organization or company is listed as registrant, else 0

impact of feature selection and the choice of classifiers in each phase. The statistics are as shown in Table IV as well as visually presented in Figure 2. The results demonstrate that the two-phase model significantly outperforms the baseline models, achieving an overall accuracy of **90.47**%, precision of **90.25**%, recall of **95.67**%, and F1-score of **92.88**%. The ablation tests indicate that both phases contribute to the model's performance, with feature selection playing a crucial role in enhancing the effectiveness of the XGBoost classifier in Phase II.

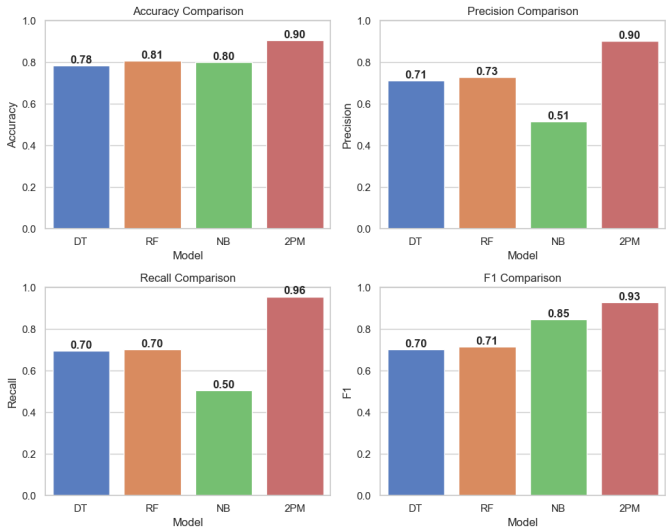

Fig. 2. Comparison of Accuracy, Precision, Recall, and F1 across four models.

TABLE IV
PERFORMANCE COMPARISON OF MODELS AND ABLATION STUDIES

Compare against Basedline Models				
Model / Phase	**Accuracy**	**Precision**	**Recall**	**F1-score**
Decision Tree	0.7836	0.7123	0.6954	0.7038
Random Forest	0.8067	0.7291	0.7012	0.7148
Naive Bayes	0.7998	0.5141	0.5045	0.4869
Ablation Test				
Phase I (entire dataset)	0.7163	0.7052	0.6927	0.6989
Phase II (entire dataset)	0.8283	0.7215	0.7215	0.7156
Phase I (triage)	0.9014	0.7967	0.8874	0.8310
Phase II (suspicious)	0.9141	0.8914	0.8684	0.8791
Phase I + II w/o FS	0.8645	0.8984	0.8926	0.8955
Complete Model	**0.9047**	**0.9025**	**0.9567**	**0.9288**

D. Discussion

Using logistic regression on lexical features is fast and efficient, with time complexity of $O(n * m^2 + m^3)$ [10]. Despite its efficiency, ablation tests show that applying it to the entire dataset lowers the accuracy. However, by making it a triage classifier by excluding ambiguous intermediate samples, it improves results significantly. On the other hand, fetching dynamic data adds delay, as well as applying genetic algorithms for feature selection also takes significant time. A more powerful classifier, XGBoost with higher time complexity: $O(Kd\|\mathbf{x}\|_0 \log n)$ [11] is applied only to the reduced dataset, this, reducing overall computational time with improved accuracy. This approach finds a good balance of models efficiency and accuracy, thus achieving our objective.

V. CONCLUSION AND FUTURE WORK

This paper presented a Two-Phase Phishing Website Detection framework, proofing the improvement in accuracy over traditional baseline classifiers. It also decreases the computational resource consumption by applying complex model and algorithms only to the ambiguous websites. However, the accuracy of 90.41% shows that it has much potential in improving the detection work. Some of the improvement can be carried out, by conducting more research on lexical feature creation, acquiring actual WHOIS information, adding html contents feature (such as popups) and most effectively visual elements, that deceive users. Feature selection can be performed more effectively by exploring different metaheuristic algorithms. In terms of classifier, automated hyperparameter tuning can be integrated to maximize the performance.

REFERENCES

[1] Anti-Phishing Working Group (APWG). [Online]. Available: https://apwg.org/.

[2] W. Li, S. Manickam, Y. -W. Chong, W. Leng and P. Nanda, "A State-of-the-Art Review on Phishing Website Detection Techniques," in IEEE Access, vol. 12, pp. 187976-188012, 2024, doi: 10.1109/ACCESS.2024.3514972.

[3] Joshi, M., Tan, C. H., Wang, Y., & Samuel, J. (2019). Using lexical features for malicious URL detection: A machine learning approach

[4] Mulder, C. J. (2020). Improving early phishing detection using SSL & WHOIS data: an application to PhishDetect for Civil Society Protection

[5] Sadaf, K. (2023, February). Phishing website detection using XGBoost and Catboost classifiers.

[6] Birthiriya, S. K., Ahlawat, P., & Jain, A. K. (2025). Phishing website detection with XGBoost and adaptive hyperparameter optimization using the Bat Algorithm.

[7] Jovanović, L., Jovanović, D., Antonijević, M., Bacanin, N., Živković, M., & Stanimirović, I. (2023). Improving phishing website detection using a hybrid two-level framework for feature selection and XGBoost tuning

[8] Anti-Phishing Community, "PhishTank: Join the fight against phishing." [Online]. Available: https://phishtank.org/.

[9] Keggle, "Phishing and Legitimate URLS" [Online]. Available: https://www.kaggle.com/datasets/harisudhan411/phishing-and-legitimate-urls.

[10] Kaggle Community, "Computational Complexity of Machine Learning Models – II," Kaggle Discussions. [Online]. Available: https://www.kaggle.com/discussions/general/263127

[11] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, San Francisco, CA, USA, Aug. 2016, pp. 785–794.