From Phishing Behavior Analysis and Feature Selection to Enhance Prediction Rate in Phishing Detection

Asmaa Reda Omar, Shereen Taie, Masoud E.Shaheen
Computer Science Department,
Faculty of Computers and Information,
Fayoum University, Fayoum 63514, Egypt

***Abstract*—Phishing incidents have captured the attention of security experts and end users in recent years as they have become more frequent, widespread, and sophisticated. The researchers offered a variety of strategies for detecting phishing attacks. Over time, these approaches suffer from insufficient performance and the inability to identify zero attacks. One of the limitations with these methods is that phishing techniques are constantly evolving, and the proposed methods are not keeping up, making it a hard nut to crack. The objective of this research is to develop a URL phishing detection model that can demonstrate its robustness against constantly changing attacks. One of the most significant contributions of this paper is the selection of a novel combination of features based on literal and recent phishing behavior analysis. This makes the model competent sufficient to recognize zero attacks and able to adjust to changes in phishing attacks. Furthermore, eleven machine learning classification techniques are utilized for classification tasks and comparative objectives. Moreover, three datasets with different instance distributions were constructed at different times for the model's initial construction and evaluation. Several experiments were carried out to investigate and evaluate the proposed model's performance, effectiveness, and robustness. The experiments' findings demonstrated that the GaussianNB method is the most durable, capable of maintaining performance even in the absence of retraining. Additionally, the LightGBM, Random Forest, and GradientBoost algorithms had the highest levels of performance, which they were able to maintain by routinely retraining the model with newer types of attacks. Models that employed these three suggested algorithms outperformed other current detection models with an average accuracy of about 99.7%, making them promising.**

Keywords*—*Gradient boosting; light GBM; machine learning; phishing; phishing URL; random forest

I. Introduction

Phishing is a crime to steal personal data and financial account credentials by employing social engineering and technical deception. This type of attack leads victims to deal with counterfeit websites and fool them into believing that they are legitimate and trusted ones by using deceptive e-messages with deceptive e-addresses. These sites trick recipients into revealing extensive financial and personal information, leading to significant aggregate identity theft and financial losses. These attacks could also instill malware onto victims' computers to directly steal credentials, often using systems that intercept victims' account data, user names, and passwords, or misdirect consumers to counterfeit websites [9]. Phishing is a significant threat to Internet users. It also causes pecuniary loss and reputational impairment to the targets, like universities, companies, charities, and government entities. The first phishing attack was on E-Gold in June 2001 [25]. Although it was not considered successful, it planted a vital seed, and it established the basics of how phishers would operate going forward and still do, in large part, today. Phishers in late 2003 registered many domains that looked like legitimate sites such as eBay and PayPal. By the beginning of 2004, they were achieving considerable success that included attacks on banking sites. Since then, they have improved their methods to be more sophisticated, but they all still work on the same basic concept, which has proven to be quite effective. Phishing attacks result in a colossal loss of sensitive/personal information and even funds whose total amount could be billions of dollars in one year [31].

Since the beginning of 2020, the Anti-Phishing Working Group[1] (APWG) was tracking between 68,000 and 94,000 attacks each month. In the fourth quarter of 2021, APWG saw 888,585 attacks, which was the previous high. March 2022 had the highest monthly total in APWG's reporting history with 384,291 attacks. APWG recorded a total of 1,025,968 phishing attacks in the first quarter of 2022. This quarter's phishing activity was the worst that the APWG has ever recorded, and it was also the first time that the quarterly total exceeded one million. The number of phishing attacks has more than tripled every year.[8] According to studies on the user experiences of phishing attacks [33], [20], computer users are susceptible to phishing for the following reasons: Users improve their confidence and vulnerability as a result of decreasing their chances of falling victim to a phishing attack. Additionally, they lack a thorough understanding of URLs, are unaware of trustworthy websites, and are unable to view the complete URL of a web page because of redirectors or hidden URLs. They do not have much time to check the URL or access certain online pages mistakenly. They are unable to discriminate between legitimate and phishing web pages. Regardless of how important caution and experience are to the user, it is not entirely possible to prevent users from being caught in phishing attacks using their expertise. Technological advancement has provided phishers with better tools to launch dangerous and sophisticated attacks, making even the savviest internet users vulnerable. [18] For instance, Examining URLs carefully and avoiding sites that do not have

[1] https://apwg.org/

an SSL certificate have been one of the main recommendations for avoiding phishing sites for many years. A website that has ”HTTPS” in the URL is one that is secured by the HTTPS encryption protocol and has an SSL certificate. This method, however, is no longer effective for identifying suspicious websites. According to APWG's report [9], SSL was used by 84 percent of the phishing sites that were examined in the fourth quarter of 2020. This with quarterly increases of about 3%. To increase the success of phishing attacks, attackers have considered end-user personality traits, particularly the ability to deceive experienced users. A spear phishing attack is one that targets a specific organization, business, or individual. This type of attack is not typically carried out by random attackers, but rather by criminals seeking financial gain, trade secrets, or military information. Furthermore, some active attackers constantly innovate and learn how to circumvent new defensive methods, causing attacks to evolve on a daily basis and luring victims into gaining access to their accounts and financial information.

Since 2004, researchers have been working to combat phishing, which has become such a severe menace that it has caused significant damage. As the term 'phishing' revealed on DBLP[2] (Digital Bibliography & Library Project), the number of research articles released about detecting phishing attacks increases year after year. Phishing attacks exploit human users' weaknesses, and attackers are always devising new strategies to avoid detection. As a result, additional assistance systems are required to secure the systems/users. As decision support tools for users, software-based approaches are preferred. These approaches are classified as list-based, search engine-based, visual similarity-based, and machine learning-based. In dealing with phishing attacks, the machine learning-based strategy is the most successful. All researchers work for the same objectives: high detection accuracy, detection stability, fast detection, zero-day detection, language independence, and real-time detection. There are however some drawbacks that researchers must contend with, such as restricted datasets and the requirement for up-to-date information as phishing strategies evolve; additional features are difficult to obtain, slow, third-party dependant, and time consuming. As a result, certain machine learning systems need a significant amount of computing to acquire and calculate the features of diverse sources. In addition, the solution must be constantly improved to deal with changes in attack technique.[18] On the attacker's side, the technologies' support to attackers allows them to effortlessly deceive the victims. Consequently, phishing is one of the most persistent and rapidly growing online threats; identifying phishing attacks is one of the ongoing issues, and the hunt for a better solution continues.

This paper proposes new models for detecting URL phishing using a new set of fourteen robustness features. These features were chosen after observing the most recent and previous phishing attacks and focusing on URL phishing detection models and the literature features in order to consider the most important features and build a robust classification model that can deal with ever-evolving attacks. Furthermore, three new datasets were created at various points in time, one for building the model and the others for testing and measuring the model's performance and robustness. Eleven different machine learning algorithms were evaluated to determine the model's best classification performance. Several experiments were carried out in order to assessed the models. The main contributions of this paper are as follows:

- Introducing a novel combination of phishing URL detection features based on observations of old and recent phishing attacks. To the best of our knowledge, this is the first paper to analyze such criteria while constructing a feature set for the phishing detection system. The main objective of this approach is to ascertain how the phishing feature set can be sufficiently integrated into an effective countermeasure that can handle constantly changing attacks.
- Implementing a phishing URL detection model that is difficult for attackers to avoid and outperform other existing detection models. The proposed model could maintain its performance and detect any phishing URL whether it came from the pretrained or new datasets, even if the trained dataset was outdated, making it a promising solution for the phishing detection problem.
- Developing a robustness test utilizing three new URL datasets (phishing / genuine) gathered at different times over a three-year period to assess the performance of the proposed model.
- Examining the effect of retraining on model performance to emphasize the importance of regular model retraining for newer types of attacks, as well as having a robustness feature for dealing with constantly evolving attacks.

The remainder of this paper is organized as follows; the next section provides a literature review. Our methodology is proposed in Section III. Section IV highlights the experiments and evaluations of our proposed model. Section V concludes this paper, along with future work and directions.

II. Related Work

Various methods and approaches have been investigated in order to understand and address phishing attacks. There are two types of phishing attack detection methods: user education-based and software-based. User education-based approaches try to improve users' ability to detect phishing attacks. These approaches teach people how to distinguish between authentic and phishing websites and emails. Software-based approaches are preferred as decision support systems for the user; these approaches are further classified into four types: blacklisting, visual similarity, machine learning, and hybrid methods. The widely used approaches to detect phishing websites is those based on machine learning. Classification, one of the primary areas of machine learning algorithms, is a widely used strategy for detecting phishing websites. The four stages of classification are typically preprocessing, feature generation, feature selection, and classification. The primary issue of classification algorithms is improving accuracy. Improving each categorization process may result in increased overall accuracy. In this section, we will concentrate on the most relevant and significant publications, as well as the existing methodologies for detecting phishing attacks that have been proposed in the literature.

[2] https://dblp.org/

Authors at [10] examined three important machine learning classifiers, Artificial Neuron Network (ANN), K-Nearest Neighbor (K-NN), and Decision Tree (C4.5), to cast with Random Forest Classifiers (RFC) in order to present a prototype for detecting phishing attacks on a website using the machine learning algorithm. According to the study, RFC outperforms other classifiers in terms of detection accuracy, scoring 97.33%. 4898 legitimate and 6157 phased websites were used in the experiment, and the researchers came to the conclusion that adding more variables to the process will increase the detection accuracy.

In this research [32], the authors employ a machine learning technique to handle the phishing problem, producing a model that uses 30 different features for phishing identification with three different algorithms: Random Forest RF, Support Vector Machine SVM, and classification tree CT. They create five alternative classification situations, including each algorithm alone, the combination of AND Techniques, and the combination of OR Techniques. The experiment results reveal that the Classification Trees technique has the most effectiveness in predicting whether a URL is secure or not, with an accuracy of 90% across a set of website links.

Authors at [21] examine different machine learning algorithms, including K-Nearest Neighbors (KNN), Decision Tree (DT), Support Vector Machines (SVM), Logistic Regression (LR), Random Forest (RF), and Extra Trees, to determine the best technique for detecting phishing websites. After comparing all of these techniques, authors decided that the Random Forest Classifier is the best for Phishing Website Detection. The authors at [27] compared the use of Random Forest, probabilistic neural networks, and XGBOOST in detecting Phishing and discovered that XGBOOST produced the best results in terms of MCC, F.score, and accuracy. This research [22] examines hyperlinks in HTML source code to detect phishing websites. Authors present a novel phishing detection approach that is client-side, language-independent, and achieves more than 98.4% accuracy when the Logistic Regression algorithm is used.

Authors at [7] propose a machine learning-based detection model and compare various algorithms. They also used various feature selection tools to select the most valuable features in 20 of the 48 features. According to their conclusion, Random Forest is the most effective classifier to use because it detected a phishing attack with 98.11 accuracy in 2.44 seconds. Fifteen features from various classes were chosen in this paper [34]. Five machine learning classifiers were tested, and it was discovered that random forest had the highest detection accuracy (94.79%). This study investigates the importance of each feature class and all potential combinations of feature classes. Authors in this research [19] created a phishing detection approach that only requires nine lexical features to detect phishing attacks. Their dataset contains 11964 instances of legitimate and phishing URLs. They tested their approach on various machine learning classifiers, including Random Forest, k-Nearest-Neighbor, support vector machine, and logistic regression, and found that the Random Forest algorithm had the highest accuracy of 99.57%. The authors claim that their approach's main contributions are third-party independence, real-time detection, detection of new websites, and use of limited features.

The authors of this paper [29] show that a machine learning model trained on old datasets can perform well when tested on those same old datasets, but when tested on new datasets, using the same features in both cases, its performance noticeably degrades. They also show that SVM is the most resistant to the new tactics employed by the current phishing attacks among the widely used machine learning algorithms. With the newly created dataset, their experimental findings revealed that Random Forest is the most effective strategy among all methods that were tested, including Support Vector Machines (SVM), k-Nearest Neighbors (kNN), Naive Bayes (NB), and Logistic Regression (LR).

Authors at [11] identify an effective machine learning approach for phishing URLs detection based on precision, false-positive rate, and false-negative rate. To ascertain the classification accuracy in phishing detection, different classifiers including Random Forest, Linear SVM, SVM Polynomial Kernel, and SVM Sigmoid Kernel were used. With an accuracy of about 97.42%, the result showed that Random Forest outperformed the other three machine learning algorithms. For similar purposes, six distinct machine learning classification techniques are used to identify phishing websites in this study [23]. the Gradient Boost Classifier had the best possible accuracy of 94.75%, while the Random Forest Classifier got the highest possible accuracy of 97.17%. Provisioning accuracy for the Decision Tree classifier is 94.69%. In contrast, SVM has a provisioning accuracy of 56.04%, KNN has a provisioning accuracy of 60.45%, and Logistic Regression has a provisioning accuracy of 92.76%.

Authors of [6] investigated the predictive performance of a number of machine learning techniques, such as Random Forests (RF), Logistic Regression (LR), Classification and Regression Trees (CART), Neural Networks (NNet), Support Vector Machines (SVM), Bayesian Additive Regression Trees (BART), and other models of AI algorithm. Based on their comparison results, the Gradient Boosting Classifier and Random Forest Classifier had the highest accuracy. Using a dataset from a phishing website called ”phising.csv,” authors in this article [14] examined the accuracy of XGBoost Classifier, Decision Tree Classifier, Random Forest Classifier, SVM Classifier, KNN Model, Logistic Regression Model, and AdaBoost Classifier methods. XGBoost Classifier & Random Forest Classifier had better accuracy according to their results, even after applying SMOTE and PCA Techniques to the dataset to account for accuracy discrepancies.

This study [16] suggests a hybrid feature-based anti-phishing technique that only uses client-side URL and hyperlink data to extract features. In order to conduct experiments utilizing well-known machine learning classification algorithms, they also create a new dataset. Their test results demonstrate that the suggested phishing detection method is superior than conventional methods, with a detection accuracy of 99.17% using the XG Boost technique. Random Forest, Decision Tree, Light GBM, Logistic Regression, and Support Vector Machine methods are compared in this study [5] to evaluate and choose the best classification algorithm for the phishing problem. Their findings demonstrate that the Light GBM algorithm delivered the greatest results.

Considering the feature selection, the authors of this research [30] discuss the efficacy of two feature selection methods, Omitting Redundant (FSOR) and Filtering Method (FSFM), in detecting Phishing Websites and compare the efficacy of three different machine learning algorithms: Naive Bayes (NB), Multilayer Perceptron (MLP), and Random Forest (RF). According to their empirical data, the optimized Random Forest (RFPT) classifier with feature selection by the FSFM outperforms all other strategies. Moreover, a framework for feature selection was described by the authors at [13] ,[12]. They presented an empirical hybrid framework with two stages that takes into account the filter and wrapper method. Those researches involve applying models with optimized (hyperparameter) parameters, such as Artificial Neural Network, XGBoost Classifier, and Random Forest Classifier, on two phishing datasets. The outcomes demonstrated that the XGBoost Classifier performed better than other classifiers.

Authors In this paper [26], proposed a strategy to identify the critical features by combining correlation and recursive feature elimination. The first scenario combines power predictive score correlation and recursive feature elimination, and the second scenario combines the maximal information coefficient correlation and recursive feature deletion. The third scenario combines recursive feature removal and Spearman correlation. According to their experimental findings, even with the lowest feature subset, all three scenarios from the combined findings of the offered approaches reach a high level of accuracy. Additionally, they discovered that Random Forest (RF) performs more accurately in identifying phishing websites.

A systematic review of phishing detection systems based on machine learning was carried out at [17]. The authors noted that studies that include more features have higher performance findings, studies that contain more features are more often used, and runtime performance was overlooked by most systems. In [15], the authors conducted a similar systematic review on machine learning-based phishing detection systems. They ranked classifiers based on the number of studies that used them. However, their conclusion is based solely on the statistical analysis of the studies under consideration. Moreover, the authors of [4] provide a systematic review of existing studies concentrating on Machine Learning and Deep Learning based phishing website detection in order to identify the major gaps and provide appropriate solutions. Their findings show that the imbalanced dataset use, issues with appropriate feature selection techniques, source selection, train-test split ratios, dataset size, inclusion and exclusion of website features, and run-time analysis are the main contributors to these flaws. Moreover, the results show that Random Forest, in the vast majority of peer-reviewed research articles, has the best overall accuracy.

In summary, the majority of the studies reviewed in this paper concentrate on the classification phase. Well-known machine learning algorithms like KNN, SVM, XGBoost Classifier, Decision Tree, Logistic Regression (LR), and Random Forest were used in the majority of the research. The Random Forest and the XGBoost Classifier algorithms are consistently yielding the best performance results when they were compared to other algorithms. The other part of the previous works is worked on feature selection phase through evolutionary and metaheuristic algorithms, and also some authors proposed hybrid feature selection models. The feature set can be derived from a variety of sources, including the page source, search engine, URL, website traffic, and DNS. High detection accuracy, detection stability, quick detection, zero-day detection, language independence, and real-time detection are the universal goals shared by all researchers.

Additionally, these methods have a number of limitations that must be addressed in order to detect phishing URLs. To begin, the limited datasets and the requirement for updated datasets as phishing techniques develop; the majority of the work has employed preclassified and smaller datasets, which do not produce exact efficiency and precision when applied to great and real-world datasets. additionally these approaches suffer from insufficient performance and the inability to identify zero attacks over time; as phishing techniques are always evolving, and the proposed methods are not keeping up. Second, the previously extracted features are comprehensive, with the limitation that such extraction requires a significant amount of time. Third, certain approaches used statistical methods to choose relevant features, while others proposed their own features; researchers often did not consider how their features can be defeated. Although these strategies have been effectively implemented in various approaches, they generate inaccurate results when domain knowledge is not amplified. Fourth, the previous methodologies offered lack advanced evaluation measures; the majority of the offered solutions don't concentrate at robustness and accuracy over time. To improve the classification accuracy of phishing websites, our suggested methodology concentrated on the feature selection phase as well as the classification phase. Furthermore, various datasets gathered over time and various experiments are used to test the model's performance and robustness.

Phishing incurs significant financial costs and can harm a company's, government entity's, or university's reputation. It also harms the systems of web hosts, email providers who must protect users from phishing spam, and responders tasked with defending networks and users. The number of phishing attacks discovered is constantly increasing. Phishing remains one of the most persistent and rapidly evolving online threats. As a result, the search for a better solution to overcome the limitations of existing solutions continues.

III. Methodology

A. Datasets

There are no benchmark datasets for detecting phishing websites. This is due to the limited lifespan of phishing websites and the inability of content-based analysis to exploit dead URLs. Furthermore, the majority of datasets are restricted to experimental feature values with no URL references. This prevents datasets from being reproduced or tested with different features. Moreover, the authors of this study [18] noticed a decline in performance when previous methods were evaluated on a current dataset, even after retraining. This drop in performance highlights the necessity of using a broad, high-quality, and up-to-date dataset when creating models. It is critical to have a robust model that can deal with constantly shifting attacks. Training classification algorithms on one dataset and then testing on a different recent one is one strategy to assess the robustness of the detection model.

As a result, three URL datasets (phishing and legitimate URLs) were collected over a three-year period. The first dataset (24,200 URLs) was collected in June 2020 for use in model building (training and initial testing), followed by the second (16,028 URLs) and the third (15,974 URLs) in October 2021 and January 2022, respectively. Both datasets will be used later to test the model's robustness without and with retraining. The classification of these datasets is displayed in Tables I, II, III.

Legitimate webpage URLs are gathered from Alexa[3], University of New Brunswick open databases[4], and Mendeley Data repository[5]. For Alexa it only recommends top-ranked domains without mentioning sub-domains or paths. As a result, for the diversity of URLs, those lists cannot be used directly, especially when features such as subdomains and paths are used. To address this issue and provide a realistic dataset, the collected domains are used as seeds for crawling 10 URLs per domain. It was then processed through to remove duplicate and domain-only URLs, allowing for more representative samples. Phishtank is used to collect phishing URLs. All duplicate and defunct URLs are deleted during preprocessing of the collected URLs, and a maximum of 10 URLs with the same domain name are preserved.

TABLE I. Dataset 1 (24,200 URLs) June 2020

Database	Number of instances	Phishing/legitimate
PhishTank dataset	4,010 out of 6,233	Phishing
Alexa top-ranked websites	2,019 ended with 20,190	Legitimate

TABLE II. Dataset 2 (16,028 URLs) October 2021

Database	Number of instances	Phishing/legitimate
PhishTank dataset	2,652 out of 4,862	Phishing
Alexa top-ranked websites	1,000 ended with 10,000	Legitimate
Mendeley data	3367	Legitimate

TABLE III. Dataset 3 (15,974 URLs) January 2022

Database	Number of instances	Phishing/legitimate
PhishTank dataset	2,659 out of 4,312	Phishing
the New Brunswick university datasets	2315	Legitimate
Alexa top-ranked websites	1,100 ended 11,000	Legitimate

B. Preprocessing

After collecting each dataset, it must be prepared for the feature extraction procedure. As soon as the dataset is collected, the preparation procedure begins. Three datasets and three distinct preparation processes have been completed. After gathering the dataset, the next stage is data preprocessing, which involves:

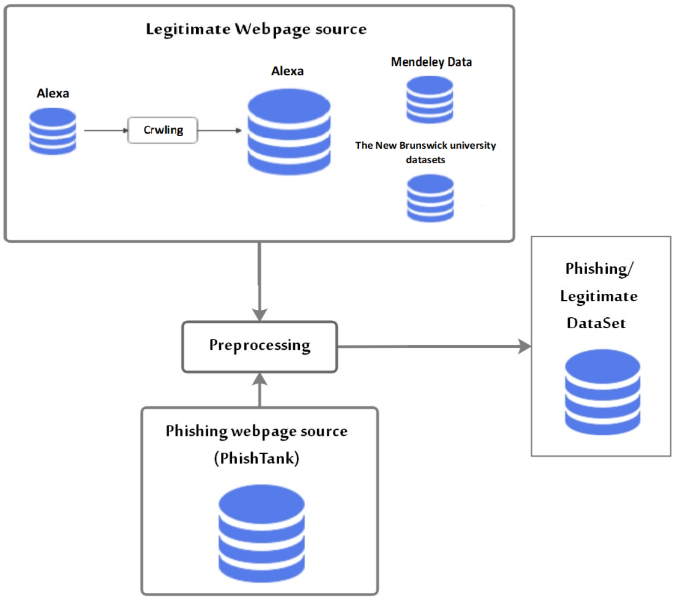

Legitimate Webpage source
Alexa
Crwling
Alexa
Mendeley Data
The New Brunswick university datasets
Preprocessing
Phishing/ Legitimate DataSet
Phishing webpage source (PhishTank)

Fig. 1. Dataset construction process.

1. Eliminating redundant URLs and ensuring there are no intersections between the URLs in the three datasets.
2. Filtering all URLs to avoid broken URLs to ensure that the required accurate features are extracted during the feature extraction procedure.

It is important to note that using a fresh dataset is necessary because the majority of URLs provided by PhishTank or other providers won't likely be active for three months or less. The collected phishing URLs were 4,010 out of 6,233 in the first dataset, 2,659 out of 4,312 in the second dataset, and 2,652 out of 4,862 in the third dataset. For the legitimate URLs, to ensure there are no redundancies, the URLs were collected with different ranking values. As a result, we obtained three distinct datasets. The general procedure used to create the dataset is shown in Fig. 1. The dataset is now ready for the next stage.

C. Feature Selection

The process of feature selection is crucial because correctly chosen features can improve classifier performance, while poorly chosen features can have the opposite effect. Feature selection is frequently accomplished by extracting as many features as possible and weighting them statistically in order to select the most weighted and vital aspects. This procedure generates dataset-dependent features that result in the best model performance. This performance, however, is not permanent because the features gradually lose significance and weight with time and a use of new datasets. This method of selection is one of the primary causes of the reduction in performance observed by prior works when their models were tested on new datasets. As a result, selecting robust features is a key goal to ensure that the classification model's performance is maintained over time. A feature or collection of features

[3] https://www.alexa.com/
[4] https://www.unb.ca/cic/datasets/url-2016.html
[5] https://data.mendeley.com/datasets/c2gw7fy2j4/3

is considered robust if a phisher is unable to readily build a phishing website with features that mimic those of a reliable website.

In the next subsection, a phishing newest behavior is evaluated to serve as a guide in the selection process in order to identify a robust feature collection and efficiently detect phishing attacks. It is based on the most recent APWG reports as well as the most recent annual study of the scope and distribution of phishing by Interisel Consulting Group[6]. This part aims to increase understanding of the rate at which phishing is evolving, gather and assess the most recent phishing attack characteristics, and pinpoint which characteristics suggest more effective ways to combat phishing. Furthermore, it recommends which vulnerable features should be changed or ignored, as well as which additional phishing features are necessary.

1) Phishing Behaviour: In regard to phishing and how it has evolved, phishers always develop and vary their methods and approaches in order to avoid being detected and enticing victims. Thus, phishing is a significant hazard to millions of consumers, and it remains one of the most rapidly growing and persistent online threats confronting today's businesses. [1] So, staying up to date on the latest phishing strategies and phisher habits will keep us one step ahead of phishing attacks.

According to the authors of this report [28], the average lifetime of a phishing attack from start to the last victim is only 21 hours. Moreover, the Anti-Phishing Working Group's Global Phishing Survey [3], indicates that when victims begin accessing phishing sites, antiphishing entities take an average of 8 hours and 44 minutes to detect the attack. During this time, 63% of victims are exploited before the attack is detected and stopped. Therefore, in order to detect zero phishing attacks, you must not rely only on phishing blacklists. Additionally, you must work with a fresh dataset to ensure that you have the necessary features before the URL expires.

According to the data in [1], many phishing sites go undiscovered for days, if not months, allowing them to carry out their attacks. Around 78% of malicious sites were identified during the first year of registration, and 22% of phishing domains were older than a year. According to the authors of [1], the majority of malicious domains are used for phishing within the first three days of registration, while some domains are used within 14 days. Phishers typically employ them quickly to escape discovery. Some phishers recently waited more than 90 days after registering their domains to move out of the new domain status, which earns low reputation scores from security and anti-spam firms. According to their findings, 17% of maliciously registered domains were not used within 90 days of registration. **In this instance, the "Domain age" feature is still a good option from this standpoint**.

According to [24], 42% of phishing domains are compromised, and 58% of them have malicious registrations. Furthermore, the authors of [1] showed that 61% of the 99,412 domains utilized for phishing during their study period were maliciously registered, with the remaining 39% classed as compromised. In contrast to phishing that occurs on compromised (hacked) domains held by innocent parties, maliciously registered domains are domain names registered by phishers to conduct phishing sites. According to the authors of a recent Interisel Consulting Group analysis [2], phishers hosted more attacks on compromised sites than malicious domains (a 53:47 ratio). This is consistent with the idea that hacked hostnames are appealing to phishers since they are more difficult to detect.

Many malicious domains have distinguishing features that may be utilized to rapidly and accurately detect them. In the event of hacked domains, however, it has legitimate features that will result in a high percentage of false-negative rates. To address this issue, we must distinguish between hacked and legitimate domains. We have principally based on three aspects, from the most recent common and best practices in the field: **1. A WHOIS-based feature (the domain's age), 2. An engine-based feature (web traffic), and 3. A feature based on HTML (if it has fake forms, broken hyperlinks, or foreign hyperlinks). All of these factors can be used to distinguish between legitimate and compromised domains**.

The majority of phishing attacks target just a few domain registries, domain registrars, and hosting companies.[24] About 9% of Phishing happens at a small number of providers that provide subdomain services [1]. As a member of the APWG, RiskIQ continuously analyzes the domain name system for instances of phishing. They discovered that out of the 6,153 distinct phishing URLs submitted to the APWG's eCrime Exchange in Q4 2020, 3,598 were hosted on unique second-level domains and 15 more were hosted on unique IP addresses without domains.[9]

In addition, Axur (an APWG member company) discovered that 63 percent of phishing domain names lacked a catchy keyword or contained or imitated brand names (like "accountupdate "or "sale"). In Q2 of 2020, it was 58%, while in 2019, it was 33%. To be clear, phishers aim to escape detection by utilizing generic terms instead of brand names in their selected domain names since telltale words in domain names are easier for defenders to locate.[9] Instead, phishers attempt to fool Internet users by taking advantage of the fact that characters in different language scripts may be virtually (or entirely) identical, allowing the phisher to impersonate a brand name. Phishers do employ them on occasion, though, since they can mislead the human eye and avoid detection by security tools that do not identify the words they are designed to represent [8].

In certain situations, **lexically-based characteristics, such as equal or hexadecimal in the URL and digits in the domain name,** might be used as a warning sign for phisher deceit. Moreover, a good signal for a phishing URL may be found in **the WHOIS-based feature; Registrar Name,** which phishers hide to avoid being blacklisted. Furthermore, the usage of subdomains that lead to phishing URLs is also indicated by **URL-based characteristics including URL length, host length, path length, and the number of dots in the URL**.

According to [9] several deception strategies phishers use to deceive consumers include encryption designed. In Q4 2020, 84 percent of phishing sites used SSL/TLS certificates, up 3 percent quarter over quarter, and 10% year over year. Furthermore, they discovered that 89 percent of the certificates used in phishing were Domain Valid "DV" certificates; these are routinely offered for free, and because they do not need

[6] https://www.interisle.net/index.html

human authentication, simply the domain name being used, they provide the lowest type of certificate validation. **In this case, the https/http check is not a sign of a phishing attack.** however, with a little motivation, we can still take use of this feature and try to leverage this clue to provide a low false-positive rate.

Based on these indications, fourteen features are in a great position to detect and prevent the bulk of phishing that occurs on maliciously and compromised domains. The structure, content, behavior, and URL of phishing websites have been considered. The selection of these features is one of the paper's main contributions. A novel fourteen-feature combination was presented to improve the detection accuracy of phishing URLs and verify resilience to deal with ever-evolving attacks.

D. Feature Extraction

This phase extracts features from the URL dataset. The extracted features are categorized as HTML-Based, URL-Based, Lexical-Based, WHOIS-Based, and Engine-Based Features, for a total of 14 features.
A data collector script was created. APIs are used for features that rely on a third party, such as WHOIS and Engine function. HTML parsing was necessary for the HTML features. Other features, such as URL and Lexical, were extracted quickly. Next the extraction and storage of these features for each URL, literature-based heuristics were employed to construct the feature vector, as illustrated below. To produce the labeled dataset, each URL needs its own feature vector. The feature vector corresponding to each URL is specified as F = F1, F2, F3,..., F14. Each attribute generates a value in the form of 1 or 0, with 1 indicating phishing and 0 indicating legitimate. Finally, the feature vectors were stored by the script into the database to be used later in the classification stage.

URL-BASED FEATURES:

- Feature 1 (F1): URL length
Although some phishers employ the accessible URL shortening tool, others continue to use the lengthy URL in the address bar to conceal the brand or company name. Legitimate URLs are often short in order to be easily remembered. Many phishing URLs, on the other hand, are lengthier since they rely on clicking on the phishing URL, and the phisher usually hides the redirected information in that long URL. URLs that are longer than 54 characters are given 1 (phishing), otherwise 0 (legitimate).

- Feature 2 (F2): Sub-domain
The URL of the majority of phishing websites has more than two subdomains. Each domain is separated by a dot (.), and it is uncommon to see more than one subdomain in the URL of a legitimate site. So, URLs with more than three dots are assigned 1 (phishing), otherwise 0 (legitimate).

- Feature 3 (F3): Secure Connection
Although it is simple to obtain a free SSL certificate from free sources such as Let's Encrypt, some phishers continue to avoid utilizing the HTTPS protocol. With a little drive, we can still make use of this feature. A flowchart for the security check was shown in Fig. 2. Simultaneously, the vast majority of trustworthy websites are secure. Furthermore, even if the supplied URL is not secure, the URL is tested after replacing HTTP with HTTPS to determine whether the provided domain is safe or not, resulting in a more accurate extraction by lowering the false-positive and false-negative rate. Furthermore, for each domain having a certificate, we validate the certificate. Depending on whether the certificate is genuine, the value assigned to this feature is 1 (phishing) or 0 (legitimate).

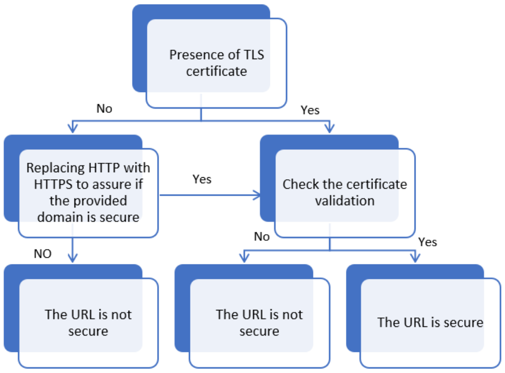

Presence of TLS certificate
No
Yes
Replacing HTTP with HTTPS to assure if the provided domain is secure
Yes
Check the certificate validation
NO
No
Yes
The URL is not secure
The URL is not secure
The URL is secure

Fig. 2. Flowchart for secure connection check.

- Feature 4 (F4): Host Length
Like URL length, genuine URLs are frequently short in order to be easily remembered and swiftly published, but phishing URLs are longer in order to conceal the identity of the site. So, URLs with host length more than 20 characters are allocated 1 (phishing), else 0 (legitimate).

- Feature 5 (F5): Path Length
The same as URL and host length, should not be too lengthy to ensure that the whole URL is not too long as a result. So, URLs with path length more than 35 characters are assigned 1 (phishing), otherwise 0 (legitimate).

HTML-BASED FEATURES:

- Feature 6 (F6): Fake Form
Following HTML processing, look for the page form. Assume the page has forms with external actions. In such instances, it is a fake form since it is most likely a phishing form that takes data and transmits it to an external processing website. So, depending on whether the page contains external form activities or not, the value assigned to this feature is 1 (phishing) or 0 (Legitimate).

- Feature 7 (F7): Broken Hyperlinks
The majority of phishers are just interested in one page that they are releasing. Most likely, most of the hyperlinks on the website are broken, thus if we discover that the majority of the hyperlinks are broken, it is an indication that the URL may be a phishing

URL. So, if the percentage of broken hyperlinks is greater than 25%, the feature is assigned 1 (phishing), otherwise it is assigned 0. (Legitimate).

- Feature 8 (F8): Foreign Hyperlinks
To prevent having broken hyperlinks on the website, most phishing pages use external functional URLs. So, if we discover that the majority of the hyperlinks are foreign, it is a strong sign that the URL is a phishing URL. If the percentage of foreign hyperlinks is more than 50%, the characteristic is assigned 1 (phishing), otherwise it is assigned 0. (Legitimate).

LEXICAL-BASED FEATURES:

- Feature 9 (F9): Equal
Because "=" is utilized to obtain input from the end-user, most professional websites are no longer used due to the risk of data sniffing. It is preferable to avoid URLs with this complex and hazardous lexical character by assigning 1 (phishing) to URLs with "=" else 0 (Legitimate).
- Feature10 (F10): Hexadecimal
Because URL encoding substitutes unsafe ASCII characters with "%" followed by two hexadecimal numbers, it is best to avoid using "%" in URLs such that URLs containing "%" symbols have a value of 1 (phishing) otherwise 0. (Legitimate).
- Feature11 (F11): Digit
Domains with numerals are perplexing. Furthermore, professional websites do not use domain names that include digits, and many phishers utilize numbers to fool end-users, such as naming the domain app1e.com instead of apple.com. If the IP address is present in the URL, it is a malicious URL, and it will be allocated 1; otherwise, it is a genuine URL, and it will be assigned 0.

WHOIS-BASED FEATURES:

- Feature12 (F12): Host Age
The normal procedure for registering a domain is to register domain and then construct the website as rich as possible, which will take some time. The founder begins to declare and publicize the URL, or does not announce at all and instead relies on search engines such as Google to do so. On the contrary, most phishers register a domain and utilize it rapidly in order to evade discovery. As a result, it is preferable to avoid domains with an age of less than 90 days by assigning it 1 (phishing) else 0 (Legitimate).
- Feature13 (F13): Registrar Name
The registrar is the entity where the domain name is registered. According to the Internet Corporation for Assigned Names and Numbers (ICANN), there were over a thousand ICANN-accredited registrars globally by the middle of 2017, with the number steadily rising. Unless they hide it for any reason, we can determine who owns most domains. If you get a URL as an announcement with an unknown registrant name, it is most likely a phishing URL and will be assigned 1 (phishing), else 0 (no phishing) (Legitimate).

ENGINE-BASED FEATURES:

- Feature14 (F14): Web traffic
The web traffic, which can be collected from the Alexa database, is the total number of users who have visited a URL or webpage. Assume the website is among the top 500 thousand. In such instances, unless it is hacked to be used as a phishing website, it is difficult to be a phishing website. The likelihood of a phishing website growing as its popularity decreases. As a result, URLs with a rank more than 500 thousand are allocated 1 (phishing), while others are assigned 0. (Legitimate).

E. Classification Phase

In this phase, eleven classification algorithms; which were found to be the most adaptable in phishing websites detection, including Random Forest (RF), Gradient Boosting (GBoost), LightGBM (LGBM), Support Vector Machines (SVM), Logistic Regression (LR), k-nearest neighbors (KNN), Gaussian Naive Bayes(GaussianNB), CatBoost, Decision Tree (DT), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA), are used for classification activities and comparison purposes.

IV. Experimental and Evaluation

A. Evaluation Criteria

To evaluate the effectiveness of the proposed model, there are numerous assessment tools available. Calculating the accuracy rates will be accurate and efficient due to the used binary datasets. In order to assess how accurate our model is, we pay attention to its correctness.

Accuracy (A): It measures the overall percentage of predictions that come true as in Eq. (1).

$$Accuracy(A) = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Properly identified cases are denoted by the letters TP, correctly rejected instances by the letters TN, incorrectly identified instances by the letter FP, and wrongly rejected instances by the letters FN.

Moreover, security is paramount in the world of cybersecurity since phishing attempts can cause significant harm to end users. Therefore, the key goal is to protect users from phishing attacks and drastically reduce misclassification to avoid any challenges faced by the user when utilizing services. Consequently, we include Precision, Recall, and F1-Score in our evaluation.

Precision evaluates the proportion of occurrences properly identified as phishing compared to all instances identified as phishing as in Eq. (2).

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall: It measures the proportion of phishing incidents that are accurately identified compared to all phishing incidents as in Eq. (3).

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

F1-Score: It is a weighted average of Precision and Recall as in (4).

$$F1\_Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

B. Experiments and Results

Two stages of the experiments were carried out. The first step is to assess the classifiers and choose the algorithm that performs the best. The robustness and generalizability of the proposed models are evaluated in the second stage by training the model on recent datasets without retraining it. It also conducts additional experiments to investigate the impact of retraining on model performance by testing the model with a recent dataset after retraining. Each of these stages will be thoroughly illustrated in the following subsections.

1) Releasing the Best Classification Algorithm: These experiments aim to determine the classification process' best performing algorithm. Random Forest (RF), Gradient Boosting (GBoost), LightGBM (LGBM), Support Vector Machines (SVM), Logistic Regression (LR), k-nearest neighbors (KNN), Gaussian Naive Bayes(GaussianNB), CatBoost, Decision Tree, Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA) algorithms are trained and tested on the first dataset; 24,200 phishing and legitimate URLs, as shown in Table II. The dataset is randomly split into 90% and 10% of the samples for the training set and testing set, respectively. On the training set, a randomized cross-validation (10-fold) search was performed with a maximum of 1000 iterations. These algorithms' classification results after 10, 100, and 1000 iterations are shown in Table IV. All algorithms improved in performance as the number of iterations increased, as can be shown. The best performance for the LightGBM and Decision Tree algorithms was achieved after 100 iterations and remained constant after 1000 iterations. The results also show a performance competition among RF, LGBM, and GBoost classifiers. To the best of our knowledge, they outperform currently available state-of-the-art phishing detection systems designed and reach the best performance with nearly identical results. The highest accuracy of the classifiers is shown in Table V, and Fig. 3.

TABLE IV. Performance of the Classifiers using 10, 100, and 1000 Iterations

Algorithm	10 iterations	100 iterations	1000 iterations
RF	99.63	99.71	99.79
GBoost	99.42	99.67	99.84
LGBM	99.67	99.84	99.84
SVM	96.66	96.87	96.99
LR	96.28	96.82	97.07
KNN	99.42	99.42	99.59
GaussianNB	98.06	98.47	98.76
CatBoost	95.83	96.49	96.82
DT	96.16	96.61	96.61
LDA	95.87	96.32	96.94
QDA	83.80	84.88	85.95

TABLE V. The Best Performance of the Classifiers

Algorithm	Accuracy	Precision	Recall	F1-score
RF	99.79	1.00	1.00	1.00
GBoost	99.84	1.00	1.00	1.00
LGBM	99.84	1.00	0.99	1.00
SVM	96.99	0.94	0.95	0.94
LR	97.07	0.95	0.92	0.94
KNN	99.59	1.00	0.99	0.99
GaussianNB	98.76	0.97	0.90	0.98
CatBoost	96.82	0.95	0.93	0.94
DT	96.61	0.94	0.93	0.94
LDA	96.94	0.96	0.91	0.94
QDA	85.95	0.43	0.50	0.46

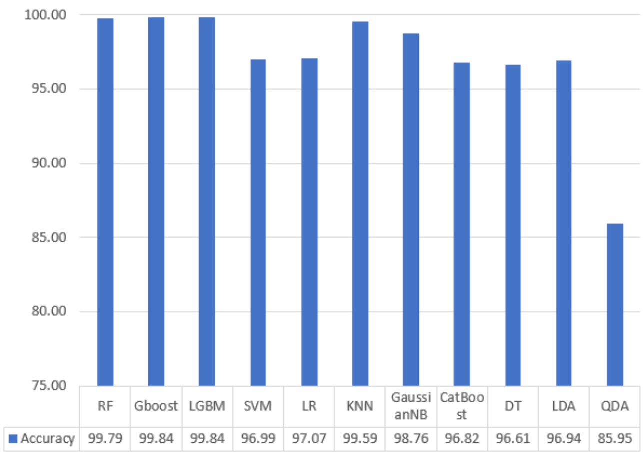


Fig. 3. The best classifier's accuracy.

2) Future Attacks: The goal of these experiments is to ensure that we have a robust model that can deal with ever-changing attacks. One method is to train the model on one dataset and then test it on a different, more recent one. As a result, two new datasets were used in these experiments. As shown in Tables II and III, we referred to them as the second and the third datasets.

These experiments were carried out in three steps:

1- Measuring model performance with new attacks.

2- Investigating the impact of retraining on model performance.

3- Emphasizing the significance of the model's regular retraining.

a) Measuring Model Performance with New Attacks: The first stage is to evaluate the model's performance with the new attacks, which is done by evaluating the suggested models with the second and third datasets, using all models demonstrated in the prior section. Table VI present the findings. According to the results, most of the classifiers' performance is slightly lower than in the previous experiment, with the exception of the CatBoost and DT algorithms, which both have slightly higher performance. Given that these datasets were not used in the training set, that each dataset is unique with no common URLs, and they were gathered over a three-year period, these findings could be considered good. Furthermore, the results show a decrease in performance for both the KNN and LightGBM algorithms. In terms of best performance, the second dataset indicated a competition amongst three classifiers: Gaussian NB, Random Forest, and Gradient Boost. Furthermore, the Gaussian NB classifier performs better with

TABLE VI. Performance of Classifiers on the Second, and the Third Datasets

Algorithm	On the second dataset	On the third dataset
RF	**98.5**	**98**
GBoost	**98.51**	**97.9**
LGBM	95.7	83.5
SVM	95.8	95.7
LR	95.98	96.3
KNN	93.795	91
GaussianNB	**98.5**	**98.67**
CatBoost	97	97.1
DT	97	97.1
LDA	96	95.5
QDA	83.4	83.5

TABLE VII. Performance of the Retrained Models

Algorithm	On the new testing dataset	On the third dataset
RF	**99.751**	**98.31**
GBoost	**99.727**	**98.3**
LGBM	**99.776**	95.57
SVM	96.943	96.13
LR	96.619	96.47
KNN	99.528	94.31
GaussianNB	**98.608**	**98.46**
CatBoost	97.042	97
DT	98.423	88.79
LDA	96.396	95.84
QDA	85.235	83.43

TABLE VIII. Performance of the Retrained Model on the New Testing Dataset

Algorithm	Accuracy
RF	**99.72**
GBoost	**99.72**
LGBM	**99.73**
SVM	96.98
LR	96.62
KNN	99.59
GaussianNB	98.38
CatBoost	96.85
DT	96.98
LDA	96.21
QDA	84.77

the third dataset. Random Forest and Gradient Boost are placed second and third in terms of performance, respectively.

The modelâ€™s efficacy and robustness are ensured by the features and classification technique used. This experiment highlights the value of the features chosen for the classification models, demonstrating that they were more resilient to new attacks and that active phishers are unable to overcome them. It also shows that the Gaussian NB algorithm's performance has held steady over time, implying that it is the most resilient to fresh phishing attack strategies. In a close race for second place, the Random Forest and Gradient Boost algorithms both performed well.

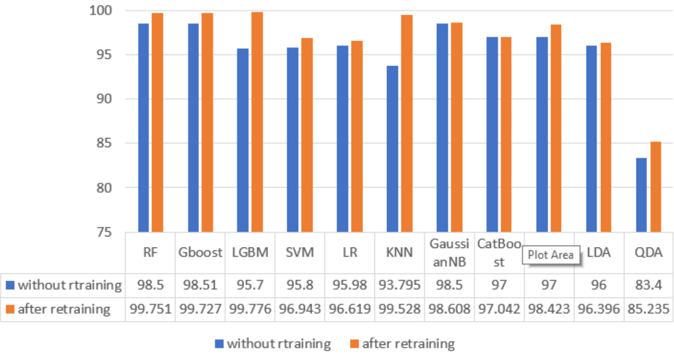
b) Investigating the Impact of Retraining on Model Performance: The second step in these experiments is to investigate the effect of retraining on the model performance, which is accomplished by testing the models again after they've been retrained with the new phishing attacks. The second dataset was divided into two parts: train and test. The model was then retrained with the new training set (90% of the second dataset), with the same set of features, and tested using the new testing set (the remaining 10% of the second dataset) and the third dataset. Table VII shows the classification results.

For the new testing set (the remaining 10% of the second dataset), all of the classifiers' performance increased and was very near to the first findings released from the initial model, emphasizing the need of retraining operations in order to preserve model performance over time. LightGBM was the most significantly improved algorithm, with its performance dropping from 99.84% accuracy to 95.68% when tested on the new dataset without retraining and returning to 99.78% accuracy when tested again but after retraining on fresh phishing attacks. For the third dataset, most classifiers' performance increased when compared to the results of the prior test with the same dataset without retraining, as shown in Table VI. LightGBM benefitted the most after retraining, increasing its accuracy from 83.5% to 95.57%, although it was not the best accuracy classifiers received. In terms of greatest accuracy, the GaussianNB Algorithm ranks first with around 98.46%, followed by both GradientBoost and Random Forest in second place with approximately 98.3%. Despite the modest decline in performance, the GaussianNB Algorithm maintained its performance from the start of the experiments. After retraining, the performance of the CatBoost, QDA, and DT algorithms all decreased.

This experiment emphasizes the need of retraining for algorithms like LightGBM, which lose performance while dealing with fresh datasets without retraining. Algorithms such as Random Forest and GradientBoost can deal with new attacks with a minor reduction in performance; this is regarded a satisfactory outcome if they do not regularly retrain with fresh datasets, but they are able to retain performance after retraining. The GaussianNB algorithm is the most immune to new phishing attacks without retraining and can even function effectively while being retrained.

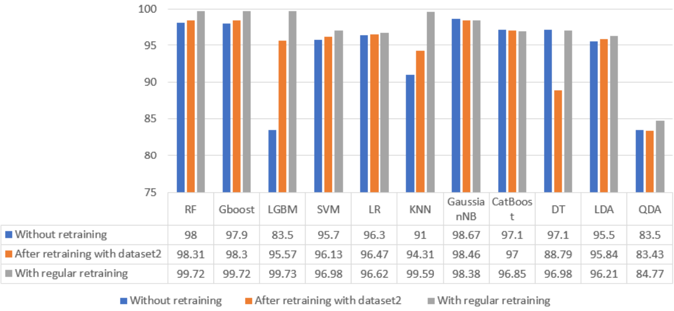
c) Emphasizing the Significance of the Model's Regular Retraining: The third phase emphasizes the importance of continual model retraining for more current attack types. As indicated in previous findings; Table VII, the classifiers' performance was as excellent as it was for the first model due to retraining the model using recent attacks from the same time period as the test set. To ensure this, the third dataset is separated into train and test sets. The model was then retrained with the new training set. The new testing set is then used to test it. The classification result is shown in Table VIII below. The performance of most classifiers has improved once again, like in the prior experiment. LightGBM classifier returns to top place in terms of accuracy, followed by Random Forest and GradientBoost at the second level. The third and fourth levels are occupied by KNN and GaussianNB, respectively.

When these results were compared to the previous test results for the third dataset without any retraining, it was discovered that all algorithms outperformed their previous performance with the exception of GaussianNB, CatBoost, and DecisionTree algorithms, whose performance had a slight slip after retraining. The same result was observed while testing the third dataset after retraining the model using a portion of the second dataset, except that DecisionTree method performance rose after retraining the model with recent attacks from the same time period. Fig. 4, 5 compare models performance with and without retraining.



	RF	Gboost	LGBM	SVM	LR	KNN	GaussianNB	CatBoost	Plot Area	LDA	QDA
without rtraining	98.5	98.51	95.7	95.8	95.98	93.795	98.5	97	97	96	83.4
after retraining	99.751	99.727	99.776	96.943	96.619	99.528	98.608	97.042	98.423	96.396	85.235

Fig. 4. The models performance on the second dataset with and without retraining.



	RF	Gboost	LGBM	SVM	LR	KNN	GaussiannNB	CatBoost	DT	LDA	QDA
Without retraining	98	97.9	83.5	95.7	96.3	91	98.67	97.1	97.1	95.5	83.5
After retraining with dataset2	98.31	98.3	95.57	96.13	96.47	94.31	98.46	97	88.79	95.84	83.43
With regular retraining	99.72	99.72	99.73	96.98	96.62	99.59	98.38	96.85	96.98	96.21	84.77

Fig. 5. The models performance on the third dataset with and without retraining.

These studies illustrate the need of having a diverse set of characteristics that can counter ever changing attacks. It also underlines the need of selecting a classification method that performs well with both previously trained and fresh datasets. Furthermore, to safeguard the model's performance, it should be retrained on a regular basis for newer sorts of attacks. Furthermore, these experiments demonstrated that the LightGBM, Random Forest, and GradientBoost algorithms performed the best and maintained their performance with regular retraining of the model with newer types of attacks. Furthermore, the GaussianNB method is the most resilient algorithm, capable of maintaining its performance even without retraining, and its performance is considered good in comparison to current best and common practices used in the field. These four proposed models demonstrate that it is tough for attackers to avoid, since it is capable of dealing with the ever-changing nature of phishing attacks. Furthermore, they outperform the other detection models currently available. To the best of our knowledge, these are the first detection models that have demonstrated their resilience and generalization by being assessed with fresh up-to-date datasets without and with retraining and indicating that they can sustain and continue to perform well.

V. Conclusion and Future Work

Based on the observation of historical and contemporary phishing attacks, this article developed a unique combination of phishing URL detecting features which offered novel detection models that used the proposed features in concert with a machine learning algorithm. Eleven alternative machine learning algorithms were examined for classification tasks and comparative objectives. For the initial model creation and the model evaluation, three datasets were created. These datasets were gathered over a three-year interval to guarantee the model's generalizability and resilience to new datasets and phishing attacks.

Through a variety of experiments, beginning with assessing the classifiers to pick the best-performing algorithm and ending with emphasizing the relevance of the model's frequent retraining, the model performance, generalization, and robustness were evaluated using appropriate evaluation metrics. Based on the experimental data, the key conclusion is that the suggested models; which utilize LightGBM, Random Forest, or GradientBoost algorithms, have the best performance with an average accuracy rate of 99.7%, outperforming all other model in the literature. Furthermore, when evaluated with newer datasets, Random Forest, and GradientBoost models comes in the second level after the GaussianNB model which is the most durable without retraining. Additionally, it demonstrated that these models are able to maintain its performance with regular retraining with newer types of attacks and with the same set of features, which is regarded an extraordinary achievement and a step forward in phishing detection technologies. Adapting various parallel processing approaches to lower the time necessary to extract the features is one potential future attempt. Furthermore, we intend to employ deep learning algorithms in a performance evaluation. Moreover, we plan to expand our work on social media platforms such as Facebook, Instagram, and others.

References

[1] Greg Aaron, Lyman Chapin, David Piscitello, and Dr.Colin Strutt. Phishing landscape 2020, a study of the scope and distribution of phishing, 13 October 2020.

[2] Greg Aaron, Lyman Chapin, David Piscitello, and Dr.Colin Strutt. Phishing landscape 2022, a study of the scope and distribution of phishing, 19 July 2022.

[3] Greg Aaron, iThreat Cyber Group, and Rod Rasmussen. Global phishing survey:trends and domain name use in 2016.

[4] Kibreab Adane and Berhanu Beyene. Machine learning and deep learning based phishing websites detection: The current gaps and next directions. *Review of Computer Engineering Research*, 9(1):13–29, 2022.

[5] SK Hasane Ahammad, Sunil D Kale, Gopal D Upadhye, Sandeep Dwarkanath Pande, E Venkatesh Babu, Amol V Dhumane, and Mr Dilip Kumar Jang Bahadur. Phishing url detection using machine learning methods. *Advances in Engineering Software*, 173:103288, 2022.

[6] Ammar Almomani, Mohammad Alauthman, Mohd Taib Shatnawi, Mohammed Alweshah, Ayat Alrosan, Waleed Alomoush, and Brij B Gupta. Phishing website detection with semantic features based on machine learning classifiers: A comparative study. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 18(1):1–24, 2022.

[7] Mohammad Almseidin, AlMaha Abu Zuraiq, Mouhammd Al-Kasassbeh, and Nidal Alnidami. Phishing detection based on machine learning and feature selection methods. 2019.

[8] APWG. Phishing activity trends report (1st quarter 2022), 7 June 2022.

[9] APWG. Phishing activity trends report (4th quarter 2020), 9 February 2021.

[10] Abdul Basit, Maham Zafar, Abdul Rehman Javed, and Zunera Jalil. A novel ensemble machine learning method to detect phishing attack. In *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pages 1–5. IEEE, 2020.

[11] Anuja Bhosale, Gayatri Gadas, Muskan Chavan, and Seema Hadke. Detection of phishing websites using machine learning. *International Journal of Advanced Research in Computer and Communication Engineering*, 6:490–494, 2022.

[12] Pankaj Bhowmik and Pulak Chandra Bhowmik. A machine learning approach for phishing websites prediction with novel feature selection framework. In *Proceedings of International Conference on Fourth Industrial Revolution and Beyond 2021*, pages 357–370. Springer, 2022.

[13] Pankaj Bhowmik, Md Sohrawordi, UA Ali, Pulak Chandra Bhowmik, et al. An empirical feature selection approach for phishing websites prediction with machine learning. In *International Conference on Bangabandhu and Digital Bangladesh*, pages 173–188. Springer, 2022.

[14] Mr Swapnil S Chaudhari, Satish N Gujar, and Farhat Jummani. Detection of phishing web as an attack: A comprehensive analysis of machine learning algorithms on phishing dataset. *Journal of Engineering (IOSRJEN)*, 2022.

[15] Avisha Das, Shahryar Baki, Ayman El Aassal, Rakesh Verma, and Arthur Dunbar. Sok: a comprehensive reexamination of phishing research from the security perspective. *IEEE Communications Surveys & Tutorials*, 22(1):671–708, 2019.

[16] Sumitra Das Guptta, Khandaker Tayef Shahriar, Hamed Alqahtani, Dheyaaldin Alsalman, and Iqbal H Sarker. Modeling hybrid feature-based phishing websites detection using machine learning techniques. *Annals of Data Science*, pages 1–26, 2022.

[17] Zuochao Dou, Issa Khalil, Abdallah Khreishah, Ala Al-Fuqaha, and Mohsen Guizani. Systematization of knowledge (sok): A systematic review of software-based web phishing detection. *IEEE Communications Surveys & Tutorials*, 19(4):2797–2819, 2017.

[18] Ayman El Aassal, Shahryar Baki, Avisha Das, and Rakesh M. Verma. An in-depth benchmarking and evaluation of phishing detection research for security needs. *IEEE Access*, 8:22170–22192, 2020.

[19] Brij B Gupta, Krishna Yadav, Imran Razzak, Konstantinos Psannis, Arcangelo Castiglione, and Xiaojun Chang. A novel approach for phishing urls detection using lexical based machine learning in a real-time environment. *Computer Communications*, 175:47–57, 2021.

[20] Tzipora Halevi, Nasir D. Memon, and Oded Nov. Spear-phishing in the wild: A real-world study of personality, phishing self-efficacy and vulnerability to spear-phishing attacks. *Innovation Law & Policy eJournal*, 2015.

[21] Sohrab Hossain, Dhiman Sarma, and Rana Joyti Chakma. Machine learning-based phishing attack detection. *International Journal of Advanced Computer Science and Applications*, 11(9), 2020.

[22] Ankit Kumar Jain and Brij B Gupta. A machine learning based approach for phishing detection using hyperlinks information. *Journal of Ambient Intelligence and Humanized Computing*, 10(5):2015–2028, 2019.

[23] SM Mahamudul Hasan, Nirjas Mohammad Jakilim, Forhad Rabbi, Rumel Rahman Pir, et al. Determining the most effective machine learning techniques for detecting phishing websites. In *Applications of Artificial Intelligence and Machine Learning*, pages 593–603. Springer, 2022.

[24] Sourena Maroofi, Maciej Korczyński, Cristian Hesselman, Benoit Ampeau, and Andrzej Duda. Comar: Classification of compromised versus maliciously registered domains. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 607–623. IEEE, 2020.

[25] Michael Miller. *Is It Safe? ProtectIng Your Computer, Your BusIness, And Yourself OnlIne*. Que Publishing; 1st edition (June 6, 2008), 1 January 2000.

[26] Jimmy Moedjahedy, Arief Setyanto, Fawaz Khaled Alarfaj, and Mohammed Alreshoodi. Ccrfs: Combine correlation features selection for detecting phishing websites using machine learning. *Future Internet*, 14(8):229, 2022.

[27] Hajara Musa, Bala Modi, Ismail Abdulkarim Adamu, Ali Ahmad Aminu, Hussaini Adamu, and Yahaya Ajiya. Acomparative analysis of different feature set on the performance of different algorithms in phishing website detection. *International Journal of Artificial Intelligence and Applications (IJAIA)*, 10(3), 2019.

[28] Adam Oest, Penghui Zhang, Brad Wardman, Eric Nunes, Jakub Burgis, Ali Zand, Kurt Thomas, Adam Doupé, and Gail-Joon Ahn. Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale. In *Proceedings of the 29th USENIX Security Symposium*, Proceedings of the 29th USENIX Security Symposium, pages 361–377. USENIX Association, 2020.

[29] Manuel Sánchez-Paniagua, Eduardo Fidalgo, Víctor González-Castro, and Enrique Alegre. Impact of current phishing strategies in machine learning models for phishing detection. In *Computational Intelligence in Security for Information Systems Conference*, pages 87–96. Springer, 2019.

[30] Shafaizal Shabudin, Nor Samsiah Sani, Khairul Akram Zainal Ariffin, and Mohd Aliff. Feature selection for phishing website classification. *International Journal of Advanced Computer Science and Applications*, 11(4), 2020.

[31] Anjum N. Shaikh, Antesar M. Shabut, and M.A. Hossain. A literature review on phishing crime, prevention review and investigation of gaps. In *2016 10th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 9–15, 2016.

[32] A Suryan, C Kumar, M Mehta, R Juneja, and A Sinha. Learning model for phishing website detection. *EAI Endorsed Transactions on Scalable Information Systems*, 7(27):e6–e6, 2020.

[33] Melanie Volkamer, Karen Renaud, Benjamin Reinheimer, and Alexandra Kunz. User experiences of torpedo: Tooltip-powered phishing email detection. *Computers and Security*, 71:100–113, 2017.

[34] Nur Sholihah Zaini, Deris Stiawan, Mohd Faizal Ab Razak, Ahmad Firdaus, Wan Isni Sofiah Wan Din, Shahreen Kasim, and Tole Sutikno. Phishing detection system using machine learning classifiers. *Indonesian Journal of Electrical Engineering and Computer Science*, 17(3), 2020.