frontiers | Frontiers in Computer Science

TYPE Original Research
PUBLISHED 02 July 2024
DOI 10.3389/fcomp.2024.1428013

OPEN ACCESS

EDITED BY
Saqib Saeed,
Imam Abdulrahman Bin Faisal University,
Saudi Arabia

REVIEWED BY
Alex C. Ng,
La Trobe University, Australia
Elochukwu Ukwandu,
Cardiff Metropolitan University,
United Kingdom

*CORRESPONDENCE
Maruf A. Tamal
✉ tamal25-103@diu.edu.bd

RECEIVED 05 May 2024
ACCEPTED 10 June 2024
PUBLISHED 02 July 2024

CITATION
Tamal MA, Islam MK, Bhuiyan T, Sattar A and Prince NU (2024) Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning.
Front. Comput. Sci. 6:1428013.
doi: 10.3389/fcomp.2024.1428013

Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning

Maruf A. Tamal[1]*, Md K. Islam[2], Touhid Bhuiyan[1], Abdus Sattar[1] and Nayem Uddin Prince[3]

[1]Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh, [2]Faculty of Graduate Studies, Daffodil International University, Dhaka, Bangladesh, [3]School of Information Technology, Washington University of Science and Technology, Alexandria, VA, United States

Introduction: The dynamic and sophisticated nature of phishing attacks, coupled with the relatively weak anti-phishing tools, has made phishing detection a pressing challenge. In light of this, new gaps have emerged in phishing detection, including the challenges and pitfalls of existing phishing detection techniques. To bridge these gaps, this study aims to develop a more robust, effective, sophisticated, and reliable solution for phishing detection through the optimal feature vectorization algorithm (OFVA) and supervised machine learning (SML) classifiers.

Methods: Initially, the OFVA was utilized to extract the 41 optimal intra-URL features from a novel large dataset comprising 2,74,446 raw URLs (134,500 phishing and 139,946 legitimate URLs). Subsequently, data cleansing, curation, and dimensionality reduction were performed to remove outliers, handle missing values, and exclude less predictive features. To identify the optimal model, the study evaluated and compared 15 SML algorithms arising from different machine learning (ML) families, including Bayesian, nearest-neighbors, decision trees, neural networks, quadratic discriminant analysis, logistic regression, bagging, boosting, random forests, and ensembles. The evaluation was performed based on various metrics such as confusion matrix, accuracy, precision, recall, F-1 score, ROC curve, and precision-recall curve analysis. Furthermore, hyperparameter tuning (using Grid-search) and k-fold cross-validation were performed to optimize the detection accuracy.

Results and discussion: The findings indicate that random forests (RF) outperformed the other classifiers, achieving a greater accuracy rate of 97.52%, followed by 97.50% precision, and an AUC value of 97%. Finally, a more robust and lightweight anti-phishing model was introduced, which can serve as an effective tool for security experts, practitioners, and policymakers to combat phishing attacks.

KEYWORDS

cybersecurity, phishing attack, optimal feature vectorization algorithm, URL-based phishing detection, anti-phishing model, supervised machine learning

1 Introduction

In recent years, the world has experienced a profound technological revolution, resulting in greater Internet accessibility than ever before. As of January 2023, there have been 5.16 billion Internet users globally, comprising 64.4 percent of the world's population (Petrosyan, 2023). This exponential surge in Internet use has brought about significant transformations in traditional systems and people's daily lives (Hoehe and Thibaut, 2020;

Maqsood et al., 2023). Consequently, data have become the lifeblood of individuals, organizations, and complex processes.

In this data-driven landscape, prioritizing data security must be paramount. The emphasis on safeguarding information needs to be ingrained into every facet of data collection, analysis, and utilization, from the very conception of any project or initiative. Numerous studies already underscored this criticality. For instance, Li et al. (2021) warned of the security and privacy risks inherent in storing and processing large volumes of energy data in cloud environments. Kasim (2022), concerned with the sensitivity of electronic medical records (EMRs), proposed a robust ensemble architecture to ensure data privacy and security. Similarly, Deepika et al. (2021) and Yuan et al. (2021) focused on data security in cloud-based systems and medical image diagnosis, respectively, both advocating for novel approaches to thwart security threats. This emphasis on data security is not a mere theoretical concern. However, suspicious online activities are evolving and escalating, leading to a surge in both cyber-dependent and cyber-enabled crimes. Cybercrime also poses a substantial threat to the global economy, national security, social stability, and individual interests (Chen et al., 2023). According to the 2020 Official Annual Cybercrime Report, cybercrime is one of the greatest challenges that humanity will face in the next two decades (The 2020 Official Annual Cybercrime Report, 2020). These concerns are compounded by the escalating costs induced by cybercrime, which surged from around \$3 trillion in 2015 to over \$6 trillion in 2021, with projections indicating an increase to over \$10.5 trillion by 2025 (Morgan, 2020).

Amid this digital landscape, cybercriminals employ various tactics (Phillips et al., 2022) to hook their targets, and among them, phishing stands as the most common but dynamic and threatening strategy. Phishing has been defined in various ways by experts, researchers, and cybersecurity institutions due to its continuous evolution and contextual variation. Consequently, there is no universally accepted or rigid definition for the term "phishing" (Alkhalil et al., 2021). However, as proposed by Alabdan (2020), the term "phishing" has originated from the term "fishing," where attackers use bait to lure victims and illicitly access their information or trick them into downloading malware. In this phenomenon, rather than technical or coding-based approaches, attackers utilize human weaknesses and psychological manipulation; hence, the term "human hacking" is often used to refer to phishing (Klimburg-Witjes and Wentland, 2021). In essence, phishing can be defined as a cyber-enabled crime employing both social engineering and technical subterfuge to trick individuals into disclosing confidential information (e.g., credit card numbers, login credentials, or personal identification details) by posing as a trustworthy source.

Every phishing attack passes through a series of phases (see Figure 1), or more specifically, three phases: lure, hook, and catch (US-CER, 2016). In the first phase (lure), the attacker gathers sufficient information about the target (e.g., an individual or organization) and decides the attack technique to be utilized to trap the victim. Subsequently, the attacker sends legitimate-looking e-mails, messages, SMS, and QR codes with a phishing website link to the intended person or organization, containing enticing offers or creating a sense of urgency (Tang and Mahmoud, 2021). In the second phase (hook), the victim unintentionally opens the email, clicks on the link, or downloads and installs malware. In the third phase (catch), the victim loses control of their system, leading to various malicious activities, such as password changes, fraudulent transactions, missing sensitive information, or further fraud. The attacker then erases all evidence (Alkhalil et al., 2021).

Phishing is an old yet effective cybercrime due to its dynamic and ever-evolving nature (Ribeiro et al., 2024). Unlike other forms of cybercrime, where the attacker's motives are known and victim types have been consistent, phishers have varying goals, motivations, and victim types. Although there are plenty of preventive and detective strategies for combatting phishing attacks, none of them perform as a "bullet of silver" against phishing (Gupta et al., 2016). To bypass anti-phishing tools, phishers frequently change their attacking tactics and look for new and creative ways. Consequently, phishing has become one of the most organized and hard-to-detect cybercrimes of the twenty-first century (Vayansky and Kumar, 2018). This has led to a substantial increase in phishing attacks in recent years, as illustrated in Figure 2 (APWG Phishing Activity Trends Reports, 2022). As reported by the Anti-Phishing Working Group [Anti-Phishing Working Group (APWG), 2022], 1,270,883 unique phishing attacks took place in the third quarter of 2022, which was the worst the APWG had ever recorded. This rising tendency highlights the shortcomings of current anti-phishing methods, revealing that existing countermeasures are insufficient in detecting and preventing these attacks (Gupta et al., 2016; Vayansky and Kumar, 2018; Alkhalil et al., 2021; Alnemari and Alshammari, 2023; Zieni et al., 2023).

In contrast to the ever-evolving tactics of phishers, most existing phishing detection methods remain static and rigid, relying on predefined patterns and rules. This leads to high false-positive rates and often renders them ineffective against modern, tailored social engineering attacks. This discrepancy between the dynamic nature of phishing attacks and the static nature of current detection techniques underscores the urgent need for innovative approaches to combat phishing. To this end, this study aims to develop an efficient large data-driven model for detection of phishing through the optimal feature vectorization algorithm (OFVA) and supervised machine learning (SML) algorithms. Here, the key objective is to ensure a safe cyberspace for individuals as well as organizations by strengthening resilience against the growing number of phishing attacks. The overall contributions of this study can be summarized as follows:

- This study focuses specifically on the intra-URL features that possess the capability to differentiate between phishing and legitimate sites, excluding other content-based aspects such as text, message, DOM, and CSS logo.
- The study employs the optimal feature vectorization algorithm (OFVA) to extract optimal intra-URL features, 10 of which are entirely novel and previously unexplored.
- To ensure high phishing detection accuracy, this study utilizes a large dataset ($N = 2{,}74{,}446$) and 15 supervised machine learning (SML) algorithms derived from various machine learning types.
- Finally, this study introduces a lightweight anti-phishing model that aims to effectively detect phishing attacks with low computational overhead.

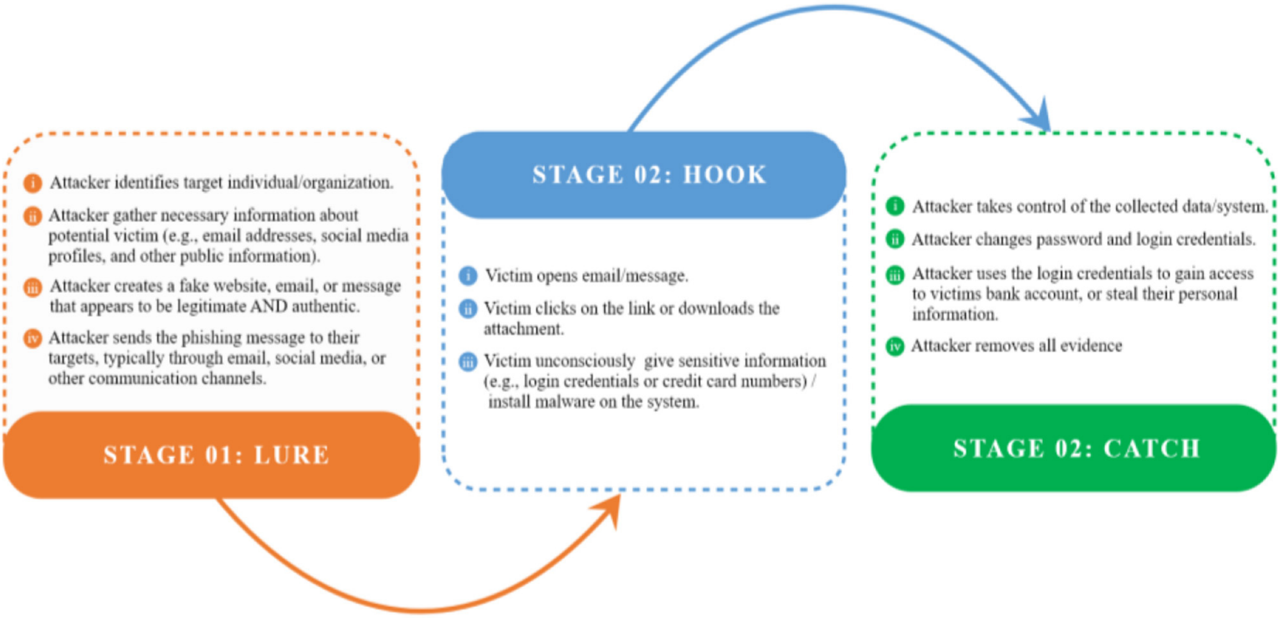

ⓘ Attacker identifies target individual/organization.
ⓘⓘ Attacker gather necessary information about potential victim (e.g., email addresses, social media profiles, and other public information).
ⓘⓘⓘ Attacker creates a fake website, email, or message that appears to be legitimate AND authentic.
ⓘⓥ Attacker sends the phishing message to their targets, typically through email, social media, or other communication channels.
STAGE 01: LURE
STAGE 02: HOOK
i Victim opens email/message.
ii Victim clicks on the link or downloads the attachment.
iii Victim unconsciously give sensitive information (e.g., login credentials or credit card numbers) / install malware on the system.
i Attacker takes control of the collected data/system.
ii Attacker changes password and login credentials.
iii Attacker uses the login credentials to gain access to victims bank account, or steal their personal information.
iv Attacker removes all evidence
STAGE 02: CATCH

FIGURE 1
Phases of a phishing attack.

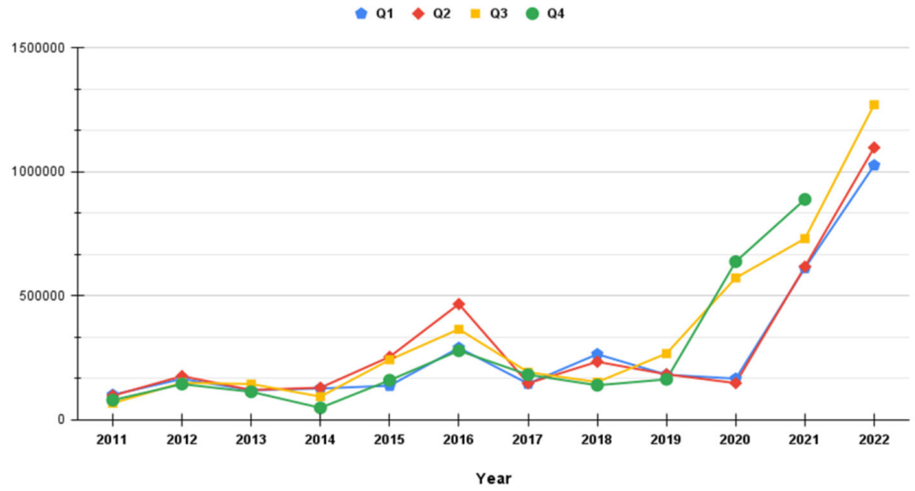


FIGURE 2
Number of unique phishing attacks (2011–2022).

2 Literature review

To combat phishing attacks, two types of approaches are typically followed: (1) preventive approach and (2) detective approach. The preventive approach relies on campaigns, awareness training, anti-phishing training, phishing simulation, seminars, press releases, and notices to increase user awareness against phishing attacks. Previous studies have found that these approaches are effective in enhancing both organizational and individual security awareness and knowledge (Dodge and Ferguson, 2006; Jensen et al., 2017; Daengsi et al., 2021; Quinkert et al., 2021; Yeoh et al., 2021; Alahmari et al., 2022). For example, Quinkert et al. conducted an empirical investigation in a real-world workplace environment and found that the implementation of awareness training initiatives resulted in a significant reduction in click rates on phishing sites (from 19% to 10%). The authors also found that some psychological vectors, such as an authoritative tone and curiosity, were more effective than others at tricking users into falling for phishing scams (Quinkert et al., 2021). Similarly, Daengsi et al. investigated the effects of age and gender on cybersecurity awareness using phishing simulations on 20,134 Thai employees working in a large financial service organization. They found that before the awareness program, 23.4% of employees opened malicious emails and 22.1% clicked on phishing links. However, after the program, there was a substantial decrease of 71.5% in the number of employees who opened phishing emails (Daengsi et al., 2021). In another study, Jensen et al. investigated the effectiveness of mindfulness training on students, faculty, and staff at a U.S. university in preventing phishing attacks (Jensen et al., 2017). The findings revealed that the training improved the ability of participants to dynamically yield attention during message evaluation, enhance contextual cues, and prevent suspicious messages by early judgment, and eventually helped avoid phishing attacks. Another training approach, based on the Transitive Memory System (TMS) theory, was proposed, where the awareness training was implemented using a bespoke app that deployed a game to deliver security training and encourage sharing (Alahmari et al., 2022). The findings of the study suggested that this app-based training approach could be an effective way to improve organizational security knowledge sharing (SKS). To explain the success of preventive approaches, Yeoh et al. emphasized the importance of phishing awareness as a continuous learning process that can strengthen individuals' behavior and equip them with the tools to effectively combat phishing attacks (Yeoh et al., 2021).

While phishing preventive approaches focus on educating individuals about the risks and tactics of phishing attacks, detective approaches mainly encompass technical measures, such as the list-based approach, rule-based approach, similarity-based approach, and ML-based approach. The list-based approach uses a list of known phishing websites to identify and block suspicious URLs. This approach involves the implementation of three distinct techniques: whitelist-based, blacklist-based, or a combination of both (Prakash et al., 2010; Li et al., 2014; Jain and Gupta, 2016; Rao and Pais, 2017; Azeez et al., 2021). In all three cases, the detection of phishing sites relies on the comparison of predefined databases containing approved and unapproved URLs, domains, IP addresses, etc. Several studies have demonstrated the effectiveness of these techniques, such as their speed and ease of use (Ludl et al., 2007; Prakash et al., 2010; Li et al., 2014; Azeez et al., 2021). However, a majority of current studies contended that these techniques might have struggled to detect unlisted phishing sites, commonly referred to as zero-hour or zero-day attacks (Sonowal and Kuppusamy, 2018; Aljofey et al., 2022; Sanchez-Paniagua et al., 2022).

The rule-based approach is another widely used method for detecting phishing attacks. It involves deriving predefined rules from known characteristics (e.g., poor grammar, spelling mistakes, and suspicious domain) of phishing websites, URLs, SMS, emails, or contents (Jain and Gupta, 2018; SatheeshKumar et al., 2022). For instance, one study (Moghimi and Varjani, 2016) proposed a rule-based approach that extracted two feature sets from phishing and legitimate websites: content-based features and document object model (DOM)-based features. The proposed model achieved a high true positive rate of 99.14% with a minimal false negative alarm rate of 0.86% in detecting phishing attacks. In a separate study, Mohammad et al. extracted 17 distinct features capable of distinguishing phishing websites from legitimate ones, which eventually can detect phishing attacks with a 4.75% error rate (Mohammad et al., 2014). Similarly, Adewole et al. (2019) proposed a hybrid rule-based model where they generated 55 rules based on 30 features of phishing websites. The findings showed an average accuracy of 96.8% in detecting suspected phishing sites (Adewole et al., 2019). While rule-based approaches were effective at identifying known phishing attack patterns, they proved less effective at detecting new or evolving phishing attacks (Vayansky and Kumar, 2018; Suleman, 2021). This is because they rely on predefined rules, which cannot be easily updated to keep up with the latest phishing tactics. Consequently, rule-based systems are difficult and time-consuming to maintain and update, especially as the number and sophistication of phishing attacks increase.

To detect phishing, scholars frequently use the visual similarity-based approach that compares the visual appearance of a suspected phishing website to the visual appearance of a legitimate website (Zieni et al., 2023). This approach considers features such as font styles, images, screenshots, page layout, logos, text content, HTML tags, text format, Cascading Style Sheets (CSS), images, and DOM to differentiate between legitimate and phishing websites (Jain and Gupta, 2017; Sattari and Montazer, 2023). For example, Ardi and Heidemann (2016) developed an efficient browser plugin named AuntieTuna that utilized cryptographic hashing of each web page's rendered DOM with more than 50% accuracy and zero false positives (FP). However, this approach failed when an attacker created different DOMs for the same website or when the website consisted solely of images. In another study, Chiew et al. (2015) proposed a logo-based phishing detection technique that extracted the logo image from a website and compared it with a database of legitimate logos to identify phishing websites. Although this approach successfully achieved a 93.4% accuracy rate in distinguishing phishing sites from legitimate ones, it presents a challenge due to its reliance on a vast database of genuine logos. Consequently, it becomes difficult to regularly update the database to include new logos and remove those that are no longer in use, hampering its practicality. Likewise, another study introduced VisualPhishNet, a visual similarity-based phishing detection technique employing a triplet convolutional neural network (CNN) to learn website profiles and identify phishing websites based on a similarity metric (Abdelnabi et al., 2020). While VisualPhishNet boasts a remarkable accuracy of 99.7%, it is crucial to consider that its training data were small (only 155 websites with 9363 screenshots). Hence, its effectiveness against real-world phishing websites remains an open question. In contrast, Khan et al. proposed SpoofCatch, a client-side solution that leverages the overall visual similarity of web pages to detect phishing attempts and provided impressive results (96% success rate) and minimal user experience impact (Khan et al., 2021). However, visual similarity-based detection has drawbacks. Recent studies (Aljofey et al., 2022; Zieni et al., 2023) highlighted potential limitations, including computational burdens, implementation hurdles, time-consuming nature of the analysis, and the challenge of identifying novel phishing attempts (potentially leading to high false-negative rates).

Among all the approaches employed to detect phishing, machine learning (ML)-based approaches have been extensively utilized by scholars and security experts globally (Mewada and Dewang, 2022; Safi and Singh, 2023). Considering phishing detection as a binary classification problem, both supervised ML algorithms (Nagaraj et al., 2018; Rao and Pais, 2018; Sahingoz et al., 2019; Zamir et al., 2020; Balogun et al., 2021; Kasim, 2021) and deep learning algorithms (Adebowale et al., 2020; Singh et al., 2020; Anitha and Kalaiarasu, 2022; Saeed, 2022; Aldakheel et al., 2023; Dhanavanthini and Chakkravarthy, 2023) have been used to distinguish phishing sites from legitimate ones. However, ML-based phishing detection approaches also face many challenges. For example, one of the primary challenges faced by ML models is the necessity for comprehensive training on large, high-quality datasets that encompass a diverse array of phishing attacks. This training data must accurately mirror the real-world scenarios, allowing ML models to generalize effectively to new phishing threats. Furthermore, the efficacy of these models is contingent on the quality of features extracted from phishing sites. Insufficient or biased data, coupled with less predictive features, can lead to subpar performance. Therefore, the volume of data, identification of high-quality features, proper data preprocessing, and precise hyperparameter tuning play pivotal roles in ensuring the efficacy of ML models against the constantly evolving landscape of phishing threats (Salihovic et al., 2018; Luca et al., 2022).

While various preventive and detective approaches exist against phishing attacks, the majority often struggle with limitations such as small datasets, static nature, difficulty in detecting zero-hour attacks, and high rates of false positives and false negatives. To address these challenges, this study introduces a novel phishing detection solution by harnessing the power of the optimal feature vectorization algorithm (OFVA) in conjunction with supervised machine learning (SML) algorithms. Unlike static methods (e.g., list-based and rule-based), our approach thrives on adaptability and scalability, enabling it to keep pace with the ever-evolving tactics employed by phishers. By continuously learning from an extensive dataset and leveraging machine learning techniques, our approach aims to address the shortcomings of traditional approaches and achieve a higher level of robustness and accuracy in distinguishing phishing sites from legitimate ones.

3 Materials and methods

The methodological flow of the proposed approach is depicted in Figure 3, which contains the following four phases: (a) dataset acquisition, (b) data preprocessing (c) model selection and evaluation, and (d) model deployment.

3.1 Dataset acquisition

In dataset acquisition, raw unstructured phishing and legitimate URLs were acquired and merged from different reliable and valid sources. As data volume and quality are always crucial for machine learning-based approaches (Wu et al., 2021), this study utilized a large volume of data to address data insufficiency, bias or class imbalance that could lead to poor or inaccurate approximation. Among the 274,446 URLs (before undergoing preprocessing), 48,009 legitimate URLs and 48,009 phishing URLs were obtained from Aalto University's research data (Marchal, 2014), while 86,491 phishing URLs were collected from OpenPhish (OpenPhish-Phishing Intelligence, 2023) and 91,937 legitimate URLs collected from DomCop (Download List of top 10 Million Domains Based on Open Data from Common Crawl and Common Search, 2023). These URLs were in their original form (e.g., https://www.facebook.com/), lacking any specific structure or organization where analysis can be performed.

3.2 Feature generation

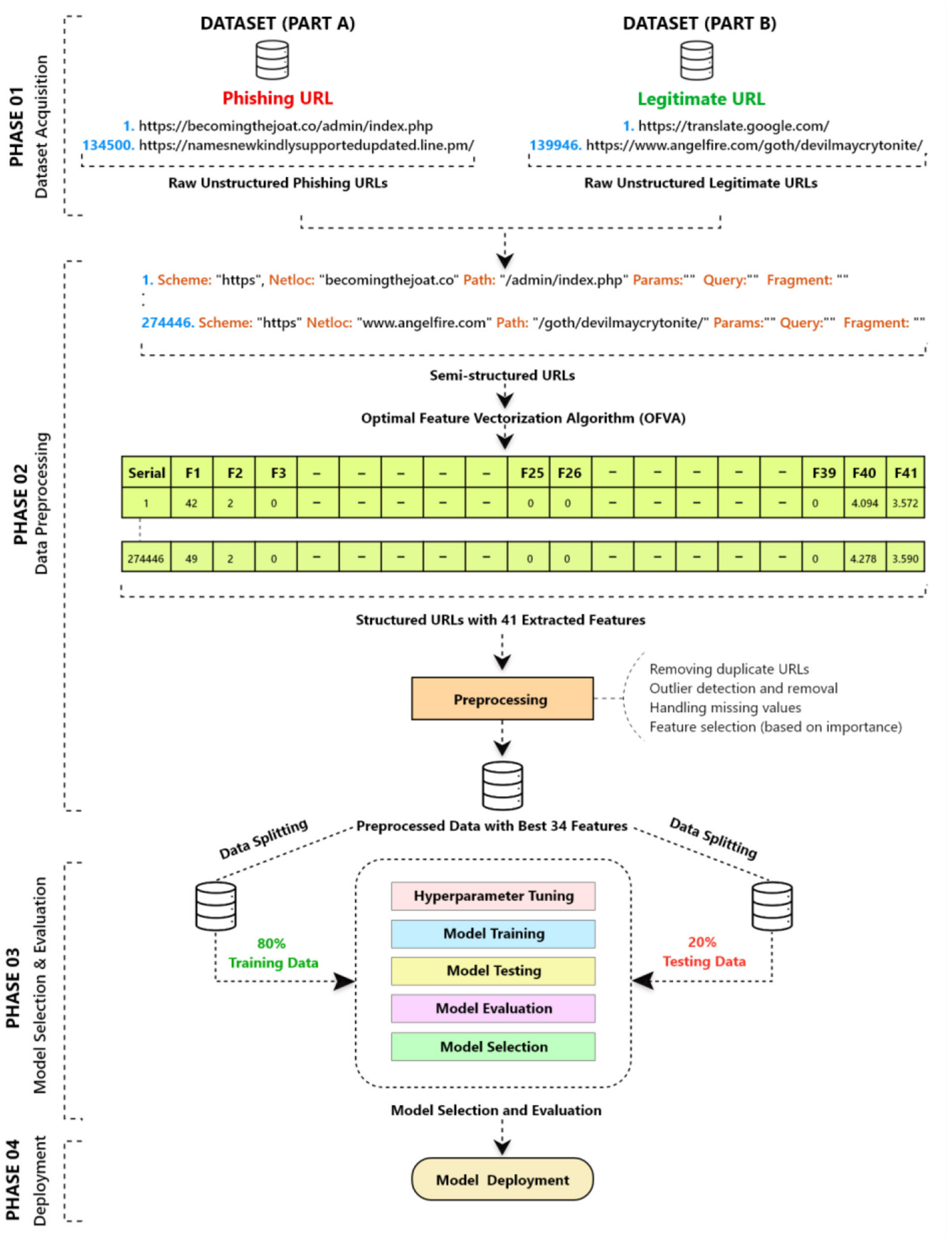
In the second phase, unstructured raw URLs (strings) were initially transformed into semi-structured components (scheme, network location, path, etc.) using the "urllib.parse" python module (Urllib.parse- Parse URLs Into Components Python Documentation, 2023). Subsequently, a list of 41 features was extracted to generate a particular feature vector ($x = F_1, F_2, F_3, \ldots\ldots\ldots\ldots F_{41}$) for each of the URLs to create a labeled dataset using a self-developed OFVA (see Appendix A). The key purpose of the OFVA was to extract the optimal intra-URL features from a given semi-unstructured URL list (see Phase 2 of Figure 3). Table 1 depicts the extracted feature list with a detailed explanation. In particular, features $F_1 - F_2$, $F_4 – F_{21}$, $F_{25} – F_{26}$, $F_{30} - F_{33}$, $F_{35} – F_{39}$ were considered, as suggested in previous studies (Jeeva and Rajsingh, 2016; Singh, 2020; Vrbančič et al., 2020; Mourtaji et al., 2021); however, we have modified and adjusted them to get better outputs. In contrast, features F_3, $F_{22} – F_{24}$, $F_{27} - F_{29}$, F_{34}, $F_{40} - F_{41}$ are completely novel and are proposed in this study based on the observation of phishing and legitimate URLs.

3.3 Data cleansing and curation

After feature generation, data cleansing and curation were performed to enhance the predictive performance of the SML classifiers. As data were obtained from multiple sources, there was a possibility of having duplicate URLs. Hence, in order to achieve optimal data quality, the data cleansing phase involved the removal of a total of 9,725 duplicate URLs. Moreover, to ensure the disproportionate effect of outliers on SML classifiers, rigorous outlier detection and removal were undertaken using the interquartile range (IQR) and box plot method, specifically targeting URL length (see Figure 4). Through this process, data points that were deemed to be outliers were systematically identified and subsequently removed from the dataset (n = 16,771). Hence, the final dataset which is uploaded in Mendeley Data (Tamal, 2023) comprised 2,47,950 records (phishing URLs = 119409; legitimate URLs = 128541).

3.4 Dimensionality reduction

To enhance the efficiency and accuracy of the SML classifiers and reduce model complexity, this study employed dimensionality reduction techniques to identify the most informative features. Specifically, the random forest algorithm (Breiman, 2001) was utilized to quantify the importance of each feature concerning the overall model's performance. The importance score of each feature was determined by assessing the extent of impurity reduction that resulted from splitting the data based on that feature. Figure 5


PHASE 01
Dataset Acquisition
DATASET (PART A)
Phishing URL
1. https://becomingthejoat.co/admin/index.php
134500. https://namesnewkindlysupportedupdated.line.pm/
Raw Unstructured Phishing URLs
DATASET (PART B)
Legitimate URL
1. https://translate.google.com/
139946. https://www.angelfire.com/goth/devilmaycrytonite/
Raw Unstructured Legitimate URLs
PHASE 02
Data Preprocessing
1. Scheme: "https", Netloc: "becomingthejoat.co" Path: "/admin/index.php" Params:"" Query:"" Fragment: ""
274446. Scheme: "https" Netloc: "www.angelfire.com" Path: "/goth/devilmaycrytonite/" Params:"" Query:"" Fragment: ""
Semi-structured URLs
Optimal Feature Vectorization Algorithm (OFVA)
Structured URLs with 41 Extracted Features
Preprocessing
Removing duplicate URLs
Outlier detection and removal
Handling missing values
Feature selection (based on importance)
Preprocessed Data with Best 34 Features
Data Splitting
Data Splitting
PHASE 03
Model Selection & Evaluation
80%
Training Data
20%
Testing Data
Hyperparameter Tuning
Model Training
Model Testing
Model Evaluation
Model Selection
Model Selection and Evaluation
PHASE 04
Deployment
Model Deployment

Serial	F1	F2	F3	–	–	–	–	–	F25	F26	–	–	–	–	–	F39	F40	F41
1	42	2	0	–	–	–	–	–	0	0	–	–	–	–	–	0	4.094	3.572
274446	49	2	0	–	–	–	–	–	0	0	–	–	–	–	–	0	4.278	3.590

FIGURE 3
Methodological flow: comprising four key phases: **(A)** dataset acquisition, **(B)** data preprocessing, **(C)** model selection and evaluation, and **(D)** model deployment.

TABLE 1 Generated features by OFVA.

SN	Feature	Description	Type
F1	Url_length	Representing the number of characters in a URL, including the domain name, path, and any query parameters.	Numeric
F2	Number_of_dots_in_url	Indicating the number of dots (“.”) in the URL	Numeric
F3	Having_repeated_digits_in_url	A Boolean feature that denotes whether the URL has repeated digits (e.g., 2232)	Boolean
F4	Number_of_digits_in_url	Representing the number of digits (0–9) in the URL.	Numeric
F5	Number_of_special_char_in_url	Indicating the number of special characters (e.g., “, #, $, %, &, and ∼) in the URL.	Numeric
F6	Number_of_hyphens_in_url	Representing the number of hyphens (“-”) in the URL.	Numeric
F7	Number_of_underline_in_url	Indicating the number of underscores (“_”) in the URL.	Numeric
F8	Number_of_slash_in_url	Representing the number of forward slashes (“/”) or backward slashes (“/”) in the URL.	Numeric
F9	Number_of_questionmark_in_url	Indicating the number of question marks (“?”) in the URL.	Numeric
F10	Number_of_equal_in_url	Representing the number of equal signs (“=”) in the URL. It is a numeric feature	Numeric
F11	Number_of_at_in_url	Indicating the number of at symbols (“@”) in the URL.	Numeric
F12	Number_of_dollar_sign_in_url	Representing the number of dollar signs (“$”) in the URL.	Numeric
F13	Number_of_exclamation_in_url	Indicating the number of exclamation marks (“!”) in the URL.	Numeric
F14	Number_of_hashtag_in_url	Representing the number of hashtags (“#”) in the URL.	Numeric
F15	Number_of_percent_in_url	Indicating the number of percent signs (%) in the URL.	Numeric
F16	Domain_length	Representing the length of the domain name in the URL.	Numeric
F17	Number_of_dots_in_domain	Representing the number of hyphens (“-”) in the domain name.	Numeric
F18	Number_of_hyphens_in_domain	It is a Boolean feature that denotes whether the domain name contains special characters (e.g., !, “, #, $, %, &, and ∼).	Numeric
F19	Having_special_characters_in_domain	Having special characters (e.g.,!, “, #, $, %, &, and ∼) in domain.	Boolean
F20	Number_of_special_characters_in_domain	Indicating the number of special characters in the domain name.	Numeric
F21	Having_digits_in_domain	It is a Boolean feature that denotes whether the domain name contains digits (e.g., 0–9).	Boolean
F22	Number_of_digits_in_domain	Representing the number of digits in the domain name.	Numeric
F23	Having_repeated_digits_in_domain	A Boolean feature that denotes whether the domain name has repeated digits (e.g., 223321).	Boolean
F24	Number_of_subdomains	Representing the number of subdomains in the URL.	Numeric
F25	Having_dot_in_subdomain	Denoting whether the subdomain contains a dot (“.”).	Boolean
F26	Having_hyphen_in_subdomain	It is a Boolean feature that denotes whether the subdomain contains a hyphen (“-”).	Boolean
F27	Average_subdomain_length	Representing the average length of the subdomains in the URL.	Continuous
F28	Average_number_of_dots_in_subdomain	Indicating the average number of dots (“.”) in the subdomains.	Continuous
F29	Average_number_of_hyphens_in_subdomain	Representing the average number of hyphens (“-”) in the subdomains.	Continuous
F30	Having_special_characters_in_subdomain	Having special characters (e.g., “, #, $, %, &, and ∼) in the subdomain	Boolean
F31	Number_of_special_characters_in_subdomain	Number of special characters (e.g., “, #, $, %, &, and ∼) in the subdomain	Numeric
F32	Having_digits_in_subdomain	It is a Boolean feature that denotes whether the subdomain contains special characters (e.g., “, #, $, %, &, and ∼).	Boolean
F33	Number_of_digits_in_subdomain	Representing the number of digits in the subdomain.	Numeric
F34	Having_repeated_digits_in_subdomain	It is a Boolean feature that denotes whether the subdomain has repeated digits (e.g., 223342).	Boolean
F35	Having_path	Denoting whether the URL has a path.	Boolean
F36	Path_length	Representing the length of the path in the URL	Numeric
F37	Having_query	It is a Boolean feature that denotes whether the URL has a query.	Boolean
F38	Having_fragment	It is a Boolean feature that denotes whether the URL has a fragment.	Boolean
F39	Having_anchor	It is a Boolean feature that denotes whether the URL has an anchor.	Boolean

(Continued)

TABLE 1 (Continued)

SN	Feature	Description	Type
F40	Entropy_of_url	Representing the Shannon entropy of the URL. It is a continuous feature calculated based on the probabilities of each character in the URL. entropy_of_url, $\mathbf{E} = \sum \mathbf{P_i} * \mathbf{P_i}$. Here, $\mathrm{P_i}$ = probability of each character in the URL, and $\log_2$ is the binary logarithm.	Continuous
F41	Entropy_of_domain	Representing the Shannon entropy of the domain. It is a continuous feature calculated based on the probabilities of each character in the domain name. entropy_of_domain, $\mathbf{E} = \sum \mathbf{P_i} * \mathbf{P_i}$. Here, $\mathrm{P_i}$ = probability of each character in the domain, and $\log_2$ is the binary logarithm.	Continuous

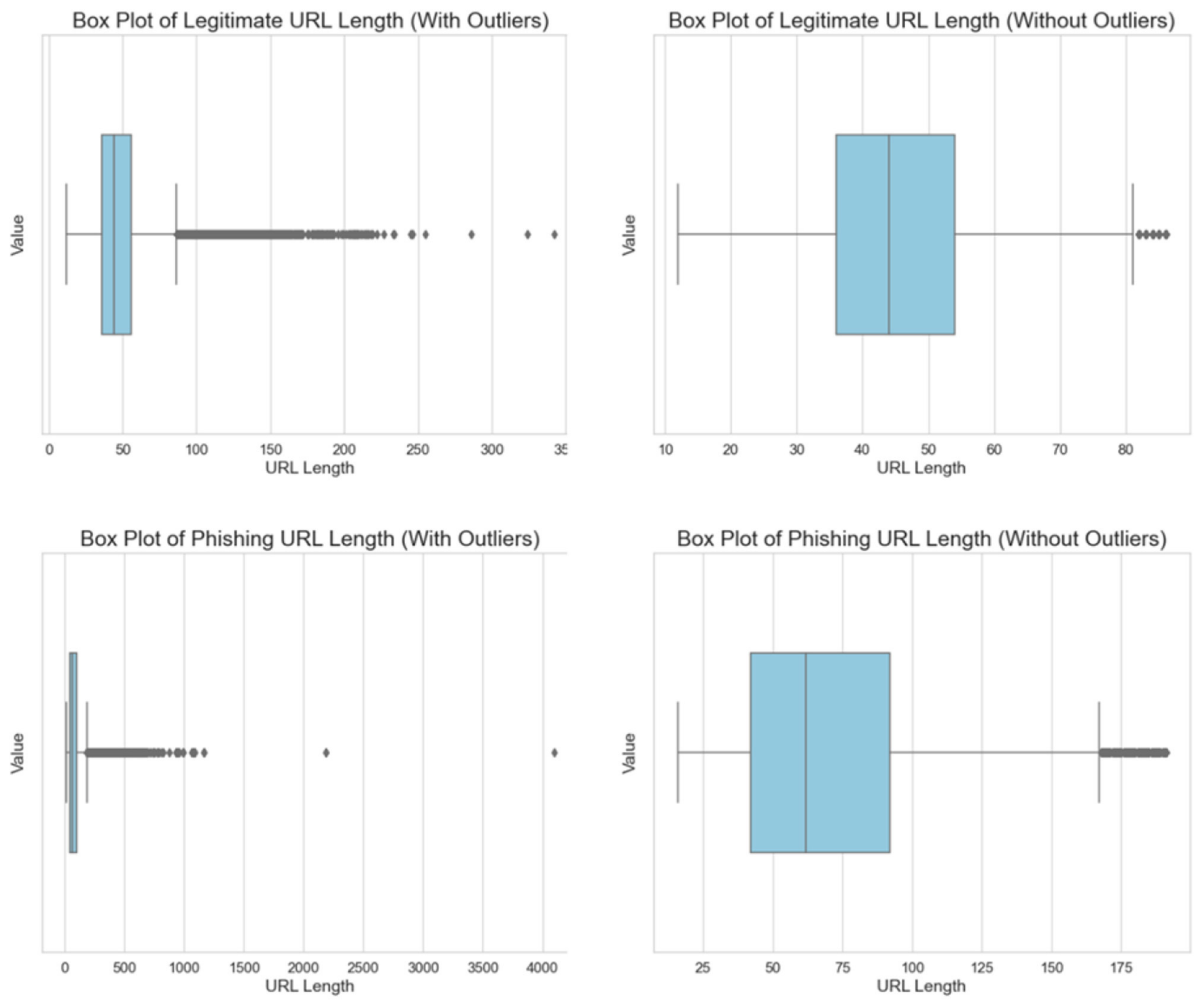


FIGURE 4
Outliers in the dataset (targeting URL length).

illustrates the relative ranking of the 41 features according to their importance scores. The results revealed that the average length of subdomains, URL length, URL entropy, and domain entropy were the most significant features in terms of their contributions to phishing classification, while the presence of a path in the URL was the least significant feature. After conducting multiple experiments, a subset of the top 34 features was selected as the most relevant and informative to train SML models. The associations between the selected features are visualized in Figure 6 using hierarchical clustering (based on their mean values).

3.5 Model selection and evaluation metrics

To find out the optimal model, this study utilized 15 mostly cited SML algorithms arising from different ML families

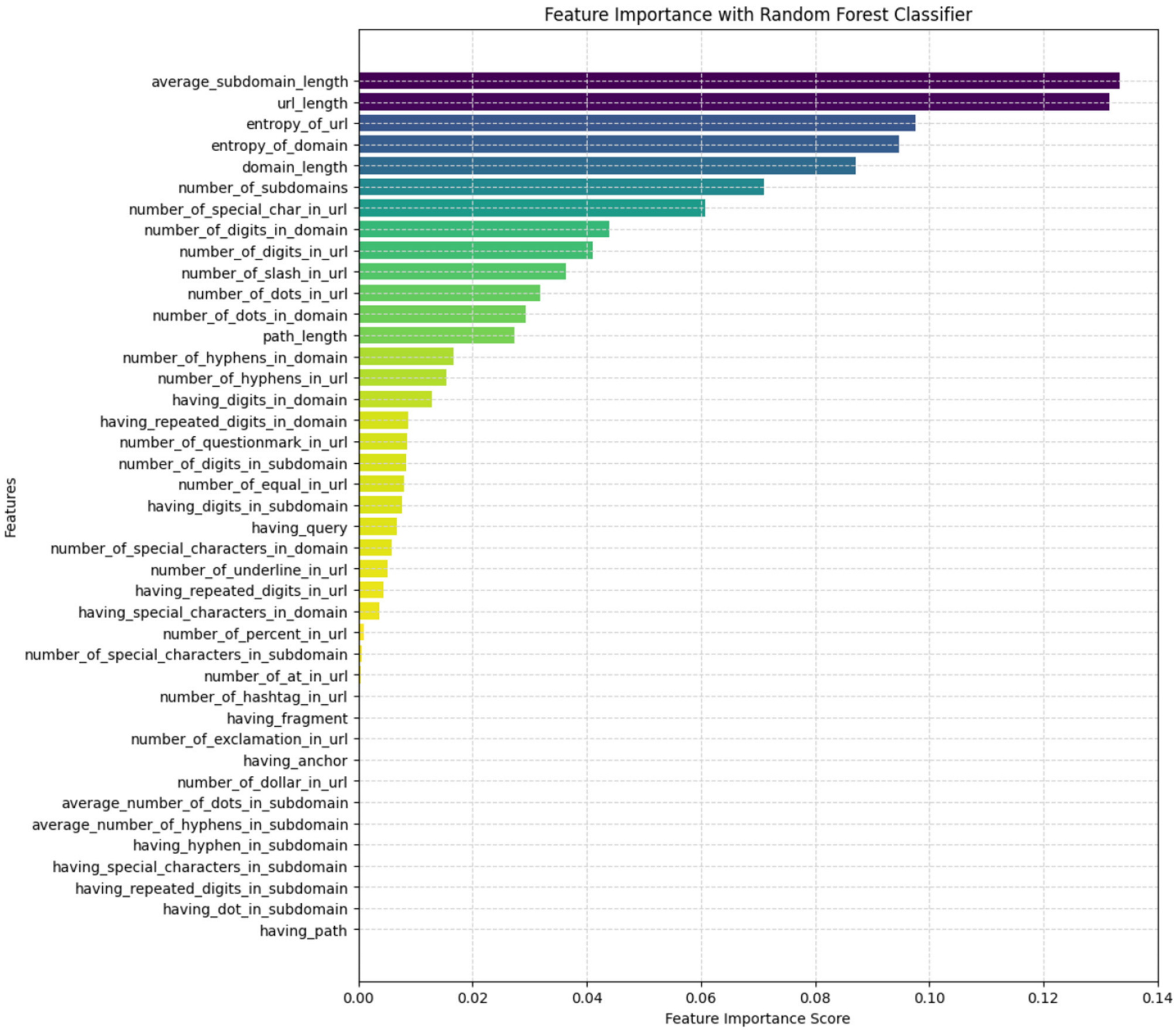


FIGURE 5
Feature importance (targeting URL length). It illustrates the relative ranking of the 41 features according to their importance scores.

(Bayesian, nearest-neighbors, decision trees, neural networks, quadratic discriminant analysis, logistic regression, bagging, boosting, random forests, and ensembles). Appendix B presents a detailed list of the selected classifiers, and the corresponding grid-search generated optimal hyperparameters that maximize accuracy. As part of the experimental setup, we considered 80% of the dataset for training the SML classifiers and 20% of the dataset for testing the classifiers. In addition, a 5-fold cross-validation technique was considered for obtaining stable performances from the classifiers. To apply different machine learning classifiers, the Scikit-learn.org package was considered (Pedregosa et al., 2011).

To assess the performance of the SML classifiers and determine the best optimal model, this study employed seven evaluation metrics (confusion matrix, accuracy, precession, recall, F1-score, ROC curve, and precision-recall curve) for reporting the results of SML prediction models, as suggested by the studies Sattari and Montazer (2023) and Zieni et al. (2023). Initially, a confusion matrix, a commonly used metric for evaluating an ML classifier's performance, is formulated for each of the classifiers. To visualize and summarize a classifier's correct and incorrect predictions, the confusion matrix utilizes four basic terminologies, namely, TP (true positive), TN (true negative), FP (false positive), and FN (false negative). Then, the accuracy scores of the classifiers are calculated (using Equation 1) to see how well they perform.

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

However, accuracy might be misleading in some circumstances when employed with imbalanced datasets; thus, there are other metrics to consider when evaluating the classifier's performance. Precession is a metric that defines the ratio between true-positive predictions (TP) and the total number of positive predictions (TP+FP) (see Equation 2). It refers to a classifier's ability to not

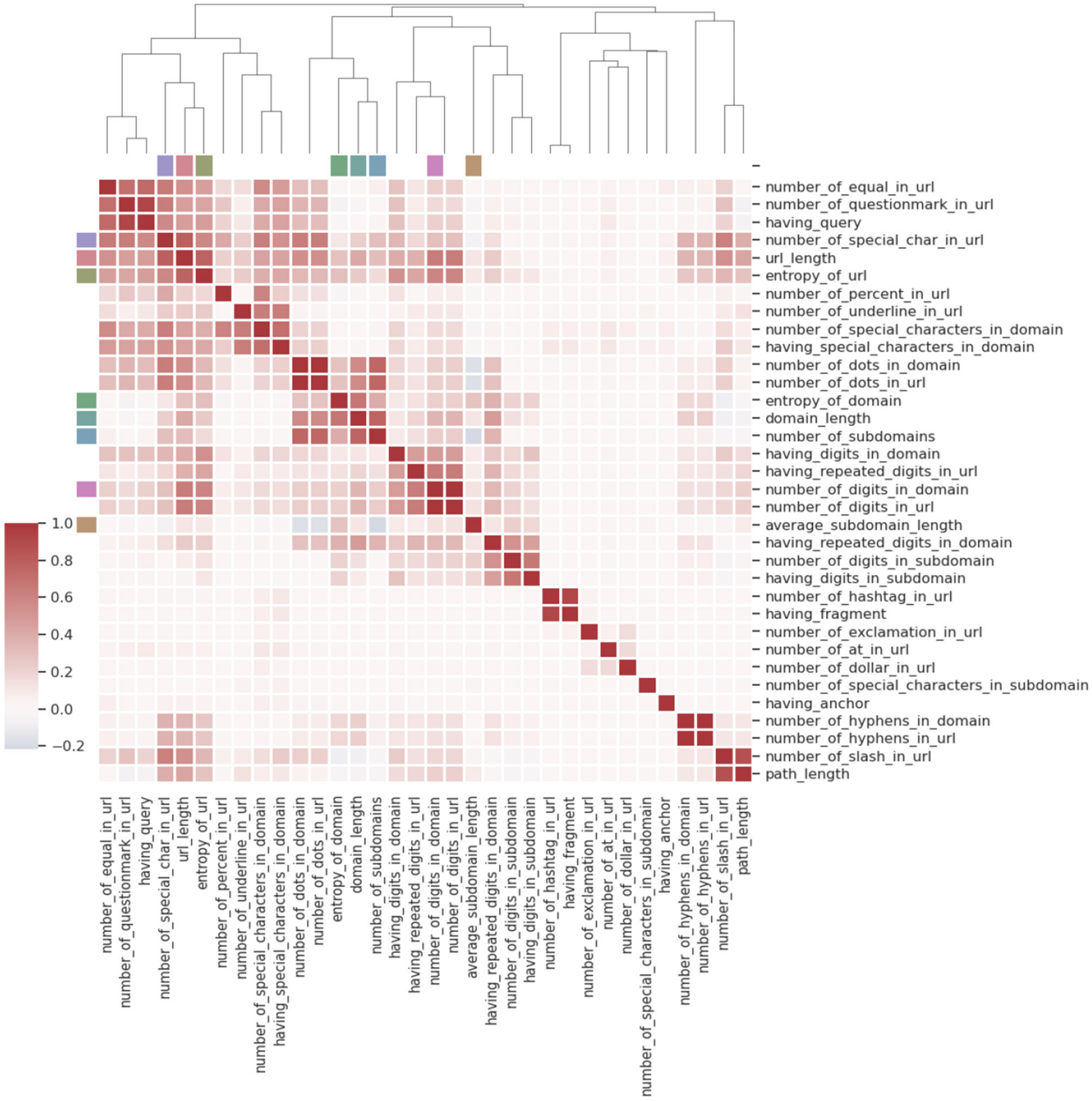


FIGURE 6
Association between the top features (based on their mean values).

categorize a negative sample as positive.

$$Precision\ (P) = \frac{TP}{TP + FP} \qquad (2)$$

This study also used recall to assess the proportion of TP cases correctly identified by the classifiers as positive (or true positive). As of Equation 3, recall is defined as the number of actual positives (TP) over the number of true positives and the number of false negatives (TP+FN).

$$Recall\ (R) = \frac{TP}{TP + FN} \qquad (3)$$

Then, to measure the classifiers' output quality, this study further utilized the F1-score, also known as the harmonic mean of the precision and recall (see Equation 4).

$$F1 - score\ (F1) = 2\,\frac{P{*}R}{P + R} \qquad (4)$$

Finally, the diagnostic capability of the classifiers was evaluated using both the receiver operating characteristic (ROC) curve and the precision-recall curve, ensuring a comprehensive assessment. While the ROC curve illustrates the trade-off between true-positive rate and false-positive rate, the precision-recall curve offers a more

TABLE 2 Distinguishing characteristics between phishing and legitimate URLs.

Key features	Phishing		Legitimate	
	Average	STD	Average	STD
URL length	71.36	36.41	46.28	13.21
Number of dots in URL	3.31	2.21	2.51	0.82
Number of digits in URL	7.45	12.75	1.50	2.98
Number of special characters in URL	11.94	6.02	8.79	2.11
Number of slashes in URL	5.29	2.24	4.32	1.11
Domain length	22.61	17.68	16.93	4.34
Number of digits in the domain	7.45	12.75	1.50	2.98
Number of subdomains	2.19	1.80	2.06	0.55
Average subdomain length	7.62	5.62	4.35	2.84
Number of digits in subdomain	0.43	1.79	0.03	0.28
Entropy of URL	4.38	0.37	4.14	0.23
Entropy of the domain	3.49	0.42	3.29	0.30

nuanced view by focusing on the trade-off between precision and recall. Typically, the ROC curve plots the true-positive rate on the Y-axis against the false-positive rate on the X-axis. However, the precision-recall curve emphasizes precision on the Y-axis against recall on the X-axis. Both curves provide valuable insights into the performance of the classifiers across various thresholds and are essential tools for assessing model effectiveness.

3.6 Deployment

Based on the performances of the selected classifiers, the best ML model was proposed for deployment into a production environment to combat phishing attacks (e.g., Anti-Phishing Tool, Anti-phishing Browser Extension, and Security Recommendation Tool).

4 Results

4.1 Key feature analysis

Table 2 provides a comprehensive analysis of the distinguishing characteristics between phishing and legitimate URLs. The results shed light on several significant findings. First, phishing URLs are substantially longer (avg = 71.36) in comparison to legitimate URLs (avg = 46.28). The considerable standard deviation associated with phishing URLs (STD = 36.41) indicates a wide variation in URL length, while legitimate URLs demonstrate relatively lower variability (STD = 13.21). Second, phishing URLs display a higher number of dots (avg. = 3.31) than legitimate URLs (avg. = 2.51), indicating a discernible discrepancy between the two categories in terms of dot usage. Third, the investigation reveals that phishing URLs tend to include a significantly greater number of digits (avg. = 7.45) than legitimate URLs (avg. = 1.50). Additionally, phishing URLs exhibit a higher frequency of special characters (avg. 11.94) than legitimate URLs (avg. 8.79). Moreover, phishing URLs have a slightly higher average number of slashes (avg = 5.29) than legitimate URLs (avg = 4.32). Phishing URLs have a longer domain (avg = 22.61) than legitimate URLs (avg = 16.93). Interestingly, phishing URLs also manifest a significantly higher average number of digits (avg = 7.45) in their domains relative to legitimate URLs (avg = 1.50). Concerning the number of subdomains, phishing URLs demonstrate a slightly higher average count (avg = 2.19) than legitimate URLs (avg = 2.06). However, phishing URLs have a longer subdomain (avg = 7.62) than legitimate URLs (avg = 4.35). Additionally, phishing URLs display a slightly higher average number of digits (avg=0.43) in their subdomains in comparison to legitimate URLs (avg = 0.03). In terms of the entropy (randomness or complexity) of the URLs, phishing URLs and domains show a higher level of randomness than legitimate URLs and domains.

4.2 Performance comparison of the 15 SML algorithms

Tables 3, 4 present a detailed assessment of the performance of 15 supervised machine learning classifiers in a binary classification task focused on identifying phishing URLs. The evaluation encompasses key performance metrics such as accuracy, 5-fold cross-validation (CV) accuracy, precision, recall, F1 score, training time, and testing time for each classifier. The primary objective was to determine the most effective classifier for accurately detecting phishing URLs.

The results reveal that the overall accuracy of the classifiers varies between 62.8% and 97.52%. In a binary classification scenario, RF and ETC exhibited superior performance when compared to other classifiers. Specifically, RF with the Gini criterion achieved an accuracy of 97.5%, 5-fold CV accuracy of 97.52%, precision of 97.50% (legitimate = 0.97; phishing = 0.98), recall of 97.5% (legitimate = 0.98; phishing = 0.97), F1-score of 98% (legitimate = 0.98; phishing = 0.98), training time of 47.9 s, and testing time of 2.9 s. ETC also demonstrated the second-highest accuracy at 96.7%, with a 5-fold CV accuracy of 96.63%, precision of 96.5% (legitimate = 0.96; phishing = 0.97), recall of 97% (legitimate = 0.97; phishing = 0.98), F1-score of 97% (legitimate = 0.97; phishing = 0.97), training time of 33.63 s, and testing time of 3.05 s. Moreover, BAG with base estimator DT secured the third-best performance with an accuracy of 96.00%, 5-fold CV accuracy of 96.00%, precision of 96% (legitimate = 0.95; phishing = 0.97), recall of 96% (legitimate = 0.97; phishing = 0.95), F1-score of 96% (legitimate = 0.96; phishing = 0.96), training time of 16.58 s, and testing time of 0.23 s. DT followed closely as the fourth-best classifier with an accuracy of 95.4%, 5-fold CV accuracy of 95.36%, precision of 96.5% (legitimate = 0.96; phishing = 0.97), recall of 97.0% (legitimate = 0.98; phishing = 0.96), F1-score of 97% (legitimate = 0.97; phishing = 0.97), training time of 2.5 s, and testing time of 0.03 s. Additionally, KNN, MLP, and HGBS

TABLE 3 Confusion matrixes of 15 classifiers.

Classifier	True positive (TP)	False positive (FP)	False negative (FN)	True negative (TN)
Random forest (RF)	25,227	585	688	23,090
Decision tree (DT)	22,639	1,019	1,283	24,649
K-nearest neighbors (KNN)	21,518	1960	2,404	23,708
Gaussian naive Bayes (GNB)	12,908	2,002	11,014	23,666
MultinomialNB (MNB)	10,273	4,801	13,649	20,867
ComplementNB (CNB)	10,310	4,825	13,612	20,843
SGDClassifier (SGDC)	20,354	8,455	3,568	17,213
Bagging (BAG)	22,606	678	1,316	24,990
ExtraTreesClassifier (ETC)	22,878	619	1,044	25,049
Adaboost (AB)	19,157	3,113	4,765	22,555
GradientBoostingClassifier (GBC)	19,722	2,581	4,200	23,087
HistGradientBoostingClassifier (HGBS)	20,751	1,924	3,171	23,744
Quadratic Discriminant Analysis (QDA)	9,775	972	14,147	24,696
Logistic regression (LR)	17,207	3,365	6,715	22,303
Multi-layer perceptron (MLP)	20,922	1,985	3,000	23,683

classifiers exhibited commendable performance, achieving overall accuracy scores ranging from 89.9% to 91.4%. Conversely, MNB and CNB classifiers emerged as the least performing, recording an accuracy score of 62.8%. The accompanying figure illustrates the summarized analysis of ROC curves among the 15 classifiers (see Figure 7). The ROC curve's vertical axis represents the true-positive rate, while the horizontal axis signifies the false-positive rate. A higher AUC value of 0.97 for both RF and ETC suggests superior performance compared to other classifiers, as depicted in Figure 7. This trend is consistent in the precision-recall curve, indicating high precision and recall for RF and ETC (see Figure 8). The overall evaluation offers a comprehensive understanding of the performance of the selected 15 machine learning classifiers in phishing classification, highlighting variations in their effectiveness for specific classes (both legitimate and phishing).

5 Discussion

Initially, this study revealed several significant distinguishing characteristics between phishing and legitimate URLs, as presented in Table 2. From this analysis, it is evident that phishing URLs are significantly longer and exhibit more variation in length than legitimate ones. They also contain more dots, digits, and special characters, potentially aiming to appear more complex and making them harder to detect at a glance. Additionally, the higher number of dots in phishing URLs could be due to the inclusion of subdomains or additional path components, which may be used to mimic legitimate website structures or redirect users to malicious pages. Interestingly, while the number of subdomains is not drastically higher for phishing URLs, the subdomains themselves are longer and contain more digits. Finally, the higher entropy observed in phishing URLs suggests a greater level of randomness or complexity, possibly resulting from automated generation techniques used by cybercriminals to create a large volume of unique phishing URLs quickly.

Along with revealing these distinguishing characteristics, this study also examined the performance of different classifiers from various machine learning families in classifying phishing URLs, as shown in Tables 3, 4 and Figures 7, 8. High-performing classifiers such as RF and ETC demonstrated exceptional performance, with average precision (AP) and area under the curve (AUC) values both reaching 0.95 and 0.97, respectively. This robustness is likely due to their use of ensemble techniques that average multiple decision trees, effectively reducing variance and capturing complex patterns. DT classifiers also showed strong performance, with AP and AUC values of 0.93 and 0.95, respectively, owing to their ability to capture non-linear relationships and their interpretability.

Moderate-performing classifiers such as KNN, histogram-based HGBC, GBC, and MLP showed reasonable performance, with AP and AUC values ranging from 0.81 to 0.91. KNN benefits from its non-parametric nature, capturing local data structures without assuming specific distributions, although its performance heavily depends on the choice of distance metric and the value of k. Boosting methods such as GBC and HGBC sequentially focus on hard-to-classify instances, improving overall accuracy, while the ability of MLP to model complex, non-linear relationships through neural networks leads to high AP and AUC values, despite significant training time requirements. In contrast, low-performing classifiers such as LR, stochastic SGDC, and Naive Bayes variants (GNB, MNB, and CNB) exhibited poorer performance. This might be due to their underlying assumptions, sensitivity to data characteristics, and limited ability to capture complex patterns inherent in phishing URLs.

Our proposed approach, also compared with existing methods as presented in Table 5, demonstrates superior performance in several aspects. First, this study introduces the OFVA to

TABLE 4 Classifiers' performance in binary-class classification.

Classifiers	Accuracy	5-fold CV accuracy	Class	Precision	Recall	F1 score	Training time (S)	Testing time(S)	Support
Random forest (RF)	97.50%	97.52%	0	0.97	0.98	0.98	47.9	2.9	25,668
			1	0.98	0.97	0.98			23,922
Decision tree (DT)	95.4%	95.36%	0	0.96	0.98	0.97	2.5	0.03	25,668
			1	0.97	0.96	0.97			23,922
K-Nearest neighbors (KNN)	91.2%	91.3%	0	0.95	0.96	0.96	0.09	73.1	25,668
			1	0.96	0.95	0.95			23,922
Gaussian Naive Bayes (GNB)	73.8%	73.73%	0	0.68	0.92	0.78	0.19	0.02	25,668
			1	0.87	0.54	0.67			23,922
MultinomialNB (MNB)	62.8%	63.10%	0	0.61	0.81	0.69	0.14	0.03	25,668
			1	0.68	0.43	0.53			23,922
ComplementNB (CNB)	62.8%	63.12%	0	0.61	0.81	0.69	0.14	0.02	25,668
			1	0.68	0.43	0.53			23,922
SGDClassifier (SGDC)	75.8%	77.33%	0	0.83	0.67	0.74	17.16	0.02	25,668
			1	0.71	0.85	0.77			23,922
Bagging (BAG)	96.00%	96.00%	0	0.95	0.97	0.96	16.58	0.23	25,668
			1	0.97	0.95	0.96			23,922
ExtraTreesClassifier (ETC)	96.7%	96.63%	0	0.96	0.98	0.97	33.63	3.05	25,668
			1	0.97	0.96	0.97			23,922
Adaboost (AB)	84.1%	84.52%	0	0.83	0.88	0.85	13.4	0.4	25,668
			1	0.86	0.80	0.83			23,922
GradientBoostingClassifier (GBC)	86.3%	86.5%	0	0.84	0.90	0.87	55.98	0.11	25,668
			1	0.88	0.82	0.85			23,922
HistGradientBoosting Classifier (HGBC)	89.7%	89.8%	0	0.88	0.93	0.90	10.61	0.49	25,668
			1	0.92	0.87	0.89			23,922
Quadratic Discriminant Analysis (QDA)	69.5%	72.52%	0	0.64	0.96	0.77	0.83	0.09	25,668
			1	0.91	0.41	0.56			23,922
Logistic Regression (LR)	79.7%	80.00%	0	0.77	0.87	0.82	6.34	0.01	25,668
			1	0.84	0.72	0.77			23,922
Multi-layer Perceptron (MLP)	89.9%	89.71%	0	0.89	0.92	0.91	852.8	0.54	25,668
			1	0.91	0.88	0.89			23,922

extract 41 optimal intra-URL features. Among them, 10 novel features are entirely new, exhibiting a high distinguishing ability in classifying phishing URLs compared to existing approaches (see Figure 3). These novel features not only enhance the accuracy of phishing detection but also contribute to a deeper understanding of the underlying characteristics that can be leveraged to identify phishing attacks (see Table 2). Second, this study employs a larger, more up-to-date dataset, consisting of 274,446 instances. This extensive dataset enables a comprehensive analysis and evaluation of the proposed approach. However, most of the previous studies considered small datasets (see Table 5). Additionally, the study incorporates a thorough data preprocessing step, performs hyperparameter tuning, and utilizes 15 supervised machine learning (SML)

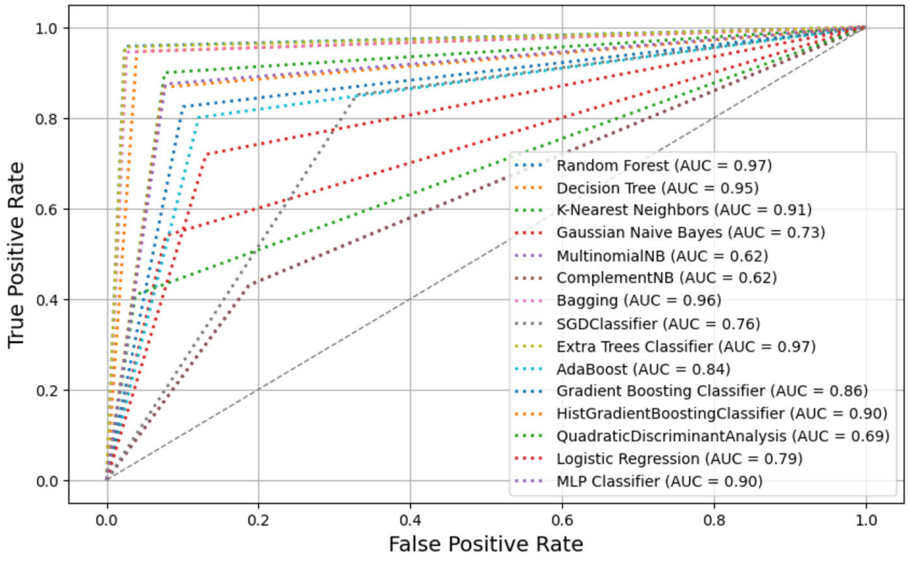


FIGURE 7
ROC curve analysis of 15 classifiers.

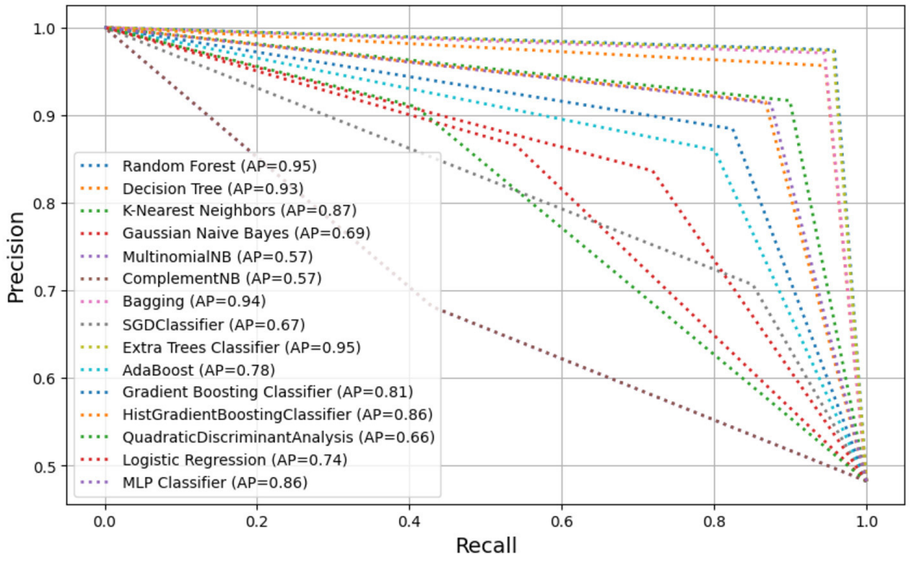


FIGURE 8
Precision-recall curves of 15 classifiers.

algorithms derived from various machine learning families. This ensures a diverse range of models for robust performance assessment and comparison (see Table 4). Lastly, this study introduces a lightweight anti-phishing model (random forest) capable of detecting phishing attacks with 97.52% accuracy. This model focuses on specific intra-URL features, offering a fast, efficient, and practical solution for combating phishing attempts. The lightweight nature of the model implies that it can be easily implemented in real-world scenarios, providing a valuable tool for organizations and individuals seeking enhanced protection against phishing threats.

6 Study limitations and future research directions

While this study presents several strengths, it is important to acknowledge its inherent limitations. One such limitation pertains to the evaluation metrics employed. Although a variety of evaluation metrics (confusion matrix, accuracy, precession, recall, F1-score, and ROC curve) were utilized in this study, it falls short of explicitly addressing the potential constraints and trade-offs associated with these metrics. It is crucial to recognize that different metrics may prioritize distinct facets

TABLE 5 Comparison of the proposed approach with existing approaches.

References	Approach	Proposed solution	Key characteristics	Advantages and disadvantages
Dutta (2021)	ML-based	This study proposed an ML-based framework to predict phishing URLs without visiting the webpage or utilizing any third-party services.	Data volume: 969241 Algorithms used: NB, LR, SVM, KNN, MLP, J48, AB, and RF No. of features: 100 Novel features: 10 Preprocessing: Yes Hyperparameter tuning: No K-fold cross-validation: No Accuracy (avg.): 93.87 (benchmark datasets)	The proposed framework demonstrates computational efficiency and attains superior accuracy, yet it lacks both hyperparameter tuning and cross-validation.
Aljofey et al. (2022)	ML-based	This study proposed a solution to detect phishing websites based on a new set of features.	Data volume: 60252 Algorithms used: XGBoost, RF, LR, NB, Ensemble, and AB No. of features: 15 Novel features: 8 Preprocessing: Yes Hyperparameter tuning: No K-fold cross-validation: No Accuracy: 96.76%	The proposed solution demonstrated improved accuracy and a low false-positive rate. Nevertheless, it lacks both hyperparameter tuning and cross-validation, raising concerns about its generalizability.
Zouina and Outtaj (2017)	ML-based	This study proposed a lightweight phishing detection system using SML algorithms and a similarity index.	Data volume: 2000 Algorithms used: NB, SVM, and PNN No. of features: 6 Novel features: 0 Preprocessing: Yes Hyperparameter tuning: No K-fold cross-validation: No Accuracy: 95.80%	The proposed system is relatively lightweight, but its performance may be limited due to the small dataset. Furthermore, it lacks both hyperparameter tuning and cross-validation, which raises concerns about its ability to generalize for widespread use.
Chiew et al. (2019)	ML-based	This study proposed an ML-based hybrid ensemble feature selection framework to detect phishing attacks.	Data volume: 5000 Algorithms used: SVM, NB, C4.5, JRip, PART No. of features: 48 Novel features: 0 Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: No Accuracy: 94.6%	The approach demonstrates innovation, and the dataset is small. Additionally, the absence of hyperparameter tuning and cross-validation raises doubts about its practical applicability.
Dutta (2021)	ML-based	This study proposed an ML-based approach using (RNN-LSTM) to detect phishing attacks.	Data volume: 13700 Algorithms used: RNN (LSTM) No. of features: Not mentioned Novel features: Not mentioned Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: No Accuracy: 97.4%	The proposed approach achieved high accuracy; however, several important metrics are missing such as hyperparameter tuning and cross-validation.
Alsariera et al. (2021)	Meta-learners-based	This study proposed three meta-learner models (ForestPA-PWDM, Bagged-ForestPAPWDM, and Adab-ForestPA-PWDM) based on Forest Penalizing Attributes (ForestPA) algorithm to detect phishing websites	Data volume: 11055 Algorithms used: ForestPA, Bagging and Boosting. No. of features: 30 Novel features: 0 Preprocessing: Not mentioned Hyperparameter Tuning: No K-fold cross-validation: Yes Accuracy: 96.26%	This study has introduced a novel method for detecting phishing websites, which has demonstrated a high level of accuracy. Nonetheless, the limited size of the dataset poses a challenge for achieving generalization.
Nagaraj et al. (2018)	ML-based	This study proposed an ensemble machine learning model for classifying phishing websites.	Data volume: 10,068 Algorithms used: Ensemble, RF, and NN No. of features: 30 Novel features: Not mentioned	This study achieved good accuracy by employing an ensemble machine-learning model. Nevertheless, the limited size of the dataset poses a challenge when it comes to generalizing the model's performance.

(Continued)

TABLE 5 (Continued)

References	Approach	Proposed solution	Key characteristics	Advantages and disadvantages
			Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: Yes Accuracy: 93.41	
Balogun et al. (2021)	Meta-learner based	This study proposed a Functional Tree (FT)-based meta-learning model for detecting phishing websites.	Data volume: 22408 (total) Algorithms used: Baseline: NB, SMO, SVM, and DTs; Ensemble: Bagging, AB, Rotation forest No. of features: 30, 48, and 10 (for datasets 1, 2, and 3 respectively) Novel features: 0 Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: Yes Accuracy: 98.51% (highest)	The proposed approach demonstrated commendable accuracy on one dataset. Nevertheless, in certain cases, the accuracy decreased significantly, reaching as low as 87.73%, which raises concerns about the stability and trustworthiness of the approach for practical applications.
Mourtaji et al. (2021)	Hybrid Rule-based	This study proposed a hybrid rule-based solution for phishing detection using CNN	Data Volume: 40000 Algorithms used: CART, SVM, KNN, CNN, and MLP No. of features: 37 Novel features: Not mentioned Preprocessing: Yes Hyperparameter Tuning: Yes K-fold cross-validation: Yes Accuracy (avg.): 93.47 (highest = 97.945)	The study found deep learning to outperform SML with good accuracy, but the dataset is too limited for a deep learning model. In addition, increasing the dataset could raise concerns about computational efficiency.
Orunsolu et al. (2022)	ML-based	This study proposed an ML-based predictive model for phishing detection.	Data volume: 5041 Algorithms used: SVM, NB No. of features: 37 Novel features: Not mentioned Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: Yes Accuracy: 99.96%	The study attained higher accuracy compared to previous research. Yet the study's small dataset raises concerns regarding its generalizability for broader applications.
Alsariera et al. (2020)	Meta-learner based	This study proposed four AI-based meta-learners to predict phishing websites.	Data volume: 11055 Algorithms used: ABET, BET, RoFBET, and LBET No. of features: 30 Novel features: 0 Preprocessing: Yes Hyperparameter Tuning: No K-fold cross-validation: Yes Accuracy: >97%	The authors claimed that the proposed models exhibited superior performance compared to existing ML-based models. Nonetheless, the limited dataset they utilized raises concerns regarding the applicability of these models in real-life scenarios.
Proposed Approach	ML-based	In this study, we have proposed a more robust, effective, sophisticated, and reliable solution for phishing detection through the optimal feature vectorization algorithm (OFVA) and supervised machine learning (SML) algorithms.	Data Volume: 274446 Algorithms used: Feature extraction: OFVA; classification: RF, DT, KNN, GNB, MNB, CNB, SGDC, BAG, ETC, AB, GBC, HGBS, QDA, LR, and MLP No. of features: 41 Novel features: 10 Preprocessing: Yes (data cleansing, curation, feature extraction, and feature selection) Hyperparameter Tuning: Yes K-fold cross-validation: Yes Accuracy: 97.52%	The approach presented in this study surpasses previous studies on several fronts. For instance, the study utilizes a previously unused, large dataset. Furthermore, it employs an organized preprocessing pipeline. To enhance accuracy and generalizability, hyperparameter tuning and cross-validation were diligently conducted.

of performance, underscoring the necessity of comprehensively considering the limitations and ramifications tied to the reliance on specific metrics. Furthermore, this study primarily focused on supervised machine learning algorithms, potentially overlooking the benefits that could arise from the incorporation of deep learning techniques for enhanced outcomes. Regrettably, due to constraints in hardware infrastructure, the exploration of deep learning was omitted. Lastly, in pursuit of a more simple, fast, and responsive model, certain content-related features such as web images or logos, DOM (document object model), as well as HTML and CSS structural elements, were excluded. While this design choice aimed to optimize speed and responsiveness, it should be acknowledged that the inclusion of these features could conceivably lead to heightened accuracy. To this end, future research could address these limitations by exploring the use of deep learning techniques, investigating the impact of content-related features, developing new evaluation metrics, and applying the findings to other types of data and tasks. Additionally, future research could investigate the impact of different training data sets, develop ensemble methods, and explore the use of explainable AI (XAI) techniques.

7 Conclusions

Currently, phishing has taken a terrifying shape globally and is considered the door for all kinds of malware and ransomware (Basit et al., 2022). Unlike other forms of cybercrime, where attackers' motives are known and victim types are consistent, phishers are likely to have varying goals, motivations, and victim types. Consequently, phishing detection has become a major challenge over time, resulting in an exponential growth of phishing attacks over the last few years. To pull the reins off the present growing trend of phishing attacks, this study employed an ML-based, real-world data-driven approach to detect phishing sites based on URL-based features. The study utilized a large dataset comprising 2,74,446 raw URLs, consisting of both phishing and legitimate URLs, and extracted 41 optimal intra-URL features using the OFVA. Among these features, 10 novel ones were introduced, exhibiting high distinguishing ability in classifying phishing URLs. Through a comprehensive evaluation and comparison of 15 SML algorithms from various machine learning families, our experiments suggested that the RF classifier outperformed the others, achieving an accuracy rate of 97.52% with high precision and an AUC value of 98%. We expect that the proposed lightweight anti-phishing model, specifically focusing on intra-URL features, will provide a fast, efficient, and practical solution for combating phishing attempts.

Data availability statement

The dataset presented in this study can be found in online repository. The name of the repository and accession number can be found below: https://doi.org/10.17632/6tm2d6sz7p.1.

Author contributions

MT: Conceptualization, Data curation, Formal analysis, Methodology, Writing – original draft. MI: Project administration, Supervision, Validation, Writing – review & editing. TB: Supervision, Validation, Writing – review & editing. AS: Conceptualization, Investigation, Project administration, Supervision, Validation, Writing – review & editing. NP: Validation, Funding acquisition, Resources, Visualization, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Acknowledgments

We would like to extend our deepest gratitude to the ICT Division, People's Republic of Bangladesh, for their invaluable support throughout this study. Their unwavering assistance and guidance have played a vital role in the successful completion of our research. And we want to thank the reviewers for their insights.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fcomp.2024.1428013/full#supplementary-material

The supplementary material submitted along with our manuscript consist of two appendices: Appendices A, B. In Appendix A, we have provided a comprehensive description of the optimal feature vectorization algorithm (OFVA). This algorithm is designed to process a list of URLs as input and subsequently extract 42 distinct features, which are then presented as output. In contrast, Appendix B is dedicated to showcasing the optimal hyperparameter settings we have identified for a set of 15 supervised machine learning classifiers. These settings have been determined through a thorough grid-search process to ensure the highest performance of these classifiers.

References

Abdelnabi, S., Krombholz, K., and Fritz, M. (2020). "VisualPhishNet: Zero-day phishing website detection by visual similarity," in *Proceedings of the 2020 ACM SIGSAC.*

Adebowale, M. A., Lwin, K. T., and Hossain, M. A. (2020). Intelligent phishing detection scheme using deep learning algorithms. *J. Enterpr. Inf. Manage.* 36, 747–766. doi: 10.1108/JEIM-01-2020-0036

Adewole, K. S., Akintola, A. G., Salihu, S. A., Faruk, N., and Jimoh, R. G. (2019). Hybrid rule-based model for phishing URLs detection. *Lecture Notes Inst. Comput. Sci. Soc. Inf. Telecommun. Eng.* 12, 119–135. doi: 10.1007/978-3-030-23943-5_9

Alabdan, R. (2020). Phishing attacks survey: types, vectors, and technical approaches. *Fut. Int.* 12:168. doi: 10.3390/fi12100168

Alahmari, S., Renaud, K., and Omoronyia, I. (2022). Moving beyond cyber security awareness and training to engendering security knowledge sharing. *Inf. Syst. e-Bus. Manage.* 21:123–158. doi: 10.1007/s10257-022-00575-2

Aldakheel, E. A., Zakariah, M., Gashgari, G. A., Almarshad, F. A., and Alzahrani, A. I. (2023). Deep learning-based innovative technique for phishing detection in modern security with uniform resource locators. *Sensors* 23:4403. doi: 10.3390/s23094403

Aljofey, A., Jiang, Q., Rasool, A., Chen, H., Liu, W., Qu, Q., et al. (2022). An effective detection approach for phishing websites using URL and HTML features. *Sci. Rep.* 12:10841. doi: 10.1038/s41598-022-10841-5

Alkhalil, Z., Hewage, C., Nawaf, L., and Khan, I. (2021). Phishing attacks: a recent comprehensive study and a new anatomy. *Front. Comput. Sci.* 3:563060. doi: 10.3389/fcomp.2021.563060

Alnemari, S., and Alshammari, M. (2023). Detecting phishing domains using machine learning. *Applied Sci.* 13:4649. doi: 10.3390/app13084649

Alsariera, Y. A., Adeyemo, V. E., Balogun, A. O., and Alazzawi, A. K. (2020). AI meta-learners and extra-trees algorithm for the detection of phishing websites. *IEEE Access* 8, 142532–142542. doi: 10.1109/ACCESS.2020.3013699

Alsariera, Y. A., Elijah, A. V., and Balogun, A. O. (2021). Phishing website detection: forest by penalizing attributes algorithm and its enhanced variations. *Arab. J. Sci. Eng.* 45, 10459–10470. doi: 10.1007/s13369-020-04802-1

Anitha, J., and Kalaiarasu, M. (2022). A new hybrid deep learning-based phishing detection system using MCS-Dnn Classifier. *Neur. Comput. Appl.* 34, 5867–5882. doi: 10.1007/s00521-021-06717-w

Anti-Phishing Working Group (APWG) (2022). *Phishing Activity Trends Report, 3rd Quarter 2022.* Available online at: https://docs.apwg.org/reports/apwg_trends_report_q3_2022.pdf (accessed May 9, 2022).

APWG and Phishing Activity Trends Reports (2022). Apwg.org. Available: https://apwg.org/trendsreports. (accessed August 30, 2022).

Ardi, C., and Heidemann, J. (2016). "Auntietuna: personalized content-based phishing detection," in *Proceedings 2016 Workshop on Usable Security*.

Azeez, N. A., Misra, S., Margaret, I. A., Fernandez-Sanz, L., and Abdulhamid, S. M. (2021). Adopting automated whitelist approach for detecting phishing attacks. *Comput. Secur.* 108:102328. doi: 10.1016/j.cose.2021.102328

Balogun, A. O., Adewole, K. S., Raheem, M. O., Akande, O. N., Usman-Hamza, F. E., Mabayoje, M. A., et al. (2021). Improving the phishing website detection using empirical analysis of Function Tree and its variants. *Heliyon* 7:e07437. doi: 10.1016/j.heliyon.2021.e07437

Basit, A., Zafar, M., Liu, X., Javed, A. R., Jalil, Z., Kifayat, K., et al. (2022). A comprehensive survey of AI-enabled phishing attacks detection techniques. *Telecommun. Syst.* 76, 139–154. doi: 10.1007/s11235-020-00733-2

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324

Chen, S., Hao, M., Ding, F., Jiang, D., Dong, J., Zhang, S., et al. (2023). Exploring the global geography of cybercrime and its driving forces. *Hum. Soc. Sci. Commun.* 10:1560. doi: 10.1057/s41599-023-01560-x

Chiew, K. L., Chang, E. H., Sze, S. N., and Tiong, W. K. (2015). Utilisation of website logo for phishing detection. *Comput. Secur.* 54, 16–26. doi: 10.1016/j.cose.2015.07.006

Chiew, K. L., Tan, C. L., Wong, K., Yong, K. S., and Tiong, W. K. (2019). A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Inf. Sci.* 484, 153–166. doi: 10.1016/j.ins.2019.01.064

Daengsi, T., Pornpongtechavanich, P., and Wuttidittachotti, P. (2021). Cybersecurity awareness enhancement: a study of the effects of age and gender of Thai employees associated with phishing attacks. *Educ. Inf. Technol.* 27, 4729–4752. doi: 10.1007/s10639-021-10806-7

Deepika, J., Rajan, C., and Senthil, T. (2021). Security and privacy of cloud-and IoT-based medical image diagnosis using fuzzy convolutional neural network. *Comput. Int. Neurosci.* 2021:6615411. doi: 10.1155/2021/6615411

Dhanavanthini, P., and Chakkravarthy, S. S. (2023). Phish-Armour: Phishing detection using deep recurrent neural networks. *Soft Comput.* 11:7962. doi: 10.1007/s00500-023-07962-y

Dodge, R. C., and Ferguson, A. J. (2006). "Using phishing for user email security awareness," in *IFIP International Information Security Conference.* Boston, MA: Springer US, 454–459.

Download List of top 10 Million Domains Based on Open Data from Common Crawl and Common Search (2023). Available online at: https://www.domcop.com/top-10-million-domains (accessed May 22, 2023).

Dutta, A. K. (2021). Detecting phishing websites using machine learning technique. *PLoS ONE* 16:e0258361. doi: 10.1371/journal.pone.0258361

Gupta, B. B., Tewari, A., Jain, A. K., and Agrawal, D. P. (2016). Fighting against phishing attacks: state of the art and future challenges. *Neural Comput. Appl.* 28, 3629–3654. doi: 10.1007/s00521-016-2275-y

Hoehe, M. R., and Thibaut, F. (2020). Going digital: how technology use may influence human brains and behavior. *Dial. Clin. Neurosci.* 22, 93–97. doi: 10.31887/DCNS.2020.22.2/mhoehe

Jain, A. K., and Gupta, B. B. (2016). A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP J. Inf. Secur.* 2016:34. doi: 10.1186/s13635-016-0034-3

Jain, A. K., and Gupta, B. B. (2017). Phishing detection: analysis of visual similarity based approaches. *Secur. Commun. Netw.* 2017, 1–20. doi: 10.1155/2017/5421046

Jain, A. K., and Gupta, B. B. (2018). Rule-based framework for detection of SMISHING messages in Mobile environment. *Procedia Comput. Sci.* 125, 617–623. doi: 10.1016/j.procs.2017.12.079

Jeeva, S. C., and Rajsingh, E. B. (2016). Intelligent phishing URL detection using association rule mining. *Hum. Centr. Comput. Inf. Sci.* 6:64. doi: 10.1186/s13673-016-0064-3

Jensen, M. L., Dinger, M., Wright, R. T., and Thatcher, J. B. (2017). Training to mitigate phishing attacks using mindfulness techniques. *J. Manage. Inf. Syst.* 34, 597–626. doi: 10.1080/07421222.2017.1334499

Kasim, Ö. (2021). Automatic detection of phishing pages with event-based request processing, deep-hybrid feature extraction and light gradient boosted machine model. *Telecommun. Syst.* 78, 103–115. doi: 10.1007/s11235-021-00799-6

Kasim, O. (2022). An efficient ensemble architecture for privacy and security of electronic medical records. *The Int. Arab J. Inf. Technol.* 19:2022. doi: 10.34028/iajit/19/2/14

Khan, W., Ahmad, A., Qamar, A., Kamran, M., and Altaf, M. (2021). SpoofCatch: a client-side protection tool against phishing attacks. *IT Prof.* 23, 65–74. doi: 10.1109/MITP.2020.3006477

Klimburg-Witjes, N., and Wentland, A. (2021). Hacking humans? Social Engineering and the construction of the deficient user in cybersecurity discourses. *Sci. Technol. Hum. Values* 46, 1316–1339. doi: 10.1177/0162243921992844

Li, L., Berki, E., and Helenius, M., and Ovaska, S. (2014). Towards a contingency approach with whitelist- and blacklist-based anti-phishing applications: What do usability tests indicate? *Behav. Inf. Technol.* 33, 1136–1147. doi: 10.1080/0144929X.2013.875221

Li, Z., Xu, W., Shi, H., Zhang, Y., and Yan, Y. (2021). Security and privacy risk assessment of energy big data in cloud environment. *Comput. Int. Neurosci.* 2021:2398460. doi: 10.1155/2021/2398460

Luca, A. R., Ursuleanu, T. F., Gheorghe, L., Grigorovici, R., Iancu, S., Hlusneac, M., et al. (2022). Impact of quality, type and volume of data used by deep learning models in the analysis of Medical Images. *Inf. Med. Unlocked* 29:100911. doi: 10.1016/j.imu.2022.100911

Ludl, C., McAllister, S., and Kirda, E., andamp; Kruegel, C. (2007). On the effectiveness of techniques to detect phishing sites. *Det. Intr. Malware Vulner. Assess.* 22, 20–39. doi: 10.1007/978-3-540-73614-1_2

Maqsood, U., Ur Rehman, S., Ali, T., Mahmood, K., Alsaedi, T., and Kundi, M. (2023). An intelligent framework based on deep learning for SMS and e-mail spam detection. *Applied Comput. Int. Soft Comp.* 2023:6648970.doi: 10.1155/2023/6648970

Marchal, S. (2014). *PhishStorm - Phishing / Legitimate URL Dataset.* Aalto: Aalto University.

Mewada, A., and Dewang, R. K. (2022). A comprehensive survey of various methods in opinion spam detection. *Multimedia Tools Appl.* 82, 13199–13239. doi: 10.1007/s11042-022-13702-5

Moghimi, M., and Varjani, A. Y. (2016). New rule-based phishing detection method. *Exp. Syst. Appl.* 53, 231–242. doi: 10.1016/j.eswa.2016.01.028

Mohammad, R. M., Thabtah, F., and McCluskey, L. (2014). Intelligent rule-based phishing websites classification. *IET Inf. Secur.* 8, 153–160. doi: 10.1049/iet-ifs.2013.0202

Morgan, S. (2020). *Cybercrime to cost the World $10.5 Trillion Annually by 2025. Cybercrime Magazine, November 13, 2020.* Available online at: https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/ (accessed May 26, 2023).

Mourtaji, Y., Bouhorma, M., Alghazzawi, D., Aldabbagh, G., and Alghamdi, A. (2021). Hybrid rule-based solution for phishing URL detection using convolutional neural network. *Wireless Commun. Mob. Comput.* 2021, 1–24. doi: 10.1155/2021/8241104

Nagaraj, K., Bhattacharjee, B., and Sridhar, A., and, G. S., S. (2018). Detection of phishing websites using a novel twofold ensemble model. *J. Syst. Inf. Technol.* 20, 321–357. doi: 10.1108/JSIT-09-2017-0074

OpenPhish-Phishing Intelligence (2023). Available online at: https://openphish.com/ (accessed May 31, 2023).

Orunsolu, A. A., Sodiya, A. S., and Akinwale, A. T. (2022). A predictive model for phishing detection. *J. King Saud Univ. Comput. Inf. Sci.* 34, 232–247. doi: 10.1016/j.jksuci.2019.12.005

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2828. doi: 10.48550/arXiv.1201.0490

Petrosyan, A. (2023). *Number of Internet and Social Media Users Worldwide as of January 2023.* Available online at: https://www.statista.com/statistics/617136/digital-population-worldwide/ (accessed June 27, 2023).

Phillips, K., Davidson, J. C., Farr, R. R., Burkhardt, C., Caneppele, S., Aiken, M. P., et al. (2022). Conceptualizing cybercrime: definitions, typologies and taxonomies. *Foren. Sci.* 2, 379–398. doi: 10.3390/forensicsci2020028

Prakash, P., Kumar, M., Kompella, R. R., and Gupta, M. (2010). "PhishNet: predictive blacklisting to detect phishing attacks," in *2010 Proceedings IEEE INFOCOM*.

Quinkert, F., Degeling, M., and Holz, T. (2021). Spotlight on phishing: a longitudinal study on phishing awareness trainings. *Det. Intr. Malware Vuln. Assessment* 341–360. doi: 10.1007/978-3-030-80825-9_17

Rao, R. S., and Pais, A. R. (2017). An enhanced blacklist method to detect phishing websites. *Inf. Syst. Secur.* 12, 323–333. doi: 10.1007/978-3-319-72598-7_20

Rao, R. S., and Pais, A. R. (2018). Detection of phishing websites using an efficient feature-based machine learning framework. *Neur. Comput. Appl.* 31, 3851–3873. doi: 10.1007/s00521-017-3305-0

Ribeiro, L., Guedes, I. S., and Cardoso, C. S. (2024). Which factors predict susceptibility to phishing? An empirical study. *Comput. Secur.* 136:103558. doi: 10.1016/j.cose.2023.103558

Saeed, U. (2022). Visual similarity-based phishing detection using deep learning. *J. Electr. Imag.* 31:1607. doi: 10.1117/1.JEI.31.5.051607

Safi, A., and Singh, S. (2023). A systematic literature review on phishing website detection techniques. *J. King Saud Univ. Comput. Inf. Sci.* 35, 590–611. doi: 10.1016/j.jksuci.2023.01.004

Sahingoz, O. K., Buber, E., Demir, O., and Diri, B. (2019). Machine learning based phishing detection from urls. *Exp. Syst. Appl.* 117, 345–357. doi: 10.1016/j.eswa.2018.09.029

Salihovic, I., Serdarevic, H., and Kevric, J. (2018). The role of feature selection in machine learning for detection of spam and phishing attacks. *Adv. Technol. Syst. Appl.* 3, 476–483. doi: 10.1007/978-3-030-02577-9_47

Sanchez-Paniagua, M., Fernandez, E. F., Alegre, E., and Al-Nabki, W., andGonzalez-Castro, V. (2022). Phishing URL detection: A real-case scenario through login urls. *IEEE Access* 10, 42949–42960. doi: 10.1109/ACCESS.2022.3168681

SatheeshKumar, M., Srinivasagan, K. G., and UnniKrishnan, G. (2022). A lightweight and proactive rule-based incremental construction approach to detect phishing scam. *Inf. Technol. Manage.* 23, 271–298. doi: 10.1007/s10799-021-00351-7

Sattari, Y., and Montazer, G. (2023). Intelligent methods in phishing website detection: a systematic literature review. *Research Square [Preprint]*. doi: 10.21203/rs.3.rs-2518632/v1

Singh, A. K. (2020). Malicious and benign webpages dataset. *Data Brief* 32:106304. doi: 10.1016/j.dib.2020.106304

Singh, S., Singh, M. P., and Pandey, R. (2020). "Phishing detection from urls using deep learning approach," in *2020 5th International Conference on Computing, Communication and Security (ICCCS)*.

Sonowal, G., and Kuppusamy, K. (2018). MMSPhiD: a phoneme based phishing verification model for persons with visual impairments. *Inf. Comput. Secur.* 26, 613–636. doi: 10.1108/ICS-12-2017-0091

Suleman, T. (2021). A survey on web phishing detection techniques. *Int. J. Electr. Crime Inv.* 5, 25–36. doi: 10.54692/ijeci.2021.050279

Tamal, M. (2023). Phishing Detection Dataset, Mendeley Data. V1. doi: 10.17632/6tm2d6sz7p.1

Tang, L., and Mahmoud, Q. H. (2021). A survey of machine learning-based solutions for phishing website detection. *Mach. Learn. Know. Extr.* 3, 672–694. doi: 10.3390/make3030034

The 2020 Official Annual Cybercrime Report (2020). Available online at: https://www.herjavecgroup.com/the-2019-official-annual-cybercrime-report/ (accessed May 23, 2023).

Urllib.parse- Parse URLs Into Components and Python Documentation (2023). Available online at: https://docs.python.org/3/library/urllib.parse.html (accessed November 12, 2023).

US-CER (2016). *IRS and US-CERT Caution Users: Prepare for Heightened Phishing Risk This Tax Season.* Available online at: https://www.us-cert.gov/ncas/tips/ST15-001 (accessed December 21, 2023).

Vayansky, I., and Kumar, S. (2018). Phishing – challenges and solutions. *Comput. Fraud Secur.ty* 2018, 15–20. doi: 10.1016/S1361-3723(18)30007-1

Vrbančič, G., Fister, I., and Podgorelec, V. (2020). Datasets for phishing websites detection. *Data Brief* 33:106438. doi: 10.1016/j.dib.2020.106438

Wu, X., Zheng, W., Xia, X., and Lo, D. (2021). Data quality matters: a case study on data label correctness for security bug report prediction. *IEEE Trans. Softw. Eng.* 48, 2541–2556. doi: 10.1109/TSE.2021.3063727

Yeoh, W., Huang, H., Lee, W. S., Al Jafari, F., and Mansson, R. (2021). Simulated phishing attack and embedded training campaign. *J. Comput. Inf. Syst.* 62, 802–821. doi: 10.1080/08874417.2021.1919941

Yuan, J., Liu, Y., and Yu, L. (2021). A novel approach for malicious URL detection based on the joint model. *Secur. Commun. Netw.* 2021:4917016.doi: 10.1155/2021/4917016

Zamir, A., Khan, H. U., Iqbal, T., Yousaf, N., Aslam, F., Anjum, A., et al. (2020). Phishing web site detection using diverse machine learning algorithms. *The Electr. Libr.* 38, 65–80. doi: 10.1108/EL-05-2019-0118

Zieni, R., Massari, L., and Calzarossa, M. C. (2023). Phishing or not phishing? A survey on the detection of phishing websites. *IEEE Access* 11, 18499–18519. doi: 10.1109/ACCESS.2023.3247135

Zouina, M., and Outtaj, B. (2017). A novel lightweight URL phishing detection system using SVM and similarity index. *Hum. Centr. Comput. Inf. Sci.* 7:981. doi: 10.1186/s13673-017-0098-1