Phishing Attack in Communication Networks is exposed using a Multi-Stage Machine Learning Approach

Ademola Philip Abidoye[1] and **Boniface Kabaso**[2]

ABSTRACT: Phishing is a cyber-attack that uses disguised email as a weapon and has been on the rise in recent times. If innocent Internet users click on a fraudulent link, it may cause them to fall victim divulging personal information such as credit card pin, login credentials, banking information, and other sensitive information. There are many ways in which attackers can trick victims revealing their personal information. In this article, we select important phishing URL features that can be used by an attacker to trick Internet users into taking the attacker's desired action. We use two machine learning techniques to accurately classify our datasets. We compare the performance of other related techniques with our scheme. The results of the experiments show that the approach is highly effective in detecting phishing URLs and attained an accuracy of 96.3% with a 17.2% false-positive rate, a 23.7% false-negative rate, and an error rate of 3.70%. The proposed scheme performs better compared to other selected related work. This shows that our approach can be used for real-time applications in detecting phishing URLs.

Keywords: Phishing Attack, Datasets, Phishing URL, URL Classification, Attackers, Machine Learning, Classifiers, Internet

DOI: 10.37936/ecti-cit.2021153.240565

Article history: received April 27, 2020; revised November 12, 2020; accepted February 26, 2021; available online November 24, 2021

1. INTRODUCTION

In the last decade, Internet usage has been increasing tremendously and it makes our lives easy, simple, and transform our daily lives. It plays a major role in the areas of communication, education, business activities, and commerce [1, 2]. A lot of useful data, information, and knowledge can be obtained from the Internet for personal, organizational, economic, and social development. Positive and productive use of the Internet will assist users to become successful in their careers and businesses. The Internet makes it easy to provide many services online and enables us to access various information at any time, from anywhere around the world. Examples include online banking, transferring money between accounts, online bill paying, and so on. These services have become very prevalent as more financial institutions start to provide almost free online services. Presently, about 40% of the world population is connected to the Internet [3]. The main purpose of the Internet is to provide worldwide access to various types of data for advancing research in engineering, science, design, and medicine as well as in maintaining global defense and surveillance [4]. However, as more people are using the Internet globally, different kinds of attacks have been identified including denial-of-service and distributed denial of service attacks, drive-by attacks, man-in-the-middle attacks, password attacks, eavesdropping attacks, and phishing attacks [5]. Over the last decade, phishing has skyrocketed to staggering proportions and will continue to increase due to the various phishing groups which use different methods of attack. Therefore, it is imperative to comprehensively study the mode of operation of attackers. The word *phishing* was coined from the fact that cyber-attackers are fishing for sensitive data and information. The "ph" comes from the advanced methods the phishers employ to distinguish their activities from the more simplistic *fishing*. The concept of phishing is a form of social engineering and can be traced back to the early 1990s via America Online (AOL) [6].

Phishing is the act of sending fake email, mes-

[1,2]The authors are with Department of Information Technology, Cape Peninsula University of Technology, Cape Town, South Africa., E-mail: abidoyea@cput.ac.za and kabasob@cput.ac.za

sages, or building malicious websites to trick the recipient/Internet users into divulging sensitive personal information such as the personal identification number (PIN) and password of their bank account, credit card information, birthdates, and social security numbers. To perpetuate this type of attack, the attacker usually poses as a trustworthy organization. For instance, an attacker may send an email that looks like it is from a financial institution or reliable credit card company requesting their account information by tricking the target by claiming there is a problem or a need to update his/her data within a stipulated time. There were 112163 unique phishing attacks and 60889 unique phishing sites reported in the U.S. in June 2019 [7]. Phishing attacks affect hundreds of thousands of Internet users across the globe. Individuals and organizations have lost a huge sum of money and private information through phishing attacks [8].

What differentiates phishing from other Internet attacks is the form the message takes: the attackers are disguised themselves as a real person, a trusted entity of some kind, or an organization the target might transact business with. It is one of the fastest-growing types of cyber-attacks and is widespread due to the financial gain the attackers derive from any successful phishing. The attackers capitalize on some recipients' desire to respond to urgent requests from their "financial institutions" by clicking a link or downloading an attachment provided in the spoofed email that looks "official", but it is linked to fraudulent website(s) which may result in financial losses, identity theft, or other fraudulent activity.

1.1 Statistics of Phishing Attacks

The sudden attack of phishing against financial institutions was first known in July 2003. Since then, commercial banks, E-gold, and E-loan companies are the main target of the phishers. Among financial institutions that have been attacked in the U.S., commercial banks account for 91 percent of the attacks while insurance companies account for 7 percent. Similarly, about 39 percent of the total retail banking activities and 25 percent of the credit card services were the main business lines that have been attacked in 2018 [9].

The number of global phishing attacks rose to 129.9 million during the second quarter of 2019. It increased by 21% more than the same quarter of 2018. Greece has the highest number of phishing attacks at 26.2%, followed by Venezuela, Brazil, Australia, and Portugal. In terms of financial institutions and establishments, commercial banks had the highest percentage of phishing emails at 30.7%, followed by payment systems at 20.1%, worldwide Internet portals at 18%, and social networks at 9% [10]. The act of phishing is not limited to a particular country. It occurs everywhere and every day. This is so because phishers are using the Internet to phish unsuspecting Internet users for financial gain [11]. Phishing information flow is shown in Fig. 1.

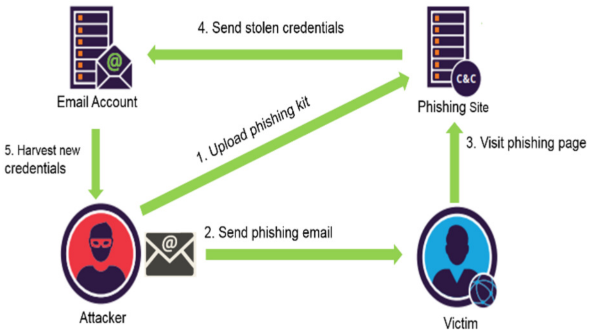


Fig.1: *Phishing information flow [12].*

Phishers continually look for more effective and advanced ways to launch phishing attacks. They are constantly developing new techniques of attack and improving old ones. Thus, with the advancement in technology, they have refined their attacks both in the usage of websites and emails. They can develop more innovative and effective methods of targeting innocent victims. It is essential to note that different phishers have various methods they use for phishing, but all have similar techniques and tools. These methods are majorly grouped into three types: impersonation, forwarding, and popups [13].

In recent years, researchers and stakeholders have paid much attention to the problem of phishing and how it could be solved. They have developed different approaches in the literature for detecting malicious uniform resource locators (URLs) and emails. Some of these approaches are presented below.

Blacklisting and whitelisting are two widely used security methods that have been deployed to manage which entities get access to our system.

A blacklist is a list of suspicious or forbidden URLs that should be blocked or denied access to a network or system. This method is very simple to implement. It is based on identifying the known and suspected URLs and denying them access to the network. However, this method is too weak to detect the majority of phishing incidents since new threats are many and constantly emerge every day, including zero-day attacks. With this approach it is impossible to detect or stop any new kind of attack. It requires keeping a comprehensive list of suspicious websites and their reports, which consume a lot of system resources [14]. Phishers sometimes design URLs specifically to evade detection by tools that use a blacklist system. Finally, this approach will fail to identify an attack that targets a particular user such as a profitable organization.

On the other hand, a whitelist allows some list of websites to be accessed and blocks other websites that are not on the list. It denies any new URL unless it is proven to be benign (legitimate). Whitelist applications can be used to identify websites by their file name, size, and directory path. Thus, whitelisting access control is more effective than blacklisting, as the default is to block websites and only let in those that are proven to be legitimate. However, its implementation is more complex and hard to assign because it requires more information on the application being used to create the whitelist. In addition, it is infeasible to create a whitelist that contains the list of all legitimate sites due to their large number [15]. Another challenge of the whitelisting approach is that a user must remember to check the interface each time he surfs any site. Thus, there is a need to develop innovative methods that are capable of detecting any recent methods the phishers are using for phishing.

1.2 Aim of Research

This work aims to develop a technique that can detect all forms of phishing strategies created by attackers in communication networks. We generate our set of rules which rely on our observations and machine learning techniques. We gather different methods and tricks used by attackers to entice unsuspecting victims to fabricated web pages and use those attributes to design our rule datasets.

1.3 The Significance of the Study

In recent times, there is an increasing need to identify phishing URLs and emails because of the negative effect they have on their targets. Researchers have developed various methods and applications for exposing phishing websites and detecting malicious emails, but only a few scholars have used machine learning methods to detect phishing websites. In this study, we are using a multi-stage machine learning technique to detect and classify the datasets obtained into phishing URLs and benign URLs with a minimum false positive rate. This approach provides up-to-date protection against zero-day phishing attacks.

1.4 Problem Statement

Phishing detection methods do suffer from detection in accuracy and have high positive false alarm rates, particularly when new phishing techniques are invented. In addition, a blacklist is a common method for phishing URLs detection but it is ineffective in responding to new phishing attacks. Since it is now very easy to register a new domain, no comprehensive blacklists can ensure an adequate up-to-date database.

Researchers have developed various approaches to detect phishing websites using different learning algorithms but this problem still needs more attention by researchers because new phishing websites are being deployed every day and phishers are using different techniques to carry out their attacks. Most of the solutions provided for phishing attacks were based on a small experimental dataset. The accuracy and effectiveness of these algorithms on real large datasets cannot be ascertained. Thus, the number of malicious websites increases very fast. How to detect phishing websites from a large number of legitimate websites in real-time with high accuracy must also be addressed. It is imperative to design intelligent anti-phishing algorithms that are capable of detecting the ever-increasing phishing attacks. We use both Support Vector Machine (SVM) and Naïve Bayes classifiers to classify the datasets since no single classifier is perfect. SVM scales relatively well to high dimensional data and error can be explicitly controlled. In addition, it is very easy to implement. However, it does not scale very well for a large dataset. A naïve Bayes classifier is used to overcome the weakness in SVM. This classifier is capable to handle large datasets and scales linearly with the number of predictors and data points.

1.5 Contributions

This research work uses a multi-stage machine learning technique to accurately classify our datasets into either phishing or benign URLs in communication networks. These two classifiers are used together because strengths in one classifier complement the weaknesses in the other classifier. We use 30 features to model our classifiers to achieve high precision and to provide a better accuracy trade-off. We observe that using these features increases the overall classification success rate across all the datasets and minimizes the error rate. This shows that the proposed approach can be used for near real-time applications in detecting phishing URLs.

The rest of this article is organized as follows. In section 2 related work is discussed. Section 3 discusses the proposed approach. Data used for the experiments, relevant features in predicting phishing URLs, and the classifiers used are discussed in this section. In section 4, we present the various experiments conducted and also discuss the performance evaluation of the two machine learning techniques used. Finally, the conclusion is presented in section 5.

2. RELATED WORK

The recent increase in suspicious URLs has attracted the attention of many researchers and they have developed different techniques for website phishing detection. The definition of phishing is constantly adjusting to the way phishing is performed. Email and websites are the two major methods phishers used for phishing. These two methods have the same goals but there are some differences between the two.

Aburrous et al. [16] proposed an intelligent system for phishing webpage detection in e-banking. They developed a model that combines fuzzy logic with a data mining algorithm to detect phishing websites and categorize the phishing type using 10-fold cross-validation. This model achieved 86.38% grouping accuracy. However, this model has a high percentage of false positives.

Basnet et al. [17] proposed a heuristic-based approach to group phishing URLs by using the data available only on URLs. The authors used a binary classification method to detect phishing URLs and grouped URLs into phishing URLs and legitimate URLs. The results of the experiments show that the proposed approach is very effective in detecting phishing URLs compared to related work. However, this approach is only tested on a dataset that has less than 300 URLs. It may not be effective on a large dataset.

Jain and Richariya [18] developed a new method for detecting phishing emails using link-based features. A prototype web browser was used as a means to process each incoming email to detect a phishing attack. A combination of the prototype and their algorithm let the system users be notified of possible attacks and prevents them from clicking any malicious URLs.

Mahmood and Rajamani [19] proposed an anti-phishing detector (APD) technique based on association rule mining for detecting phishing websites. APD dynamically traces out any possible phishing attacks during message transmission between computer users. Also, the authors developed an algorithm to extract frequently occurring words and forward the information to APD for further processing. The results of the approach have been shown to be effective.

Ajlouni et al [20] proposed a method for detecting phishing websites based on associative classification algorithms. It is an improvement over [16]. The results of the experiment show that the method achieved 98.5% accuracy in detecting phishing webpages. However, there is no information about how many rules they used for the extraction.

Zhang et al [21] proposed a new classification method based on the Sequential Minimal Optimization classifier algorithm that consists of features of websites. The results show that the algorithm performs better than the selected baseline. However, this approach can only detect phishing web pages in the Chinese language.

A new rule-based approach for detecting phishing attacks in Internet banking is presented in [22]. The authors used two feature sets that have been developed to find webpage identity and a support vector machine algorithm to classify webpages. The proposed features are independent web browser history or search engine results. The results of the experiments show that the method can detect phishing webpages with an accuracy of 99.14% true positive and only 0.86

Ramesh et al. [23] developed a method for detecting phishing web pages. The webpage is scrutinized and classifies all the indirect and direct links associated with the page. Indirect link features are extracted from the search engine result while direct links are extracted from the page contents. In addition, they used a third-party DNS lookup to match the domains of the malicious webpage and phishing target to the corresponding IP address. The results of this approach achieve 99.62% accuracy. However, the efficiency of this method depends largely on the speed of the search engine and DNS lookup time which can affect its performance.

Kumar and Gupta [24] presented a new approach that can expose phishing attacks using hyperlinks information found in the source code of various websites. The proposed approach combines different unique hyperlink-specific features to detect a phishing attack. The dataset obtained contained 12 different features which are used to train the machine learning algorithms. The authors used various classification algorithms to classify the datasets into phishing and non-phishing websites. The proposed approach achieved high accuracy in the detection of phishing websites.

An efficient approach for phishing detection using Machine Learning is proposed in [25]. The authors shortlist a set of features using a feature selection technique so that high-performance classification models can be designed in less time. They conducted experiments on a phishing dataset containing 11,055 with 30 features. Several machine learning algorithms are used for obtaining accurate results and also to improve the build time of classification models for phishing detection without compromising their accuracy.

A comparison of related work that has been used to detect phishing URLs in the literature with our work is presented in Table 1.

Table 1: *Classification results obtained by classifiers.*

Work	Technique	A	B	C	D
[17]	Binary Clarification	Yes	No	Yes	Yes
[21]	Sequential Minimal Optimization	No	No	Yes	Yes
[22]	Rule-based	Yes	Yes	Yes	No
[23]	Domain Identification	Yes	No	Yes	No
[24]	Logistic Regression	Yes	Yes	No	Yes
[25]	Feature Selection	Yes	Yes	Yes	No
	Proposed approach	Yes	Yes	Yes	Yes

Where A = Zero-day phishing detection,
B = 3rd-party service sovereignty,
C = Search engine sovereignty,
D = Language sovereignty

3. PROPOSED APPROACH

In this section, we present in detail our method for detecting malicious URLs. The approach is divided into two parts. The output of the first part is an input to the second part, as shown in the proposed framework in Figure 2.

The first part is based on data collection which needs data description, processing of datasets, and URL feature extraction. We consider different heuristic features in the structure of URLs, ranging from generic social engineering features, lexical features in the URL, multiple alphabets, and phishing target brand names. The feature vector is constructed with the features listed in Table 3 to model our classifiers. The second part is based on the classification of datasets using machine learning classifiers to evaluate our approach. We performed different experiments and the results show that our scheme achieves 96.3% accuracy on average. The description of each part is briefly discussed in the following subsections.

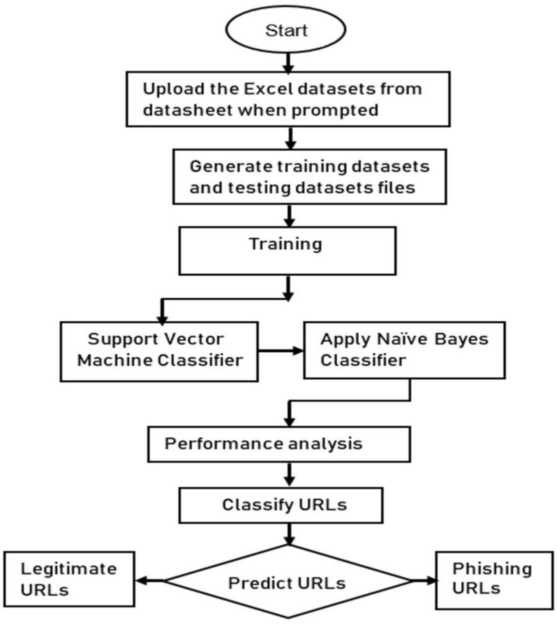

Start
Upload the Excel datasets from datasheet when prompted
Generate training datasets and testing datasets files
Training
Support Vector Machine Classifier
Apply Naïve Bayes Classifier
Performance analysis
Classify URLs
Legitimate URLs
Predict URLs
Phishing URLs

Fig.2: *Proposed framework for detecting phishing URLs.*

3.1 Processing of datasets and URL feature extraction

For our large dataset, 36,874 URLs were collected. Details are presented in sub-section 3.4. The datasets consist of three types of URLs: phishy URLs, suspicious URLs, and legitimate URLs. The phishy URLs are designed with the main aim to trick innocent Internet users into revealing their confidential information, which may lead to financial loss. Suspicious URLs are considered phishy and could have links, malicious codes, and/or pictures attached to them. Legitimate URLs are web pages with real source code which do not contain any malicious code. These datasets are processed to make them suitable for this study. The processing involved many stages. These include webpage feature extraction, data standardization, and attribute weighing. These steps are very important so that the classifiers can understand the datasets and appropriately categorize them into their respective classes. The classifier is regularly retrained with new phishing web pages to learn new trends in phishing. The outcome of this phase is used as input to the next part of the appropriate classifiers.

3.2 Assessment of Classifiers

The assessment of classifiers is needed in this research to determine the performance achieved by the proposed method. To do this, a set of tests consisting of datasets with known tags is used. Each individual classifier has a training dataset and MySql is used as a database to store our datasets. Thereafter, we compared their performances with related work.

We use both SVM and Naïve Bayes algorithms to create models from training datasets which consist of extracted features and class labels to effectively classify the phishing URLs based on the information available to individual URLs. Phishing URLs are treated as a binary classification problem with benign URLs belonging to the negative class and phishing URLs belonging to the positive class. We collected our phishing and benign URLs from PhishTank, Yahoo directory, and Google engine to form our datasets. Next, we extract several features that have proved to be effective in predicting phishing URLs by employing different publicly available resources to classify the datasets into their respective classes [26, 27].

The datasets contain 36,874 URLs with their related features. We wrote Python scripts to automatically download certified phishing URLs from PhishTank.

3.3 Phishing datasets

PhishTank is a joint project to which people can submit suspicious phishing URLs for confirmation. It is a public clearinghouse for phishing URLs [28]. Suspicious URLs are further scrutinized by many people before being confirmed as phishing URLs and added to a blacklist. PhishTank provides a comprehensive list of current and active phishing URLs.

Researchers and developers can download phishing URLs from the Phish-Tank after signing up. The URLs can be downloaded in different file formats with an API key.

The datasets were collected for several months to collect a range of datasets. The first dataset is referred to as DTS1 and it contains 14,298 phishing URLs. They were collected from March 4, 2019, to April 19, 2019, based on the reports in [29] which show that phishing attacks were usually higher during this period compared to the preceding months. In addition, we observe that phishers constantly develop new tactics to get personal information from unsuspecting users. Exploring various recent methods, the attackers are using motivated us to collect the second set of data which is referred to as DTS2. It contains 7,350 phishing URLs. They were collected from November 1 to December 4, 2019. We chose this period because of a special day in this period called “Black Friday” (November 29, 2019). On this day, many people are eager to buy cheap goods from stores online using their credit or debit cards. Phishers also use this period as an opportunity to display their tactics and launch different attacks on unsuspecting users. A total of 21,648 phishing URLs were collected from the PhishTank Website.

3.4 Legitimate datasets

Our benign URLs were collected from the Yahoo directory. Yahoo provides a generator that arbitrarily produces a URL in its directory each time the Web page is visited. This service is used to randomly choose a URL and download the contents of the Web page with the server header information. This service was used to collect 9,045 random URLs from May 6, 2019, to June 10, 2019. The list consists of URLs from financial institutions, e-commerce, online services, cloud storage, second religious organizations in order to get different URL structures and Web page contents [30]. To provide more learning instances for legitimate URLs, we chose 6,181 legitimate URLs from the Open Directory Project (DMOZ) Web directory [31]. DMOZ is a multilingual open-content directory of World Wide Web links containing more than three million URLs.

Google tool is used to analyze the list of benign URLs collected and crawled the URLs. These URLs are legitimate web pages, based on the assumption that all the URLs extracted were benign since they were downloaded from legitimate Internet sources.

Python and Java scripts were used to parse the legitimate and phishing URLs and extract the features discussed in subsection 3.2. Web pages that we could not extract the features from were discarded to get only valid URLs for our datasets. The total number of URLs in our datasets is presented in Table 2. The percentages for phishing and non-phishing datasets are computed as shown in the table.

Table 2: *Datasets for Phishing URLs Detection.*

Dataset	Phishing	Non-phishing	Total datasets	Percentage
DTS1	14,298	9,045	23,343	63.3%
DTS2	7,350	6,181	13,531	36.7%
DTS1 + DTS2	21,648	15,226	36,874	100%
Percentage	58.7%	41.3%	100%	

3.5 Data Authentication

Datasets collected need to be authenticated to ascertain the real status of the URLs, particularly in the case of phishing websites as it is known that phishing websites only last a few weeks [32]. Thus, every URL needs to be authenticated before processing.

In this section, we present some of the features that are effective in predicting phishing websites. All the features used for this work are listed in Table 3. We computed the mean and the standard deviation for each feature as shown in the table. Table 4 presents the percentage of occurrences of each feature in the dataset in a pie chart.

Generic salutation

Phishers use generic greetings in their messages such as “Sir”, “Dear Bank Customer”, “Dear Customer”, and “Dear Member” to address their target victims. The content of the message is always threatening such as “please update your bank account to prevent it from being blocked”, “Your account has been compromised!”, “Urgent action required!”, “Your account will be closed!” These intimidation strategies are becoming more common than the promise of “instant riches”. Taking advantage of victims’ anxiety and concern to get them to provide their personal information is the modern popular strategy.

$$Rule \begin{cases} \textbf{if } \textit{the greeting is directed to the} \\ \quad \textit{account owner and} \\ \quad \textit{do not require supplying} \\ \quad \textit{personal information via a link in the} \\ \qquad \textit{message} \rightarrow \textit{Legitimate} \\ \textbf{else if } \textit{the greeting is generic} \rightarrow \\ \qquad \textit{Suspicious} \\ \textbf{else } \textit{update your information} \\ \qquad \textit{via a given link} \rightarrow \textit{Phishing} \end{cases}$$

Lexical features explain lexical patterns of phishing URLs such as long IP addresses, special characters, number of dots, and so on.

IP-based URL

An Internet Protocol (IP) address is one of the ways to hide the webpage address. If an IP address is used instead of a Domain Name System (DNS) address in the URL, it will be difficult for innocent users to ascertain where they are being directed to when they click the link or press the Enter key on their system to load the page. Another reason for using an IP address is that phishers would not like to spend money to buy a domain for their phony web pages.

$$Rule: \begin{cases} \textbf{If } \textit{the domain name has an IP Address} \\ \qquad \rightarrow \textit{Phishing} \\ \textbf{else} \rightarrow \textit{Legitimate} \end{cases}$$

Long URL to hide the fake part

Attackers can use lengthy URLs to mask the fake part in the address bar. For instance,

"http://prudentbank.com/2k/ab51e2e319e51502f416dbe46b773a5e/?cmd=_home&amp;dispatch=11004d58f5b74f8dc1e7c2e8dd4105e811004d58f5b74f8dc1e7c2e8dd4105e8@phishing.net.html"

We computed the length of URLs in our datasets and determined their average length to ensure the accuracy of our research. The findings showed that if the URL length is less than 52 characters, it is classified as legitimate. It is suspicious if the length is between 52 and 73 characters. It is a phishing URL if it is more than 73 characters. A method based on frequency has been used to update this feature rule, which improves its accuracy.

$$Rule{:}\begin{cases}\textbf{If } URL\ length < 52\ characters \rightarrow Legitimate\\ \textbf{else if } URL\ length \geq 52\ and\ \leq\ 73\ characters\\ \quad \rightarrow\ Suspicious\\ \textbf{else} \rightarrow Phishing\end{cases}$$

Shortened URL "TinyURL"

A short URL allows reducing a long link from social networks and top sites on the Internet. This is achieved by the service provider through an "HTTP Redirect" on a domain name that is short and redirects to the corresponding long URL [33]. For instance, a URL for Wiki's article "`http://en.wikipedia.org/wiki/URL_shortening`" contains 64 characters, and its corresponding short URL is http://bit.ly/c1htE. It contains 16 characters with Bitly's default domain name "bit.ly" and the hash "c1htE" as the back-half. A hash only consists of letters and numbers "a-z, A-Z,0- 9". Attackers use this shortened URL feature to hide links to infected websites or phishing.

$$Rule:\begin{cases}\textbf{if } TinyURL \rightarrow Phishing\\ \textbf{else} \rightarrow Legitimate\end{cases}$$

URL's having "@" Symbol Using an "@" symbol within the URL causes the Web browser to read the right side of the browser address and ignore everything preceding the "@" symbol. For instance, in this URL www.prudentbank.com@www.google.com, the browser will ignore "`www.prudentbank.com`" and only read www.google.com. This technique may be used to hide a phishing URL.

$$Rule:\begin{cases}\textbf{if } URL\ having\ @\ symbol \rightarrow Phishing\\ \textbf{else} \rightarrow Legitimate\end{cases}$$

Hovering of a Mouse over Hyperlink Feature

One of the tactics of phishers is that they use legitimate domain names for their links to send messages to their potential victims while the destination URLs are hidden from them using HTML code. For instance, a phisher may send this link $<$ a href = "http://phishing.com">www.prudentbank.com$<$ / a $>$ to unsuspecting Internet users which looks like a Prudent Bank Website. The destination URL "http://phishing.com" is hidden from the user. If the user clicks the link "www.prudentbank.com" it will take him to "http://phishing.com" thinking that they are surfing a

$$Rule{:}\begin{cases}\textbf{if } destination\ URL\ is\ the\ same\ as\ the\ domain\\ \quad name\ and\ the\ link\ leads\ to\ the\\ \quad homepage\ \rightarrow Legitimate\\ \textbf{else if } the\ destination\ URL\ cannot\ be\\ \quad determined \rightarrow Suspicious\\ \textbf{else } the\ destination\ URL\ does\ not\ the\ same\\ \quad as\ the\ domain\ name\ \rightarrow\ Phishing\end{cases}$$

Redirecting using "//" The presence of "//" in the URL path shows that an innocent user will be redirected to another infected website. For example, `http://www.legitimate.com-` //`http://www.phishing.com` This study examines the position of "//" in a legitimate URL. If the URL begins with "HTTP" then "//" should appear in the 6th position or in the 7th position if it begins with "HTTPS".

$$Rule:\begin{cases}\textbf{if } the\ position\ of \text{``}//\text{''}\ in\ the\ URL\ >\ 7\\ \rightarrow Phishing\\ \textbf{else} \rightarrow Legitimate\end{cases}$$

Domain name separated by a dash symbol

It is very rare for a legitimate domain name to be separated by a dash symbol (-). Phishers use this method to trick Internet users by adding a dash symbol (-) within the domain name so that users will think that they are surfing a legitimate webpage. For instance, http://www.pay-pal.com/.

$$Rule:\begin{cases}\textbf{if } Dash\ symbol\ (-)\ is\ part\ of\ a\ domain\\ \quad name \rightarrow Phishing\\ \textbf{else} \rightarrow Legitimate\end{cases}$$

Subdomain of a subdomain

A URL might include an Internet country code top-level domain (ccTLD) to identify a particular country. For instance, http://www.prudentbank.com.za/login/. "za" is a ccTLD, and the ".com" por-tion of the extension shows that the domain name is a commercial entity. Taking the two extensions together ". com.za" is called a second-level domain (2LD) and "prudent bank" is the real domain name. To minimize rules for extracting this feature, first, we remove the subdomain "www" from the URL and ccTLD if the extension is part of the URL. Thereafter, the number of dots in the URL is counted. If the number of dots is one, then the URL is legitimate. It is suspicious if the number of dots is two since the URL has one subdomain. It is declared phishing if the number of dots is more than two since it will contain many subdomains.

$$Rule : \begin{cases} \textbf{if } the\ number\ of\ dots\ in\ domain \\ \quad portion = 1 \rightarrow Legitimate \\ \textbf{else if } dots\ in\ domain\ portion = 2 \\ \rightarrow Suspicious \\ \textbf{else} \rightarrow Phishing \end{cases}$$

A domain name containing multiple alphabets
It is possible to register domain names in other alphabets such as Chinese, Arabic, French, German, or anything that can be represented with the Unicode standard since 1998. Phishers have taken advantage of this unique feature by finding characters in other alphabets that look similar to Latin ones to lure users into a phishing website. For instance, in this URL "HTTPS://apple.com", the domain name can be regis-tered with "xn–pple-43d.com". The URL is equiva-lent to "`HTTPS://xn--pple-43d.com`". Thus, most users will fall for this trick because their browsers will show the green padlock icon, showing that the user is on a secure connection but in fact, a bunch of Cyrillic characters is embedded within the multiple alphabets.

$$Rule : \begin{cases} \textbf{if } domain\ name\ containing\ multiple \\ \quad alphabets \rightarrow Phishing \\ \textbf{else} \rightarrow Legitimate \end{cases}$$

Phishing website longevity
We believe that legitimate websites will be hosted and regularly paid for one or more years in advance. It has been shown that a phishing website exists for a short period to avoid being detected [34]. In our datasets, the longest fake domains that have been used are only for six months.

$$Rule : \begin{cases} \textbf{if } domains\ expire \leq sixmonths \\ \rightarrow Phishing \\ \textbf{else} \rightarrow Legitimate \end{cases}$$

Presence of "HTTPS" Token in the Domain Part of the URL
The phishers may add the "HTTPS" token to the domain part of a URL to trick innocent Internet users. For instance, `http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/`

$$Rule : \begin{cases} \textbf{if } using\ HTTP\ token\ in\ the\ domain \\ \quad part\ of\ the\ URL \rightarrow Phishing \\ \textbf{else} \rightarrow Legitimate \end{cases}$$

Abnormal URL
This feature can be extracted from the WHOIS database. The identity is typically part of its URL for a legitimate website.

$$Rule : \begin{cases} \textbf{if } the\ hostname\ Is\ not\ included\ In\ URL \\ \rightarrow Phishing \\ \textbf{else} \rightarrow Legitimate \end{cases}$$

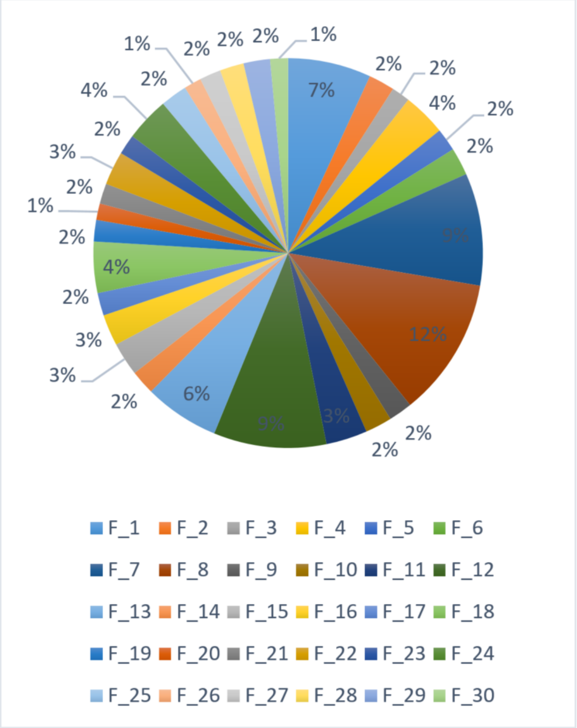
All the features, the number of appearances, and the percentages are presented in Table 3.

Table 3: *Table of Training data consisting of 30 features.*

S/N	Features	Mean	Standard Deviation
F_1	Generic salutation	0.2846	0.6169
F_2	IP-based URL	0.7950	0.7341
F_3	Long URL to hide the fake part	0.1564	0.4349
F_4	Shortened URL	0.1321	0.7919
F_5	URL's having "@" Symbol	0.7981	0.8311
F_6	Hovering a mouse over the hyperlink feature	0.1545	0.5315
F_7	Redirect pages	0.2118	0.8889
F_8	Domain name separated by a dash symbol	0.2556	0.3269
F_9	Subdomain of a subdomain	0. 5283	0.5011
F_10	Domain name having multiple alphabets	0.4816	0.4131
F_11	Phishing website longevity	0.1864	0.3651
F_12	Presence of "HTTPS" Token in the Domain Part of the URL	0.0659	0.5701
F_13	Anomalous Request URL	0.1502	0.2729
F_14	Using forms with the 'Submit' button	0.1742	0.5129
F_15	Spelling errors	0.1224	0.6251
F_16	Copying Website	0.0956	0.3731
F_17	Anomalous cookie	0.0694	0.5351
F_18	Website Traffic	0.1067	0.7621
F_19	Using Non-Standard Port	0.2158	0.9289
F_20	URL of Anchor	0.7502	0.7271
F_21	Disabling right-click button	0.1621	0.3919

F_22	Adding Prefix or Suffix	0.3627	0.6021
F_23	Status Bar Customization	0.1216	0.5131
F_24	Age of Domain	0.0935	0.6941
F_25	Google Index	0.1719	0.6899
F_26	Server Form Handler (SFH)	0.4532	0.6991
F_27	Number of Links Pointing to Page	0.3651	0.5781
F_28	Using Hexadecimal Character Codes	0.6435	0.7941
F_29	Replacing Similar Characters for URL	0.1162	0.4671
F_30	Using the pop-up window	0.1523	0.8939

Table 4: *Percentage of Each Feature in The Dataset.*



4. DETECTION OF PHISHING URLS

A feature vector matrix is built from the datasets presented in Table 2. Each vector-matrix consists of 30 lexical features described in Table 3. We use two variables to classify the datasets: -1 for a legitimate URL and 1 for a phishing URL as shown in equation (1). This gives a feature matrix vector of 36,874 rows denoting the total size of the dataset.

There are many machine learning classification algorithms. We classified our datasets using the following classification algorithms. Metrics for classification are discussed thereafter.

4.1 Training of datasets

The dataset that is presented in Table 2 is stored on our local computer in a CSV (a common delimited) file. The data features used were those presented in Table 3. These features were selected to different websites as phishy or legitimate. These two tags were created and assigned the following values -1 and 1. Phishy is denoted by -1 and 1 means legitimate. Thereafter, the dataset is divided into a training dataset that the classifiers can use to make a prediction and a testing dataset that we can use to evaluate the accuracy of the model. The ratio of the training dataset to the test dataset is 70:30. Scikit-Learn, a library in Python, is used to implement both the SVM and Naïve Bayes algorithms. We trained the classifiers and in each feature. The ordinate (Y-axis) shows the number of occurrences for each behavior. The abscissa (X-axis) shows the behavior of the feature as either phishy or legitimate.

4.2 Support Vector Machines (SVMs) Classifiers

In any classification process, both a parameter and a model technique should be chosen to achieve a high level of performance. Recent methods enable different kinds of models of varying complexity to be selected.

This study uses a linear classifier of the form: $f(X_i) = W.X_i + b$ where. represents the dot product, W denotes the weight vector, X_i is the input data we want to classify, and b is the linear coefficient estimated from the training data.

Let $\{X_i\}$ denote the features of our datasets for all $i = 1, 2, 3, \ldots, n$, $X_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ denote class labels (indicator variable). Our goal is to classify the datasets correctly. The following mathematical equations need to be satisfied to achieve this goal as shown in equation (1). SVM dataset classification is shown in Algorithm 1.

$$\begin{gathered} f(X_i) = \begin{cases} \geq \ 0 \ y_i = +1 \\ < \ 0 \ y_i = -1 \end{cases} \\ W.X_i + b \geq 1 \\ W.X_i + b < 1 \\ y_i(W.X_i + b) \geq 1, \text{for all } i \end{gathered} \tag{1}$$

Algorithm 1: SVM Data Classification

```
Begin
Given a hyperplane $W.X + b$
$f(X_i) = W.X_i + b$ for all $i = 1,2,3, \dots \dots n$
The classifier can be expressed as
$f(X_i) = \widetilde{W}.\widetilde{X_i} + w_o = W.X_i$
where $W = (\widetilde{W}, w_o)$, $X_i = (\widetilde{\widetilde{X_1}}, 1)$
Let $W = 0$
Considering the datasets and class labels,
 $\{X_i, y_i\}$
 $f(X_i) = sign\,(\sum w[i]\,x[i] + b)$
if $X_i$ is wrongly classified then
    $W \leftarrow W + \beta * sign(W.X_i + b)$
else
    Continue until all the datasets are correctly classified
end if
End
```

4.3 Naïve Bayes Classifiers

Naive Bayes classifiers are a group of classification algorithms based on Bayes' Theorem. The underlying assumption of these classifiers is that all the features used for the classification are autonomous of each other. In other words, it assumes that the existence of a specific feature in a dataset is unrelated to the existence of any other feature. The Bayes classifier can consider all the features of datasets and correctly classify them. It provides a way of determining posterior probability $P_r(yX_i)$ from $P_r(X_i)$, $P_r(y)$, and $P_r(X_iy)$ as shown in equation (2).

Figure (3) shows the process of experimenting before arriving at our results.

$$P_r(y|X_i) = \frac{P_r(X_i|y) * P_r(y)}{P_r(X_i)} \tag{2}$$

$P_r(y|X_i)$ is defined as the posterior probability of class (legitimate or phishing URL) given the predictor (feature).

$P_r(X_i)$ is the probability of a predictor.
$P_r(y)$ is the probability of the class.
$P_r(X_iy)$ is the probability of the predictor given class.

The variable $y = y_k$ denotes the class defined above and variable X_i denotes the features of our datasets such that

$$X_i = (X_1, X_2, X_3, \dots, X_n)$$

Substituting for X_i in equation (3) and expanding using the chain rule gives Equation 3.

$$P_r(y|X_1, X_2, \dots, X_n) = \frac{P_r(X_1|y)P_r(X_1|y) \dots P_r(X_n|y)P_r(y)}{P_r(X_1)P_r(X_2) \dots P_{r(X_n)}} \tag{3}$$

The value of the denominator remains static for all values in our dataset. Thus, the denominator is eliminated and proportionality is introduced as seen in Equation 4.

$$P_r(y|X_1, X_2, \dots, X_n) \propto P_r(y) \prod_{i=1}^{n} P_r(X_i|y) \tag{4}$$

The function is further used to classify our datasets, X_i, into two classes: legitimate or phishing URLs.

4.4 Metrics used for Evaluation

The following metrics are used for the evaluation of the proposed scheme in order to eliminate or minimize misclassification in our datasets. We assume that a legitimate website is negative (N) and a phishing website is positive (P). The metrics we used are defined next.

The confusion matrix is defined as the total number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) recognized by the classifiers.

We express P = TP + FN and N = TN + FP. The prediction outcomes are summarized in Table 5.

Accuracy is defined as the proportion of instances that are classified correctly versus the total number of instances. The rate is mathematically expressed in Equation 5.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

True positive is phishing classified as phishing. It is defined as the proportion of legitimate websites that are correctly classified as legitimate. The rate is mathematically expressed in Equation 6.

$$TP_{rate} = \frac{TP}{TP + FN} \tag{6}$$

False-negative is phishing is classified as phishing. It is defined as the proportion of phishing websites that are correctly classified as phishing. The rate is mathematically expressed in Equation 7.

$$FN_{rate} = \frac{FN}{TP + FN} \tag{7}$$

Table 5: *Prediction Outcomes for Phishing URLs Detection.*

		Expected class	
		Positive Prediction	Negative Prediction
Classes	Positive	True positive (TP)	False-negative (FN)
	Negative	False-positive (FP)	True negative (FN)

In a False-positive phishing classified as legitimate. It is defined as the proportion of phishing websites that are wrongly classified as legitimate websites. The rate is mathematically expressed in Equation 8.

$$FP_{rate} = \frac{FP}{FP + FN} \tag{8}$$

True negative rate is defined as the proportion of legitimate websites that are wrongly classified as phishing websites. The rate is mathematically expressed in Equation 9.

$$TN_{rate} = \frac{TN}{TN + FP} \tag{9}$$

Recall is defined as the proportion of instances correctly identified by the classifiers as relevant divided by the total number of true positives and false negatives. It is expressed in Equation 10.

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

F-measure is defined as the harmonic mean of precision and recall. It is expressed in Equation 11.

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{11}$$

Geometric Mean is the square of true negative and recall. It is expressed in Equation 12.

$$G_{Mean} = \sqrt{\frac{TP}{TP + FN} * \frac{TN}{TN + FP}} \tag{12}$$

Balanced Detection Rate (BDR) measures the number of minority class instances that were correctly classified and penalize appropriately using the incorrectly classified instances of the majority class.

$$BDR = \frac{TP}{1 + FP} \tag{13}$$

Error rate (ERR) is determined as the number of all wrong predictions divided by the total number of elements in the dataset. The rate is mathematically expressed in Equation 14 and 15.

$$\text{Error rate} = \frac{FP + FN}{TP + TN + FN + FP} = \frac{FP + FN}{P + N} \tag{14}$$

$$\text{Error rate} = 1 - \text{Accuracy rate} \tag{15}$$

5. EXPERIMENTS

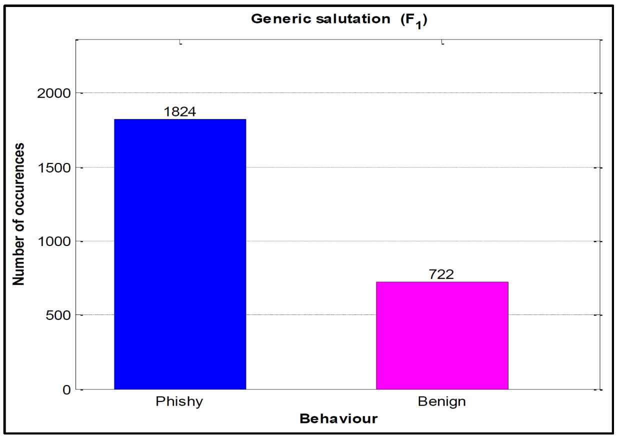
In order to evaluate the proposed scheme, we used two machine learning techniques Support Vector Machines (SVM) and Naïve Bayes to, classify our training datasets into two classes. Many experiments were performed on the datasets to test whether the input URLs are malicious or benign. The URL were entered into the python program and it extracted the URLs features. The features are classified into phishy and benign as shown in Table 6 and the graphical representation is presented in Fig. 3.

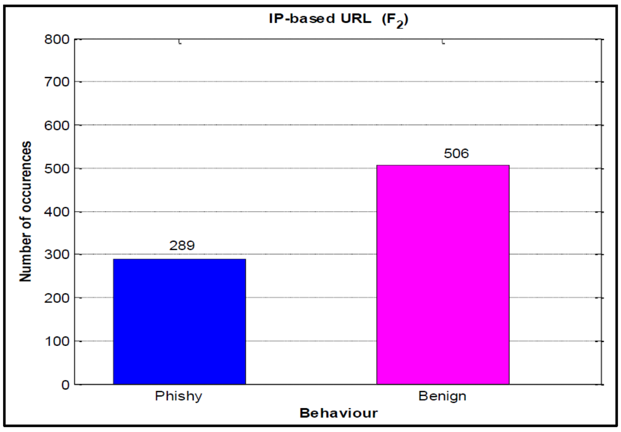
Table 6: *Number of Features in the first 13 Category and their Classifications.*

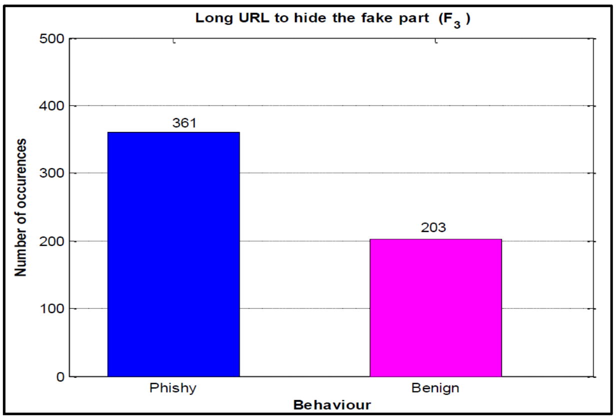
Features	Phishy	Benign	No. of Occurrences
F_1	1824	722	2546
F_2	289	506	795
F_3	361	203	564
F_4	267	1054	1321
F_5	439	259	698
F_6	276	569	845
F_7	2381	1057	3438
F_8	2356	1900	4256
F_9	249	479	728
F_10	189	627	816
F_11	341	923	1264
F_12	1098	2353	3451
F_13	1469	833	2302

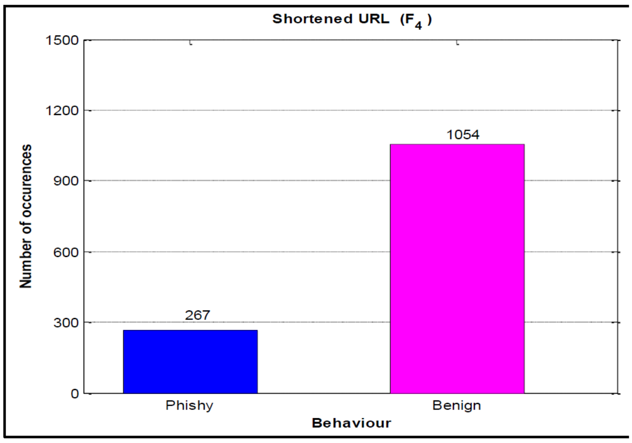
5.1 Behaviours of Selected Datasets

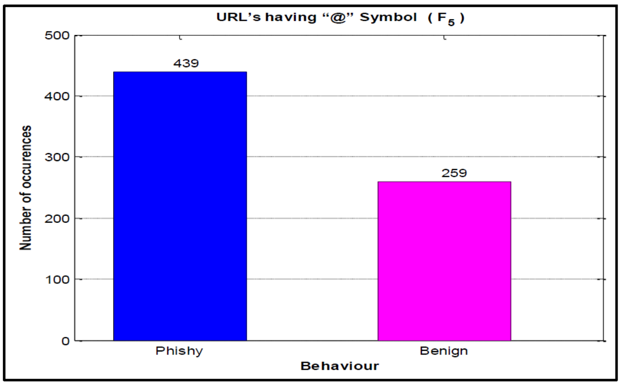
To provide further information about confidence intervals of URL classification, each classifier runs for 100, 150, 200, and 250 iterations. Table 7 shows the 5th percentile, 95th percentile, median, and standard deviation (SD) values for the accuracy of each classifier. We can see that the Naïve Bayes classifier performs better than the SVM classifier in all of the experiments.

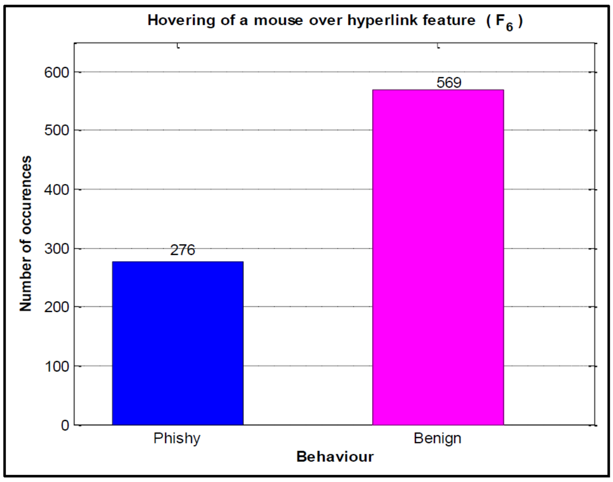


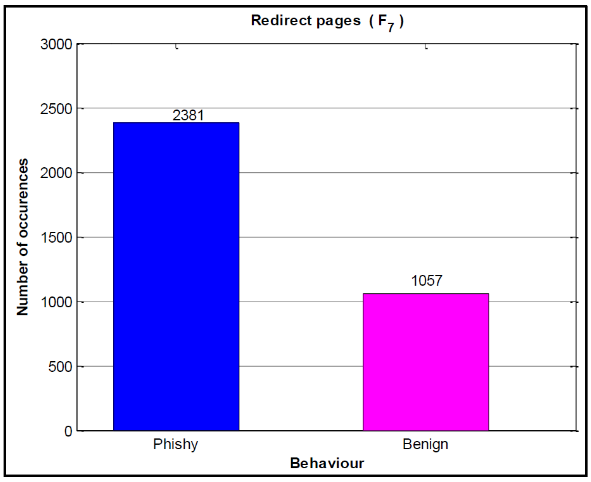

IP-based URL (F2)
Number of occurences
800
700
600
500
400
300
200
100
0
289
506
Phishy
Benign
Behaviour


Long URL to hide the fake part (F3)
Number of occurences
500
400
300
200
100
0
361
203
Phishy
Benign
Behaviour


Shortened URL (F4)
Number of occurences
1500
1200
900
600
300
0
267
1054
Phishy
Benign
Behaviour


URL's having "@" Symbol (F5)
Number of occurences
500
400
300
200
100
0
439
259
Phishy
Benign
Behaviour


Hovering of a mouse over hyperlink feature (F6)
Number of occurences
600
500
400
300
200
100
0
276
569
Phishy
Benign
Behaviour


Redirect pages (F7)
Number of occurences
3000
2500
2000
1500
1000
500
0
2381
1057
Phishy
Benign
Behaviour

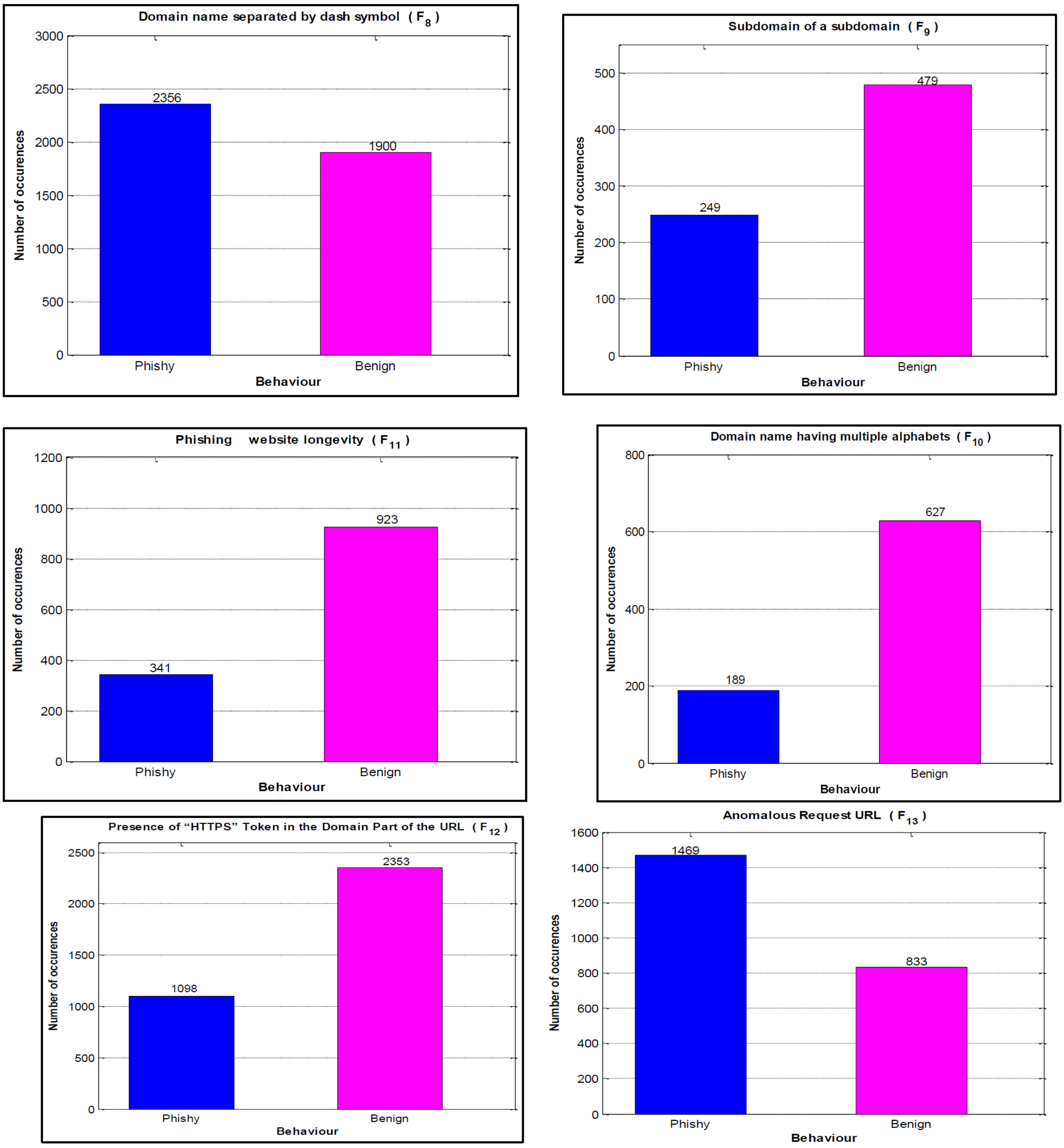


Fig.3: *Number of Occurrences vs Behaviours of the URL.*

Table 7: *Classification Results for the Classifiers.*

No of Runs	Classifier	5th Percentile	95th Percentile	Median	SD
100	SVM	95.25	97.31	96.78	0.31
	Naïve Bayes	96.37	98.42	97.81	0.27
150	SVM	92.95	94.10	93.45	0.42
	Naïve Bayes	95.07	95.29	94.62	0.38
200	SVM	89.09	90.73	90.39	0.57
	Naïve Bayes	91.51	93.48	94.62	0.49
250	SVM	86.37	88.06	87.41	0.67
	Naïve Bayes	89.28	91.50	90.59	0.61

In order to test the accuracy of the algorithms, we obtained the following experimental results which we present in a tabular form in Table 8.

Table 8: *Experimental Results of the Phishing Classifiers.*

Experiment	Phishy	Benign	Selected URLs
Exp1	657	343	1000
Exp2	983	1017	2000
Exp3	1648	1352	3000
Exp4	1850	2150	4000
Exp5	3266	1734	5000
Exp6	2843	3157	6000
Exp7	3108	3892	7000
Exp8	3871	4129	8000
Exp9	5618	3382	9000
Exp10	4796	5204	10000
Exp11	6741	4259	11000
Exp12	8798	3202	12000
Exp13	8371	4629	13000
Exp14	6417	7583	14000
Exp15	5722	9278	15000
Exp16	11038	8962	20000
Exp17	16897	8103	25000
Exp18	16705	13295	30000
Exp19	21163	13837	35000

The first two columns denote the rate of correct classification and incorrect classification of the URLs using the following metrics: $\mathrm{TP}_{\mathrm{rate}}$, $\mathrm{FP}_{\mathrm{rate}}$, $\mathrm{FN}_{\mathrm{rate}}$, $\mathrm{TN}_{\mathrm{rate}}$. Precision, F-measure, and Accuracy. The results are presented in Table 9.

Table 9: *URL classification results.*

Class	Class. as Phishy	Class. as Benign	Precision	F-measure	Accuracy
Phishing	94.2% (TP)	23.7% (FN)	98.01%	95.8%	96.3%
Benign	17.2% (FP)	98.4% (TN)			

We compared four different classifiers, namely SVM, Naïve Bayes, K-Nearest Neighbours (KNN), and Regression Tree in terms of TP rate, accuracy, and FP rate as shown in Fig. 4. The experimental results show that the Naïve Bayes classifier can classify the datasets more accurately than the other classifiers

The performance of the proposed approach (a Multi-stage machine learning approach) is compared with three other state-of-the-art approaches for phishing detection as shown in Fig. 5.The selected related approaches are: Feature Selection approach [25], Logistic Regression approach [24], and Rule-based approach [22]. The percentage for True positive, True negative, Precision, F1 scores, and Accuracy is determined for each of the approaches. The values obtained are presented in Table 10 and the graphical representation is shown in Fig. 5. Our proposed approach achieved the best accuracy of 96.3% in detecting phishy URLs.

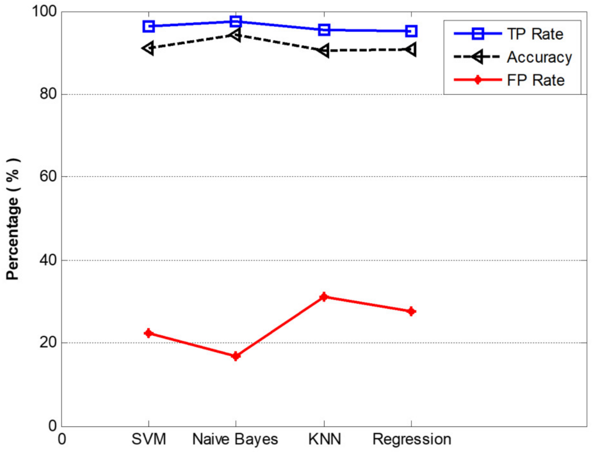


Fig.4: *Performance evaluation of four classifiers .*

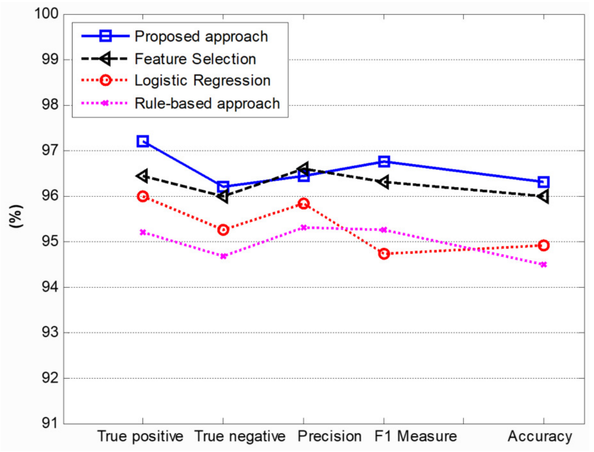


Fig.5: *Performance comparison of the proposed with three other related approaches.*

Table 10: *Performance comparison of four approaches.*

Approach	True Positive (%)	True Negative (%)	Precision (%)	F1 Score (%)	Accuracy (%)
Proposed approach	97.21	96.20	96.43	96.74	96.30
Feature Selection	96.43	95.98	96.58	96.30	96.00
Logistic Regression	95.98	95.24	95.83	94.42	94.90
Rule-based	92.21	94.68	95.31	95.26	94.05

6. CONCLUSION AND FUTURE WORK

Phishing is a type of social engineering attack often used to steal user personal information. In this project, we explore several tactics which phishers use to trick innocent Internet users into divulging their personal information. We added new features to our design and included some important features we identified in the literature. An efficient approach was developed for detecting malicious URLs. Two different machine learning algorithms were used to classify the datasets. Several experiments were performed to determine the efficiency of our scheme. These experiments showed we achieved better performance than the competition and achieved a classification accuracy of 96.3% with a low false-positive rate of 17.2%.

In the future, we will consider more machine learning algorithms to compare our work to their accuracy and false-positive rates.

ACKNOWLEDGMENT

This work is supported by the Centre for Postgraduate Studies, Cape Peninsula University of Technology, Cape Town, South Africa.

References

[1] E. Soegoto and M. Rafi, "Internet role in improving business transaction," in *Proceedings of the IOP Conference Series: Materials Science and Engineering*, pp.012059, 2018.

[2] M. Graham and W. H. Dutton, *Society and the internet: How networks of information and communication are changing our lives:* Oxford University Press, 2019.

[3] Miniwatts Marketing Group. (2019, November 11,). *Internet World Starts.* Available: `https://www.internetworldstats.com/stats.htm`

[4] M. Büchi, N. Just, and M. Latzer, "Caring is not enough: the importance of Internet skills for online privacy protection," *Information, Communication & Society*, Vol.20, No.8, pp. 1261-1278, 2017.

[5] A. Wang, W. Chang, S. Chen, and A. Mohaisen, "Delving into internet ddos attacks by botnets: Characterization and analysis," *IEEE/ACM Transactions on Networking (TON)*, Vol.26, No.6, pp. 2843-2855, 2018.

[6] J. A. Chaudhry, S. A. Chaudhry, and R. G. Rittenhouse, "Phishing attacks and defenses," *International Journal of Security and Its Applications*, Vol.10, No.1, pp. 247-256, 2016.

[7] APWG. (2019, November 13,). *Anti-Phishing Working Group Phishing Activity Trends Report.* Available: `https://docs.apwg.org/reports/apwg_trends_report_q2_2019.pdf`

[8] J. Hong, "The current state of phishing attacks," 2012.

[9] A. Bouveret, *Cyber risk for the financial sector: a framework for quantitative assessment:* International Monetary Fund, 2018.

[10] Kaspersky. (2019, November 25). *How to protect yourself against spam email and phishing.* Available: `https://www.kaspersky.co.za/resource-center/threats/spam-phishing`

[11] K. L. Chiew, K. S. C. Yong, and C. L. Tan, "A survey of phishing attacks: their types, vectors and technical approaches," *Expert Systems with Applications*, Vol.106, pp. 1-20, 2018.

[12] L. Lazar. (2020, 20 October). *Our Analysis of 1,019 Phishing Kits.* Available: `https://www.imperva.com/blog/our-analysis-of-1019-phishing-kits/`

[13] V. Suganya, "A review on phishing attacks and various anti phishing techniques," *International Journal of Computer Applications*, Vol.139, No.1, pp. 20-23, 2016.

[14] L.-H. Lee, K.-C. Lee, H.-H. Chen, and Y.-H. Tseng, "Poster: Proactive blacklist update for anti-phishing," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp.1448-1450, November 2014.

[15] L. Li, E. Berki, M. Helenius, and S. Ovaska, "Towards a contingency approach with whitelist-and blacklist-based anti-phishing applications: what do usability tests indicate?," *Behaviour & Information Technology*, Vol.33, No.11, pp. 1136-1147, 2014.

[16] M. Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert systems with applications*, Vol.37, No.12, pp. 7913-7921, 2010.

[17] R. B. Basnet, A. H. Sung, and Q. Liu, "Learning to detect phishing URLs," *International Journal of Research in Engineering and Technology*, Vol.3, No.6, pp. 11-24, 2014.

[18] A. Jain and V. Richariya, "Implementing a web browser with phishing detection techniques," *arXiv preprint arXiv:1110.0360*, 2011.

[19] M. A. Mahmood and L. Rajamani, "APD: ARM Deceptive Phishing Detector System Phishing Detection in Instant Messengers Using Data Mining Approach," *Global Trends in Computing and Communication Systems*, Vol.269, No.1, pp. 490-502, 2011.

[20] M. Ajlouni, W. e. Hadi, and J. Alwedyan, "Detecting phishing websites using associative classification," *image*, Vol.5, No.23, pp. 36-40, 2013.

[21] D. Zhang, Z. Yan, H. Jiang, and T. Kim, "A domain-feature enhanced classification model for the detection of Chinese phishing e-Business websites," *Information & Management*, Vol.51, No.7, pp. 845-853, 2014.

[22] M. Moghimi and A. Y. Varjani, "New rule-based phishing detection method," *Expert systems with applications*, Vol.53, pp. 231-242, 2016.

[23] G. Ramesh, I. Krishnamurthi, and K. S. S. Kumar,"An efficacious method for detecting phishing webpages through target domain identification," *Decision Support Systems*, Vol.61, pp. 12-22, 2014.

[24] A. K. Jain and B. B. Gupta, "A machine learning based approach for phishing detection using hyperlinks information," *Journal of Ambient Intelligence and Humanized Computing*, Vol.10, No.5, pp. 2015-2028, 2019.

[25] E. Gandotra and D. Gupta, “An Efficient Approach for Phishing Detection using Machine Learning,” in *Multimedia Security*, ed: Springer, 2021, pp. 239-253.

[26] L. M. Ellram and W. L. Tate, “The use of secondary data in purchasing and supply management (P/SM) research,” *Journal of purchasing and supply management*, Vol.22, No.4, pp. 250-254, 2016.

[27] PhishTank. (2019, November 25,). *Statistics about phishing activity and PhishTank usage.* Available: http://www.phishtank.com/stats.php

[28] R. B. Basnet, A. H. Sung, and Q. Liu, “Feature selection for improved phishing detection,” in *Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pp.252-261, 2012.

[29] H. N. Security. (2019). *Phishing attacks at highest level in three years.* Available: https://www.helpnetsecurity.com/2019/11/07/phishing-attacks-levels-rise/

[30] J. LaCour, “Phishing Trends and Intelligent Report,” 2019.

[31] R. Verma and A. Das, “What’s in a url: Fast feature extraction and malicious url detection,” in *Proceedings of the 3rd ACM on International Workshop on Security and Privacy Analytics*, pp.55-63, 2017.

[32] S. Wedyan and F. Wedyan, ”An Associative Classification Data Mining Approach for Detecting Phishing Websites,” *Journal of Emerging Trends in Computing and Information Sciences*, Vol.4, No.12, 2013.

[33] S. Le Page, G.-V. Jourdan, G. v. Bochmann, J. Flood, and I.-V. Onut, “Using url shorteners to compare phishing and malware attacks,” in *Proceedings of the 2018 APWG Symposium on Electronic Crime Research (eCrime)*, pp.1-13, 2018.

[34] L. James, *Phishing exposed.* Canada.: Syngress, 2005.

Philip Abidoye received his Master’s and Ph.D. degrees in Computer Science from the University of Ibadan, Ibadan, Nigeria, and the University of the Western Cape, Cape Town, South Africa in 2006 and 2015 respectively. He is a Lecturer in the Department of Information Technology, Cape Peninsula University of Technology, Cape Town, South Africa. He was a research fellow in the same deparment between 2019 and 2020. Previously, he was a Lecturer in the Department of Computer Science, School of Computing, University of South Africa (UNISA), Johannesburg, South Africa. His research interests include secure wireless sensor networks, cybersecurity, the Internet of Things (IoT), and machine learning. He is also interested in solving security challenges in Cloud Computing and Cyber-Healthcare.

He has a strong record of accomplishment of authoring novel articles in accredited and peer-reviewed journals, conference papers, and book chapters. Abidoye is a member of the Association for Computing Machinery (ACM), Institute of Electrical and Electronics Engineers (IEEE), South African Institute of Computer Scientists and Information Technologists (SAICSIT), and Computer Professionals Registration Council of Nigeria (CPN).

Boniface Kabaso received a Ph.D. degree in Information Technology from the Cape Peninsula University of Technology, Cape Town, South Africa in 2014. Currently, he is the Head of the Department of Information Technology Department, Cape Peninsula University of Technology, Cape Town, South Africa. His research interests include software development, soft computing, the Internet of Things (IoT), and Cloud Computing. He has authored many articles which are published in international journals of high repute.