SN Computer Science (2020) 1:191
https://doi.org/10.1007/s42979-020-00194-z

ORIGINAL RESEARCH

# Phishing Email Detection Based on Binary Search Feature Selection

**Gunikhan Sonowal**[1]

Received: 7 April 2020 / Accepted: 14 May 2020 / Published online: 6 June 2020


**Abstract**
Phishing has appeared as a critical issue in the cybersecurity domain. Phishers adopt email as one of their major channels of communication to lure potential victims. This paper attempts to detect phishing emails by using binary search feature selection (BSFS) with a Pearson correlation coefficient algorithm as a ranking method. The proposed method utilizes four sets of features from the email subject, the body of the email, hyperlinks, and readability of contents. Overall, 41 features were selected from the aforementioned four dimensions. The result shows that the BSFS method evaluated the accuracy of 97.41% in comparison with SFFS (95.63%) and WFS (95.56%). This exploration shows that the SFFS requires more time to ascertain the optimum features set and the WFS requires the least time; however, the accuracy of WFS is very low in comparison with other algorithms. The significant finding of the experiment is that the BFSF requires the least time to evaluate the best feature set with better accuracy even though few features are removed from the feature corpus.

**Keywords** Phishing · Cyber-crime · Anti-phishing · Binary search feature selection · Social engineering · Pearson correlation coefficient (PCC)

## Introduction

The email has emerged as one of the reliable and real-time communication mediums using which a huge number of individuals and organizations share their messages and data. According to the *Radicati Group* [58], the number of overall email users was approximately *2.6 billion* in 2015, and it is estimated to be *2.9 billion* by the end of 2019. With the prolific expansion in the number of email users, phishers exploit the email in different manners to spur the users to reveal their credentials [23, 39, 53]. As per the anti-phishing working group (APWG) [9] report on (February 24, 2020), the number of unique phishing emails from the customer was 45,072 in December 2019 and 42,424 in November. Starting in mid-March, 2020, cybercriminals propelled an assortment of COVID-19 themed phishing and malware assaults against workers, healthcare facilities, and the recently jobless.

Phishers send fraudulent emails to the users to gain an individual's credentials [41, 78]. The fraudulent emails are designed to look like genuine emails with the incorporation of elements such as legitimate brand's logo, ID, signature. This presents the users to effortlessly come into a trust and uncover their credentials [2, 49, 71]. Phishing is a form of social engineering-based attack that primarily adopts the following techniques [4, 17, 38, 46, 57] to accomplish their objective:

- *Luring Emails* Phishers send the email that contains lucrative offers with eye-catching contents. They design their messages in an alluring manner so that the victims effortlessly fall prey to it: for example, winning prizes, lottery, fortunate customer offer, and others.
- *Urgent Emails* Phishers send the warning email with constrained time to the victims so that phishers can rapidly accomplish their job. As a gvf, the phishing emails live only for a few days. The warning includes contents such as the suspension of the account.
- *Link to Another Website* Phishers send embedded phishing website links with emails to the users, and the link appears as a genuine site. For example, < *a* href="http://phishingsite.com"> http://bank.com< /*a* >, the visible text shows the name of a genuine bank; however, the actual link is redirected to the phishing site. In most cases, users fall prey to phishing on examining the visible text.

✉ Gunikhan Sonowal
gunikhan.sonowal@gmail.com

[1] Department of Computer Science, Pondicherry University, Puducherry, India

- *Generic Names* Usually, phishers send random emails to millions of victims, and hence, they lack the knowledge of the victim's name. For this reason, they employ the generic name such as Dear customer, and others.

However, to alleviate the phishing issue, several anti-phishing techniques have been developed to protect the users [18, 62, 70]. Two methodologies are fundamentally adopted among them: *link-based approach* and *word list-based approach* [8]. In a link-based approach, the hyperlinks are examined through blacklist [51], Google safe browsing [27], SiteAdvisor [63], whitelist [5, 16, 64] and heuristic-based methods [24, 67, 73, 75] to decide whether the email is a phishing email or legitimate. On the other hand, the words list-based approach examines the frequent keywords. In most instances, phishers employ these keywords to manipulate the victims [10, 43, 55].

Machine learning approaches have been attempted to detect phishing. They employed several novel features with the end goal to achieve better accuracy. This paper initially experimented with the dataset using WFS where WFS represents without feature selection. As the method collected 41 features from different directions, this method employs all the features together without using any feature selection algorithm. After that, the SFFS was an experiment that is explained in "Feature Ranking Algorithm" section. Although these algorithms provide good accuracy, the major challenge is to select the best features with a minimum time among all the features to optimize detection accuracy. This paper applies the binary search feature selection algorithm, which employs the Pearson correlation algorithm (PCC) to rank the features and binary search to search the best features set with minimum time complexity.

The major objectives of this paper are as listed below:

- To generate word-based features by analyzing frequently appearing words of email's subjects and contents.
- To generate link-based features by examining the URL links embedded in an email.
- To generate the readability-based features using eight well-known readability algorithms and applying them to discriminate the text contents of phishing emails and legitimate emails.
- To search the optimum features set using Pearson correlation algorithm (PCC) with binary search as well as the sequential forward search algorithm.
- To verify the best features set by comparing with other feature selection algorithms on the basis of time, accuracy and number of features.
- To justify the method by comparing the results with the existing approaches.

The structure of the paper is as follows: "Related Works" section provides an overview of the background of email phishing detection research. "The Proposed Method" section analyzes the phishing emails and legitimate emails in order to elicit discriminative features for the method. "Experimental Evaluation" section builds the features and depict the experimentation results. "Discussion" section discusses the outcomes of the experimentation. "Conclusion and Future Work" section summarizes the paper and indicates the future directions for this research.

## Related Works

Several studies have been developed to detect phishing emails using different machine learning approaches. Many novel features are introduced to filter phishing emails from legitimate emails. This section discusses various approaches, which were proposed by researchers to mitigate phishing emails.

One of the interesting methods titled *PILFERS* was proposed by Fette et al. [24] based on ten features to detect phishing emails. They evaluated the accuracy using *random forest* on a set of 860 phishing emails and 6950 legitimate emails and identified over 96.00% of the phishing emails, and error rate was 0.1% of the legitimate emails.

Another study employed hyperlink and structural properties of emails alongside whois information on hyperlinks as profile classes [74] was also attempted by a study. They employed two classification algorithms *BoosTexter* and *Support Vector Machine* for experimentation. The outcomes demonstrate that profiling should be possible with a significantly high accuracy using hyperlink information.

Another study has been carried out with 16 relevant features including keyword features, which employs six different machine learning methods [10]. The result of their experiment shows that the biased support vector machine (BSVM) and artificial neural networks offered the equivalent accuracy of 97.99%.

A novel phishing email classifier [13] that focused on fundamental features, external features, model-based features, and image processing has also been proposed. They proposed a new feature trained by machine learning techniques using the dynamic Markov Chain (DMC) feature and latent class topic model (CLTOM). From the investigations, they discovered that the proposed strategies beat other published methodologies for classifying phishing messages.

Another study by Khonji et al. [40] has endeavored to develop a robust phishing email classification model by examining several feature subset selection methods, which primarily used beforehand proposed phishing features and classification algorithms. By assessing different feature subset selection strategies, a viable feature subset made of 21

features was picked out of the set of 47 full features. The result of the experiment shows that utilizing the feature subset, RF classifiers accomplished an F1 score of 99.396%.

To detect phishing emails on zero day using a multilayer hybrid strategy was proposed by Chowdhury et al. [20]. They applied a novel method for pruning the ensemble using ranking-based, clustering-based and optimization-based pruning. The result revealed that multilayer hybrid strategy (MHS) was effective and produced superior outcomes with the F-measure of 0.98%. MHP (multilayer hybrid pruning) performed superior to other pruning methods in two layers of MHS. The outcome illustrated that the accuracy of filtering decreased for the more distant time span.

Text mining-based approach is also an important technique in order to detect phishing emails. Zareapoor et al. [76] employed three distinct feature selection techniques, namely *Chi-square, InfoGain, and GainRatio*, and five different well-known classifiers, namely Naïve Bayes, random forest, support vector machine, Ripper, and AdaBoost. From the experiment, they discovered that the proposed method requires less preprocessing, less training time and yields good performance.

Distinct structural features from phishing emails are applied to detect phishing emails [17]. They experimented with these features with a limited corpus of 400 emails using a support vector machine classifier and showed that the proposed approach can distinguish an extensive variety of phishing emails with a minimum performance overhead.

A novel method of using text mining and data mining to detect phishing emails [55] extracted 23 keywords from a dataset of 2500 phishing and nonphishing emails. Further, they selected 12 keywords using t-statistic-based feature selection and experimented with multiple machine learning classifications with and without feature selection. From the result, they discovered the higher phishing prediction accuracy with fewer numbers of features.

An intelligent classification technique was proposed by Yasin and Abuhasan [72], which detects phishing emails using knowledge discovery, data mining, and text processing techniques. They utilized the preprocessing phase by applying text stemming and WordNet ontology. The model employed knowledge discovery procedures using five popular classification algorithms and achieved 99.1% accuracy using the random forest algorithm.

One interesting technique was provided by Olivo et al. [52], which employed 11 relevant features to yield the minimum set of significant features providing reliability, good performance, and flexibility to the phishing detection engine. The experimental results demonstrate that the proposed technique optimized the detection engine of the anti-phishing scheme.

Information gain is used to extract hybrid features to detect phishing emails that were proposed by Ma et al. [44]. The result of their experiments shows the selected features evaluated improved performance as the original features. They tested five machine learning algorithms and compared the performances of each other, and the result shows that the decision tree evaluated the best performance.

A multitier classification model was proposed by Islam and Abawajy [37] based on a weighting of message content and message header, and the features were selected according to the priority ranking. The results from the experiments showed that the algorithm reduced the FP problems substantially with lower complexity.

The existing anti-phishing models present numerous advanced methods to recognize phishing emails. Researchers continuously operated several features from hyperlinks, keywords to enhance the accuracy of the models. To date, a considerable number of distinctive features are prepared to counter phishing emails. It is commonly accepted that all features are not relevant to the particular task because the attackers continually develop novel features. Hence, one critical issue is to remove irrelevant features.

For the aforementioned issue, some models practiced several feature selection algorithms to lessen the dimension of the features. The primary weakness of these models is the selection of insufficient features for implementing feature selection algorithms which produce a challenge to recognize the performance of the feature selection algorithm. However, many studies undetermined about the time for searching the best feature set.

This paper introduces a model called binary search feature selection (BSFS) for detecting phishing emails using a novel feature selection algorithm, which requires minimum time to ascertain the best feature set. The decision of the best features set is performed using two parameters: better accuracy and a smaller dimension of features. This study combined one more parameter, that is, time.

## The Proposed Methods

The overall architecture of the proposed method is shown in Fig. 1. In this figure, the proposed method preprocessed the subjects, body contents, hyperlink, and readability scores of texts. Subsequently, the method extracts the features from phishing as well as legitimate emails, which is explained in "Features Extraction" section and generates the feature vector space. The extracted features are assigned to the feature ranking algorithm in order to evaluate the rank of the features against the decision attribute as explained in "Feature Ranking Algorithm" section. Finally, the feature search algorithms search the best features set using the machine learning algorithm.

Below, all the features are briefly explained, which are accepted in the feature corpus.

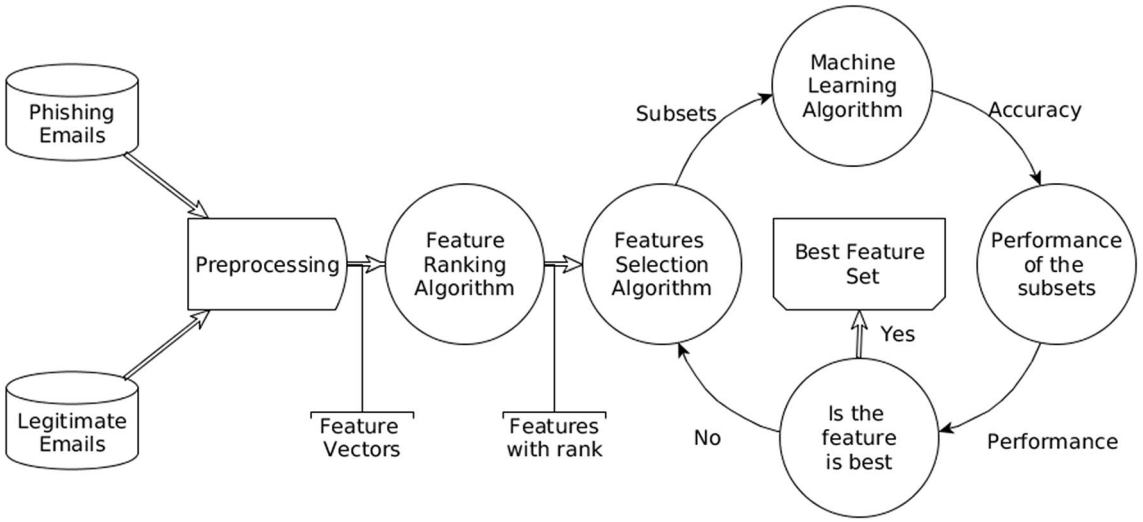


**Fig. 1** Proposed method architecture

## Word Features of Subjects

The subject of an email is considered to play a significant role in phishing emails [32, 69], as users perceive the emails through the subject of the emails. In this instance, phishers employ attractive messages in the subject so that users easily come into a conviction on the email. In this manner, the subject is likewise taken into the feature's corpus. Therefore, the frequent keywords are also investigated in this section.

From the dataset (phishing email and legitimate email), 465 phishing emails and 841 legitimate emails are selected, and the proposed method carries out the following steps to extract the keyword:

- Extracts the subjects from both types of emails: phishing and legitimate.
- Finds the pattern of the keywords and converts all the keywords into the lower case.
- Eliminates the stop words from the subject, such as is, an, and others.
- Extracts the top frequently used keywords from phishing emails using Eq. (1).

$$F(k,s) = \frac{f_{k\in s}}{N_s} \tag{1}$$

where $k$ is the keyword, $s$ is the subject of the email, $f_{k\in s}$ frequency of the keyword in subjects, and $N_s$ is the number of emails.

- Compares the phishing keywords with legitimate emails by searching the frequency of occurrence in legitimate email's subjects.

Table 1 shows the 12 keywords' frequency of the subjects on analyzing the phishing emails as well as legitimate emails and generates the words features for subject. The value for this feature is {0, 1}. If the keyword is present in emails, then the proposed method returns 1 otherwise 0.

## Words Features for Contents of Emails

In this section, the proposed method applies similar steps as explained in "Words Features of Subjects" section to analyze the keywords in the contents. However, this section analyzes the contents of emails; therefore, the method eliminates the subjects and header portions from the emails.

Table 2 shows the keywords' frequencies of content of phishing emails and legitimate emails. In this analysis, the

**Table 1** Keywords frequency of emails subject

| Keywords | Phishing emails | Legitimate emails |
|---|---|---|
| Account | 47.1 | 0.0 |
| Update | 18.06 | 2.49 |
| Security | 13.98 | 0.71 |
| Important | 10.97 | 0.12 |
| Resent | 9.46 | 0.0 |
| Notice | 9.25 | 0.12 |
| Verify | 6.45 | 0.0 |
| Please | 6.24 | 0.36 |
| Verification | 6.02 | 0.0 |
| Credit | 5.38 | 0.0 |
| Bank | 5.16 | 0.0 |
| Online | 5.16 | 0.24 |

**Table 2** Keywords frequency of emails content

| Keywords | Phishing emails | Legitimate emails |
|---|---|---|
| Account | 332.04 | 2.62 |
| Update | 219.14 | 70.16 |
| Information | 257.42 | 23.31 |
| Transfer | 190.32 | 57.43 |
| Post | 153.33 | 88.94 |
| Credit | 133.55 | 01.43 |
| Priority | 109.03 | 09.51 |
| User | 400.22 | 374.32 |
| Resent | 302.8 | 146.73 |
| Security | 238.71 | 77.05 |
| Status | 194.19 | 21.05 |
| Address | 139.14 | 12.96 |
| Access | 125.38 | 07.49 |
| Time | 107.53 | 47.09 |

method had 14 keywords. However, some keywords were similar to subject's keywords. The data type is used in this feature as {0, 1}; if the keywords are present in emails, then the proposed method returns 1, otherwise 0.

## Features from Hyperlinks

Hyperlink of emails is considered as associating a website page using the URL of the page. Individuals or companies employ the hyperlink through several techniques such as an icon, text in their emails. A hyperlink is a combination of two components: the visible text, which is visible to users, and the actual link, which is an actual destination address. For example $< a\ href = \}\}$http://go.microsoft.com/?linkid=3D9724456$'' >$ *clickhere* $< /a >$, the visible text is *click here* and actual link is http://go.microsoft.com/?linkid=3D9724456. The actual link of the hyperlink is a Uniform Resource Locator (URL) of Web sites. It has six components, namely: addressing scheme, network location, path, parameters, query, and fragment identifier. The structure of the URL is "scheme://netloc/path;parameters?query#fragment." From the above example http://go.microsoft.com/?linkid=3D9724456, then the scheme='http', netloc='go.microsoft.com', path='/', params='', query='linkid=3D9724456', fragment=''.

The hyperlink is utilized by attackers to manipulate the victims to visit their sites. Phishers insert the hyperlink into the phishing emails that seem like a legitimate link to achieve trust from the users. On clicking the link by the users, it navigates to the phishing page and demands the credentials from users. According to *SecAware*, the 23% of victims open phishing emails and 11% of victims click on attachments.[1]

**Table 3** Comparison between the actual link and visible texts

| Components | Phishing | Legitimate |
|---|---|---|
| Addressing scheme | 22 | 25 |
| Network location | 1 | 19 |

Many anti-phishing techniques are proposed based on the hyperlink. However, in our research, the proposed method analyzes the novel and existing discriminative features of phishing and legitimate emails from our selected dataset. For this purpose, the proposed method extracts the hyperlinks from phishing emails as well as legitimate emails and investigates the features to classify phishing emails from legitimate emails. The features are listed in the remaining part of this section.

- *Link in Visible Text* In a legitimate hyperlink, most visible texts provide proper information regarding the actual link and usually, no link is shown in the visible text of the legitimate hyperlink. However, most of the phishers provide a legitimate link in the visible text so that users come to trust in it. From the investigation of both phishing and legitimate hyperlinks, the proposed method informs that phishing emails contain the links in visible texts which are 6.67% in comparison with legitimate emails 3.09%.
- *Mismatch Link* It has been observed that some legitimate emails also provide a link to the visible text. Therefore, the proposed method explores the mismatch between the actual link and visible texts. The phishing hyperlinks display the visible texts as a genuine link; however, the actual link is connected to a phishing site so that users easily prey fall in phish on looking at the visible text of the hyperlink. With respect to the illustration, $< a\ href = \}\}$http://page.paypal.com$'' >$http://paypal.com$>$, the actual link (http://page.paypal.com) belonged to phishing URL. However, the visible text (http://paypal.com) contains a legitimate URL. This feature is also used in Basnet et al. [10], Alkhozae and Batarfi [7], Chen and Guo [19]

  The proposed method investigates the similarity between the actual link and visible text on the basis of two components of the URL, namely *addressing scheme* and *network location*. Initially, the proposed method compares the addressing scheme between two links (visible text and actual link); if the two links are identical, then forward the similarity investigation to the network

[1] https://secaware.co.uk/.

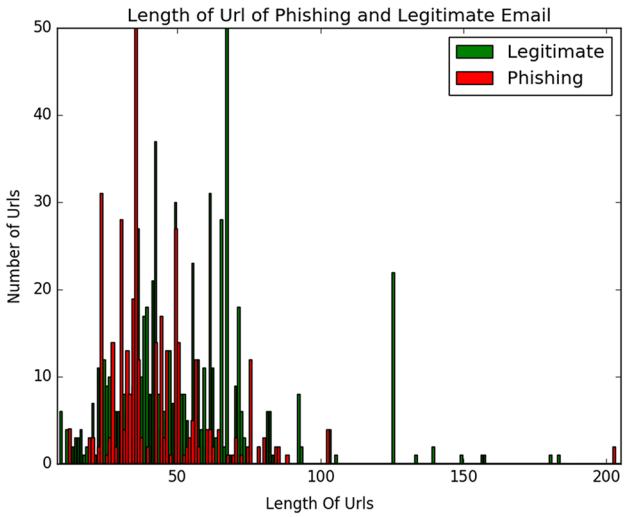


**Fig. 2** Length of phishing emails and legitimate emails

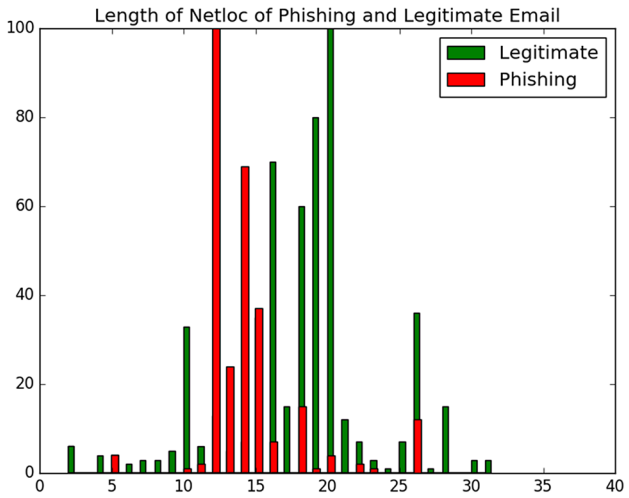


**Fig. 3** Network location length of phishing emails and legitimate emails

location similarity testing. The proposed method has discovered that the number of hyperlinks present in visible texts for selecting phishing and legitimate emails is 31 (phishing emails) and 26 (legitimate emails).

Table 3 shows that among 31 hyperlinks in visible texts of phishing emails, only 22 are identical in addressing scheme comparison and one is identical in networking location comparison. However, in legitimate emails, 25 are identical in addressing scheme and 19 are identical in the network location from 26 hyperlinks. This feature is also used in Alkhozae and Batarfi [7].

- *IP-Based URL* Internet users identify legitimate sectors through the domain names, as the DNS server provides a unique domain name to every Internet sector. Usually, phishers replicate the page of the legitimate site; however, the domain of the page is unique. For this reason, phishers employ the IP-based URL. Most users unnoticed the URL of the page, and they highly preserve attention on the page contents. In addition, in IP-based URL, phishers are free from DNS server registration. From the investigation, the proposed method observes that the phishing emails contain 20 IP-based domains in comparison with legitimate emails which contain null. As a result, no legitimate emails provide IP based on the hyperlink. This feature is also used in Alkhozae and Batarfi [7], Basnet et al. [10], Garera et al. [26], Basnet et al. [11], Zhang et al. [77], Moghimi and Varjani [47], Sonowal and Kuppusamy [65].
- *Length of URL* Length of URL is regarded as an important feature to classify a phishing URL from a legitimate URL. Figure 2 shows the length of both legitimate and phishing URLs. From the figure, the method selects the length 54 as discrimination length. This feature is also used in Moghimi and Varjani [47], Mohammad et al. [48], Moghimi and Varjani [47].
- *Length of Network Location of URL* Network location, lengths are as well as important features to differentiate between phishing URLs and legitimate URLs. Figure 3 illustrates the length of the network location of both legitimate and phishing URLs. In this feature, the method selects the length 16 as the discrimination length. This feature is also used in Garera et al. [26], Basnet et al. [11]
- *Hyphen in Network Location* Most legitimate URLs ignore the hyphen in the domain name. In our investigation, phishing URLs contain the hyphen 4.95% in comparison with legitimate URL having 0.47%. This feature is also used in Basnet et al. [10], Zhang et al. [77], Mohammad et al. [48].
- *Number of Dots in the URL* Phishing utilizes dots to hide the phishing domain in the URL by adding the legitimate domain. However, the URL is redirected to a phishing page. In this instance, a majority of users notice only the legitimate domain and believe as legitimate URL and fall prey in Phish. In our investigation, the proposed method counts the dots of phishing emails and legitimate emails as shown in Fig. 4. From the analysis, the method selects three dots for discrimination. This feature is also used in Basnet et al. [10, 11], Zhang et al. [77], Mohammad et al. [48], Moghimi and Varjani [47], He et al. [35].
- *Img Tag in Visible Texts* Phishers utilize the icon of the legitimate brands in visible text so that it looks and feels similar to legitimate links; however, in the actual link, they feed the phishing site link. In our investigation, the proposed method has discovered 13 “img” tag in phishing in comparison with legitimate 0. This feature is also

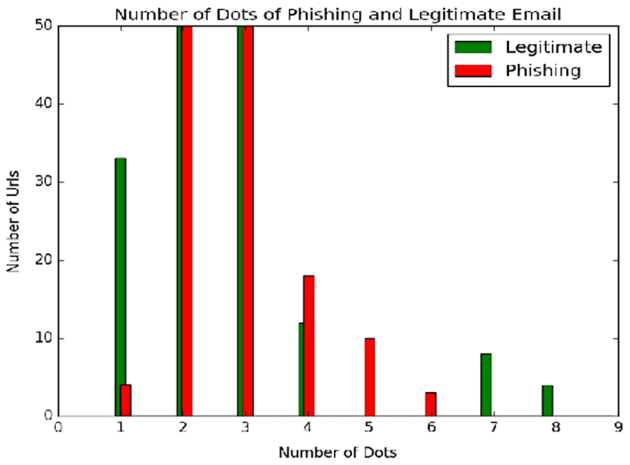


**Fig. 4** Number of dots in phishing and legitimate URLs

used in Alkhozae and Batarfi [7], Basnet et al. [10], Zhang et al. [77].

- Unsubscribed link: A majority of legitimate emails provide a link to unsubscribe the message which is irrelevant for the users. However, many phishing emails exclude any particular category of an unsubscribed link. The proposed method has observed legitimate emails consist of 35 in comparison with phishing emails which contain null.
- *Empty Visual Text* From the dataset, it has been observed that the hyperlink has an empty in visible text. This feature is put into the proposed method feature's corpus to classify the phishing from legitimate emails.
- *Invalid URL in Actual Link* The proposed method analyzes the URL of phishing emails and legitimate emails, and it is observed some href tags' return URL is invalid to parsing. In this scenario, the method initially analyzes the "http" pattern matching; if it is valid, then it overanalyzes the network location validation. This feature is also used in Pan and Ding [54], Mohammad et al. [48], He et al. [35]

## Features from Readability Algorithms

Readability score assists individuals to compute how hard to peruse a piece of texts. Usually, companies or organizations maintain their standard of writing text in emails, and before sending any specific emails to their customers, they analyze the style of the emails so that the customers easily understand the text of the emails.

Readability is one of the important aspects of accessibility [60], and it plays an important role in phishing emails, as it has different text writing styles [1, 22, 34]. This section analyzes the phishing emails and legitimate emails by well-known eight readability algorithms as follows:

### Automated Readability Index

The automatic readability index is used to calculate the readability score on the premise of readability of English text [61]. The equation of the automatic readability index is shown in Eq. (2)

$$\mathrm{ARI} = 4.71\left(\frac{C}{W}\right) + 0.5\left(\frac{W}{S}\right) - 21.43 \tag{2}$$

where $C$ is the number of letters and numbers, $W$ is the number of spaces, and $S$ is the number of sentences.

### Coleman Liau Index

Meri Coleman and T. L. Liau developed the Coleman–Liau index to calculate the readability score [21]. The equation of the Coleman–Liau index (CLI) is shown in Eq. (3)

$$\mathrm{CLI} = 0.0588L - 0.296S - 15.8 \tag{3}$$

L denotes the average number of letters per hundred words and S denotes the average number of sentences per hundred words.

### Flesch–Kincaid Readability Test

Rudolf Flesch developed the Flesch–Kincaid Readability Test, which is used to indicate how difficult a text in English is to understand [25]. Two tests are conducted: Flesch–Kincaid Grade Level and Flesch Reading Ease Score.

The equation of the Flesch–Kincaid Grade Level (FKGL) is shown in Eq. (4)

$$\mathrm{FKGL} = 0.39\left(\frac{\mathrm{TW}}{\mathrm{TS}}\right) + 11.8\left(\frac{\mathrm{Tsy}}{\mathrm{TW}}\right) - 15.59 \tag{4}$$

Flesch Reading Ease Score (FRES) test is shown in Eq. (5)

$$\mathrm{FRES} = 206.835 - 1.015\left(\frac{\mathrm{TW}}{\mathrm{TS}}\right) - 84.6\left(\frac{\mathrm{Tsy}}{\mathrm{TW}}\right) \tag{5}$$

where TW is the total words, TS is the total sentence, Tsy is the total syllables, and Tsy is the total syllables

### Gunning Fog Index

Robert Gunning, an American businessman, developed this readability test [29].

The equation of the Gunning Fog Index is shown in Eq. (6)

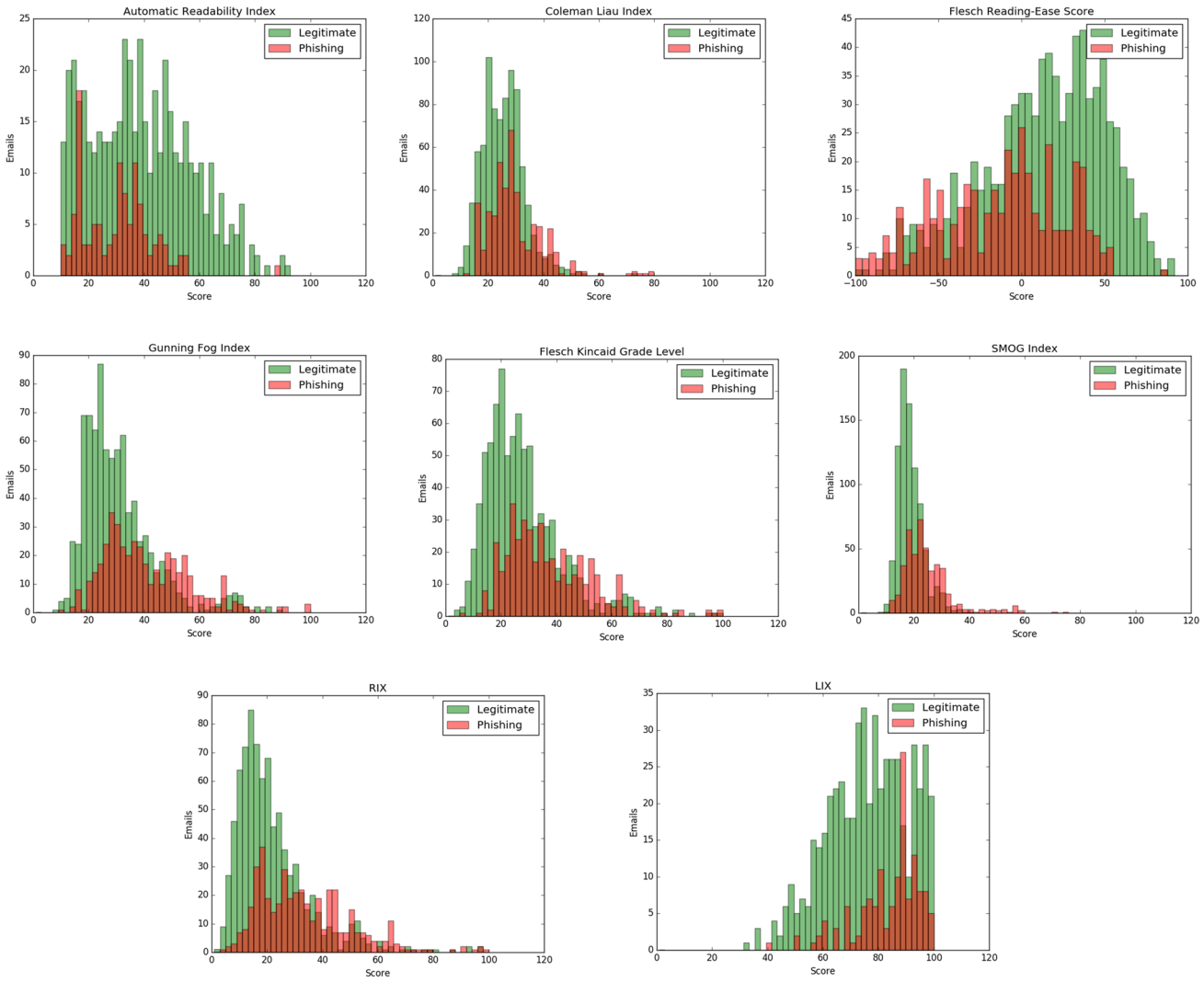


**Fig. 5** Readability scores of phishing emails and legitimate emails using the eight well-known readability algorithms

$$\text{GFI} = 0.4\left[\left(\frac{\text{words}}{\text{Sentences}}\right) + 100\left(\frac{\text{Complex Words}}{\text{Words}}\right)\right] \quad (6)$$

### SMOG Index

G. Harry McLaughlin developed this SMOG index [45], and SMOG is primarily employed for testing the health messages. The equation of Smog to test readability score is shown in (7).

$$\text{SMOG} = 1.0430\sqrt{\text{TP} \times \frac{30}{\text{TS}}} + 3.1291 \quad (7)$$

where TP is the total number of polysyllables and TS is the total sentence.

### LIX Readability Score

The Swedish scholar Carl–Hugo Björnsson developed this readability test [14]. The equation of this test is shown in (8).

$$\text{LIX} = \frac{W}{P} + \frac{\text{LW}.100}{W} \quad (8)$$

where $W$ is the number of words, $P$ is the periods, and LW is the long words containing more than six letters.

### RIX

The equation of the readability test RIX is shown in (9).

$$\text{RIX} = \frac{\text{Number of Long word}}{\text{Number of Sentence}} \quad (9)$$

Figure 5 shows the readability score of eight well-known readability algorithms. In this figure, the first row from left to right shows an *automatic readability index (ARI), Coleman–Liau index (CLI), Flesch Reading Ease Score (FRES)* and second row *Flesch–Kincaid Grade Level (FKGL), SMOG Index (SMI) and Gunning Fog Index (GFI)* and third row *RIX and LIX*. From Fig. 5, the method selects 65 for ARI, 36 for FRES, 39 for FKGL, 41 for GFI, 21 for SMI, 33 for CLI, 12 for Lix, and 28 for Rix as a discrimination boundary to distinguish legitimate and phishing.

#### Pearson Correlation Algorithm (PCC)

This paper has employed primarily Pearson correlation algorithm (PCC) to rank the features. It measures the linear correlation between two features [12, 36]. It assesses three classes of correlation: positive linear correlation is considered as 1, no linear correlation is 0, and negative linear correlation is −1. Several researchers have adopted Pearson correlation coefficient (PCC) to determine the relevant features [30, 31].

Assume $X = \{x_1, x_2, \ldots x_n\}$ and $Y = \{y_1, y_2, \ldots y_n\}$ are two sets of features. The PCC is defined by $\rho$, and equation is shown in (10).

$$\rho(X, Y) = \frac{\text{cov}(X, Y)}{\sigma X, \sigma Y} \tag{10}$$

where $\text{cov}(X, Y)$ is the covariance of $X$, $Y$, and $\sigma X$ is the standard deviation of $X$ and $\sigma Y$ is the standard deviation of $Y$.

$$\text{cov}(X, Y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \tag{11}$$

where $\bar{x}$ and $\bar{y}$ are mean of $X$ and $Y$ is denoted by Eq. (12)

$$\bar{x} = \frac{1}{n} \sum_{x=1}^{n} x_i \tag{12}$$

The standard deviation ($\sigma$) is defined by Eq. (13)

$$\sigma(x) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}} \tag{13}$$

### Feature Ranking Algorithm

The target of using a feature selection algorithm is to minimize the dimension of the features. The proposed method employs the Pearson correlation coefficient (PCC) to eliminate the irrelevant features from the feature set as explained in "Pearson Correlation Algorithm (PCC)" section. Assume $F = \{f_i | i = 0, 1, 2, \ldots n\}$ to be the features set where $F \neq 0$. Subsequently, the method evaluates the Pearson correlation coefficient (PCC) scores ($Y$) of all the features with the decision attribute ($d_i$) using Eq. (14):

$$y_i \leftarrow \text{PCC}(f_i, d) \tag{14}$$

where $f_i \in F$ and $y_i \in Y, Y = \{y_0, y_1, y_2 \ldots y_n\}$ and $d_i$ denotes the level of $f_i; i = 1, 2, \ldots n$.

This method evaluates the rank of all the features using Eq. (14), and the highest distance manages the high relevance to the particular assignment and forwards the features with rank to the feature selection algorithm which is explained in the "Features Selection" section where the method implements multiple features selection algorithms on the basis of the time complexity, dimensions of features, and others.

**Table 4** Datasets

| Total emails | Phishing emails | Legitimate emails |
|---|---|---|
| 3428 | 1824 | 1604 |

### Machine Learning Classification

Several machine classification techniques are used to classify phishing emails from legitimate emails. The classifiers learn from a set of features, which is called training datasets, and predict the output. In this scenario, the method classifies the emails into phishing and legitimate by learning the features from phishing and legitimate emails [3]. In this paper, the proposed method employs random forest classifier [6, 15], which is widely used for phishing email classification and provides a superior accuracy rate. The random forest algorithm is explained as follows:

Random forest builds several decision trees randomly in order to classify a new class. All the trees give votes for that class and choose the classification having the most votes. Assume there is $N$ number of the training sets; then, the $N$ decision tree is made randomly. $M$ is the input variables for testing, and $m < M$ variables are selected randomly from $M$. The best split of these "$m$" is used to split the node.

## Experimental Evaluation

### Data Collection

We gathered a dataset of legitimate emails from csmining group [28] and phishing emails from Jose Nazario's dataset [50] as shown in Table 4.

### Features Extraction

To extract the features for the method, the method employs vector space technique [59]. The vector space

technique utilizes a matrix; each row corresponds to emails $\{d_1, d_2, \ldots d_n\}$, and each column corresponds to the features $F = \{f_1, f_2, \ldots f_m\}$. Each cell in the matrix represents the corresponding feature in the corresponding email; that is, the feature $f_j \in F$ is present in the corresponding email of $d_j \in D$. The method computes the matrix as follows:

$$\begin{matrix} d_1f_1, d_1f_2, \ldots, d_1f_m, b \\ d_2f_1, d_2f_2, \ldots, d_2f_m, b \\ \ldots \quad \ldots \\ d_nf_1, d_nf_2, \ldots, d_nf_m, b \end{matrix} \tag{15}$$

where $b$ is the level in which $b \in \{0, 1\}$, that is, 0 for legitimate level, and 1 for phishing level. In the training dataset, both phishing emails and legitimate emails are collected. The dimension of the features is defined by $\{m \times n\}$.

## Features Selection

This section discusses the features selection algorithms to generate the subset to reduce the number of features and iteration. However, the accuracy would be improved and equal to all features of the feature corpus.

### Sequential Forward Feature Selection (SFFS)

The sequential forward feature selection algorithm adds to the feature's set one by one the high-rank features from the features corpus [42, 56], which is shown in Algorithm 1. The algorithm maintains one threshold value that is the accuracy of the all feature's accuracy ($F$), and initially, the feature set is assigned with empty ($S \leftarrow \emptyset$); afterward, the algorithm adds the features to the features set ($x^+ \leftarrow \max(F_i)$), which have the highest rank and evaluate the accuracy using machine learning algorithm ($\text{Acc}(S + x^+)$). If the current accuracy ($C$) is above the threshold value, then terminate the flow of the algorithm and return the accuracy with the number of features; otherwise, the algorithm is continuously adding the features to the feature set ($S + x^+$).

### Binary Search Feature Selection (BSFS)

The sequential forward feature selection (SFFS) algorithm was presented in study (Sonowal and Kuppusamy [66]). However, the significant issue of the SmiDCA is the acceptance of the sequential forward feature selection algorithm where in every iteration, the features are added to the best feature set one by one. In a situation, the dimension of features is enormous, and then, it expects much time to produce the best feature set.

To handle this issue, this paper introduces a novel algorithm named binary search feature selection (BSFS) algorithm, which explores the best feature set with the least time and better accuracy. The binary search feature selection is inspired by the binary search algorithm which is shown in Algorithm 2. This algorithm initially selects half of the features from the feature's corpus ($(f_a – f_m)$) where $m$ denotes the midpoint that is half of the features; the accuracy is evaluated ( $C \leftarrow \text{Acc}(S + x^+)$). If the accuracy is above the threshold value (the method used the same threshold value of sequential forward feature selection algorithm), then the method examines the first half of the midpoint and upgrades the threshold value with current accuracy and in the same way runs the algorithm. If the method is unable to ascertain the better accuracy than the threshold value, then the method investigates the adjacent half by assigning the midpoint with the initial point with the same threshold value.

## Performance Metrics

The proposed method employs a set of metrics to measure the performance using machine learning classifications. Assume $N_{\text{ham}}$ denotes the number of legitimate emails and $N_{\text{phish}}$ denotes the number of phishing emails. The four parameters used to compute the metrics are as follows: $N_{\text{phish}\rightarrow\text{phish}}$ = TP: number of phishing emails correctly classified by phishing, $N_{\text{ham}\rightarrow\text{ham}}$ = TN: number of legitimate emails correctly classified by legitimate, $N_{\text{ham}\rightarrow\text{phish}}$ = FP:

**Algorithm 1** Sequential Forward Feature Selection

```
SFFS(F_i = {f_1, f_2, ... f_n})
S ← ∅
for i = 1 to i < size(F_i) do
    x^+ ← max(F_i)
    C ← Acc(S + x^+))
    if C ≤ Thld then
        S = S + f_i
        Continue
    else
        Thld ← C
        return Thld
    end if
end for
```

number of phishing emails classified to legitimate, $N_{\text{phish}\rightarrow\text{ham}}$ = FN: number of legitimate emails classified to phishing.

**Algorithm 2** Binary Search Feature Selection

```
BSFS(F_i = {f_1, f_2, ... f_n})
S ← ∅
l ← size(F_i)
a ← 0
for l ≥ a do
    m ← div((a + l), 2)
    x^+ ← (f_a to f_m)
    C ← Acc(S + x^+)
    if C ≤ Thld then
        S ← S + x^+
        a ← m + 1
        go to step 5
    else
        Thld ← C
        S ← S + x^+
        l ← m − 1
        go to step 5
    end if
    return Thld
end for
```

The four performance metrics are shown below:

- *Accuracy* The overall correctly classified accuracy is shown in Eq. (16)

$$\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \tag{16}$$

- *Precision* The precision is shown in Eq. (17)

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{17}$$

- *Recall* The recall is given in Eq. (18)

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{18}$$

- *F1-score* The $f$1-score is given in Eq. (19)

$$F1\text{-score} = 2\frac{\text{Precision.Recall}}{\text{Precision}+\text{Recall}} \tag{19}$$

## Experimental Result

Once the features are extracted, the method employs the three proposed algorithms, namely without feature selection (WFS), sequential forward feature selection (SFFS), and binary search feature selection (BSFS). The result of the experiment is presented in Table 5. The result reveals that the proposed method BSFS offers superior accuracy of 97.41%. Furthermore, the misclassification rate of the proposed method (BSFS) was compared with other methods such as WFS (4.44%) and SFFS (4.37%) where the BSFS required minimum misclassification rate that is 2.59%. Therefore, the feature selection algorithm performs an essential purpose of minimizing the misclassification rate. Further analysis showed that the method uses a minimum number of iterations to determine the best feature set. As a result, the method requires minimum time to detect phishing emails.

Finally, the result of the proposed method is compared with those of the traditional methods as shown in Table 6. It is found from Table 6 that the proposed method performs well, producing good results.

The method computed a precision of 96.24%, a recall of 99.67%, and a $f$1-score of 97.78% of the best feature set. As the precision and recall are inversely proportional to each other, that is, increasing one of them decreases the other one, $F$1-measure is used to evaluate the efficiency of

**Table 5** Feature selection algorithms

| Feature selection algorithms | Number of features | Number of iterations | Accuracy |
|---|---|---|---|
| Without feature selection | 42 | 1 | 95.56 |
| Sequential forward feature selection | 29 | 29 | 95.63 |
| Binary search feature selection | 37 | 6 | 97.41 |

**Table 6** Comparison with other methods

| Methods | Authors | Accuracy (%) |
|---|---|---|
| Obtaining the threat model for email phishing | Olivo et al. [52] | 94.89 |
| Hybrid features detection on phishing email | Hamid and Abawajy [32] | 96.00 |
| Hybrid feature selection approach | Hamid et al. [33] | 94.00 |
| Semantic feature selection | Verma and Hossain [68] | 95.00 |
| Binary search feature selection | Our method | 97.41 |

**Table 7** Efficiency of the proposed model

| Method | Precision | Recall | *F*1-score | Accuracy |
|---|---|---|---|---|
| Binary search feature selection | 96.24 | 99.67 | 97.78 | 97.41 |

the method. Table 7 shows the efficiency of the proposed method as 97.78%.

## Discussion

The paper aims to detect phishing emails using the best feature set that has high accuracy with minimum features. Hence, this paper proposed a method (BSFS) that evaluated better accuracy with minimum features and search time. This section discusses the limitation of all the algorithms on the basis of time complexity, number of features, and accuracy.

The WFS algorithm requires a very less time to ascertain the accuracy of the features; however, the major drawback of the algorithm is accuracy. Comparing the accuracy with other algorithms such as SFFS and BSFS, it has been observed that the other algorithms offered better accuracy, and in addition, the WFS included all the features in their features corpus to evaluate the accuracy. Hence, another issue of this algorithm is feature dimension and this exploration shows that more features are unnecessary to evaluate the optimum solution.

The SFFS evaluated better accuracy than WFS, and one more advantage of this algorithm is the number of features. The SFFS utilized minimum features in comparison with the other algorithms; however, the major limitation of this algorithm is the time complexity. In order to search the best features set, it requires more time because it adds one by one features to the features set and there are a large number of features in the feature space. Therefore, in practice, this is inapplicable to ascertain the optimum solution.

Finally, the proposed algorithm BSFS overcomes these issues as discussed in this section. However, the major limitation of the BSFS is it generates the feature set based on the ranking algorithm; hence, the low-rank feature may evaluate the high accuracy with the combination of other features. The significance exploration of this paper mostly focuses on searching the best feature set with a minimum number of features, time complexity that evaluates the highest accuracy. Therefore, it can be concluded from this exploration is that the BSFS offers the optimum solution to detect the phishing emails with high accuracy and the least number of features.

## Conclusion and Future Work

With the steep increase in the number of phishing emails, many researchers have been developing anti-phishing techniques to reduce the momentum of phishing activities. In this paper, the objective of the proposed method was to ascertain the best features set from the collection of 41 relevant existing features and novel features. This method has employed the features ranking algorithm in order to rank the features and applied it to the features search algorithm to search the best features set. The result of the experiment shows that the BSFS offers the better accuracy (97.41%) than WFS (95.56%) and SFFS (95.63%) and the SFFS algorithm as well provides the better accuracy; however, the time complexity is maximum in comparison with BSFS. From the exploration, it can be concluded that the BSFS is the optimum solution to search for the best feature set with time complexity and minimal features to detect phishing emails.

In the future, more features shall be included and advanced feature selection techniques shall be applied to derive the best feature set.

**Funding** The author received no financial support for the research, authorship, and/or publication of this article.

### Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1. Abdallah EE, Abdallah AE, Bsoul M, Otoom AF, Al-Daoud E. Simplified features for email authorship identification. Int J Secure Netw. 2013;8(2):72–81.

2. Abdelhamid N, Ayesh A, Thabtah F. Phishing detection based associative classification data mining. Expert Syst Appl. 2014;41(13):5948–59. https://doi.org/10.1016/j.eswa.2014.03.019.
3. Abu-Nimeh S, Nappa D, Wang X, Nair S. A comparison of machine learning techniques for phishing detection. In: Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit. ACM; 2007. p. 60–9.
4. Aburrous M, Hossain MA, Dahal K, Thabtah F. Predicting phishing websites using classification mining techniques with experimental case studies. In: 2010 seventh international conference on information technology: new generations (ITNG). IEEE; 2010. p. 176–81.
5. Afroz S, Greenstadt R. Phishzoo: detecting phishing websites by looking at them. In: 2011 fifth IEEE international conference on semantic computing (ICSC); 2011. p. 368–75. https://doi.org/10.1109/ICSC.2011.52.
6. Akinyelu AA, Adewumi AO. Classification of phishing email using random forest machine learning technique. J Appl Math. 2014;2014:425731.
7. Alkhozae MG, Batarfi OA. Phishing websites detection based on phishing characteristics in the webpage source code. Int J Inf Commun Technol Res. 2011;1(6):283–91.
8. Almomani A, Gupta B, Atawneh S, Meulenberg A, Almomani E. A survey of phishing email filtering techniques. IEEE Commun Surv Tutor. 2013;15(4):2070–90.
9. APWG. Phishing activity trends report. http://www.apwg.com/. Accessed Mar 2020.
10. Basnet R, Mukkamala S, Sung AH. Detection of phishing attacks: a machine learning approach. In: Soft computing applications in industry. Springer; 2008. p. 373–83.
11. Basnet RB, Sung AH, Liu Q. Rule-based phishing attack detection. In: International conference on security and management (SAM 2011), Las Vegas, NV; 2011.
12. Benesty J, Chen J, Huang Y, Cohen I. Pearson correlation coefficient. In: Noise reduction in speech processing. Berlin: Springer; 2009. p. 1–4. https://doi.org/10.1007/978-3-642-00296-0_5.
13. Bergholz A, De Beer J, Glahn S, Moens MF, Paaß G, Strobel S. New filtering approaches for phishing email. J Comput Secur. 2010;18(1):7–35.
14. Björnsson CH. Lesbarkeit durch Lix. Stockholms skolförvaltn: Pedagogiskt centrum; 1968.
15. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32.
16. Cao Y, Han W, Le Y. Anti-phishing based on automated individual white-list. In: Proceedings of the 4th ACM workshop on digital identity management. ACM, DIM'08; 2008. p. 51–60. https://doi.org/10.1145/1456424.1456434.
17. Chandrasekaran M, Narayanan K, Upadhyaya S. Phishing email detection based on structural properties. In: NYS cyber security conference; 2006. p. 1–7.
18. Chen C, Wen S, Zhang J, Xiang Y, Oliver J, Alelaiwi A, Hassan MM. Investigating the deceptive information in twitter spam. Future Gener Comput Syst. 2017;72:319–26. https://doi.org/10.1016/j.future.2016.05.036.
19. Chen J, Guo C. Online detection and prevention of phishing attacks. In: First international conference on communications and networking in China, 2006. ChinaCom'06. IEEE ;2006. p. 1–7.
20. Chowdhury M, Abawajy J, Kelarev A, Hochin T. Multilayer hybrid strategy for phishing email zero-day filtering. Concurr Comput Pract Exp. 2016;29(23):e3929.
21. Coleman M, Liau TL. A computer readability formula designed for machine scoring. J Appl Psychol. 1975;60(2):283–4.
22. Cooley S, McCorkendale B. Misspelled word analysis for undesirable message classification. US Patent 2015;8,973,678.
23. Cova M, Kruegel C, Vigna G. There is no free phish: an analysis of "free" and live phishing kits. WOOT. 2008;8:1–8.
24. Fette I, Sadeh N, Tomasic A. Learning to detect phishing emails. In: Proceedings of the 16th international conference on World Wide Web. ACM; 2007. p. 649–56.
25. Flesch R. A new readability yardstick. J Appl Psychol. 1948;32(3):221.
26. Garera S, Provos N, Chew M, Rubin AD. A framework for detection and measurement of phishing attacks. In: Proceedings of the 2007 ACM workshop on recurring malcode. ACM; 2007. p. 1–8.
27. Google-Safe Browsing. https://code.google.com/p/google-safe-browsing/. Accessed Dec 2016.
28. Group C. http://csmining.org/index.php/spam-email-datasets-.html. Accessed Jan 2017.
29. Gunning R. The technique of clear writing. McGraw-Hill; 1952.
30. Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res. 2003;3(Mar):1157–82.
31. Guyon I, Nikravesh M, Gunn S, Zadeh LA. An introduction to feature extraction. Berlin: Springer; 2006. p. 1–25. https://doi.org/10.1007/978-3-540-35488-8_1.
32. Hamid IRA, Abawajy J. Hybrid feature selection for phishing email detection. In: International conference on algorithms and architectures for parallel processing. Springer; 2011. p. 266–75.
33. Hamid IRA, Abawajy J, Kim Th. Using feature selection and classification scheme for automating phishing email detection. Stud Inf Control. 2013;22(1):61–70.
34. Han Y, Shen Y. Accurate spear phishing campaign attribution and early detection. In: Proceedings of the 31st annual ACM symposium on applied computing. ACM; 2016. p. 2079–86.
35. He M, Horng SJ, Fan P, Khan MK, Run RS, Lai JL, Chen RJ, Sutanto A. An efficient phishing webpage detector. Expert Syst Appl. 2011;38(10):12018–27. https://doi.org/10.1016/j.eswa.2011.01.046.
36. Inomata A, Rahman M, Okamoto T, Okamoto E. A novel mail filtering method against phishing. In: PACRIM. 2005 IEEE Pacific Rim conference on communications, computers and signal processing, 2005. IEEE; 2005. p. 221–4. https://doi.org/10.1109/PACRIM.2005.1517265.
37. Islam R, Abawajy J. A multi-tier phishing detection and filtering approach. J Netw Comput Appl. 2013;36(1):324–35. https://doi.org/10.1016/j.jnca.2012.05.009.
38. Jagatic TN, Johnson NA, Jakobsson M, Menczer F. Social phishing. Commun ACM. 2007;50(10):94–100.
39. Jøsang A, AlFayyadh B, Grandison T, AlZomai M, McNamara J. Security usability principles for vulnerability analysis and risk assessment. In: Twenty-third annual computer security applications conference, 2007. ACSAC/IEEE; 2007. p. 269–78.
40. Khonji M, Jones A, Iraqi Y. A study of feature subset evaluators and feature subset searching methods for phishing classification. In: Proceedings of the 8th annual collaboration, electronic messaging, anti-abuse and spam conference. ACM; 2011. p. 135–44.
41. Khorshed MT, Ali AS, Wasimi SA. A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing. Future Gener Comput Syst. 2012;28(6):833–51. https://doi.org/10.1016/j.future.2012.01.006.
42. Kittler J, et al. Pattern recognition. A statistical approach; 1982.
43. L'Huillier G, Hevia A, Weber R, Ríos S. Latent semantic analysis and keyword extraction for phishing classification. In: 2010 IEEE international conference on intelligence and security informatics (ISI). IEEE; 2010. p. 129–31.
44. Ma L, Ofoghi B, Watters P, Brown S. Detecting phishing emails using hybrid features. In: Symposia and workshops on ubiquitous, autonomic and trusted computing, 2009. UIC-ATC'09. IEEE; 2009. p. 493–7.
45. Mc Laughlin GH. Smog grading-a new readability formula. J Read. 1969;12(8):639–46.

46. Mishra S, Soni D. Smishing detector: A security model to detect smishing through SMS content analysis and url behavior analysis. Future Gener Comput Syst. 2020;108:803–15.
47. Moghimi M, Varjani AY. New rule-based phishing detection method. Expert Syst Appl. 2016;53:231–42.
48. Mohammad RM, Thabtah F, McCluskey L. An assessment of features related to phishing websites using an automated technique. In: 2012 international conference for internet technology and secured transactions. IEEE; 2012. p. 492–97.
49. Mohammad RM, Thabtah F, McCluskey L. Tutorial and critical analysis of phishing websites methods. Comput Sci Rev. 2015;17:1–24.
50. Nazario J. https://monkey.org/~jose/wiki/doku.php?id=PhishingCorpus. Accessed Jan 2017.
51. Netcraft. http://toolbar.netcraft.com/. Accessed Dec 2016.
52. Olivo CK, Santin AO, Oliveira LS. Obtaining the threat model for e-mail phishing. Appl Soft Comput. 2013;13(12):4841–8.
53. Owen B, Steiner J. Email filtering system and method. US Patent 2009;7,580,982.
54. Pan Y, Ding X. Anomaly based web phishing page detection. In: 22nd annual computer security applications conference, 2006. ACSAC'06. IEEE; 2006. p. 381–92.
55. Pandey M, Ravi V. Detecting phishing e-mails using text and data mining. In: 2012 IEEE international conference on computational intelligence & computing research (ICCIC). IEEE; 2012. p. 1–6.
56. Pernkopf F. Bayesian network classifiers versus selective k-nn classifier. Pattern Recogn. 2005;38(1):1–10.
57. Phishingorg. http://www.phishing.org/what-is-phishing. Accessed Jan 2017.
58. Radicati S. Email statistics report. The Radicati Group, Inc; 2016.
59. Salton G, Wong A, Yang CS. A vector space model for automatic indexing. Commun ACM. 1975;18(11):613–20.
60. San Norberto EM, Taylor J, Salvador R, Revilla Á, Merino B, Vaquero C. The quality of information available on the internet about aortic aneurysm and its endovascular treatment. Revista Española de Cardiología (English Edition). 2011;64(10):869–75.
61. Senter R, Smith EA. Automated readability index. Technical report, DTIC document; 1967.
62. Shahriar H, Zulkernine M. Trustworthiness testing of phishing websites: a behavior model-based approach. Future Gener Comput Syst. 2012;28(8):1258–71. https://doi.org/10.1016/j.future.2011.02.001.
63. SiteAdvisor M. http://www.siteadvisor.com/. Accessed Dec 2016.
64. Sonowal G, Kuppusamy K. Masphid: A model to assist screen reader users for detecting phishing sites using aural and visual similarity measures. In: Proceedings of the international conference on informatics and analytics. ACM; 2016. p. 87.
65. Sonowal G, Kuppusamy K. Mmsphid: a phoneme based phishing verification model for persons with visual impairments. Inf Comput Secur. 2018a;26(5):613–36.
66. Sonowal G, Kuppusamy K. Smidca: an anti-smishing model with machine learning approach. Comput J. 2018b;61(8):1143–57.
67. Sonowal G, Kuppusamy K. Phidma: a phishing detection model with multi-filter approach. J King Saud Univ Comput Inf Sci. 2020;32(1):99–112. https://doi.org/10.1016/j.jksuci.2017.07.005.
68. Verma R, Hossain N. Semantic feature selection for text with application to phishing email detection. In: International conference on information security and cryptology. Springer; 2013. p. 455–68.
69. Wang J, Herath T, Chen R, Vishwanath A, Rao HR. Research article phishing susceptibility: an investigation into the processing of a targeted spear phishing email. IEEE Trans Prof Commun. 2012;55(4):345–62.
70. Wenyin L, Fang N, Quan X, Qiu B, Liu G. Discovering phishing target based on semantic link network. Future Gener Comput Syst. 2010;26(3):381–8. https://doi.org/10.1016/j.future.2009.07.012.
71. Whittaker C, Ryner B, Nazif M. Large-scale automatic classification of phishing pages. In: NDSS; 2010. p. 10.
72. Yasin A, Abuhasan A. An intelligent classification model for phishing email detection. In: CoRR; 2016. arXiv:abs/1608.02196.
73. Yearwood J, Mammadov M, Banerjee A. Profiling phishing emails based on hyperlink information. In: 2010 international conference on advances in social networks analysis and mining (ASONAM). IEEE; 2010. p. 120–7.
74. Yearwood J, Mammadov M, Webb D. Profiling phishing activity based on hyperlinks extracted from phishing emails. Soc Netw Anal Min. 2012;2(1):5–16.
75. Yu WD, Nargundkar S, Tiruthani N. Phishcatch: a phishing detection tool. In: 2009 33rd annual IEEE international computer software and applications conference; 2009. https://doi.org/10.1109/COMPSAC.2009.175.
76. Zareapoor M, Seeja K. Text mining for phishing e-mail detection. In: Intelligent computing. Communication and devices. Springer; 2015. p. 65–71.
77. Zhang Y, Hong JI, Cranor LF. Cantina: a content-based approach to detecting phishing web sites. In: Proceedings of the 16th international conference on World Wide Web. ACM; 2007. p. 639–48.
78. Zhuge H. Special section: semantic link network. Future Gener Comput Syst. 2010;26(3):359–60. https://doi.org/10.1016/j.future.2009.10.010.