Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2022.030982
Article

URL Phishing Detection Using Particle Swarm Optimization and Data Mining

Saeed M. Alshahrani[1], Nayyar Ahmed Khan[1,*], Jameel Almalki[2] and Waleed Al Shehri[2]

[1]College of Computing and IT, Shaqra University, Shaqra, 155572, Saudi Arabia
[2]Department of Computer Science, College of Computer in Al-Leith, Umm Al-Qura University, Makkah, Saudi Arabia
*Corresponding Author: Nayyar Ahmed Khan. Email: nayyar@su.edu.sa

Received: 07 April 2022; Accepted: 19 May 2022

Abstract: The continuous destruction and frauds prevailing due to phishing URLs make it an indispensable area for research. Various techniques are adopted in the detection process, including neural networks, machine learning, or hybrid techniques. A novel detection model is proposed that uses data mining with the Particle Swarm Optimization technique (PSO) to increase and empower the method of detecting phishing URLs. Feature selection based on various techniques to identify the phishing candidates from the URL is conducted. In this approach, the features mined from the URL are extracted using data mining rules. The features are selected on the basis of URL structure. The classification of these features identified by the data mining rules is done using PSO techniques. The selection of features with PSO optimization makes it possible to identify phishing URLs. Using a large number of rule identifiers, the true positive rate for the identification of phishing URLs is maximized in this approach. The experiments show that feature selection using data mining and particle swarm optimization helps tremendously identify the phishing URLs based on the structure of the URL itself. Moreover, it can minimize processing time for identifying the phishing website instead. So, the approach can be beneficial to identify such URLs over the existing contemporary detecting models proposed before.

Keywords: Phishing; particle swarm optimization; feature selection; data mining; classification; cloud application

1 Introduction

A new trend related to internet scammers called phishing has emerged recently. In this process, a fraudster tries to contact the victim with the help of an email message. The appearance of the message and sender profile appears to be similar to a financial institution. The victim tries to connect with the links provided in the invitation email. The website appears similar to the original website for the financial organization. A similar CSS/HTML/JS element is expected to encounter in this fake URL. Once the user inputs his information into this website process of phishing starts. Depending on the fraudster, the process can take place in three different methods.

Impersonation: A fake website is created by the fraudster. The link to this website is presented via mail sent to the user. When the user inputs his credentials on this site, the credentials are revealed to the fraudster. The original website is opened with them and the user does not even suspect that he has been trapped. Now with the credentials of the user, the fraudster misuses them to cause financial or reputational loss to the genuine user.

Forwarding: The phishing e-mail itself asks for the login details. When these are entered, they lead the user to the original website. The fraudster gets hold of the user credential. However, the hacker does not even have to take the effort of creating a mirror website in this case.

Pop-up: The phishing e-mail contains a URL link, which is the phishing link. It opens the original website with a fake pop-up created by the fraudster when clicked. The pop-up asks for the credentials. The credentials are saved by the fraudster in the database and open the genuine website. The users are redirected to the genuine website and they do not even realize that something has gone amiss or their credentials have been compromised. This attack is not prevalent nowadays as pop-up blockers are available at the browser level.

In the proposed research, the concentration is on phishing caused by impersonation attacks, as these are the most prevalent and frequent attacks. Here, fake or mirror websites are created, which have the complete look and feel of the original. The main task is to distinguish between phishing/malicious sites and genuine sites. The advent of technology and the internet has caused an instant spur in this kind of attack. By looking at these websites, generic users would not be able to make out that it is a phishing website address. The phishing websites ask the users for their account user name and password, which are read by the fraudster. He then uses the credentials to perform malicious operations on the original website. The features selected are analyzed using particle swarm optimization and classification technique. The results retrieved are further tested with various algorithms to confirm their authenticity and accuracy. Finally, the study derives a conclusion and also suggests directions for future works.

2 Background

Recent research has shown that data mining has been used extensively to analyze the different URL features and detect phishing/fake URLs. [1] suggested a very interactive approach to detect web form spam. This technique makes use of fuzzy logic. The topic modeling framework suggested by [2] draws our attention to using the URL structure and developing the model for identifying phishing. Furthermore, [3] proposed the use of deep learning for the detection of spam SMS. However, various approaches did not focus on the data mining technique to detect spamming or phishing. Data Mining is the field of computer science where the computer learns from examples given by the user, where both the input and output are given [4]. This is called the training phase. In the testing phase, the only input is supplied to the system and the output is computed by the system based upon the logic it generates after learning from the examples in the training phase.

Data Mining is used to perform Classification (Supervised learning) and Clustering (Unsupervised Learning) [4]. The system learns from the training phase examples (labeled data) in classification problems and applies the learning logic to the testing data. In Clustering, there is no training phase. The system is directly given the test data (unlabeled data). As there are no previous examples based on which it can classify the data, the system clusters the data into different categories based on the similarity of the data. When Data Mining is applied to the web security area, especially to the phishing sub-domain, Classification (Supervised Learning) techniques are preferred in the literature. Labeled data can be easily generated like genuine and phishing websites with their features. The merging trend in web security research, particularly in phishing website detection, is machine learning techniques. [5] suggested the technique of machine learning classifiers. Classifications using SVM, Bayesian networks, Naïve Bays classifier, etc., are being explored [6]. All this has also been quite fruitful in detecting phishing websites. But the fraudsters are also acquiring knowledge about how this detection is made and, day by day, improving their standards. Thus, tool kits (templates) for creating phishing websites may be rarely used. URLs can also be skillfully crafted to have a minimal deviation from the original. So, the URL-based and image-based classification of phishing sites can be problematic in the future. Techniques like content-based classification [7], which considers the entire website content like the text, hyperlinks and images, seems to be a better idea as making a site that can deceive the security analyst and score to con in all these three aspects are difficult.

A combination of two techniques and a fusion of results is used in this study. Such research shows more precision in detecting phishing websites, as the flaws of one technique need not shadow the results of the other. Even if one method is not able to detect a discrepancy, the other might detect and report it. [8] suggested a brilliant framework for fraud detection in job sites with the help of the gradient boosting method. Applying unique ideas like web phishing detection [9], which is used in philosophy and analyzing websites [10], shows how researchers are thinking out of the box to catch the culprits. Such innovations are needed in this internet era where fraudsters are only one step behind us. The proposed work first carried out a thorough analysis of the phishing URL features and the best-suited feature selection [11] and classification algorithm. Tree-based classifiers are best suited as individual classifiers for this problem. An analysis of using a hybrid methodology for the same issue was done. PSO is being used to adjust the weights of the neural networks to classify phishing URLs.

3 Proposed Architecture

In the proposed system, phishing URLs are recognized by analyzing the URL structure. There is no requirement to click on and put the phishing site. The time required to handle the information and analyze it for any vulnerabilities is thereby reduced. URL web page content need not be intelligently analyzed in this case. Fig. 1 depicts the architecture of the proposed work. During the phase of training the model, the data is passed in the beginning to the training set. This data is classified and made ready to identify the URL's fishing nature. The feature selection takes place after the process of classification. The PSO technique is used to classify the URLs. These classified URLs are used for training artificial neural networks. Once the complete training is done, optimization of the model is done. The results after optimization are forwarded to the decision module, where necessary action is decided from the rule base. The final result is presented to the user after identifying the nature of the phishing URL. For any unknown data received from the interface, the rule base tries to identify the phishing nature of the data. Based on the inferences given by testing data, the rule base information is passed to the decision module.

3.1 Particle Swarm Optimization (PSO) Application on URL Structure

PSO algorithm provides a more robust means to classify data. Unlike genetic algorithms, it does not use any mutation or crossover techniques. Instead, the entire algorithm focuses on collaborating and identifying the similar candidate value from the bulky data. Every candidate in this algorithm is called a particle. A clear fitness function is applied to all the particles, which provides a fitness value. Two values are maintained in the algorithm for every particle, viz "pbest" and "gbest." Gbest represents that value of the fitness function, which yields the hygienist factor amongst all the particles. Pbest corresponds to the highest output value from the fitness function related to the neighborhood of a particle. The main target to achieve in this method is to identify the highest value of pbest and gbest. The main reason to use this type of technique in the study is the nature of finding the most required candidate amongst the various particles available. The fitness function for particle swarm optimization relates as:

$$R^n \to R \tag{1}$$

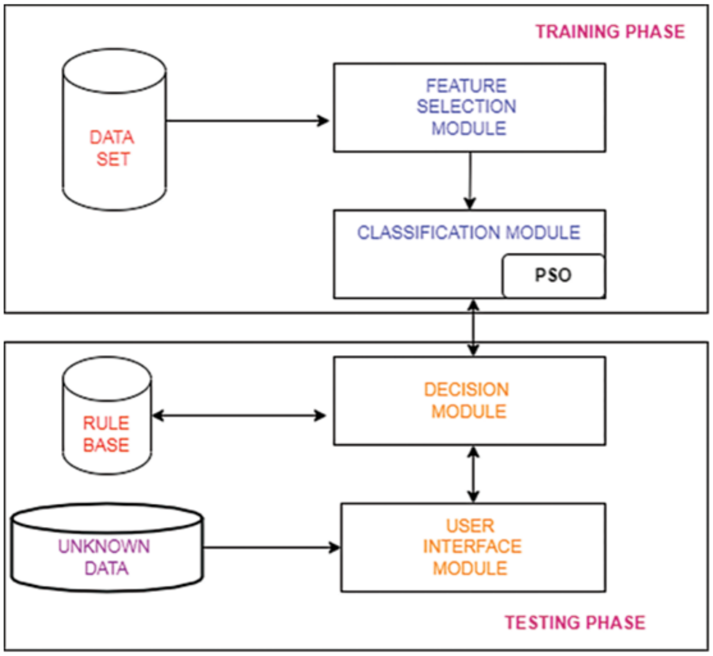

Figure 1: System architecture for phishing URL detection

This function measures the quality of a particular solution existing with the associated value of the particle [12]. All the candidates are individually existing and randomly placed in the hyperplane represented with the position vector x_i, where:

$$x_i \in R \tag{2}$$

The velocity with which a particle moves towards the solution is represented as:

$$v_i \in R^n \tag{3}$$

For any given particle value, there exists a new value, which is supposed to be closer to the desired solution. The successor value for this particle with the changed position after the next iteration is completed to the pbest and gbest. A similar calculation holds good for the particle's velocity from any initial position towards a final position with updated pbest and gbest velocity vectors. Fig. 2 provides a simple illustration of the particle's motion in the hyperplane. The new locations for this particle depend upon the fitness function. Based on the pbest and gbest values, the most optimum result is identified after necessary iterations.

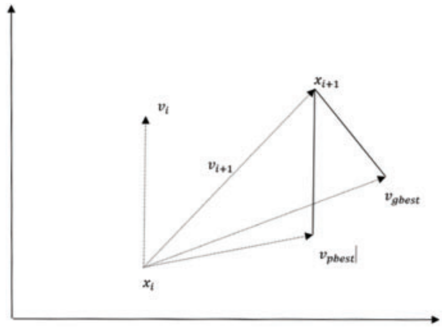

Figure 2: A particle's optimum solution based on "Pbest and Gbest values"

The velocity function, which is responsible for the movement of a particle towards its most optimum solution value, is given as:

$$v_i^{k+1} = wv_i^k + c_1rand_1\left(\ldots..\right) x \left(pbest_i - s_i^k\right) + c_2rand_2\left(\ldots..\right) x \left(gbest_i - s_i^k\right) \tag{4}$$

Here, w is the weight value; c_i corresponds to the weighting factor for the particle r and$_i$ is a uniformly distributed value between 0 and 1. Then, the wait for the next upcoming value is calculated with the help of the equation:

$$w = w_{initial} - \left[\left(w_{initial} - \; w_{final}\right) i\right] / max\left(i\right) \tag{5}$$

$w_{initial}$ is termed as the initial weight, w_{final} is called the final weight and max (i) is the maximum possible iterations that can take place in this case. Finally, the position of a particle in the hyperplane is represented with the help of the equation as:

$$x_i^{k+1} = s_i^k + v_i^{k+1} \tag{6}$$

When all the fitness function values corresponding to the particles are calculated, the groups of particles are identified as Swarms. These groups are expected to travel towards the optimum solution with the velocity vector. This one, which is most likely to reach the destination, is selected as the most suitable candidate or precisely the optimum value. The methodology proposed in this schema uses the PSO technique for adjusting the weights of the underlying artificial neural network. By using the global optimization toolbox in MATLAB, significant results can be achieved. The proposed algorithm for this technique is depicted in the pseudocode:

Pseudo Code for Selecting Particle Values in PSO.

```
Initialize the Particle Values from all existing values
    For all the Particles calculate value.fitnessFunction( )
      if value.fitnessFunction( ) > pbest
            pBest = value.fitnessFunction
    For all Particles [in range] value.fitnessFunction
      if p(i) = maximum [all values]
            gBest = max ( ) value
    For all the Particles [in range] velocity value
      If value.VelocityVector = max (All velocity values)
      Max (velocity) = value.VelocityVector
  Store pBest, Gbest
  Store Max(Velocity)
End
```

The vital parameter for this problem is the fitness function for valuation. It is determined on the miscalculation rate of the artificial neural network system.

3.2 Dataset

In the URL, www.abc.com/xyz, abc is the hostname, .com is the top-level domain, and xyz is the path. In many cases, phishing is done by trying to fake a famous and frequently used website. Users tend to overlook some minute discrepancies and click and enter the malicious site. For example, consider the URL [http://www.abcxyz.com/https.Paypal.com/secure.paypalv/login.php] appears to be a link to the paypal.com site. But a closer examination shows that Paypal.com is present in the path part of the URL and the actual hostname is abcxyz. This is a phishing site that, when clicked, leads to a fake page. The deceived user would unknowingly fill in the details on the site. This credential is passed on to the fraudster who uses them to log in to the real PayPal site and commit the fraud. Therefore, phishing URLs can be identified using their structure. Hence, phishing URLs are recognized in this proposed system by analyzing the URL structure without entering the phishing site.

For classification in this study, 10,000 URLs are collected. This entire set of information comprises of 6000 genuine connection links and 4000 phishing URLs. These links are under the Public license of the DMOZ repository. The data and information are available for ethical use. These data repositories are considered as one of the gigantic directories of digital data on the web [13]. The dataset comprises of the URLs that are tested manually for authenticity. A subset of the fishing URLs is taken anonymously from [14], considered a community-based repository. Users of various regions contribute to the fake URLs and vote for the authenticity and ethical nature of the URL site. The fishing nature of different URLs present in this repository is applied by various well-known sites like Kaspersky, Yahoo, Vimeo, etc., to restrict the fake URLs. Some famous methods like lexical analysis, domain-based, network-oriented and feature-based URL identification techniques have evolved recently. Almost 27 features fall under the category of these methods. Certain features like age, number of dots, security-sensitive word presence, etc., were calculated during this study. For the introspection of the existing set of URLs, they were classified and broadly divided into four different domains as suggested in [15]. The macro classification contains-Gaming, Banking, News and Advertising and Online Shopping websites. Fig. 3 represents the classification of the URLs:

For the training phase, an initial set of 250 URLs was taken (comprising of 125 fake and 125 real authentic URLs). For the second training phase, another set was taken containing 500 URLs divided into 250 fake and 250 genuine. Subsequently, the third set comprises of 1000 and the fourth set comprises of 2000 URLs for training purposes. The final set, makes use of 6000 genuine and 4000 nongenuine URLs.

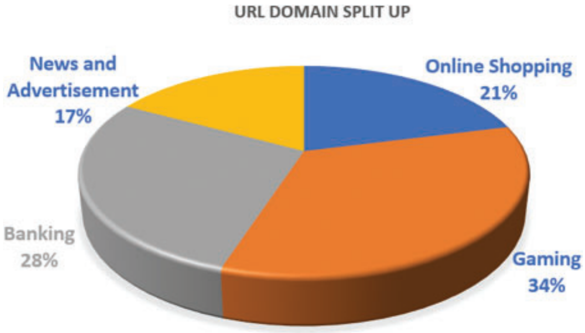

Figure 3: Domain split on sample URL’s

3.3 Features

The standard approach for producing phishing URLs is with the help of bot programs. These programs try to generate various phishing links that refer to a target website URL. Just in case one of the URLs is identified as phishing, a parallel copy of the variant URL from the bot program gets activated as the successor of the URL, which is referred to as fake. Negligible change in the URL structure occurs, which is difficult to identify easily. One of the factors related to the bot programs refers to the similarity of URL structure [16]. This becomes the main point to identifying the phishing nature of the URL along with its sister concern URL. On a general note, the appearance and key features of all the URLs created by a similar bot program appear approximately the same. The feature in Tab. 1 are used in the current study to avert fake URL phishing.

Table 1: List of features

Attribute	Data type
IP address presence	Nominal {0, 1}
Unknown noun presence	Nominal {0, 1}
Suspicious URLs	Nominal {0, 1}
Out of position top level domain	Nominal {0, 1}
No of dots in the URL	Numeric
Security sensitive word presence	Nominal {0, 1}
No of links to this site	Numeric
Real traffic rank of the site	Numeric
Age of the domain	Numeric
Genuine	Nominal {Y, N}

3.3.1 Lexical Features

Use of IP Address: quite often, when a URL is created, a name server-oriented domain name system is used to provide the name for the website. But in the case of fake URLs use of IP addresses is widespread. The domain name does the masking of the IP address for genuine URLs. This lacks in fake URLs as such. The presence of an IP address [17] represents a probabilistic chance for the URL to be fake. As an example, where IP address is present in the URL itself is shown below:

http://185.28.22.67/bchileperfilamiento/Process?MID=&#x0026;AID=LOGIN-0004&#x0026;RQI=5001435125BE97 Unknown Noun Presence: This study portrays the new "Unknown noun Presence" technique. e.g., [http://emmmhhh.ru:8080/forum/links/column.php]. A close examination of the URL structure focuses on identifying random letters towards the beginning of the URL. The creation of domain names is not done with such kinds of words or alphabets. Usually, they are common or proper nouns representing an organization or real-world entity. The classification of such characters helps identify the URL's phishing nature.

Count the number of dots present in URL: Certain studies [18] figured out that the availability of multiple dots in a URL structure can be considered as a phishing URL. Therefore, this parameter has also been included in the present study.

3.3.2 URL Based Features

Three features from the URL are extracted in this work.

Presence of Security Sensitive Word: If the URL has any of the following words, confirm, account, banking, secure, web-src, login, and sign-in, then the URL can be classified as phishing as per earlier works [17].

Suspicious Symbol Presence: Programmatically, the use of the "@" symbol is done with text and email addresses. It is also worth mentioning that the text before is supposed to be ignored whenever this symbology is used. e.g., www.paypal.com@abc.com. Even though this looks like the link to paypal.com, the user is taken to abc.com [18]. Furthermore, the (-) symbol, also termed as (dash: -) in various websites, is discouraged.

Misplaced Top Domain: e.g., http://a9s7px4x2ys3ciy4x.0pu.ru/https/www.paypale.fr/Client/754198204/

A close look analysis of the URL given above shows that the URL seems to derive from the famous PayPal. However, the misplacement of the domain is done, which refers to a hypothetical fake domain giving rise to phishing [17]. It is also worth mentioning that the word PayPal is also misspelled in the example. So, it makes a clear indication of a fake URL.

3.3.3 Network-Based Features

URL Site connections: It is most likely that if a URL is connected to a large number of pages, then it is also genuine [19].

Traffic Received: certain websites measure the incoming and outgoing traffic once they are connected to a specific URL example, Alexa (a subsidiary of Amazon.com). The data collected twice such services can help identify phishing sites [19]. Once a website is marked as fake, the traffic generated reduces to a large extent.

3.3.4 Domain-Based Features

Domain Age: Various phishing websites are reported and blocked in a concise span of time. The domain creation date can be easily monitored in the WHOIS properties. It can be derived that if the site is older, its chances of being phished will be lesser [19].

4 Experimental Evaluation

4.1 Feature Selection

To improve the results of classification, feature selection has been employed. This study selects the most relevant features. By using feature selection, redundant data is removed and accuracy is also improved. The problem of over-fitting is also eliminated. WEKA tool has been used to perform feature selection and classification. The feature selection techniques [20] used are:

4.1.1 Subset Valuation

The prediction ability and degree of redundancy for all the considered features are used to calculate the weight of each subset of features. There is a high correlation between all the subsets [20]. Every feature selection technique is used along with a search algorithm. Here, the best First search is used along with the Subset valuation. The attributes selected using this feature selection mechanism are

- Security sensitive word presence
- Unknown noun Presence
- Out of positioning Top Level Domain
- Age of the domain
- Suspicious URLs
- IP Address Presence
- Number of links to this site

4.1.2 Correlation Attribute Valuation with Ranker Search

This algorithm evaluates the importance of an attribute by measuring its correlation with the other attributes in the class. The weighted average is calculated to determine the overall correlation. The merit value for a subset feature S having n features is given by:

$$Merit\ S_n = \frac{na_{cf}}{\sqrt{n + (n-1)\, a_{ff}}} \tag{7}$$

The correlation attribute valuation (CAE) is defined as:

$$CAE\ Factor = s_n \left[\frac{a_{cf1} + a_{cf2} + a_{cf3} + a_{cf4} + \cdots + a_{cfn}}{\sqrt{n + 2(a_{f1f2} + a_{f2f3} + a_{f3f4} + \cdots a_{fifj} + a_{nf1}}} \right] \tag{8}$$

where the correlations are defined in [16,20]. The top five attributes selected in this scenario are:

- Security sensitive word presence
- Unknown noun Presence
- Dot's pattern/reoccurrence in the URL
- Out of position in the Top-Level Domain
- Age of the domain

4.1.3 Gain Ratio Attribute Valuation with Ranker Search

This parameter evaluates attribute importance value by comparing the gain ratio [20] in accordance with the class.

$$Gain\ Ratio\ (Class,\ Attributes) = \frac{(H\,(Class) - H\,(Class/Attributes\,))}{H\,(Attributes)} \tag{9}$$

The top five attributes selected using this technique are:

- Security sensitive word presence
- Out of Top position Level
- Unknown noun Presence
- Number of dots in the URL
- Number of links to this site

4.1.4 Information Gain Attribute Valuation with Ranker Search

This parameter valuates the worth of an attribute by comparing information gain [20] concerning the class.

$$InfoGain\ (Class,\ Attributes) = H\ (Class) - H\ (Class/Attributes) \tag{10}$$

The top five attributes selected using Information gain feature selection:

- Number of links to this site
- Security sensitive word presence
- Occurrence of multiple dots in the URL.
- Unknown noun Presence
- Out of expected position
- Top-Level Domain

4.1.5 Statistical Results

Hypothesis- The "Unknown Noun" feature that has been proposed in this research work is consistently among the top five features during feature selection. This hypothesis has been tested using chi-square and t-test attribute selection methods. The T-test is used to test if the sample means significantly differs from the hypnotized value. The implementation of these two feature selection mechanisms was done in the Tanagra tool. The features that were selected from the t-test are:

- Security sensitive word presence
- Unknown noun Presence
- Out of position Top Level Domain
- Age of the domain
- Suspicious URLs

Chi-square is a standard feature selection algorithm that has been used to rank the features in the order of relevance by comparing the observed and hypothetical proportions of a value. The features that are delivered as output for this test are:

- Security Sensitive Word Presence
- Unknown Noun
- Out of Position Top Level Domain
- Suspicious URLs
- IP Address Presence

After performing Correlation, Gain Ratio and Information Gain feature selection, the attributes are ranked as shown in Fig. 4.

It is observed that the new feature proposed, Unknown Noun Presence, is ranked among the top 3 features in all the feature selection techniques.

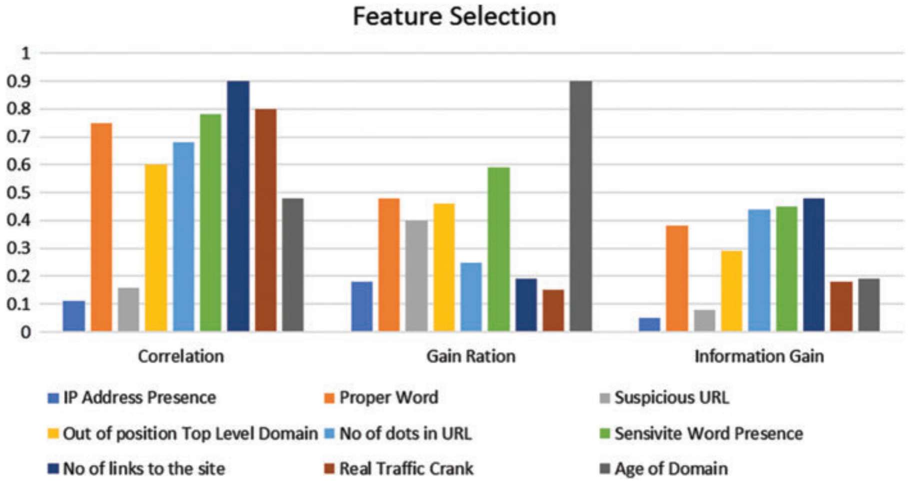

Figure 4: Features selected after applying feature selection algorithms

4.2 Classification

The data have been thoroughly scrutinized and refined. Now classification is performed on this data. First, the feature selection process was executed using the WEKA tool. Then, the classification process in conjunction with PSO was implemented using De Jong's fifth function in the MATLAB from the Global optimization toolbox [21]. The classification techniques exploited to analyze the better feature selection technique are Multi-layer Perceptron (MLP) and Random Tree.

The accuracy of both the classifiers after applying the different feature selection techniques is shown in Tab. 2. The accuracy has improved after applying feature selection. Random Tree provides better accuracy when compared to MLP. From Tab. 3, it can be inferred that the time taken from the classification is also less when a random Tree is used. Based on this result, it can be concluded that random Tree gives better results with information gain as the feature selection criteria. To ascertain the improvement of accuracy because to the inclusion of the new feature, classification of the data is first performed without including the new feature and then compared with the accuracy obtained after including the feature. Tab. 4 shows that the accuracy has improved significantly with the inclusion of the proposed feature, Unknown Noun Presence in the features.

Table 2: Classifier accuracy

Accuracy (with cross validation 10 folds) in %	Without feature selection	Subset valuation	Correlation	Gain ratio	Information gain
MLP	92.83	92.63	92.1	91.84	92.53
Random tree	93.63	93.63	93.23	93.7	93.77

Table 3: Time taken for classification

Time taken (in seconds)	Without feature selection	Subset valuation	Correlation	Gain ratio	Information gain
MLP	2.97	2.26	1.83	2.68	1.27
Random tree	1.02	0.69	0.63	0.53	0.50

Table 4: Classifier accuracy with and without including the proposed feature (Unknown noun presence)

Accuracy (with cross validation 10 folds) with random forest classifier	Subset valuation	Correlation	Gain ratio	Information gain
Without unknown noun feature	92.067%	92%	92%	92.93%
With unknown noun feature	93.63	93.63	93.23	93.7

WEKA tool is used to classify the data after feature selection using Naïve Bays, Multi-layer Perceptron, J 48 Tree, LMT, Random Forest, Random Tree, C 4.5, ID 3, C-RT and K-Nearest Neighbor algorithms [22].

The classification accuracy, precision, and recall values are higher for the Tree-based classification algorithms than the other frequently used algorithms from Tab. 5. Therefore, the subdomains of the data are loaded to the Tree-based classifiers for further study.

Table 5: Classifier accuracy

Classification algorithm	Training accuracy (%)	Cross validation (10-fold) (%)	Cross validation (3 folds) (%)	Leave one out (%)
Naïve bays	89.73	89.63	88.16	89.73
J 48 tree	93.3	93.46	92.83	92.26
LMT	94.16	94.86	93.13	93.53
Random forest	95.5	95.07	95.17	95.93
MLP	92.53	92.83	91.8	91.63
Random tree	95.6	95.63	96.67	96.4
C 4.5	92.97	91.07	91.3	91.93
ID 3	91.17	90.33	93.87	92.13
C-RT	92.7	91.53	92.47	91.97
K-nearest neighbor	92.6	92.5	92.47	92.5

Classification accuracy (Fig. 5) for the different categories of phishing URLs is around 94% to 95% in various subdomains. This leads to the conclusion that the classification of phishing URL can be best achieved with the help of Tree-based classifiers (Fig. 6).

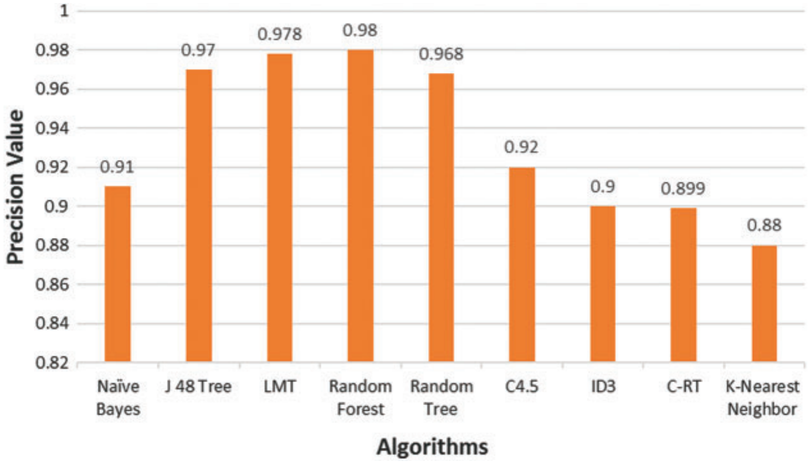

Figure 5: Comparison of precision for various algorithms

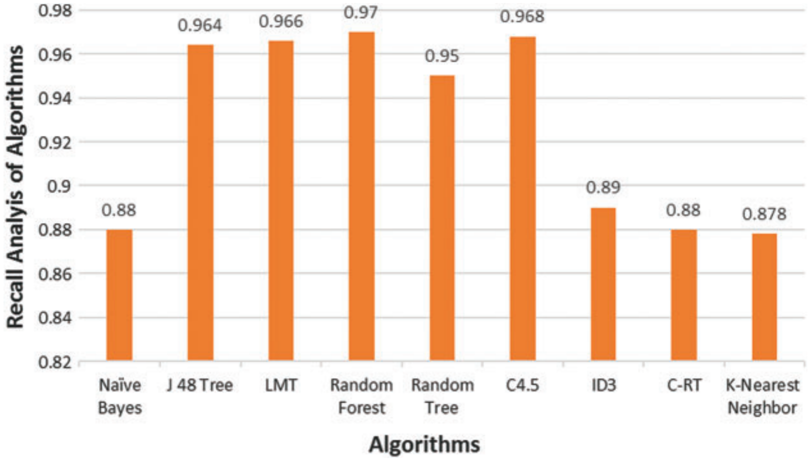

Figure 6: Recall analysis of different classification algorithms

Table 6: Classifier accuracy–URL domains

Domains classified	Accuracy (%)			
	J48 tree	LMT	Random forest	Random tree
Gaming section	92.98	92.97	95.02	94.89
Banking section	93.28	93.63	95.39	97.99
News and advertising	93.9	93.7	94.7	94.82
Online shopping section	93.70	94.69	95.68	94.99

Classification with PSO

In this part of the study, the impact on the classification by introducing the PSO algorithm is computed.

Table 7: Classification techniques comparison for PSO

Algorithm	Accuracy (%)	False positive rate
Naïve bays	88.16	0.6
K-nearest neighbor	92.47	0.5
ID tree	93.87	0.45
SVM	91.5	0.58
NN with PSO	98.7	0.21

In the observation set in Tab. 6 and Tab. 7 above, the false-positive rate and the accuracy are studied for classification technique algorithms, where PSO adjusts weights.

There is a considerable increase in accuracy with the help of PSO as per Fig. 7. The false-positive rate is reduced. If we consider the domain, the classification responding to FP rate is essential. The precision and recall values are shown in Fig. 8 and from this value, it can be inferred that as the dataset size increases, the values become more elevated, which is a good indication.

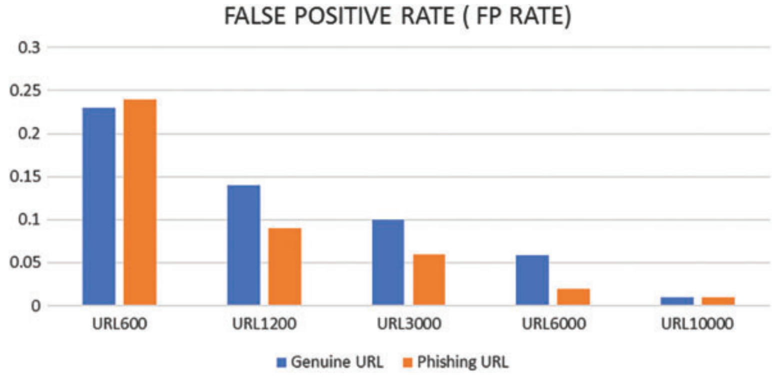

Figure 7: Graph showing the false positive rate

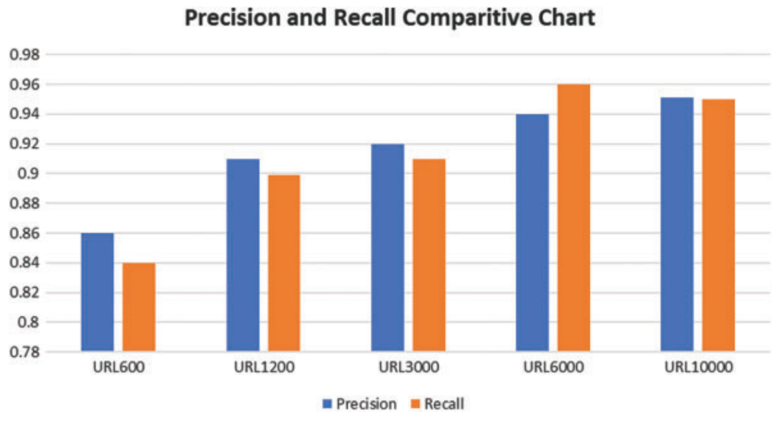

Figure 8: Graph showing the precision and recall values

5 Conclusion

Phishing is a problem that is constantly troubling internet security analysts. New attacks keep sprouting despite current research being carried out in this field. Extensive research needs to be performed in this field to bridge the gap. In the proposed methodology, certain unique features have been selected and the accuracy has improved by using feature selection techniques. The time taken to perform the model building and then the classification is also reduced considerably. The application of hybrid methods like a combination of PSO with neural networks has given better results when compared to the traditional classification techniques. The data mining technique applied in this study provides good results and performance in identifying URL phishing. Classification of the dataset is done with the help of machine learning algorithms to find the best possible features. These features are trained with a machine learning model. The dataset training was completed using various algorithms and the results are explained. A collective comparison is made and results are recorded to identify the performance of the proposed model. The precision values received by the model's help were satisfactory and acceptable. The model yields a substantial decrease in the false-positive rates of the phishing URL structure based on the features selected by the classification techniques. Almost all the classifiers have given more than 91% results in identifying the URL phishing under this model. This is a considerable result and it provides more than 98% accuracy in identifying the phishing nature of the URL. The model is sufficient to prove the best results, but more enhanced algorithms from data mining can be applied as future work to the existing model. The study identifies only a limited future for feature selection and there can be more improvement to the features available. The model is not yet tested with more classification algorithms and this can be a further next level of study in the future. Processing time for identifying URL phishing is also one of the future aspects of this study.

Acknowledgement: Authors would like to thank College of Computing and IT, Shaqra University for their support during this study. We would like to extend our thanks to the entire department and faculty of Computer Science for their motivation.

Funding Statement: The authors would like to thank the Deanship of Scientific Research at Shaqra University for supporting this work.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] P. Kaur and A. Gosain, "An intelligent oversampling approach based upon general type-2 fuzzy sets to detect web spam," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3033–3050, 2021.

[2] S. Abri and R. Abri, "Providing a personalization model based on fuzzy topic modeling," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3079–3086, 2021.

[3] O. Karasoy and S. Balli, "Spam SMS detection for turkish language with deep text analysis and deep learning methods," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1–17, 2021.

[4] S. Durugkar, R. Raja, K. Nagwanshi and S. Kumar, "Introduction to data mining," in *Data Mining and Machine Learning Applications*, John Wiley & Sons, pp. 1–19, 2022.

[5] V. Gaur and R. Kumar, "Analysis of machine learning classifiers for early detection of DDoS attacks on IoT devices," *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1353–1374, 2022.

[6] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel *et al.,* "A survey of intrusion detection techniques in cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[7] A. Abbasi, F. M. Zahedi and S. Kaza, “Detecting fake medical web sites using recursive trust labeling,” *ACM Transactions on Information Systems*, vol. 30, no. 4, pp. 1–36, 2012.

[8] A. Mehboob and M. S. I. Malik, “Smart fraud detection framework for job recruitments,” *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3067–3078, 2021.

[9] E. Rashidy, A. Mohamed, “A smart model for web phishing detection based on new proposed feature selection technique,” *Menoufia Journal of Electronic Engineering Research*, vol. 30, no. 1, pp. 97–104, 2021.

[10] S. Wen, Z. Zhao and H. Yan, “Detecting malicious websites in depth through analyzing topics and webpages,” in *Proc. of the 2nd Int. Conf. on Cryptography, Security and Privacy*, Guiyang, China, pp. 128–133, 2018.

[11] B. Frénay, G. Doquire and M. Verleysen, “Estimating mutual information for feature selection in the presence of label noise,” *Computational Statistics & Data Analysis*, vol. 71, pp. 832–848, 2014.

[12] C. Kolias, G. Kambourakis and M. Maragoudakis, “Swarm intelligence in intrusion detection: A survey,” *Computers & Security*, vol. 30, no. 8, pp. 625–642, 2014.

[13] G. Matošević, J. Dobša and D. Mladenić, “Using machine learning for web page classification in search engine optimization,” *Future Internet*, vol. 13, no. 1, pp. 9, 2021.

[14] I. Saha, D. Sarma, R. J. Chakma, M. N. Alam, A. Sultana *et al.,* “Phishing attacks detection using deep learning approach,” in *Third Int. Conf. on Smart Systems and Inventive Technology (ICSSIT)*, India, pp. 1180–1185, 2020.

[15] F. Bozkurt, “A comparative study on classifying human activities using classical machine and deep learning methods,” *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1507–1521, 2022.

[16] B. Senliol, G. Gulgezen, L. Yu and Z. Cataltepe, “Fast correlation based filter (FCBF) with a different search strategy,” in *Int. Symp. on Computer and Information Sciences*, Istanbul, Turkey, pp. 1–4, 2008.

[17] G. Xiang, J. Hong, C. P. Rose and L. Cranor, “Cantina+ a feature-rich machine learning framework for detecting phishing web sites,” *ACM Transactions on Information and System Security*, vol. 14, no. 2, pp. 1–28, 2011.

[18] A. Chonka, Y. Xiang, W. Zhou and A. Bonti, “Cloud security defense to protect cloud computing against HTTP-DoS and XML-DoS attacks,” *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1097–1107, 2011.

[19] R. Basnet, S. Mukkamala and A. H. Sung, “Detection of phishing attacks: A machine learning approach,” in *Soft Computing Applications in Industry*, Berlin, Heidelberg: Springer, pp. 373–383, 2008.

[20] E. Blessie, C. Chandra and E. Karthikeyan, “Sigmis: A feature selection algorithm using correlation based method,” *Journal of Algorithms & Computational Technology*, vol. 6, no. 3, pp. 385–394, 2012.

[21] K. Gu, N. Wu, B. Yin and W. Jia, “Secure data query framework for cloud and fog computing,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 1, pp. 332–345, 2019.

[22] Y. Alotaibi, “A new database intrusion detection approach based on hybrid meta-heuristics,” *Computers, Materials & Continua*, vol. 66, no. 2, pp. 1879–1895, 2021.