Detecting Phishing Cyber Attack Based on Fuzzy Rules and Differential Evaluation

Rawaa Mohammed Abdul-Hussein [1], Ahmed H. Mohammed [2], Amal Abbas Kadhim [2]

[1] *Computer Engineering Department, Engineering College, Mustansiriyah University, Baghdad, Iraq*
[2] *Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq*

***Abstract* - Cyber-criminal attacks witnessed various phishing attacks that blindly attract users to use their secret personal information. This research highlights this type of attack and proposes the development of a new model to detect phishing websites. Two aspects of improvement were followed. Firstly, a new feature is inferred from a combination of features with higher classification capability than the original features. Secondly, a novel method has been proposed by using differential evaluation (DE) to select the best rules among the fuzzy rule set; the DE is a viable way to optimize the fuzzy rule, improving the classification complexity. The proposed system was evaluated by applying two malicious and legitimate datasets collected from Phishtank and Alexa websites. Results show the combination feature can include a promising method for detecting phishing attacks according to the evaluation metrics in the precision of 0.973, recall of 0.933, F-measure of 0.947, and accuracy of 0.976.**

***Keywords* - phishing, cyber-attack, fuzzy rule, differential evaluation.**

DOI: 10.18421/TEM112-07
https://doi.org/10.18421/TEM112-07

Corresponding author: Ahmed H. Mohammed,
Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq.
Email: dr.ahmedh@uomustansiriyah.edu.iq

Received: 22 January 2022.
Revised: 29 March 2022.
Accepted: 05 April 2022.
Published: 27 May 2022.

1. Introduction

Today, internet applications play a crucial role in every aspect of our life, such as education, online shopping, online software services, and entertainment. Consequently, all of the user's vital information, including their online banking account, is controlled by a dedicated secure account [1]. As a result, cybercrime attracts intruders that try to mimic a legitimate website that is remarkably in the same style as the original website. The phishing cyber-attack convinces users to enter their accounts into legitimate, faked electronic pages [2]. This phenomenon is inspired by hunting or fishing in the water. However, instead of setting the bait on the hook, the hacker prepares a faked page that entices the user to enter his accounts or other confidential information [3]. Eventually, the hacker will acquire all the credentials and sensitive information and use it for malicious purposes [4]. This type of attack is called web phishing, affecting financial transactions or access to personal information about the users. The phisher will embed images, HTML code, and other keywords referring to the legitimate target website. The attacker begins by compromising the server to host his phished web pages pretending to be a legitimate website [5]. Furthermore, an email is sent to a specific receiver with particular concern to the attacker. Finally, the victim opens this imitative page that asks the user to enter his user name or password through click on a link that will relate all user information to the spoofed websites [6].

Moreover, the phisher may put attachments used to inject a spyware program into the user's computer to mine sensitive information [7]. This type of attack showed a considerable increase in recent years [8]. Therefore, light should be shed on this type of attack to protect users from its effects, primarily on e-commerce applications and overall business transactions [9]. The term phishing comes up from fishing to trap someone in an attacker's area. The number of unawareness with security concerns on the internet has been increased recently [10]. Therefore, this attack targets humans instead of machines.

Additionally, the last technology gives the attacker an advanced phishing kit that helps spread the faked pages [11].

The reason behind the evolvement of this attack is its dependency on social engineering techniques [12]. Following the recent statistics of the anti-phishing working group, It is possible to deduce that there is a noteworthy development in the number of phishing websites that can be detected per month [13]. According to the investigations done by the Internet Crime Complaint Center, they declared that the frequency of phishing attempts has grown considerably in the recent few years with over a 1.2-billion-dollar in losses. Moreover, their report stated that people who have been phished with hypertext transfer protocols increased noticeably [14]. One of the most effective methods to decrease the negative impacts of trapping in this attack is to increase the user's awareness of the attackers' methods to recognize and prevent faked links [15]. Consequently, there is a significant necessity for anti-phishing techniques to recognize phishing web pages from legitimate ones. Web pages can be represented by multiple sets of critical terms repeated over web pages such as text, title, and domain name [16].

The contribution of this paper can be summed up twofold:

1. A novel feature has been generated by combining the phishing data set to introduce more efficient features; these inferred features increase the accuracy and the speed of processing by eliminating the less important features.
2. Appling fuzzy criteria are utilized to detect phishing websites using the adopted data set. A novel method has been proposed by using differential evaluation to select the most appropriate rules from among the fuzzy rule set, improving the classification complexity.

The rest of this paper is organized as follows. Section 2 presents the related work; then Section 3 elaborates on preliminary concepts related to fuzzy logic. In Section 4, the proposed SMS method is introduced. The results and discussion of the proposed method are clarified in Section 5. Finally, Section 6 provides the conclusion and recommendations for future research

2. Related Work

Cyber-attacks are very complicated, such that there is no decisive approach to prevent all types of attacks effectively. Essentially, there are two basic kinds of security approaches improving user awareness and using additional programmed tools [17]. Over the years, a series of methods in the last decade for identifying phishing websites have been developed to respond to different forms of phishing schemes intended to get people to provide their personal information. Phishing cyber-attack may generally be detected using one of many approaches, including list-based detection, machine-based learning detection, heuristic detection, or deep learning methods [18].

However, more new approaches have taken machine learning algorithms and adapted them to assess a website's validity. In this section, recent detection approaches have been surveyed to strengthen the detection of phishing websites by the use of machine learning techniques .

Rajithaet and VijayaLakshmi (2016) presented Oppositional Cuckoo search and fuzzy logic classification to detect malicious cyber-attack. In the feature selection stage, the OCS algorithms have been used to select the essential features among the four types of features. The chosen features are trained with FLC in the second stage to determine the fuzzy score. Finally, the testing stage uses a fuzzy score to detect malicious URLs [19].

Rao et al. (2020) suggested a heuristic strategy using a twin support vector machine. This method detects malicious phishing sites registered on vulnerable servers by matching the difference between hyperlink and URL features for both the URL of the visited page and the home page to classify phishing websites [20]. Tan et al. (2020) have extracted a new characteristic to maximize the precision of phishing detection. The main stage of the suggested process begins with hyperlinks extraction of the webpage and a set of associated webpages. The page linking information was collected during this process to construct a web graph to generate a classifier to identify phishing web pages [21]. The research work conducted by ALI et al. (2020) detects various features' weighting based on the particle swarm optimization (PSO) technique, it is possible to detect phish websites. The results of their experiments revealed that proposed machine learning models improve the accuracy with which phishing websites are identified using fewer website features [22]. Aljofey et al. (2020) have used convolutional neural network (CNN) for detecting phishing websites. URL strings are captured without having previous knowledge about phishing. They then use the sequential pattern functionality to speed up the current URL classification [23]. Wei et al. (2020) suggested a deep neural network with convolutional layers. This work relies only on evaluating URL text to detect malicious URLs. Contrary to the previous studies, this method detects zero-day attacks faster. It can also be used on handheld devices without significantly impacting their performance [24]. FENG et al. (2020) proposed a hybrid deep learning network to detect phishing websites, including extraction and representation paradigms. Firstly, the model looks at HTML, DOM (Document Object Model) and URL architectures as a string of characters. Secondly, a representational technology is used to learn webpages' representation automatically; then, these representations are passed to a deep learning hybrid network consisting of a bidirectional memory network and a coevolutionary neural network for retrieving local and global features [25].

Anupam and Kar (2020) attempted to use various features (IP address length, HTTP request) of URL to portend if a website was legitimate or not depending on a support vector machine (SVM) binary classifier. Besides the assistance of four optimization algorithms, the Firefly, the Bat, Grey Wolf, and the Whale are presented to find the optimal hyperplane of the SVM [26]. Mahdavifar and Ghorbani (2020) have built a knowledge base by using Deep Embedded Network Expert Systems (DeNNeS) to extract refined rules from a trained deep network (DNN) architecture in order to replace the expert system's knowledge base [27].

Kumar and Indrani (2020) suggested a phishing detection technique using a deep neural network classifier and fuzzy logic to classify websites into three types, Phishing, Non-Phishing, and Suspicious through the use of the optimum collection of features and rules. Also, a greedy selection algorithm (GSA) has been developed by the Frequent Rule Reduction algorithm (FRR) to detect the best subset of rules that have an efficient prediction of phishing websites [28]. Sankhwar et al. (2020) have proposed an anti-phishing model for an enterprise using an artificial neural network. Two ANN (Levenberg-Marquart and feed-forward backpropagation) approaches are used to develop the URL classification processes and generate outcomes with inaccurate data on social characteristics using the Fuzzy Inference System. This methodology is successful in determining if phishing emails are known or unknown to minimize phishing cyber-attack [29]. Tharani and Arachchilage (2020) showed and discussed the collection of phishing URLs to discover the technique used to simulate It has the potential to attract people to perform malicious things, such as clicking on false links. Twofold machine learning (ML) methods and IG and Chi-Squared feature selection methods are used on a phishing dataset [30]. Haynes et al. (2021) have suggested applying phishing detection based lightweight algorithms by relying on URLs in the mobile device, deep transformers (BERT and ELECTRA) are used to detect phishing websites solely [31].

There are three primary forms of phishing cyber-attacks: spear phishing, clone phishing, and whaling attacks. The first type targets people or multiple organizations through monitoring information about the victim and user details to increase the probability of a successful attack. The second type of attack spins from an already infected member, legitimately and historically detected by receiving mirrored or cloned emails with attachments. The third type is spear phishing formulated for senior executives and other outstanding ratings. The text sent to the target is drafted with an overhead manager [32].

3. Preliminary Concepts

The theoretical background of fuzzy logic and differential evolution is described in this section.

3.1. Fuzzy Logic

Professor Lotfi Zadeh invented the term "fuzzy logic" in 1965. [33]. Fuzzy logic is a branch of mathematics that was developed from fuzzy set theory, where evaluation is an approximation rather than precise values. Finite-state fuzzy logic is an extension of classical logic that offers mechanisms for approximation (approximate reasoning) and inference (decision-making) under uncertainty [34]. Four major components make up the fuzzy controller: the fuzzification interface, the knowledge base, the inference mechanism, and the defuzzification interface. Each input to the fuzzy logic has a degree of membership specified by the number and is always restricted to a range between 0 and 1 because of the membership function. [35].

3.2. Basics of Differential Evolution Algorithm

Today, researchers recognize that Evolutionary Algorithms (EA) are the best optimization algorithms specially applied for the extensive problem domain. Differential Evolution (DE) is one of the EA derived by Storn and Price in 1997. The DE has been widely used in many optimization problems due to a large number of optimization issues that may be solved using its global solid search technique[36].

The DE algorithm is not an inspired algorithm like EA algorithms since it has no natural paradigm [37].

The DE solution is stochastic and can find the global solution in an individual space like a genetic algorithm. The basic operators of DE consist of four stages: Initialization, Mutation, Crossover, and Selection [38]. The mutation is the primary operator in DE to get better solutions. At the same time, the genetic algorithm relies on the crossover operator.

In the same way as other EA algorithms, DE starts with a set of preliminary solutions updated in each step until reaching the stop criteria. The mutation phase is used for new individual generations. The crossover phase generates a new trial element. This phase is responsible for specifying how many mutations are there in the existing population [39]. Finally, the DE algorithm will select the new element that forms the next generation.

4. Methodology for Phishing Detection

The following are the key steps of the suggested approach, as discussed in more detail below.

4.1. Feature Extraction

Feature extraction has a significant impact because it affects the efficiency of classifiers for phishing detection. Six features F={f1,f2,,,,f6} are extracted from the URL of the dataset as illustrated in the following.

1. URL length ratio is defined as the number of characters in the URL divided by the maximum URL length in the dataset, as calculated in Eq.(1)

$$f1 = \frac{L(URL)}{Max\ l} \tag{1}$$

Where
L is the length of the URL
Max l is the maximum length of the URL in the dataset.

2. Rank ratio specifies the website popularity that depends on Alexa Rank, which gives 1 for the most popular websites. Alexa rank is determined by combining the site's prediction, traffic and visitor interaction have increased significantly over the previous three months. The rank ratio is expressed in Eq.(2).

$$f2 = \frac{R}{MaxR} \tag{2}$$

Where
R is the Alex Rank
MaxR maximum rank in the dataset.

3. Cardinality ratio, this measure refers to the originality of URL determined by the number of words in the remaining URL, as expressed in Eq.(3), is calculated by dividing the cardinality of the URL by the maximum cardinality in the dataset.

$$f3 = \frac{C}{MaxC} \tag{3}$$

Where
C is the URL cardinality.
maxC is the maximum cardinality in the dataset.

4. Associated ratio determines the ratio of the association between words in the remaining URL, Eq.(4) expresses how this feature was calculated.

$$f4 = \frac{AR}{MaxAR} \tag{4}$$

Where
AR represents the associated ratio for the URL.
MaxAR represents the maximum associated ratio in the dataset.

5. Related ratio indicates the ratio of the remaining URL's related words; Eq. (5) expresses how this feature was calculated.

$$f5 = \frac{RR}{MaxRR} \tag{5}$$

Where
RR is the related ratio.
MaxRR is the maximum related ratio in the dataset.

6. Jacard ratio, this feature depend on many features that are related to Jacard as follows:
   a. Jacard RR represents the ratio of related words for the registered domain and the remaining URL.
   b. Jacard RA represents the Jacard index between the related words for the registered domain and the associated words for the remaining URL.
   c. Jacard AR represents the Jacard index between associated words for the registered domain and the related words for the remaining URL.
   d. Jacard AA represents the Jacard index between associated words for the registered domain and the associated words for the remaining URL.
   e. This feature can be calculated as in Eq.(6) and Eq.(7).

Jratio=Jacard RR+Jacard RA+Jacard AR+Jacard AA. (6)

$$f6 = \frac{Jratio}{MaxJratio} \tag{7}$$

Where
MaxJratio is the maximum ratio in the dataset.

4.2. Fuzzy rule as Phishing Classifier

The feature extraction was applied to the dataset to extract the six mentioned features and assign a numerical value to each feature. The feature vector F is fed to the fuzzy logic to transfer each numerical value to the corresponding linguistic value, which consists of three values: High(H), Medium(M), and Low(L). The Trapezoid membership function as expressed in Eq. (8) [40] is used in this paper to transfer each feature score to a degree of membership.

The Trapezoid is characterized by its bottom and upper boundaries, a and d, and its nucleus, b and c, as illustrated in Figure 1.

$$T(X)=\left\{\begin{matrix}0 \quad if\ X\le\ a\ or\ X\ge d\\ \frac{X-a}{b-a} \qquad if\ x\in(a,b)\\ 1 \qquad if\ x\ \in(b,c)\\ \frac{d-X}{d-c} \qquad if\ x\ \in(c,d)\end{matrix}\right\} \qquad (8)$$

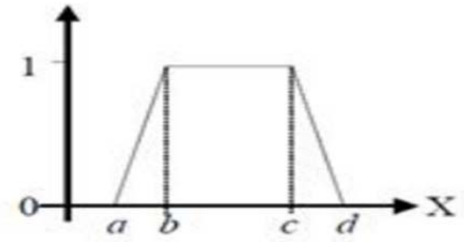

Figure 1. Trapezoid membership function

After converting the input values to the corresponding linguistic values, the inference engine generates if-then rules. The generated rules consist of two parts antecedent and consequent. The rule's antecedent shows a fuzzy area in the input. In contrast, the consequent refers to the output in the fuzzy region as either phish or legitimate. The following example explains the fuzzy rule.

If (f1) is H and (f2) is H and (f3) is L and (f4) is M, and (f5) is H, and f6 is M, then phish.

4.3. Rules Selection Using DE

Despite the efficiency of the proposed fuzzy logic for classifying web pages as phish or legitimate, too many rules are generated. Therefore a new method based on DE is used in this paper to select a set of appropriate rules from the entire set of rules to improve the suggested method's performance.

4.3.1. Individual representation and population initialization

There are many phases to implement the DE algorithm. The first significant phase is concerned with individual representation.

Binary encoding utilizes strings of 0s and 1s to represent individuals. The individual length is size 18 bits since six inputs each of the three membership (L, M, H). Each bit is 0 or 1, which indicates the value of the membership, as shown in Figure 2.

1	0	0	0	1	0	…	

⟷

Figure 2. Individual representation

There must be only one bit with one value for every feature. The remaining two bits must be 0 to indicate which membership is assigned to the corresponding feature.

DE is an optimization algorithm initiated with a population P of N solutions. P can be expressed as follows:

$$P=\{x_1,x_2,\dots,x_N\}$$

Where N is the size of the population.

4.3.2. Fitness function

The second phase involves introducing a new modification to the DE algorithm by adding a fitness function. As known, premature convergence is one of the fundamental problems of population-based algorithms.

Exploration and exploitation across the search area balancing are is one of the most crucial points in any population-based algorithm. Consequently, The fundamental DE algorithm is not a very efficient method [41]. The fitness function could be introduced to explore the whole search space and exploit the near-optimal solution. The fitness function maximizes the feature summations as in Eq.(9).

$$maximize\ \emptyset(I)=\sum_{i=1}^{6}Fi \qquad (9)$$

4.3.3. Mutation and crossover operations

DE is considered a competitive form of evolutionary algorithms. To create new solutions, DE algorithms rely on mutation solutions to scale the differences of randomly selected individuals from the population.

The choice of a proper mutation strategy is vital for the success of a DE algorithm. There are various types of differential mutation operators. The most commonly used variant is DE/rand/1 that is generated by adding a scaled difference vector F. (x_{r2} –x_{r3}) between two randomly picked individuals xr2 and xr3 to a third randomly chosen individual x_{r1}, called the base vector:

$$V_i=x_{r1}+F.(x_{r2}-x_{r3}) \qquad (10)$$

Where

r1,r2 and r3 are random number in [1,N].

F parameter is called the scaling factor. It was obtained for each individual to realize a log-normally distributed random variable.

$$F=\exp(\beta) \qquad (11)$$

Where β in [0,1]

Crossover is the next phase of the DE algorithm. In this phase, the trial vector is defined by combining the mutant vector v_i with a DE individual x_i given as Eq.(12).

$$U_{i,j}=\begin{cases}V_{i,j}\ if\ randi,j\le CR\ or\ j=jrand\\ x_{i,j} \qquad otherwise\end{cases} \qquad (12)$$

where

CR is a random number in [0,1] that determines the value of the trial vector U which is inherited from the donor V.

4.3.4. Selection operation

First, the trial vector is created by choosing the individual for the mutation operation. Second, the best parent or offspring is selected based on their fitness value for the next generation. The parent is selected if it has a higher fitness function than the offspring.

$$U_{i,j}(G+1) = \begin{Bmatrix} U_{i,j}(G) \ if \ fitness\left(U_{i,j}(G)\right) > fitness\left(X_{i,j}(G)\right) \\ X_{i,j}(G) \qquad\qquad\qquad\qquad otherwise \end{Bmatrix} \quad (13)$$

The Pseudocode for DE strategy is described in the following algorithm.

```
Algorithm: DE algorithm.
 Initialize the number of population NP
While the stop condition is not true, do
   For each individual $x_{i,j}$ ϵ NP do
      Evaluate the fitness($x_{i,j}$)
      Apply mutation operator to create the trial vector
v
      Create offspring $U_{i,j}$ by applying the crossover.
      If   fitness($x_{i,j}$) > fitness ($U_{i,j}$) then Add $x_{i,j}$ to P
      Else add $U_{i,j}$ to the P.
   End for
End while.
```

5. Experimental Results and Discussion

5.1. Phishing Dataset

The suggested approach was tested on two datasets. One of these is a malicious dataset, and the second is a legitimate dataset. The dataset contains 10,000 cases for malicious and 10,000 cases for legitimate. The malicious dataset was generated using information from the PhishTank site. (https://www.phishtank.com). Phishtank is a phishing website repository that is available to users for free.

Due to the general use of PhishTank, people can submit websites, so the dataset is frequently updated. Before being confirmed as malicious and placed on a blacklist, suspected phishing URLs are checked by numerous people. PhishTank offers lists of effective phishing URLs. The legitimate URLs dataset was collected from Alexa (https://www.alexa.com). Each URL in the two datasets is processed to extract features later used for classification purposes.

5.2. Evaluation Measures

The result of the proposed model was evaluated using confusion matrix as shown in Table 1. and the well-known standard evaluation metrics for the classification algorithms; namely, Precision (P), Recall (R), F-Measure (F), Accuracy (ACC), False Negative rate (FNR), and Receiver Operating Characteristic (ROC) curve is used to check the the effectiveness of the presented model cost against the recall.

$$P = \frac{TP}{TP+FP} \quad (14)$$

$$R = \frac{TP}{TP+FN} \quad (15)$$

$$F - Measure = \frac{2*P*R}{P+R} \quad (16)$$

$$Acc = \frac{TP+TN}{TP+TN+FN+FP} \quad (17)$$

$$FNR = \frac{FN}{TP+FN} \quad (18)$$

Table 1. displays the phishing website confusion matrix

	Classified as Phishing	Classified as Legitimate
Phishing Website	TP	FN
Legitimate Website	FP	TN

Where

TP is is the number of correctly classified phishing websites.

TN is the number of correctly classified legitimate websites.

FP is the number of incorrectly classified phishing websites.

FN is the number of incorrectly classified legitimate websites.

5.3. Results and Discussion

A trial and error mechanism is used to select the best proposed DE parameters to produce the most efficient detection result. The values of the parameters are reported in Table 2.

Table 2. DE parameter

Parameter	Value
Population size	200
Maximum number of Iteration	150
Crossover probability, CR	0.7
Mutation Rate, F	0.5

It is obvious from Figure 3 that the suggested method's ability to distinguish between phishing and legal websites is very effective where the ROC with the AUC equals 0.97.

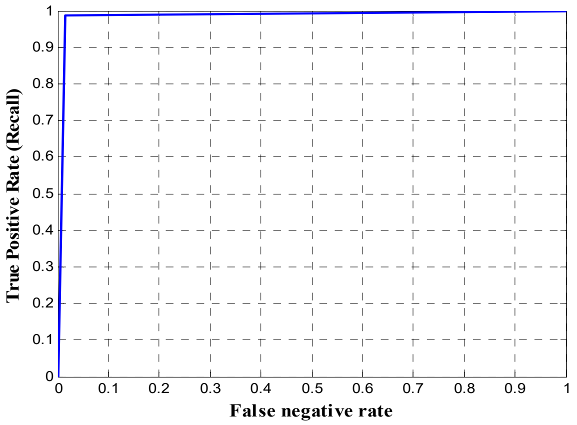

Figure 3. ROC for the proposed model

As mentioned previously, there are two significant contributions to the proposed method. The first contribution concerns introducing DE as a fuzzy rule selection. In contrast, the second is concerned with introducing a new feature Jacard ratio (F6). In order to precisely assess the contributions, the performances of Random forest (RF), support vector machine (SVM), k-Nearest Neighbor (KNN) were evaluated with and without the new feature and compared with the proposed method.

Table 3. shows that that the proposed method's performance outperforms the performance of other machine learning methods due to the excellent formulation of the problem based on fuzzy logic and selection of the most appropriate rules efficiently using DE.

Table 3. Comparisons between the proposed model and other models without using F6.

Model	P	R	F-measure	ACC
RF	0.901	0.921	0.91	0.913
SVM	0.911	0.882	0.896	0.902
KNN	0.862	0.834	0.847	0.851
Proposed Model	0.932	0.913	0.922	0.931

In addition, based on the formulation of the fitness function that seeks to detect the phishing website, Table 4. showed that the obtained result is very efficient for the phishing class and less efficient for the legitimate class where the precision=0.932 and the recall=0,913.

Table 4. Comparisons between the proposed model and others model with using F6

Model	P	R	F-measure	ACC
RF	0.912	0.930	0.920	0.925
SVM	0.932	0.906	0.918	0.931
KNN	0.875	0.862	0.868	0.894
Proposed model	0.973	0.933	0.947	0.976

It is clear from Figure 4. that the features are very close to each other, leading to misclassification when used separately.

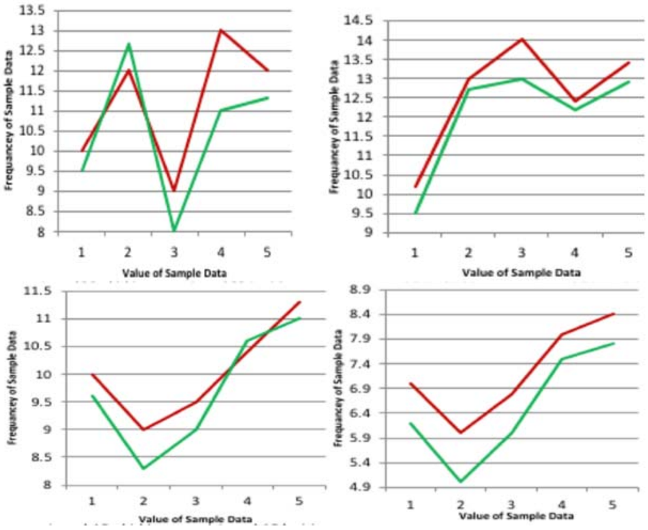

Figure 4. Frequency of feature (F6) before combination

Whereas Figure 5. shows the frequency of the feature F6 after combining the four features. The generated feature is distributed for both cases that support the classification process and increase the accuracy results.

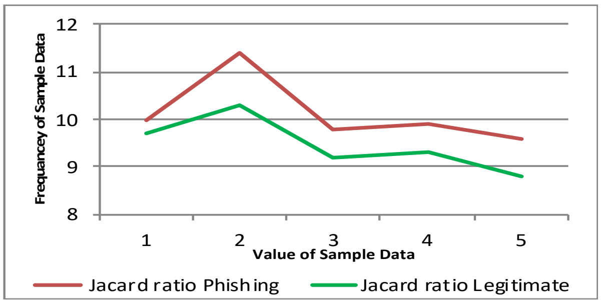

Figure 5. The frequency of feature (F6) after combination

6. Conclusion

Phishing websites have been increased dramatically as more advanced phishing kits are introduced. These kits help the attackers in spreading the faked pages. The conventional techniques that depend on separated features have extraordinary misclassification because overlapping between the selected features is also high. The phishers always attempt to manipulate these separated features to their advantage. Phishing detection techniques need development to refine the classes. Consequently, more consideration should be given to mitigate it. A novel technique for phishing detection was described in this study by using a proficient fuzzy set and the DE algorithm.

The proposed system was evaluated on two malicious and legitimate datasets collected from Phishtank and Alexa websites.

Results show that the combination feature can maintain high classification accuracy and restrict phishers from making use of the features to their advantage . Also, the DE is a more viable way to optimize the fuzzy rule and yield improvement in the classification accuracy and mean F1 score than the machine learning algorithms.

The plans to continue the proposed model are two aspects for future work. First, add a new feature combination close to each other, and evaluate the model before and after the combination. Secondly, apply Particle Swarm Optimization (PSO) algorithm to select the best fuzzy rules.

References

[1]. Ravi, R. (2020). A performance analysis of Software Defined Network based prevention on phishing attack in cyberspace using a deep machine learning with CANTINA approach (DMLCA). *Computer Communications*, *153*, 375-381.

[2]. Surwade, A. U. (2020). Phishing e-mail is an increasing menace. *International Journal of Information Technology*, *12*(2), 611-617.

[3]. Saravanan, P., & Subramanian, S. (2020). A framework for detecting phishing websites using GA based feature selection and ARTMAP based website classification. *Procedia Computer Science*, *171*, 1083-1092.

[4]. Subasi, A., & Kremic, E. (2020). Comparison of adaboost with multiboosting for phishing website detection. *Procedia Computer Science*, *168*, 272-278.

[5]. Chen, J. L., Ma, Y. W., & Huang, K. L. (2020). Intelligent Visual Similarity-Based Phishing Websites Detection. *Symmetry*, *12*(10), 1681.

[6]. Zhu, E., Ju, Y., Chen, Z., Liu, F., & Fang, X. (2020). DTOF-ANN: an artificial neural network phishing detection model based on decision tree and optimal features. *Applied Soft Computing*, *95*, 106505.

[7]. Frauenstein, E. D., & Flowerday, S. (2020). Susceptibility to phishing on social network sites: A personality information processing model. *Computers & security*, *94*, 101862.

[8]. HR, M. G., Adithya, M. V., & Vinay, S. (2020). Development of anti-phishing browser based on random forest and rule of extraction framework. *Cybersecurity*, *3*(1), 1-14.

[9]. Aleroud, A., Abu-Shanab, E., Al-Aiad, A., & Alshboul, Y. (2020). An examination of susceptibility to spear phishing cyber attacks in non-English speaking communities. *Journal of Information Security and Applications*, *55*, 102614.

[10]. Rao, R. S., & Pais, A. R. (2020). Two level filtering mechanism to detect phishing sites using lightweight visual similarity approach. *Journal of Ambient Intelligence and Humanized Computing*, *11*(9), 3853-3872.

[11]. Adebowale, M. A., Lwin, K. T., Sanchez, E., & Hossain, M. A. (2019). Intelligent web-phishing detection and protection scheme using integrated features of Images, frames and text. *Expert Systems with Applications*, *115*, 300-313.

[12]. Chiew, K. L., Tan, C. L., Wong, K., Yong, K. S., & Tiong, W. K. (2019). A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. *Information Sciences*, *484*, 153-166.

[13]. Marchal, S., François, J., State, R., & Engel, T. (2014). PhishStorm: Detecting phishing with streaming analytics. *IEEE Transactions on Network and Service Management*, *11*(4), 458-471.

[14]. Bu, S. J., & Cho, S. B. (2021). Deep character-level anomaly detection based on a convolutional autoencoder for zero-day phishing URL detection. *Electronics*, *10*(12), 1492.

[15]. Sameen, M., Han, K., & Hwang, S. O. (2020). PhishHaven—an efficient real-time ai phishing URLs detection system. *IEEE Access*, *8*, 83425-83443.

[16]. Rao, R. S., Vaishnavi, T., & Pais, A. R. (2020). CatchPhish: detection of phishing websites by inspecting URLs. *Journal of Ambient Intelligence and Humanized Computing*, *11*(2), 813-825.

[17]. Vijayalakshmi, M., Shalinie, S. M., & Yang, M. H. (2020). Web phishing detection techniques: a survey on the state‐of‐the‐art, taxonomy and future directions. *IET Networks*, *9*(5), 235-246.

[18]. Trisanto, D., Rismawati, N., Mulya, M. F., & Kurniadi, F. I. (2020). Effectiveness undersampling method and feature reduction in credit card fraud detection. *Int. J. Intell. Eng. Syst*, *13*(2), 173-181.

[19]. Rajitha, K., & VijayaLakshmi, D. (2016). Oppositional cuckoo search based weighted fuzzy rule system in malicious web sites detection from suspicious URLs. *Int J Intell Eng Syst*, *9*(4), 116-125.

[20]. Rao, R. S., Pais, A. R., & Anand, P. (2021). A heuristic technique to detect phishing websites using TWSVM classifier. *Neural Computing and Applications*, *33*(11), 5733-5752.

[21]. Tan, C. L., Chiew, K. L., Yong, K. S., Abdullah, J., & Sebastian, Y. (2020). A graph-theoretic approach for the detection of phishing webpages. *Computers & Security*, *95*, 101793.

[22]. Ali, W., & Malebary, S. (2020). Particle swarm optimization-based feature weighting for improving intelligent phishing website detection. *IEEE Access*, *8*, 116766-116780.

[23]. Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J. P. (2020). An effective phishing detection model based on character level convolutional neural network from URL. *Electronics*, *9*(9), 1514.

[24]. Wei, W., Ke, Q., Nowak, J., Korytkowski, M., Scherer, R., & Woźniak, M. (2020). Accurate and fast URL phishing detector: a convolutional neural network approach. *Computer Networks*, *178*, 107275.

[25]. Feng, J., Zou, L., Ye, O., & Han, J. (2020). Web2Vec: Phishing Webpage Detection Method Based on Multidimensional Features Driven by Deep Learning. *IEEE Access*, *8*, 221214-221224.

[26]. Anupam, S., & Kar, A. K. (2021). Phishing website detection using support vector machines and nature-inspired optimization algorithms. *Telecommunication Systems*, *76*(1), 17-32.

[27]. Mahdavifar, S., & Ghorbani, A. A. (2020). DeNNeS: deep embedded neural network expert system for detecting cyber attacks. *Neural Computing and Applications*, *32*(18), 14753-14780.

[28]. Kumar, M. S., & Indrani, B. (2021). Frequent rule reduction for phishing URL classification using fuzzy deep neural network model. *Iran Journal of Computer Science*, *4*(2), 85-93.

[29]. Sankhwar, S., Pandey, D., Khan, R. A., & Mohanty, S. N. (2021). An anti-phishing enterprise environ model using feed-forward backpropagation and Levenberg-Marquardt method. *Security and Privacy*, *4*(1), e132.

[30]. Tharani, J. S., & Arachchilage, N. A. (2020). Understanding phishers' strategies of mimicking uniform resource locators to leverage phishing attacks: A machine learning approach. *Security and Privacy*, *3*(5), e120.

[31]. Haynes, K., Shirazi, H., & Ray, I. (2021). Lightweight URL-based phishing detection using natural language processing transformers for mobile devices. *Procedia Computer Science*, *191*, 127-134.

[32]. Barraclough, P. A., Fehringer, G., & Woodward, J. (2021). Intelligent cyber-phishing detection for online. *Computers & Security*, *104*, 102123.

[33]. Zadeh, L. A., Klir, G. J., & Yuan, B. (1996). *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers* (Vol. 6). World Scientific.

[34]. Waris, S., & Ahmad, Z. (2011). Application of fuzzy logic in academic setup. *Recent Advances in Statistics*, 367.

[35]. Vlamou, E., & Papadopoulos, B. (2019). Fuzzy logic systems and medical applications. *AIMS neuroscience*, *6*(4), 266-272.

[36]. Alguliev, R. M., Aliguliyev, R. M., & Isazade, N. R. (2013). Multiple documents summarization based on evolutionary optimization algorithm. *Expert Systems with Applications*, *40*(5), 1675-1689.

[37]. Georgioudakis, M., & Plevris, V. (2020). A comparative study of differential evolution variants in constrained structural optimization. *Frontiers in Built Environment*, *6*, 102.

[38]. Do, D. T., Lee, S., & Lee, J. (2016). A modified differential evolution algorithm for tensegrity structures. *Composite Structures*, *158*, 11-19.

[39]. Zaharie, D. (2009). Influence of crossover on the behavior of differential evolution algorithms. *Applied soft computing*, *9*(3), 1126-1138.

[40]. Sharma, H., Bansal, J. C., & Arya, K. V. (2012). Fitness based differential evolution. *Memetic Computing*, *4*(4), 303-316.

[41]. Xiao, Z., Xia, S., Gong, K., & Li, D. (2012). The trapezoidal fuzzy soft set and its application in MCDM. *Applied Mathematical Modelling*, *36*(12), 5844-5855.