Mitigation of Phishing URL Attack in IoT using H-ANN with H-FFGWO Algorithm

Gopal S. B[1]*, and Poongodi C[2]
[1]Department of Electronics and Communication Engineering, Kongu Engineering College, INDIA
[e-mail: s.b.gopalece@gmail.com]
[2]Department of Computer Science and Engineering, Vivekananda College of Engineering for Women, INDIA
[e-mail: poongodi321@yahoo.co.in]
*Corresponding author: Gopal S.B

Received June 27, 2022; revised April 5, 2023; revised April 26, 2023; accepted May 26, 2023; published July 31, 2023

Abstract

The phishing attack is a malicious emerging threat on the internet where the hackers try to access the user credentials such as login information or Internet banking details through pirated websites. Using that information, they get into the original website and try to modify or steal the information. The problem with traditional defense systems like firewalls is that they can only stop certain types of attacks because they rely on a fixed set of principles to do so. As a result, the model needs a client-side defense mechanism that can learn potential attack vectors to detect and prevent not only the known but also unknown types of assault. Feature selection plays a key role in machine learning by selecting only the required features by eliminating the irrelevant ones from the real-time dataset. The proposed model uses Hyperparameter Optimized Artificial Neural Networks (H-ANN) combined with a Hybrid Firefly and Grey Wolf Optimization algorithm (H-FFGWO) to detect and block phishing websites in Internet of Things(IoT) Applications. In this paper, the H-FFGWO is used for the feature selection from phishing datasets ISCX-URL, Open Phish, UCI machine-learning repository, Mendeley website dataset and Phish tank. The results showed that the proposed model had an accuracy of 98.07%, a recall of 98.04%, a precision of 98.43%, and an F1-Score of 98.24%.

Keywords: Feature selection optimization, H-FFGWO Algorithm, IoT, Phishing attack.

http://doi.org/10.3837/tiis.2023.07.010 **ISSN : 1976-7277**

1. Introduction

The Internet of Things (IoT) is a cutting-edge technology that enables remote monitoring of items and the sharing and management of data among linked devices without the need for human intervention. Every day, more and more people connect to the internet. By the start of 2025, experts predict there will be 974 million people using the Internet. The web makes everything straightforward. IoT has applications in numerous fields like medicine, financial, automation, agriculture and so on. It has created home automation and good appliances straightforward and exciting and it is necessary to take care of the security of these IoT devices. Even if the web brings a lot of advantages to individuals, securing the information on the web from attackers may be a major challenge. Security threats to the web are increasing enormously. IoTs are vulnerable to attacks, like DDOS, spoofing, phishing, botnet and ransomware. These attacks should be resolved to avoid major losses. Several phishing attacks are reported recently.

One of the intensive issues in phishing attacks was the attackers try to steal the non-public information of a target person through fake websites or emails or calls. After stealing the credentials, they struggle to steal or corrupt, or delete the user’s information or they would install a tiny low malware program in the victim's system. They predominantly use it for fraudulent purposes or try to steal bank details. The attackers could attack a bunch of people or individuals. These types of attacks lead to a hectic threat to businesses, industries and people. The threats, along with the values and technical support to fix these kinds of attacks, are increasing which ends up in the loss of holding, loss of name and revenue and spreading of security vulnerabilities.

The users would easily fall into the traps set by attackers because they are unable to distinguish between legitimate and phishing websites due to the similar appearance of both. The steps involved in it are listed below:

- Deciding the target.
- Composing emails, messages or pretending Uniform Resource Locator (URLs).
- Attacking and installing malware in the victim’s system.
- Gathering the desired information like username, bank credentials and so on.
- Trying and accessing the initial website and corrupting the information.

The attacker creates a website that mimics an original website [1]. When a user unknowingly clicks the link they would be directed to the phishing website. As for a normal website, the phishing website would ask the user to enter his or her login credentials. After entering the details, the attackers would collect the data and enter into the actual website and try to corrupt or delete the victim’s details. Manual investigations of these attacks may take a long time. Analyzing these attacks is so complex and it requires attack tools to detect the attacks. Sometimes the manual calculations may also go wrong. Deceptive phishing is one of the most prevalent types of phishing attacks, which involves the creation of a website similar to a legitimate website and then sending an email to the target people or individuals in a genuine manner. These fraudulent emails would contain a malicious URL or link. The victim is instructed to click on the URL and the instructions are included. It would collect and forward all the login credentials and the other sensitive details of the victim to the attackers. For example, user@google.com uses the number '1' instead of the letter ‘a, and user@goog1e.com looks like an original one. The users are not able to distinguish between these and they were attacked by the attackers. Spear phishing is identical to deceptive phishing. It differs from attacking targets; it targets only one individual, not groups. It aims at one person and attacks him/her for getting his/her confidential data by customizing an email [2]. The email would contain the victim's name, business, and job title, among other identifying details, making it simple for the victim to access the malicious URL. Most spear phishing attacks occur on social media sites like LinkedIn because the phishers have easy access to information about the target's profession and personal data.

Whaling attacks occur once the photographer targets privately a leader who places himself as a business manager. The attacker would be watching the victim's profile for a substantial amount of time before playacting the attack. The attacker would send the victim an email and manipulate it to provide the attacker with personal data. Such attacks are dangerous because people in government bands have the main way. Pharming could be a type of online fraud involving malicious code and dishonorable websites on laptops. Cybercriminals install malicious code on the laptop or server. The malicious code mechanically directs the victims to fake websites, not their data. It is a hidden threat as a result of the victim's being aware of whether the website is hacked or not before making the non-public data.

The rest of the paper is organized as follows: Section II discusses works that are already done related to phishing attack detection. Section III describes the working of the Particle swarm optimization algorithm. Section IV describes the Problem statement. Section V describes the working of an Artificial Neural Network. Section VI discusses the Proposed H-ANN with the H-FFGWO model. Section VII discusses the Experimental Results of the H-FFGWO Method. Section VIII discusses the Conclusion of the work and Future Work.

2. Literature Survey

The literature survey has included the latest work done on improved optimization techniques over PSO, GWO, WHO, etc. and also the hybrid combination algorithms. Waleed Ali and Sharaf Malebary et al., [3] utilized better identification of phishing websites through PSO-based feature ranking. Web phishing attempts have been on the rise in recent years, with users losing faith in online services and portals. The rapid advancement of technology has resulted in the development of more complex strategies for attracting users. The newest and most often used phishing websites are zero-day phishing sites, which are undetectable by blacklist-based methods. Recent search experiments have been implementing machine learning algorithms to detect phishing websites. In many ways, the relevant aspects have been chosen based on the experience of human beings and the frequency of studying the website properties. In the proposed method, the distinction between distinct website functions is used in the particle swarm-based website characteristics.

Sulemana and Awan et al., have experimented with Genetic algorithms for improving the identification of fake websites based on URLs [4]. Obtaining information such as passwords, credit card numbers and account numbers is a cybercrime. Hackers fool users by redirecting them to bogus websites. Nowadays, using a machine learning method is efficient. Phishing detection based on the Uniform Resource Locator (URL) has been implemented. Machine learning has been divided into legitimate and illegitimate websites using Naive Bayes (NB), Iterative Dichotomiser3 (ID3), K-Nearest Neighbor (KNN), Decision Tree (DT) and Random Forest (RF). For feature selection, genetic algorithms are applied, which could enhance detection accuracy.

Sameena Naaz et al., have attempted to detect phishing in the Internet of Things (IoT) using Machine Learning Approach with the use of decision trees, neural networks, random forests and linear models, data could be classified as authentic, phishing, and suspicious [5]. Phishing attempts must be predicted and prevented if online transactions are to be saved. Data mining techniques can be used to mine millions of pieces of data and to get precise results. IoT refers to gadgets that are linked to the internet. Machine Learning (ML) algorithms such as Support Vector Machine (SVM) and Logistic Regression (LR) have been applied to the IoT dataset to detect phishing assaults. The outcomes are compared with the previous studies using the Phish Tank dataset as well as the other datasets.

Damodaram, Valarmathi, et al., have experimented with the optimization of Bacterial Foraging (BFO) for fake website detection [6]. This assists in overcoming the challenges of identifying phishing websites. The existing system is a smart, durable, and successful system for employing data mining algorithms for association and classification. It is utilized to define all the criteria and rules used to simplify the phishing sites and their relationships as well as to compare their performances, accuracy and the number of rules generated. The simplification model reveals a link between URL and Domain Identity, as well as security, and accurate results for phishing detection. In the proposed strategy, the BFOA meta-heuristic algorithm is utilized to find a solution for fraudulent websites and it is evaluated with the other phishing datasets.

Rana et al., have proposed a Whale optimization algorithm (WOA) based email spam feature selection method using a rotation forest algorithm for classification [7]. On the internet, email is one of the most effective modes of communication. Spam mail is a problem in the mail; it takes up a lot of bandwidth and space, and the spam mail filtering techniques fail, resulting in the misclassification of original mail as spam mail, which is a big challenge for the internet world. The forest algorithm is used in the WOA to identify the prominent features in the email corpus and to categorize the emails into spam and non-spam. The entire dataset is used for the rotation forest method assessment and feature selection is done by using WOA. The results demonstrate that following feature selection with the Whale Optimization method, the rotating forest algorithm can classify emails into spam and non-spam with high accuracy of 99% and a low FP rate of 0.0019.

A.Alellah et al., have attempted to detect DOS attacks from network traffic using Grey Wolf Optimization (GWO) algorithm [8]. As observed in commercial web server instances, a DoS attack can temporarily interrupt service or harm the system by using all the system platforms such as memory, network bandwidth, Central Processing Unit (CPU) and so on. Damages are possible to all networks that are connected to the systems. As this form of assault is so damaging to systems and networks, many researchers are working to figure out how to detect DOS attack and how to avoid it. A model for identifying DoS attacks is described in the study. To detect the attack, swarm techniques, particularly GWO, are deployed. The attack detection rate is measured by extracting data from network packets and analyzing it before entering it into the wolf algorithm. The outputs of the wolf algorithm are then analyzed and evaluated to determine the attack detection rate with false alarms and negative alarms and found to be good and satisfactory. The TCP/IP protocol is utilized with Matlab version 10 on Windows 10 operating system.

Safara F et al., have employed the Improved Spotted Hyena Optimization algorithm (ISHO) for detecting phishing websites [9]. The emergence of fraudulent or phishing sites that steal users' information is one of the primary concerns in cyberspace and the Internet of Things (IoT). A site, which has a multimedia system, allows users to operate varieties of data such as documents, images, videos and audio. Each sort of data is pruned so that fishers can use it for a phishing attempt. People get directed to the false pages in phishing assaults, and their personal information is obtained by a thief or phisher. The most extensively utilized methods for identifying websites and detecting phishing assaults are ML and DM algorithms. The feature selection approach used to find acceptable features for classification has a big impact on classification accuracy.

3. Problem Formulation

The PSO method has a poor convergence rate during the repeated process, and it is simple to slip into local Optimum in high-dimensional spaces. To overcome this drawback and to improve its performance, the proposed algorithm combines the advantages of the two different optimization algorithms like GWO and FFA. The advantage of GWO is its simple structure, which enhances its implementation and very high convergence because of its few deciding factors. Likewise, the advantages of the Firefly Optimization Algorithm are that it usually has good efficiency for certain problems and requires only a small number of iterations. The proposed H-FFGWO approach is to first select the required feature (α) from the available dataset using FFA. Then the output of FFA is given as the input to the GWO algorithm to fine tune and finalize the features required for detecting phishing attacks.

4. Problem Solution

4.1 Neural Network

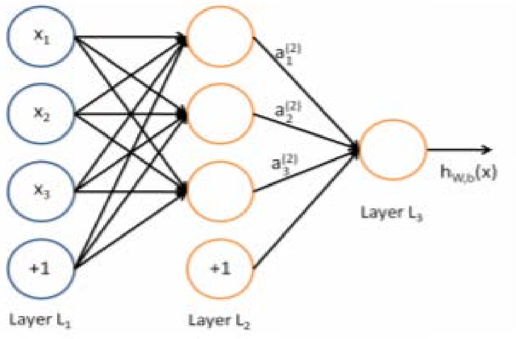

Fig. 1. The architecture of an ANN model with one hidden layer

A neural network is a computational model. It has a network architecture made up of artificial neurons. Its architecture is similar to that of the human brain. It is formed by hooking together many simple neurons so that the output of a neuron can be the input of another as shown in **Fig. 1**. It has many layers and each layer performs a specific function [10]. The neuron is a mathematical function that collects and classifies information.

4.2 Simulation Tool

Googlecolab is used for training the ANN model. It is a product of Google and it allows you to write and execute the code in the browser and it is free to use. To clean the dataset and to train the model, googlecolab lab is used. It is possible to import and run the files from google drive. The codes are stored in the drive. It is a good tool for deep learning, which gives free access to Google computing resources.

4.3 Work Flow

The Workflow of the proposed H-ANN with the H-FFGWO Algorithm undergoes the following steps and which are also shown in **Fig. 2**.

STEP 1: The URL dataset has been downloaded for the corresponding repository.

STEP 2: Preprocessing of the dataset has been done to remove the empty and NaN values from the downloaded dataset

STEP 3: URL-based features alone extracted from the dataset before giving as input to the model

STEP 4: Only required features from the available dataset based on the weighting have been extracted from the H-FFGWO feature selection model

STEP 5: Hyperparameter Optimized ANN model has been developed for better performance.

STEP 6: The extracted dataset from the H-FFGWO feature selection model is given to the H-ANN model and the Performance has been compared with the existing model.

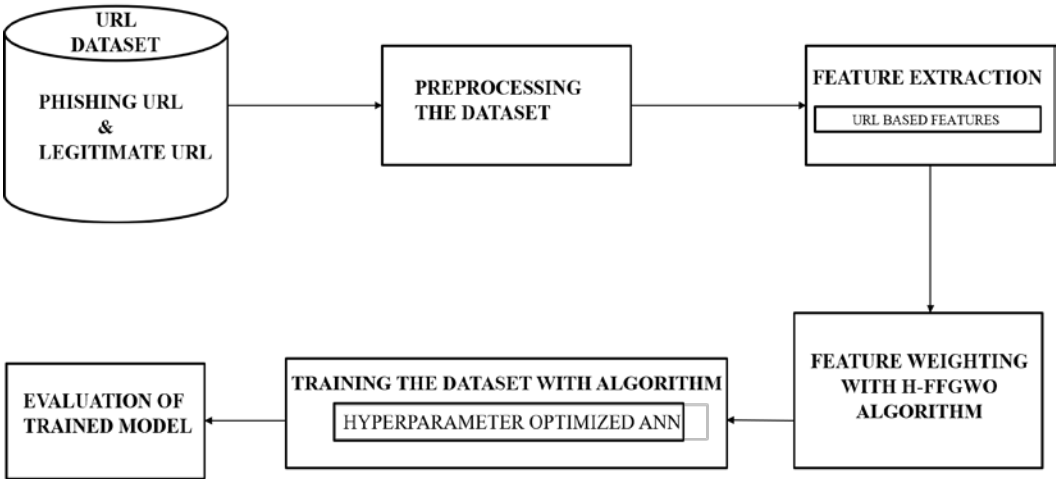
URL DATASET
PHISHING URL & LEGITIMATE URL
PREPROCESSING THE DATASET
FEATURE EXTRACTION
URL BASED FEATURES
FEATURE WEIGHTING WITH H-FFGWO ALGORITHM
TRAINING THE DATASET WITH ALGORITHM
HYPERPARAMETER OPTIMIZED ANN
EVALUATION OF TRAINED MODEL

Fig. 2. The pipeline of the proposed solution.

4.4 Dataset Description

The different Datasets which are going to be considered for the performance evaluation of our proposed model are shown in **Table 1**.

Table 1. Datasets Description

S.No	Name of the Dataset	Benign URLs	Phishing URLs
1	ISCX-URL2016	35,000	10,000
2	OpenPhish	20,025,990	85,003
3	PhishTank	48,009	48,009
4	UCI Machine learning repository	2,04,863	24,567
5	Mendeley website	58,000	30,647

4.5 Hyperparameter Optimized Artificial Neural Networks (H-ANN)

By adding more concealed levels, the ANN model can be made as more complex. The input is projected onto a new vector space using non-linear transformations at each layer, and a complicated judgment boundary is drawn to distinguish between groups. The ANN is used to model different complex functions. The steps involved in ANN are listed below

1. The weight of all the nodes are randomly initialized.
2. Then using the forward pass with the random weights, the output from each intermediate node is given to the next one, and the end output at the end node is obtained.
3. Comparison is performed with the result obtained at the last node with the target value in the training process, and use a loss function to measure the error.
4. Carry out a right-to-left reverse pass and propagate the error with backpropagation in every individual node.
5. Next, the weight adjustment has been done based on the error, to get the desired output using gradient descent.

Finally, the ANN model propagates the error gradients back starting from the last layer. The parameters, which have been set after experimenting with different scenarios in hyperparameter optimization, are given below.

- Learning Rate: 1e-3
- The number of layers: 4
- Activation function: softmax
- Regularization: L2
- Optimization: Adam
- Batch-size:100
- Epochs: 500

4.6 H-FFGWO Algorithm

The existing PSO algorithm is based on the movement of birds. Its drawback is that sometimes it is very complex to design the initial values or parameters [11]. It can converge into a certain local minimum for problems that are difficult to solve. It cannot be used for the problem of scattering [12]. It is slow when compared with the other nature-inspired algorithms.

The proposed H-FFGWO method combines the positive of the Grey Wolf and Fire Fly Algorithm. The positives of the Grey Wolf Optimization (GWO) algorithm are that it is not complex to implement and it needs less space for storage and computation [13]. It is simple in its approach and due to the hierarchy behavior, it is fast in solving a problem [14]. Exploration of the GWO algorithm is high with précised output. The fire Fly Optimization (FFO) algorithm also performs efficiently compared to the existing PSO method [15]. It requires only a limited number of iterations which reduces the time to solve certain kinds of problems [16].

In the H-FFGWO algorithm, the fitness function of the firefly algorithm is modified as shown below.

4.6.1 Modified Fitness function for Firefly

Apart from the normal working procedure of the Firefly Algorithm, the following are the values that have been set for the parameters.

ω = 0.9 Fitness function constant

Niter = 50 The Number of iterations for optimization

α = 0.5 Randomness parameter

γ = 1 Absorption coefficient

δ = 0.99 Randomness reduction coefficient

β = 0 0.2 Light amplitude

In the H-FFGWO algorithm, the best Alpha features are selected based on the working principle of light intensity as shown in **Fig. 4**. Then the resultant feature set is further analyzed by using the Grey Wolf Optimizer to fine-tune the set of features. The proposed H-FFGWO for feature selection improves the performance of the ANN model more than the existing model.

4.6.2 Modified Greywolf Optimization

In Greywolf Optimization, apart from its normal working procedure following are the changes upgraded.

The number of dimensions (d) = 1: This refers to the number of input variables or features in a problem.

The lower bound (minx) = -9.0: This is the minimum value that the input variable can take. In this case, the minimum value is -9.0.

Upper bound (maxx) = 9.0: This is the maximum value that the input variable can take. In this case, the maximum value is 9.0.

Several grey wolves (N) = 25: This parameter specifies the number of agents or individuals in the population that will be used to solve the problem. In this case, there are 25 grey wolves.

A maximum number of iterations (max_iter) = 50: This parameter specifies the maximum number of times that the algorithm will iterate or search for a solution. In this case, the algorithm will run for a maximum of 50 iterations before stopping.

The feature selection incorporates the following processes.

4.6.3 Light Intensity

The attractiveness behavior of fireflies will vary based on the intensity emitted from one firefly to another [17]. Hence the brightness is proportional to the attractiveness [18]. To find the attractiveness or intensity of light from the firefly, let i and j be the light intensity or brightness of two fireflies respectively as shown in **Fig. 3**. First, preprocessing of the dataset has been performed to remove the redundant or empty fields present in the dataset. The preprocessed dataset is given to the firefly algorithm to evaluate the fitness of all features present and based on the threshold value it divides the features into alpha, beta and gamma features. From the available features, Alpha features alone are given to the grey wolf algorithm to continue its process. At the end of grey wolf optimization only alpha features are taken and provided to the Hyperparameter optimized ANN model. Training will be done to the model based on the optimized dataset and after training is completed model will be taken as an output.

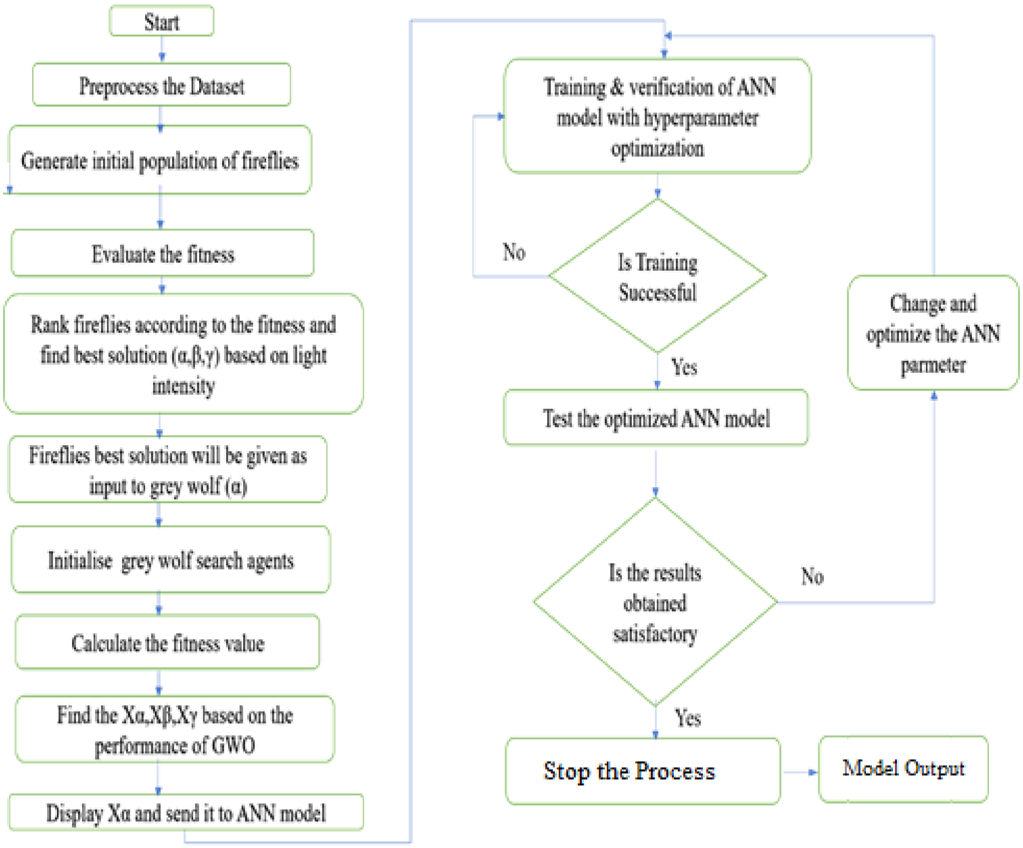
Start
Preprocess the Dataset
Generate initial population of fireflies
Evaluate the fitness
Rank fireflies according to the fitness and find best solution (α,β,γ) based on light intensity
Fireflies best solution will be given as input to grey wolf (α)
Initialise grey wolf search agents
Calculate the fitness value
Find the Xα,Xβ,Xγ based on the performance of GWO
Display Xα and send it to ANN model
Training & verification of ANN model with hyperparameter optimization
Is Training Successful
No
Yes
Test the optimized ANN model
Is the results obtained satisfactory
No
Yes
Change and optimize the ANN parmeter
Stop the Process
Model Output

Fig. 3. Flowchart of Feature selection using the H-FFGWO algorithm

Pseudocode for the Proposed H-FFGWO algorithm

```
Set the GW population as $y_j$ ( j = 1, 2, 3, 4, ……n)
Set the maximum iteration to α
Initiate the initial population
Define the objective function f(y)
Determine intensity (J) at cost (y) of each individual determined by $f(y_j)$
while (t< Inter maximum)
    for j = 1 to m
        for k = 1 to m
if $(I_k > I_j)$
                Move firefly j towards i in L dimension
            end if
                identify the new value and update (j)
        end for k
    end for j
    Find the best rank (α)
        for α search agent
    Update the position of α search agent
```

```
        end for
         Update b, B, and D
         Compute the fitness of α search agents
        Update Yα
        u = u + 1
end while
return Yα
```

Let r be the certain gap between the fireflies.

$$I(r) = \frac{I_s}{r^2} \tag{1}$$

Where $I(r)$ is the light intensity of fireflies with higher brightness.α - Randomness parameter, γ - Absorption coefficient, δ - Randomness reduction coefficient,β0 - Light amplitude.

The firefly with low intensity is attracted to fireflies with greater brightness value; every firefly is initiated with a specific brightness value, i.e., each one has an attractiveness parameter beta.

$$\beta(r) = \beta_0 e^{-\gamma r^2} \tag{2}$$

$\beta_0$denotes the attractiveness parameter of each firefly, r denotes the distance between the fireflies and γ denotes the coefficient of light absorption.

In the pseudocode of H-FFGWO, Maximum iterations are first set to the alpha variable. After defining the objective function of the firefly algorithm, based on the light intensity three different variables alpha, beta and gamma will be identified.

From the output, the best rank Alpha features will be given to the Greywolf Algorithm to find the best features from it. Based on the fitness of the variables best ranking alpha features alone will be selected and given as input to the Hyperparameter optimized ANN.

4.6.4 Movement of Firefly

The movement of firefly i, which is the lesser brighter one towards the brighter firefly, is given in (3).

$$x_i(t+1) = x_i(t) + \beta_0 e^{-\gamma r^2}\left(x_i - x_j\right) + \alpha \varepsilon_i \tag{3}$$

Where x_i is the position of the firefly which moves towards the firefly j with higher brightness. Hence this optimization technique compares the present updated attractiveness value with the older one [19]. If the updated position of the firefly produces a higher brightness parameter, the firefly is moved to a newer location; else, the position is retained [20]. The best firefly with a greater brightness value moves randomly to below (4)

$$x_i(t+1) = x_i(t) + \alpha \varepsilon_i \tag{4}$$

4.6.5 Searching

Searching for prey is usually influenced by combining the position or the location of the alpha and the other wolves. It usually splits during the searching and combines when attacking the prey [21]. In mathematical representation for searching, let a vector A which may contain some random numerical value within the limit of -1 to 1 be considered, and a parameter b which is linearly reduced from 0 to 2 influence the search and exploration. If the A vector value is greater than 1, they split to find prey and if it is less than 1, they combine to attack the prey.

4.6.6 Encircling

As given in **Fig. 4**, grey wolves involve in encircling once they find prey [11]. For mathematical representation, the distance of the prey from the wolf is calculated with the help of the position vector. Once the distance is determined, each wolf updates its location [22]. Equations (5) and (6) are used to measure the distance between the grey wolf and its prey.

$$\vec{E} = |\vec{D}.\vec{Y_p}(u) - \vec{Y}(u)| \tag{5}$$

$$\vec{Y}(u+1) = \vec{Y_p}(t) - \vec{B}.\vec{E} \tag{6}$$

Where u indicates the present iteration and Y denotes a vector that relates the location of the alpha wolf. The B and D vectors are calculated as given in (7) and (8).

$$\vec{B} = 2\vec{b}.\vec{s_1} - \vec{b} \tag{7}$$

$$\vec{D} = 2.\vec{s_2} \tag{8}$$

Where the components of a vector are reduced linearly from 2 to 0 and the random vectors in the limit between 0 and 1 are s1 and s2.

4.6.7 Hunting for Prey

The alpha wolf, which is dominant in the group guides the other wolves to involve in hunting [23]. Hence alpha is considered as the best individual solution and beta is taken to be the second best solution followed by delta [24]. Therefore, the first three fittest solutions are saved and the other individuals are forced to update their position based on the best solution obtained.

$$\vec{E_\alpha} = |\vec{D_1}.\vec{Y_\alpha} - \vec{Y}|, \vec{E_\beta} = |\vec{D_2}.\vec{Y_\beta} - \vec{Y}|, \vec{E_\delta} = |\vec{D_3}.\vec{Y_\delta} - \vec{Y}|$$

$$\vec{Y_1} = \vec{Y_\alpha} - \vec{B_1}.(\vec{E_\alpha}),$$

$$\vec{Y_2} = \vec{Y_\beta} - \vec{B_2}.(\vec{E_\beta}), \vec{Y_3} = \vec{Y_\delta} - \vec{B_3}.(\vec{E_\delta})$$

$$\vec{Y}(u+1) = \frac{\vec{Y_1}+\vec{Y_2}+\vec{Y_3}}{3} \tag{9}$$

Finally, a random circle is formed based on the fittest solution influenced by the search agents, i.e., based on the updated position of alpha and the other wolves given by (9). In a simple notation search, wolves or agents predict the location of the prey and the other agents randomly update their location by forming a circle from the prey.

4.6.8 Attacking

Finally, once the prey stops moving attacking occurs [25]. In mathematical representation, for approaching the prey, the vector is reduced linearly from a limit between 0 and 1. Therefore there is a change in a vector that has random values with the limit of -2b to +2b. Hence with the parameters proposed, the wolves upgrade their location based on the position of dominant leaders, circle the prey and lastly attack it.

Based on the above H-FFGWO detailed feature extraction procedure, the extracted dataset is given as input to Hyper-parameter optimized Artificial Neural Network and performance is compared with the existing Improved Optimization Algorithm.

5 Experimental Analysis

The performance of the proposed model is analyzed using accuracy, precision, recall and F1-score.

1. Accuracy

Accuracy [26], is defined, as the number of correctly predicted data to the Total number of tested data.

$$\frac{\text{Number of correctly predicted data}}{\text{Total number of tested data}} \times 100\%$$

2. Precision

The high value of precision indicates that the model has a low false-positive rate, which is good. In the proposed model, a precision of about 98.6% is obtained.

$$\text{Precision} = \text{True Positive} / \text{Total number of positive predictions.}$$

3. Recall

It is calculated as the ratio of the True Positive to the actual Positive. In the proposed model the recall of about 98.7 percent is obtained.

$$\text{Recall} = \text{True Positive} / (\text{True positive} + \text{False Negative})$$

4. F1 score

In the proposed model, the f1score of about .9 is obtained, which is above .5, which means that it performs better.

$$\text{F1 score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

5.1 Performance Metrics of ANN Model by Varying Epochs

The performance metrics for the ANN model are constituted by varying epochs and layers. Firstly, the ANN model is trained for various epochs and the performance metrics accuracy, precision, recall and f1 score are evaluated. **Table 2** shows the results obtained for varying epochs and hidden layers of the ANN model. For every increase in epochs, it is observed that the model gets deprived. At every layer, the model learns more about the input and the performance metrics increase gradually. It is observed that at 50 epochs, the accuracy is about 92.17 and when it is increased to 100 epochs, there is a 2.5% increase in accuracy. The ANN model performs better with 500 epochs and 4 layers which shows a 3% increase in accuracy as shown in **Table 2**.

Table 2. Varying Epochs for ANN

Epochs	Accuracy	Recall	Precision	F1-Score
50	92.17	91.54	94.15	92.81
100	94.57	94.16	95.41	94.78
200	94.96	94.5	95.16	94.81
300	95.1	95.21	95.42	95.3
400	95.6	95.36	95.71	95.53
500	95.86	95.51	95.83	95.67

5.2 Performance Metrics of ANN Model by Varying Optimizer for Loss Function

Several optimizers for loss function have been used to compile the model. Different optimizers for loss functions have been applied one by one and the performance metrics and the list of optimizers and loss functions used are evaluated. **Fig. 11** shows that when the Adam optimizer is varied for the Binary cross entropy loss function, a maximum increase in the accuracy of about 17.8% is obtained as shown in **Table 3**.

Table 3. With Epoch 500 Varying Optimizer with Binary Cross Entropy

Optimizer	Accuracy	Recall	Precision	F1-Score
Adam	98.8936	98.7921	98.9533	98.8725
AdaGrad	58.1363	55.2790	75.5226	62.0982
Adamax	98.2219	98.0041	98.4434	98.2223
Adadelta	16.5492	12.2962	56.2537	15.1244
SGD	82.6539	80.1423	86.3036	82.5882
Nadam	98.8080	98.6552	98.8535	98.7528
RMSdrop	98.2680	98.1248	98.3516	98.2370

5.3 Performance Metrics with H-FFGWO Algorithm

The performance metrics for the ANN model with H-FFGWO have been evaluated. The ANN model is trained for various epochs and the performance metrics are evaluated. **Table 4** makes it clear that when there is an increase in the number of epochs, it is observed that the ANN model gets deeper i.e., at every layer the model learns more about the input and the performance metrics increase gradually. It is observed that at 50 epochs, the accuracy is about 92.55 and when it is increased to 100 epochs, there is a 4.3% increase in accuracy as shown in **Table 4**. The model performs better at 500 epochs with 4 layers where there is a 5.7% increase in accuracy.

Table 4. Varying Hidden layers with 500 Epochs using H-FFGWO

layers	Accuracy	Recall	Precision	F1-Score
1	92.55	91.75	95.22	93.4
2	96.71	96.58	97.74	97.15
3	97.5	97.35	98.27	97.8
4	98.07	98.04	98.43	98.24

5.4 Comparison of H-FFGWO Feature Selection with Existing Feature Selection

A comparison is done for performance metrics with the H-ANN model by the proposed H-FFGWO feature selection to the Improved PSO, Improved GWO, and Improved FFA methods. It is observed that the performance metrics get improved when H-FFGWO is used for feature selection than the existing feature selection methods.

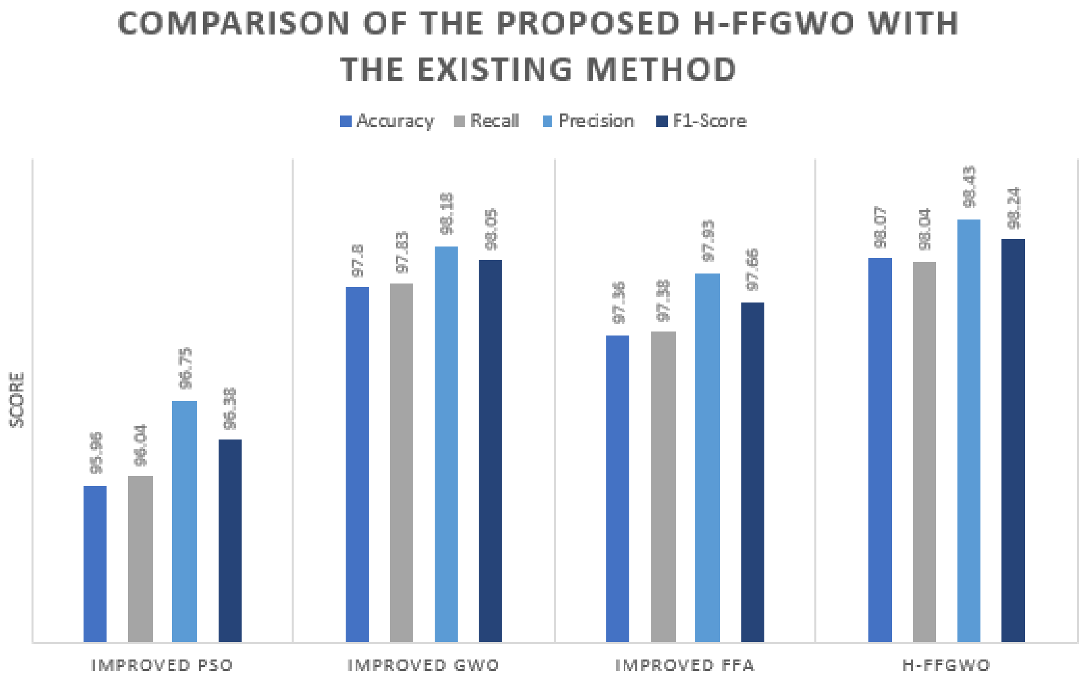

Fig. 4. Comparison of the proposed H-FFGWO feature selection method with Existing Methods

By using the H-FFGWO feature selection method combined with Hyperparameter optimized ANN model is trained and result obtained as accuracy 98.07%, a recall of 98.04%, a precision of 98.43%, and an F1-Score of 98.24% in layer 4 with epochs of 500are obtained. And it is compared with existing feature selection method.

Table 5. Comparison of the proposed H-FFGWO feature selection method with Existing Methods using Phish Tank.

Algorithms	Accuracy	Recall	Precision	F1-Score
I-PSO	95.96	96.04	96.75	96.38
I-GWO	97.8	97.83	98.18	98.05
I-FFA	97.36	97.38	97.93	97.66
H-FFGWO	98.07	98.04	98.43	98.24

Compared to the other optimization techniques, the proposed H-FFGWO has obtained an accuracy of 98.07%, a recall of 98.04%, a precision of 98.43%, and an F1-Score of 98.24% as shown in **Table 5**, which shows the improved result when compared to other existing algorithms.

Table 6. Comparison of the proposed H-FFGWO feature selection method with Existing Methods using the UCI machine-learning repository.

Algorithms	Accuracy	Recall	Precision	F1-Score
I-PSO	92.8	93.54	93.22	93.5
I-GWO	93.42	94.32	94.67	94.74
I-FFA	93.76	93.66	93.72	93.53
H-FFGWO	96.71	96.58	97.74	97.15

Compared to the other optimization techniques, the proposed H-FFGWO has obtained an accuracy of 96.71%, a recall of 96.58%, a precision of 97.74%, and an F1-Score of 97.15% as shown in Table 6, which shows the improved result when compared to other existing algorithms.

Table 7. Comparison of the proposed H-FFGWO feature selection method with Existing Methods using the Mendeley website dataset.

Algorithms	Accuracy	Recall	Precision	F1-Score
I-PSO	94.21	95.43	95.84	95.76
I-GWO	94.14	95.34	95.27	95.6
I-FFA	94.82	95.45	95.23	95.28
H-FFGWO	97.5	97.35	98.27	97.8

Compared to the other optimization techniques, the proposed H-FFGWO has obtained an accuracy of 97.5%, a recall of 97.35%, a precision of 98.27%, and an F1-Score of 97.8% as shown in Table 7, which shows the improved result when compared to other existing algorithms.

Table 8. Comparison of the proposed H-FFGWO feature selection method with Existing Methods using open phish.

Algorithms	Accuracy	Recall	Precision	F1-Score
I-PSO	89.21	89.12	93.67	91.45
I-GWO	90.14	89.66	93.72	89.26
I-FFA	89.44	89.64	93.54	89.92
H-FFGWO	92.55	91.75	95.22	93.4

Compared to the other optimization techniques, the proposed H-FFGWO has obtained an accuracy of 92.55%, a recall of 91.75%, a precision of 95.22%, and an F1-Score of 98.24% as shown in Table 8, which shows the improved result when compared to other existing algorithms.

5.4 Performance Comparison of H-FFGWO feature selection method with different phishing datasets

The proposed Algorithm has been verified using open phish, UCI machine learning repository, Mendeley website dataset and Phish tank dataset to verify the performance and the result is shown in Fig. 5.

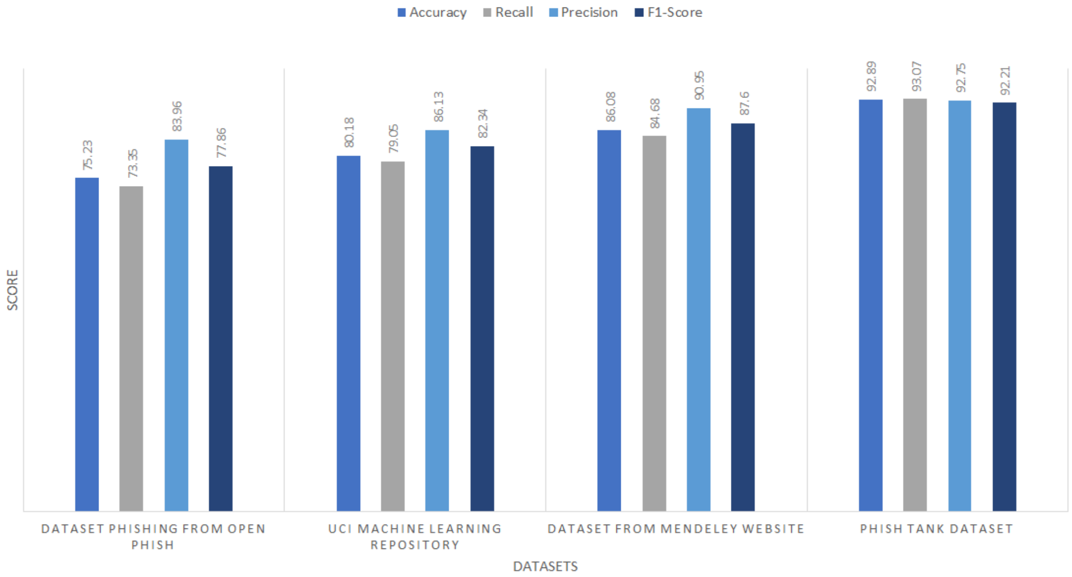

Fig. 5. Comparison of the proposed H-FFGWO feature selection method with Different Dataset

As shown in **Fig. 5** comparison has been done over a different dataset with reduced features after using H-FFGWO and verified using accuracy, precision, recall and F1-score.

Table 9. Comparison of the proposed H-FFGWO feature selection method with Different Dataset

Dataset	Accuracy	Recall	Precision	F1-Score
Open Phish	92.55	91.75	95.22	93.4
UCI	96.71	96.58	97.74	97.15
Mendeley	97.5	97.35	98.27	97.8
Phish Tank	98.07	98.04	98.43	98.24

Using the Open Phish dataset the result obtained an accuracy of 75.23%, Recall as 71.3%, Precision as 83.96% and F1-Score as 77.86%. Using the UCI machine learning repository obtained an accuracy of 80.18%, Recall of79.06%, Precision of86.13% and F1-Score of82.34%. Using Mendeley Website dataset the result obtained an accuracy of86.08%, Recall of84.68%, Precision of90.96% and F1-Score of87.6%. Using Phish Tank dataset the result obtained an accuracy of92.89%, Recall of93.07%, Precision of92.75% and F1-Score of92.21% is shown in **Table 9**.

6. Conclusion

In this paper, to identify the phishing URLs a combined model of the H-ANN model with H-FFGWO algorithms has been proposed. Different datasets have been used to train the H-ANN model. Eighty percent of the dataset has been split from the original dataset and the remaining dataset is used to evaluate the performance of the model (20%). By improving the number of hidden layers and epochs, the accuracy of the model improves. Features have been minimized using the hybrid optimization algorithm i.e., H-FFGWO. By using the reduced features, the H-ANN model is trained and an accuracy of 98.07%, a recall of 98.04%, a precision of 98.43%, and an F1-Score of 98.24% are obtained with the use of 4 layers and the epoch of 500. The H-

FFGWO algorithm shows better performance compared to the other existing hybrid algorithms based on the datasets taken for verification. The proposed model contains only a hybrid optimization algorithm for feature selection. Our future work is to incorporate the Ensemble model to compare the result with multiple feature selection algorithms.

References

[1] S. Mahdavifar and A. A. Ghorbani, “DeNNeS: deep embedded neural network expert system for detecting cyber attacks,” *Neural Computing and Applications*, vol. 32, no. 18, pp. 14753-14780, 2020. Article (CrossRef Link)

[2] S. Farrugia, J. Ellul, and G. Azzopardi, “Detection of illicit accounts over the Ethereum blockchain,” *Expert Systems with Applications*, vol. 150, p. 113318, 2020. Article (CrossRef Link)

[3] W. Ali and S. Malebary, “Particle swarm optimization-based feature weighting for improving intelligent phishing website detection,” *IEEE Access*, vol. 8, pp. 116766-116780, 2020. Article (CrossRef Link)

[4] M. T. Suleman and S. M. Awan, “Optimization of URL-based phishing websites detection through genetic algorithms,” *Automatic Control and Computer Sciences*, vol. 53, no. 4, pp. 333-341, 2019. Article (CrossRef Link)

[5] S. Naaz, “Detection of Phishing in Internet of Things Using Machine Learning Approach,” *International Journal of Digital Crime and Forensics (IJDCF)*, vol. 13, no. 2, pp. 1-15, 2021. Article (CrossRef Link)

[6] M. Radha Damodaram and M. Valarmathi, “Bacterial foraging optimization for fake website detection,” *International Journal of Computer Science & Applications (TIJCSA)*, vol. 1, no. 11, 2013. Article (CrossRef Link)

[7] N. Rana, M. S. A. Latiff, S. i. M. Abdulhamid, and H. Chiroma, “Whale optimization algorithm: a systematic review of contemporary applications, modifications and developments,” *Neural Computing and Applications*, vol. 32, no. 20, pp. 16245-16277, 2020. Article (CrossRef Link)

[8] Nam, Seung Yeob, and Sirojiddin Djuraev, “Defending HTTP web servers against DDoS attacks through busy period-based attack flow detection,” *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 8, no. 7, pp. 2512-2531, 2014. Article (CrossRef Link)

[9] M. Sabahno and F. Safara, “ISHO: Improved spotted hyena optimization algorithm for phishing website detection,” *Multimedia Tools and Applications*, vol. 81, pp. 34677-34696, 2022. Article (CrossRef Link)

[10] T. Liu and S. Yin, “An improved particle swarm optimization algorithm used for BP neural network and multimedia course-ware evaluation,” *Multimedia Tools and Applications*, vol. 76, no. 9, pp. 11961-11974, 2017. Article (CrossRef Link)

[11] M. Clerc, “The swarm and the queen: towards a deterministic and adaptive particle swarm optimization,” in *Proc. of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406) [Offline]*, IEEE, vol. 3, pp. 1951-1957, 1999. Article (CrossRef Link)

[12] N. Singh and S. Singh, “Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance,” *Journal of Applied Mathematics*, vol. 2017, 2017. Article (CrossRef Link)

[13] Khan, Muhammad Fahad, Farhan Aadil, Muazzam Maqsood, Salabat Khan, and Bilal Haider Bukhari, “An efficient optimization technique for node clustering in VANETs using gray wolf optimization,” *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 12, no. 9, pp. 4228-4247, 2018. Article (CrossRef Link)

[14] A. K. M. Khairuzzaman and S. Chaudhury, "Multilevel thresholding using grey wolf optimizer for image segmentation," *Expert Systems with Applications*, vol. 86, pp. 64-76, 2017. Article (CrossRef Link)

[15] N. Nekouie and M. Yaghoobi, "A new method in multimodal optimization based on firefly algorithm," *Artificial Intelligence Review*, vol. 46, no. 2, pp. 267-287, 2016.

[16] Wahid, Fazli, Lokman Hakim Ismail, Rozaida Ghazali, and Muhammad Aamir, "An efficient artificial intelligence hybrid approach for energy management in intelligent buildings," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 13, no. 12, pp. 5904-5927, 2019. Article (CrossRef Link)

[17] X.-S. Yang, "Firefly algorithms for multimodal optimization. in International symposium on stochastic algorithms," in *Proc. of SAGA 2009: Stochastic Algorithms: Foundations and Applications*, pp. 169-178, 2009. Article (CrossRef Link)

[18] D. Nanthiya, P. Keerthika, S. B. Gopal, S. B. Kayalvizhi, T. Raja, and R. Snegapriya, "SVM Based DDoS Attack Detection in IoT Using Iot-23 Botnet Dataset," in *Proc. of 2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, IEEE, pp. 1-7, 2021. Article (CrossRef Link)

[19] S. Łukasik and S. Żak, "Firefly algorithm for continuous constrained optimization tasks," in *Proc. of International conference on computational collective intelligence*, pp. 97-106, 2009. Article (CrossRef Link)

[20] S. K. Pal, C. Rai, and A. P. Singh, "Comparative study of firefly algorithm and particle swarm optimization for noisy non-linear optimization problems," *International Journal of intelligent systems and applications*, vol. 4, no. 10, pp. 50-57, 2012. Article (CrossRef Link)

[21] Gopal, S. B., C. Poongodi, D. Nanthiya, K. Harish, E. Divya, and N. Aarthy, "Enhancement of Routing Protocol for Low Power Lossy Network for Internet of Things," in *Proc. of 2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW)*, IEEE, pp. 354-356, 2018. Article (CrossRef Link)

[22] M. Kohli and S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems," *Journal of computational design and engineering*, vol. 5, no. 4, pp. 458-472, 2018. Article (CrossRef Link)

[23] M. A. Tawhid and A. F. Ali, "A hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function," *Memetic Computing*, vol. 9, no. 4, pp. 347-359, 2017. Article (CrossRef Link)

[24] C. Lu, L. Gao, X. Li, and S. Xiao, "A hybrid multi-objective grey wolf optimizer for dynamic scheduling in a real-world welding industry," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 61-79, 2017. Article (CrossRef Link)

[25] N. Mittal, U. Singh, and B. S. Sohi, "Modified grey wolf optimizer for global engineering optimization," *Applied Computational Intelligence and Soft Computing*, 2016. Article (CrossRef Link)

[26] S. B. Gopal, C. Poongodi, D. Nanthiya, K. Harish, E. Divya, and N. Aarthy, "Enhancement of Routing Protocol for Low Power Lossy Network for Internet of Things," in *Proc. of 2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW)*, IEEE, pp. 354-356, 2018. Article (CrossRef Link)

S. B. Gopal is an Assisatnt Professor of Department of Electronics and Communication Engineering in Kongu Engineering College, Perundurai and has 10 years of teaching experience. He hasobtained her B.E. in Electronics and Communication Engineering fromAnna University, Chennai in 2007, M.E. in Computer and CommunicationEngineering from Anna University in 2012. He has published twelve papers in Internationalconferences and seven papers in International journals and two book chapters. His research areasinclude Computer Networks, Network Secuirty, Cyber Security and IoT. His Research ID is orcid.org/ 0000-0002-3505-2412. Mail-id: s.b.gopalece@gmail.com.

C. Poongodi is a Professor of Department of Computer Science and Engineering in Vivekanandha College of Engineering for Women, Tiruchengode and has 20 years of teaching experience. She has obtained her B.E in Electronics and Communication Engineering from Bharathiar University, Coimbatore in 2001, M.E in Computer Science Engineering from Bharathiar University in 2002 and obtained Ph.D in Information and Communication Engineering from Anna University, Chennai during 2013. She has published more than forty papers in International conferences and in International Journals. She has obtained one Patent. She has written and edited four books. She has been occupying herself as a Journal Reviewer nationally and internationally since 2015. She has acted as co-chair for various International conferences. Her research areas include Internet of Things, Wireless networks and Artificial Intelligence. Her Research ID is orcid.org/0000-0002-4004-9461. Mail-id: poongodi321@yahoo.co.in.