International Journal of Innovative Research and Scientific Studies, 8(2) 2025, pages: 3703-3718

ISSN: 2617-6548

URL: www.ijirss.com

Leveraging pigeon-inspired optimizer with deep learning model on website phishing detection and classification for secure web mining

S. Jaiganesh[1], L.R.Aravind Babu[2*]

[1,2]*Department of Computer and Information Science, Annamalai University, Annamalai Nagar, Tamil Nadu, India*.

Corresponding author: L.R.Aravind Babu (*Email: er.arvee@rediffmail.com*)

Abstract

With the high growth of the Internet, the security of networks has stimulated individuals' attention. It is believed that a safe system atmosphere is an effective source for the fast and complete expansion of the Internet. Phishing is a vital type of cybercrime, which is a mischievous action of tricking consumers into clicking on phishing links, stealing consumer data, and eventually utilizing user information to fake log in with linked accounts to take assets. The models of phishing and the expertise of recognition are always being upgraded. With the progress and applications of machine learning (ML) technology, numerous ML-based solutions for detecting phishing have been developed. Some solutions depend upon the extraction of features by rubrics, while others require trusting third-party services, which can affect variability and lead to time-consuming issues in the forecasting service. Thus, this article develops a novel Pigeon Inspired Optimizer with a Deep Learning Model on Website Phishing Detection and Classification for Secure Web Mining (PIODL-WPDCWM) algorithm. The objective of the PIODL-WPDCWM technique lies in securing web mining activities and defending users from phishing attacks on websites. Primarily, the presented PIODL-WPDCWM technique involves z-score normalization to ensure that input features are standardized to a common scale. For the feature selection procedure, the brown-bear optimization algorithm (BBOA) has been employed to classify the most relevant and informative features from the data. Additionally, the self-attention-based long short-term memory and auto-encoder (S-LSTM-AE) classifier is deployed for the detection and classification of website phishing. Lastly, the pigeon-inspired optimizer (PIO) algorithm can be utilized for the hyperparameter tuning model of the S-LSTM-AE method. To certify the higher performance of the PIODL-WPDCWM technique, a wide range of simulation studies was conducted, and the attained outcomes demonstrated the improvement of the PIODL-WPDCWM technique over other existing models.

Keywords: Data normalization, Deep learning, Phishing attack, Pigeon-inspired optimizer, Web mining, Website.

DOI: 10.53894/ijirss.v8i2.6100
Funding: This study received no specific financial support.
History: Received: 3 March 2025 / **Revised:** 2 April 2025 / **Accepted:** 4 April 2025 / **Published:** 11 April 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

Phishing websites are semantic threats that aim at the user instead of the computer. They represent a comparatively innovative form of Internet crime compared to other types, such as hacking and viruses [1]. The phishing concern is challenging because it is extremely simple for attackers to generate convincing replicas of legitimate banking sites that may look quite authentic to consumers. The term "phishing" is derived from the phrase "website phishing," which itself is a play on the word "fishing" [2]. Phishing is the act of sending an email to a consumer, falsely claiming to be a legitimate business establishment, in an attempt to trick or scam the user into submitting confidential data that will be used for identity theft. The consequences include data security breaches involving the compromise of private information, and victims may ultimately suffer financial losses or other types of harm [3]. Phishing websites present a complicated issue to recognize and analyze; consequently, it intertwines social and technical problems, with no known single silver bullet to resolve it Aksu et al. [4]. Phishing threats are traditionally initiated by sending an email that appears to come from a reputable firm, requesting victims to confirm or update their data by clicking a link within the email. Figure 1 depicts the general structure of website phishing.

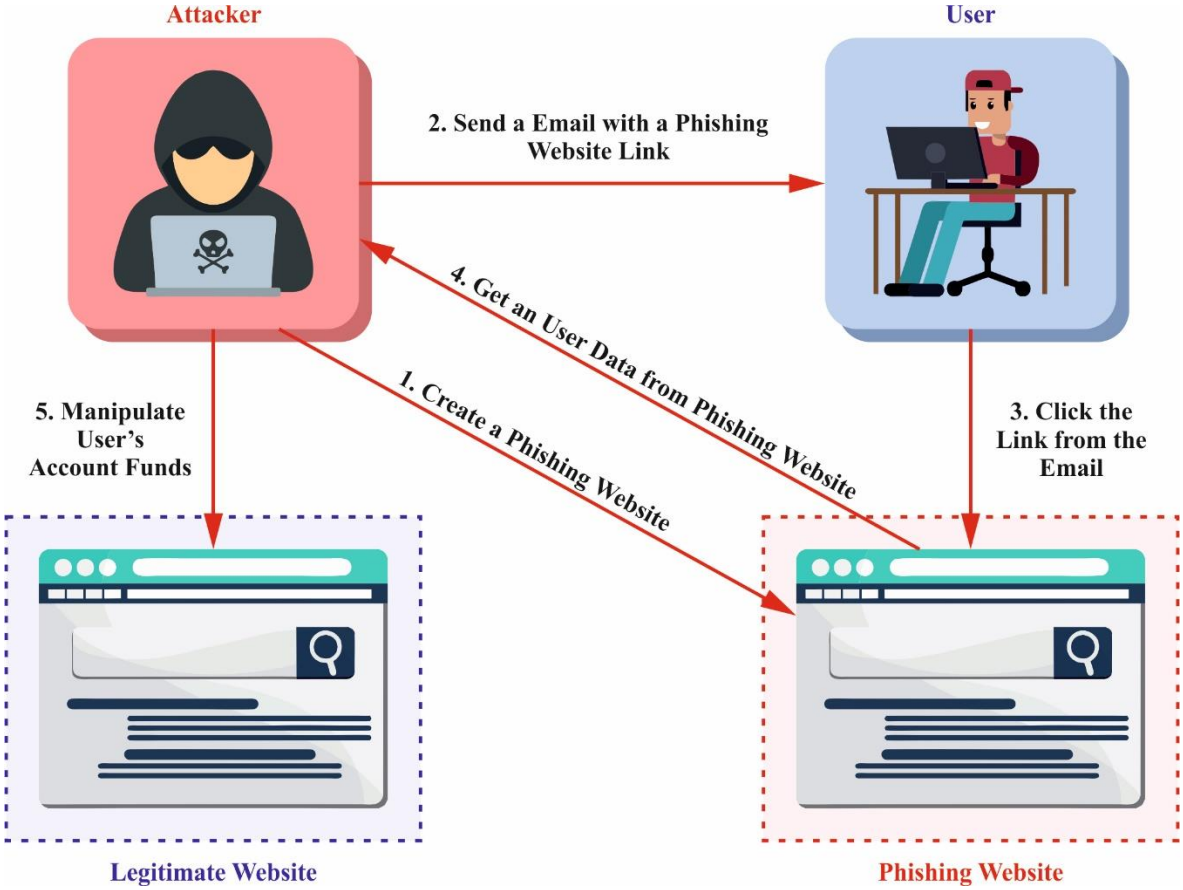

Figure 1.
General Structure of Website Phishing.

Though phishers instantly apply multiple methods for creating phishing websites to allure and fool users, all of them utilize a group of mutual features to generate phishing websites [5]. This can assist in distinguishing between phishing and honest websites depending on the feature extraction from the visited website. Recognizing phishing websites is a difficult task that requires significant specialist experience and knowledge. So far, multiple solutions have been advanced and presented to address these concerns. Data mining is a field of investigation that can make use of feature extraction from websites to discover patterns, along with relationships among others [6]. Data mining tools and models can identify e-banking phishing websites in an artificial intelligence (AI) model [7]. Classification and associative models might be extremely valuable for forecasting phishing websites. Data mining is the automatic extraction of previously unrealized data from large data resources to support actions [8]. The rapid growth of data mining has made it accessible to a broad range of models, represented by pattern recognition, statistical fields, databases, and machine learning (ML).

Currently, communication and information tools are utilized in a way that is extremely dense with data. In these conditions, multiple solution approaches for several types of difficulties have been advanced [9]. Deep Learning (DL) and Machine Learning (ML) models can be exploited in application expansion for information security. DL and ML models might be employed for classification purposes in several fields. Classification can be treated as a process to determine whether data belong to one of these categories in the dataset governed by certain rules [10]. Classification is utilized in several areas and holds a significant place and importance for information security.

This article develops a novel Pigeon Inspired Optimizer with a Deep Learning Model on Website Phishing Detection and Classification for Secure Web Mining (PIODL-WPDCWM) algorithm. Primarily, the presented PIODL-WPDCWM

technique involves z-score normalization to ensure that input features are standardized to a common scale. For the feature selection procedure, the brown-bear optimization algorithm (BBOA) has been employed. Additionally, the self-attention-based long short-term memory and auto-encoder (S-LSTM-AE) classifier is deployed for the detection and classification of website phishing. Lastly, the pigeon-inspired optimizer (PIO) algorithm can be exploited for the hyperparameter tuning model of the S-LSTM-AE method. To certify the enhanced performance of the PIODL-WPDCWM system, a wide range of simulation studies was conducted, and the attained outcomes establish the improvement of the PIODL-WPDCWM technique over other existing approaches.

2. Related Works

Sahingoz et al. [11] project that the growth of a phishing detection method depends on DL, applying 5 diverse models: CNN, ANN, BiRNN, RNN, and attention systems. This method is mainly aimed at the rapid classification of web pages utilizing URLs. To assess the performance method, a relatively broad labeled URL dataset, including nearly 5 million records, was shared and collected. In Kumar et al. [12], a Swarm Intelligence Binary Bat Algorithm (SIBBA) approach was utilized for intending the NN that are classified as the system URL websites related to the classification method. The DL-based Adam optimizer reaches higher classification precision. Alsubaei et al. [13] projected innovative DL methods, the RNT, embedded GRU, and ResNeXt approaches, rigorously advanced for real-world phishing threat recognition. The systematic method includes SMOTE for handling data imbalance through primary processing of the data. This discriminative method is capable of enhancing the process of feature extraction, while AE and ResNet (EARN) were incorporated with feature engineering.

Pillai and Sharma [14] presented a hybrid unsupervised recognition method a DL-based anomaly-based web attack recognition. While the De-Noising Autoencoder (DAE) encrypted outputs, along with Stacked AE, are incorporated and granted to GAN as an input. Thus, to classify the kind of threats, an innovative DBM-Bi LSTM-based classification method was presented. In Asiri et al. [15], a method that identifies 3 kinds of phishing threats: regular phishing threats, Browser in the Browser (BiTB), and Tiny Uniform Resource Locators (TinyURLs) is intended. This method was divided into 3 kinds: Docker container, browser extension, and DL model. Initially, a DL method is intended by utilizing Bi-LSTM and an attention method for categorizing the URL. Then, an extension of the browser is intended to remove the novel URL from the suspected webpage. Afterward, the Docker container unlocks the website and removes every URL from its JavaScript and HTML. Alohali et al. [16] implemented an innovative Metaheuristics DL-oriented Phishing Detection (MDLPD-SSE) method. Moreover, the LSTM method is employed in this paper to recognize phishing. To end with, the Bald Eagle Search (BES) optimizer model is employed to fine-tune the hyper-parameters significant to the LSTM approach.

Zhu et al. [17] projected that Phishing Detection depends on Hybrid Features (PDHF), an innovative phishing recognition method that depends on a combination of automated DL and optimum artificial features. The optimum artificial phishing characteristics were attained by extracting redundant characteristics and an enhanced bi-directional searching model. To increase the actual period of phishing recognition and deep features are gathered from the URL utilizing a disorderly quantized attention mechanism and a 1D character CNN. In Aldakheel et al. [18], an innovative approach for recognizing phishing sites with higher precision. This method employs a CNN-based method for accurate classification, which efficiently differentiates legal websites from phishing websites. This method projects a single contribution to the phishing recognition field by attaining higher precision rates and breaking existing advanced methods.

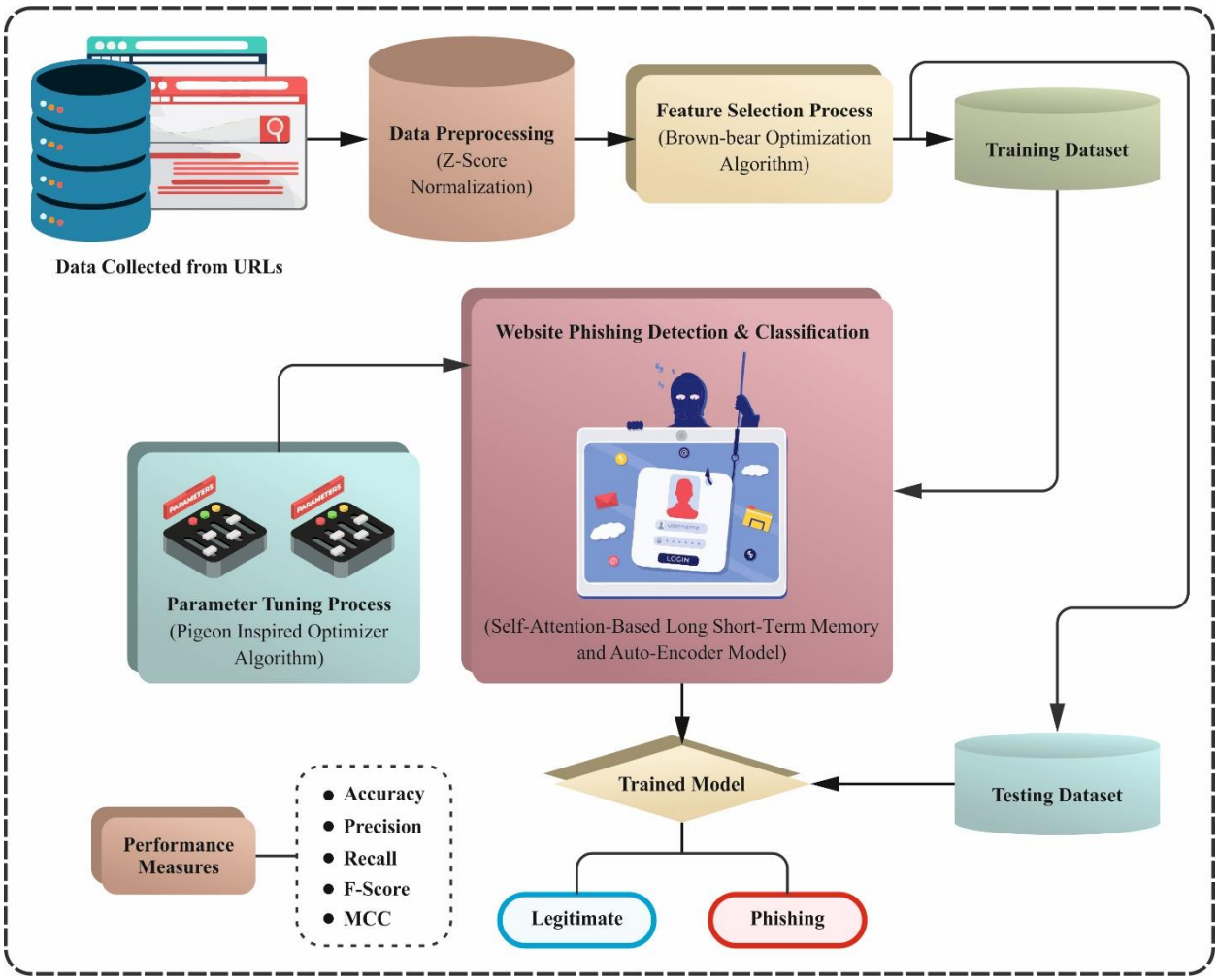

Figure 2.
Overall Flow of PIODL-WPDCWM algorithm.

3. Materials and Methods

In this article, we have developed a novel PIODL-WPDCWM algorithm. The objective of the PIODL-WPDCWM technique lies in securing web mining activities and defending users from phishing attacks on websites. It encompasses four major stages are involved as z-score normalization, BBOA-based feature subset selection, website phishing detection, and PIO-based parameter tuning process. Figure 2 represents the entire flow of the PIODL-WPDCWM algorithm.

3.1. Z-score Normalization

Primarily, the presented PIODL-WPDCWM technique involves z-score normalization to ensure that input features are standardized to a common scale. Z-score normalization is an arithmetical model that is employed in order to normalize the features of a database, certifying that they have a standard deviation of 1 and a mean of 0 [19]. In the situation of website phishing recognition, Z-score normalization is employed for the features mined from websites (like HTML tags, URL length, and link features) to eliminate any biases caused by opposing feature measures. This certifies that every feature donates similarly to the technique, enhancing the performance of ML techniques. By altering the data into an even measure, Z-score normalization aids the technique to unite faster and creates the recognition of phishing websites more precisely. It is chiefly valuable when employing intricate methods such as DL or ensemble models.

3.2. BBOA-Based Feature Subset Selection

For the feature selection process, the BBOA is employed to classify the most related and informative features from the data. The initial phases of the BBOA are inspired by the Brown-bear's sniffing and pedal scent-marking behaviors [20]. Different groups of BBs are arbitrarily produced inside an identified land, by all groups are marked by pre-determined pedal scent mark counts. This method is mathematically stated as shown:

$$P_{i,j} = P_{i,j_{min}} + \gamma.\left(P_{i,j_{max}} - P_{i,j_{\min}}\right) \quad (1)$$

Whereas P_{ii} signifies the $j\ th$ design inside the $i\ th$ group of BB paths. $P_{i,j_{\min}}$ and $P_{i,j_{max}}$ signify the lower and upper limits of the scent marking, correspondingly, and γ designates randomly generated values in the interval of 0 and 1.

When the total number of groups inside a land is np, and the complete amount of trail marks (for example, decision variable counts) in every cluster is nd, formerly the aggregate group of possible solutions (P) is described as

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,nd} \\ P_{2,l} & P_{2,2} & \dots & P_{2,nd} \\ \vdots & \vdots & \ddots & \vdots \\ P_{np,l} & P_{np,2} & \dots & P_{np,nd} \end{bmatrix} \quad (2)$$

Three attributes (i) distinguishing foot actions during the walk, (ii) caution steps, and (iii) turning of the feet into impressions at ground level are arithmetically demonstrated to imitate the pedal scent marking behavior. It is mathematically represented in Equation 3:

$$P_{i,j,k}^{n} = P_{i,j,k}^{o}, -\left(\lambda_k . c_{i,j,k}, P_{\mathrm{i,j,k}}^{o},\right) \quad (3)$$

Now, $P_{i,j,k}^{n}$, characterizes the upgraded $j\ th$ trail marks of the $i\ th$ group of BBs in the $k\ th$ iteration, $P_{i,j,k}^{o}$ signifies the previous $j\ th$ trail marks of the $i\ th$ cluster of bears inside a similar period. Additionally, $c_{i,j,k}$, characterizes numbers distributed at random amongst$(0, 1)$, related to the footmark of $i\ th$ cluster of bears for the $k\ th$ iterations, and λ_k specifies the event feature for the $k\ th$ iteration, which rises linearly through the iteration counts. The second third of each of the iterations is stated in Equation 4:

$$P_{i,j,k}^{n} = P_{i,j,k}^{o} + A_k . \left(P_{j,k}^{b} - B_k . P_{\mathrm{j,k}}^{w}\right) \quad (4)$$

Whereas $P_{j,k}^{b}$ and $P_{j,k}^{w}$ signify the $j\ th$ top rank and $j\ th$ lower rank pedal scent marking, correspondingly, detected within the $k\ th$ iterations. A_k signifies the step aspect within the $k\ th$ iteration, and B_k designates the step length in that iteration. The fitness function (FF) imitates the accuracy of the classifier and the extent of preferred features. So, the below-mentioned FF is applied to assess individual solutions. Its mathematical equation is shown in Equation 5.

$$Fitness = \alpha * ErrorRate + (1 - \alpha) * \frac{\#SF}{\#All\_F} \quad (5)$$

Here, $ErrorRate$ specifies the classifier error ratio of the chosen features. $ErrorRate$ is proposed as the ratio of inappropriate classifications considered that classification counts made, definite as a value between 0 and 1. $\#SF$ refers to the number of preferred features and $\#All\_F$ denotes the total quantity of features. α is applied to switch the implication of classifier superiority and subset length.

3.3. Website Phishing Detection using S-LSTM-AE

Additionally, the S-LSTM-AE classifier is deployed for the recognition and classification of website phishing. The AE is the unsupervised learning method intended for handling unlabeled data [21]. The important inequality in the gathered data is considered as a majority of standard information and a lack of error information, in addition to the need for manual labeling by specialists. Using an AE architecture for the task of recognition is particularly suitable. AEs normally contain dual modules: an encoder and a decoder. The encoder is directed to map input data to a lower-dimensional latent space model, whereas the decoder converts the latent model back to an input region, aiming to rebuild unique information as precisely as possible.

The encoder and decoder procedures are defined below:

$$h = g_{\theta_1}(x) = \sigma(W_1 x + b_1) \quad (6)$$

$$\hat{x} = g_{\theta_2}(h) = \sigma(W_2 h + b_2) \quad (7)$$

Now, W characterizes the weights, b indicates the biases, x embodies input data, h and $\hat{x}$ refer to the encoding and decoding output, respectively; and an activation function has been utilized, by the function of the sigmoid to be utilized in this regard.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (8)$$

The encoding gets an input X and maps it to the representation of a latent area h, whereas the decoding re-mapping h gets back to an input area for giving the rebuilt signals. The function of loss is normally described as the mean squared error (MSE), while N signifies the training sample counts. Network training aims to reduce this MSE.

$$\min \frac{1}{N} \sum_{i=1}^{N} \| x - \tilde{x} \|_2^2 \quad (9)$$

The trained AE targets for learning the accurate likelihood distribution of input data as precisely as promising. LSTM systems are a specified form of recurrent neural networks (RNNs) tailored for predicting and processing time-series data. They deal with the tasks of learning longer sequences, in addition to alleviating problems associated with gradient explosion and gradient vanishing. LSTM systems have a novel architecture of cells, which handles the information flow through a complicated gating mechanism. Unlike a conventional RNN, LSTM presents a cell state, forget, input, and output gates. This gating mechanism is observed as a fully connected (FC) layer. It enables storage of data and upgrades over this gate method, especially using the function of Sigmoid in addition to dot product processes. Whereas RNNs only transmit hidden layers (HLs), LSTM combines cell states additionally. While f_t, i_t, and 0_t characterize the forget, input, and output gates correspondingly. $Tanh$ and σ refer to the activation function of hyperbolic tangent and sigmoid, which maps data to the range [1, 1] and [0,1]. W_c, W_f, W_i, W_o characterize the weights related to the cell state, forget, input, and output gates, individually. $W_f, W_i, Bf, bi, b_c,\ b_o$ denote biased terms. $W_{f,i_{Xt}}$ signifies an input at $t$he time. h_{t-1} and h_t symbolize an output at the time t and $t-1$, correspondingly. $\mathcal{C}_t$ denotes the temporary condition. C_{t-1} and C_t characterize the state at the time t and $t-1$, correspondingly.

Forgetting Gate: It normalizes whether previous longer-range memory data must be rejected.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \quad (10)$$

Input Gate: It defines the sum of data, which can be recovered from the longer-range memory unit for outputs.

$$C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (11)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (12)$$

Output Gate: It controls the sum of data, which is recovered from the longer-range memory unit for outputs.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (13)$$

$$h_t = o_t * \tanh(C_t) \quad (14)$$

Longer-Term Memory Unit: Mainly used to store and process previous data, in addition to filtering and sifting data.

Short-Term Memory Unit: Intended to keep the current output and send this back into the system.

$$C_t = f_t * C_{t-1} + i_t * C_t \quad (15)$$

The LSTM method inputs successive data into a gate of input that inscribes the information to the longer-range memory unit as needed. The gate of forgetting controls which data must be removed, but the gate of output recovers the basic data from the longer-range memory unit for the output. The Attention mechanism has grown considerably, and it is initially used in the visual field. This mechanism aids in removing the main data related to every time window. At last, the n length sequences m output from the encoding is passing over a linear layer to gain the value matrix: $V = [V^1, V^2, \cdots, V^m]$, key matrix: $K = [K^1, K^2, \cdots, K^m]$, query matrix: $Q = [Q^1, Q^2, \cdots, Q^m]$. The computation procedure is exposed below:

$$\begin{cases} Q^i = W^q X^i \\ K^i = W^k X^i \\ V^i = W^v X^i \end{cases} \quad (16)$$

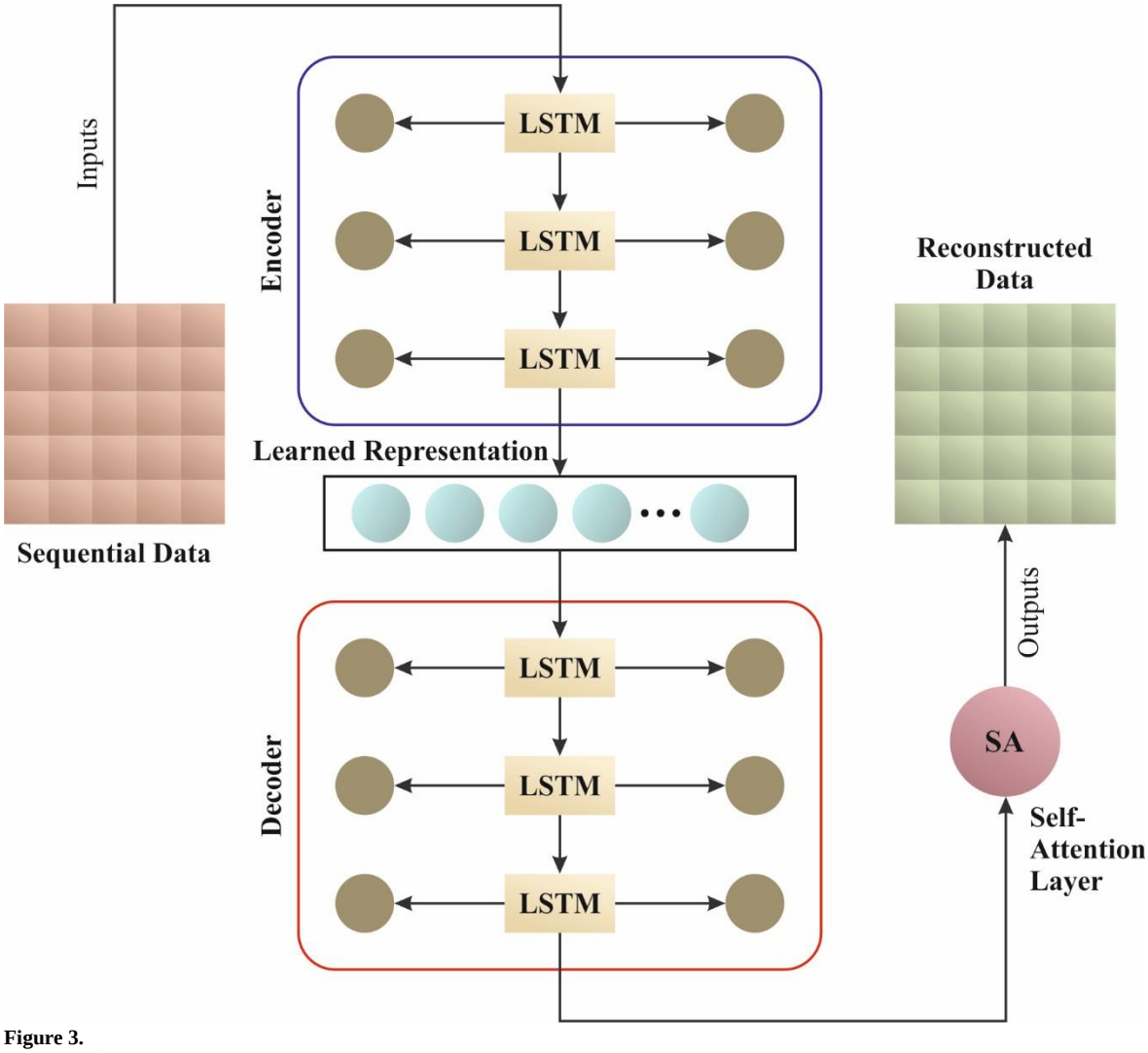

Figure 3.
Structure of S-LSTM-AE.

Subsequently, the dot-product between the vector of keywords K^i and the vector of query Q^i for every time it is computed, and the dot-product is separated by $\sqrt{d_k}$, while d_k denotes the size of the keyword vector, and the outcome is standardized utilizing $SoftMax$. The value of attention can be multiplied by the vector value V_i. Figure 3 represents the architecture of the S-LSTM-AE model.

$$Attention\ (Q,K,V) = SoftMa\chi\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (17)$$

3.4. Hyperparameter Tuning Process

Lastly, the PIO algorithm can be exploited for the hyperparameter tuning model of the S-LSTM-AE method. By mimicking homing pigeons, we presented the PIO model [22]. Landmarks and magnetic fields are exploited by pigeons to detect routes in homing. So, dual operators are presented: the landmark operator and the map and compass operator.

3.5. Map and Compass Operator

Pigeons may identify the earth's magnetic arena utilizing magnetic objects and might create cognitive maps. They exploit the sun's elevation as a compass to fine-tune their direction of travel, and the reliance on magnetic objects and the sun decreases after approaching the target.

3.6. Landmark Operator

This operator has been applied to mimic the inspiration of landmarks on pigeons during direction finding. For example, pigeons address the target, close landmarks are looked for. When the pigeons are aware of landmarks, they fly straight to the target. Or else, it follows the pigeon's flight sensible of the landmarks.

3.7. Mathematical Approach

The assumption that the search space is n-dimensioned, the pigeon i is mathematically characterized by an n-dimensioned vector $X_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})$. Also, the speed of all pigeons is stated as other n-dimensioned vectors, $V_i = (v_{i,1}, v_{i,2}, \dots, v_{i,n})$. The finest global location is characterized by $X_g = (x_{g,1}, x_{g,2}, \dots, x_{g,n})$. Formerly, every pigeon upgrades the velocity and location based on the succeeding dual Equations. (18) and (19)

$$V_i(t) = V_i(t-1)e^{-Ri} + rand\left(X_g - X_j(t-1)\right) \quad (18)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (19)$$

Whereas t characterizes the present iteration counts. R represents the map and compass feature, in an interval of 0 and 1 which controls the impact of the current on the present velocities. Finally, $rand$ refers to a randomly created number distributed uniformly in the interval of 0 and 1. Equation 18 upgrades the velocity of the pigeon based on the current velocity of the pigeons and the distance between the pigeon's present location and the globally finest location. The pigeon formerly upgrades the location by a novel speed based on Equation 19.

During this landmark operator, pigeons depend on milestones for navigating. Next, every iteration, the pigeon count is reduced by half based on Equation 20. It is far away from the target, unaware of the landmarks, and fails to recognize the route, thus, these pigeons are rejected. X_c characterizes the middle location of the residual pigeon, which helps as a landmark and guide for flight. These equations are applied in the landmark operator as shown.

$$N_p(t) = \frac{N_p(t-1)}{2} \quad (20)$$

$$X_c(t) = \frac{\Sigma_{n=1}^{N_p(t)} X_i(t) fitness\left(X_i(t)\right)}{N_p \Sigma_{n=1}^{N_p(t)} fitness\left(X_i(t)\right)} \quad (21)$$

$$X_i(t) = X_i(t-1) + rand\left(X_c(t) - X_i(t-1)\right) \quad (22)$$

Whereas N_p refers to population size; *fitness* signifies the estimation function computing the fitness of all pigeons. Equation 21 has been applied to compute the middle value of the residual pigeons, and then all pigeons fly toward a novel location based on Equation 22. Formerly, the requisite iteration counts are attained inside the landmark operator. So, the operator ends work, and also the model ends.

The workflow of the method is shown below.

Stage 1: Initializing the parameter, dimension of the pigeon N, The map and compass feature R, and the maximal iteration counts $N1$, and N2 for the landmark operator.

Stage 2: Arbitrarily make N pigeons, assess every individual, and define the finest pigeon X_g.

Stage 3: Performing the map and compass factor by upgrading every velocity and location of the pigeons, estimating the fitness of each of the pigeons, and establishing the finest pigeon X_g.

Stage 4: Checking the end criteria for the iteration; when the end criteria of the map and compass factor are encountered, go to Stage 5. Or else to Stage 3.

Stage 5: Performing the landmark operator by upgrading every velocity and location of the pigeons, estimating the fitness of each of the pigeons, and establishing the finest pigeon X_g.

Stage 6: Checking the end criteria for the iteration; when the end criteria of the map and compass factor are encountered, stop. Or else, go to Stage 5.

The PIO model originates an FF to get the amended performance of classification. It refers to a positive number to imply a better solution of the candidate solution. Here, the reduction in the classifier rate of error is reflected in FF, which is formulated in Equation 23.

$$fitness(x_i) = ClassifierErrorRate(x_i)$$

$$= \frac{No.of\ misclassified\ instances}{Total\ no.of\ instances} * 100 \qquad (23)$$

4. Performance Validation

In this section, the performance study of the PIODL-WPDCWM approach is examined under the PhiUSIIL Phishing URL dataset [23, 24]. It contains 100000 samples under two classes, as shown in Table 1. There are 54 no, of features but only 36 features are chosen.

Table 1.
Details of the database.

Class	No. of Instances
Legitimate	50000
Phishing	50000
Total Instances	**100000**

Figure 4 displays the confusion matrices formed by the PIODL-WPDCWM approach. The outcomes require that the PIODL-WPDCWM method has effective detection and identification of all classes properly.

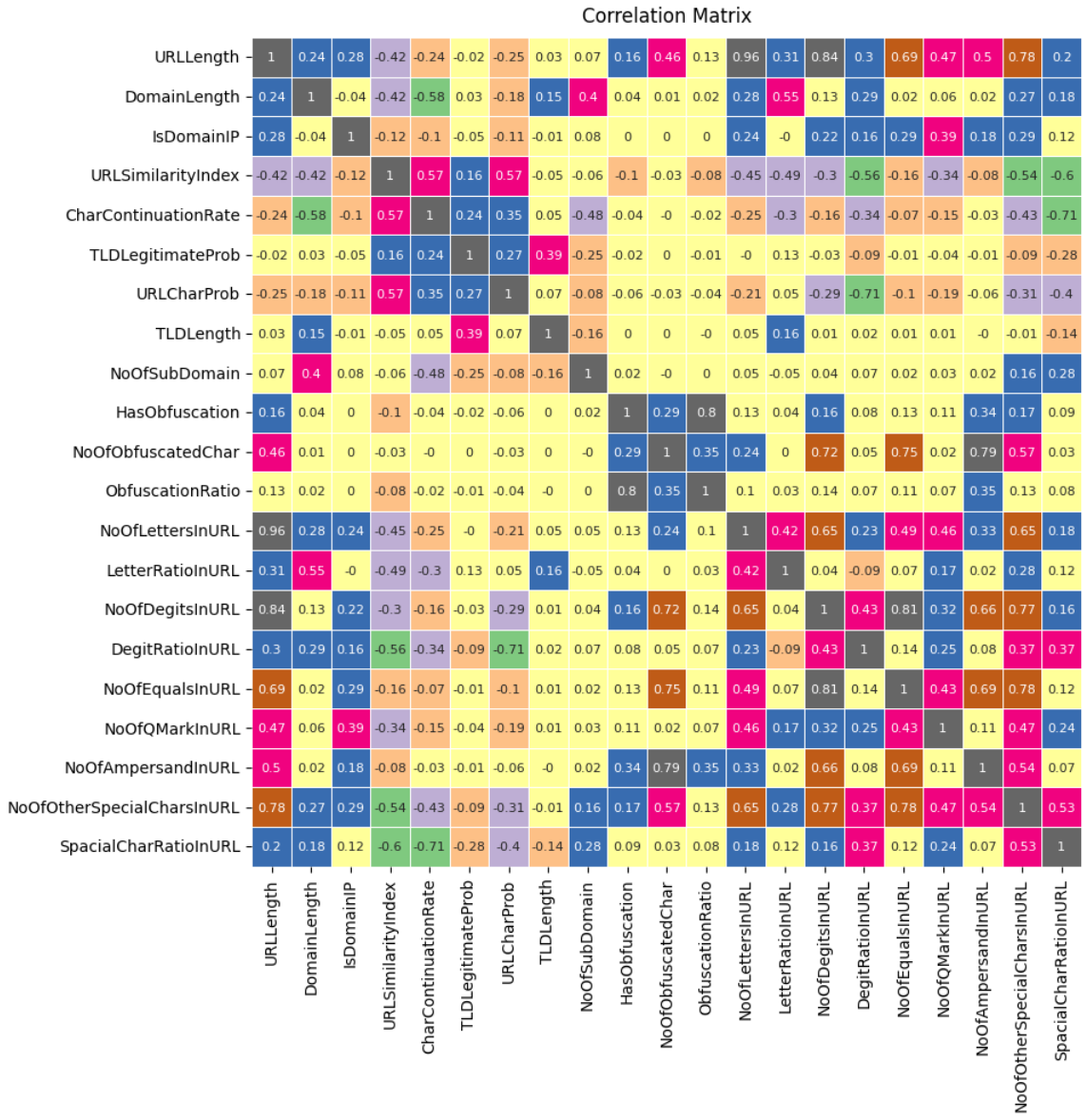

Figure 4.
Correlation matrix of PIODL-WPDCWM model.

Figure 5 establishes the confusion matrices generated by the PIODL-WPDCWM approach under different epochs. The results stipulate that the PIODL-WPDCWM system has effective detection and identification of Legitimate and Phishing classes exactly.

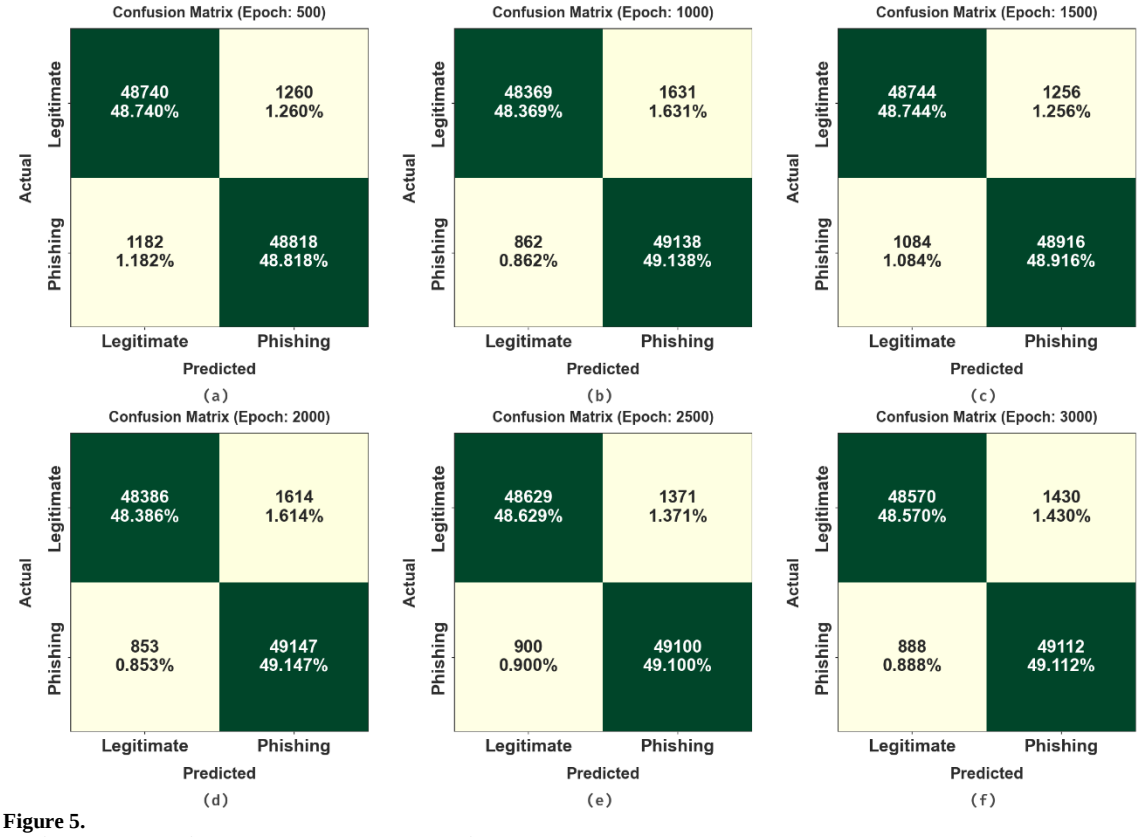

Figure 5.
Confusion matrices of BADD-SAODFF technique (a-f) Epochs 500-3000.

The Phishing website recognition of the PIODL-WPDCWM model is demonstrated under distinct epochs in Table 2 and Figure 6. The table values state that the PIODL-WPDCWM system correctly recognized all the classes. On 500 epochs, the PIODL-WPDCWM technique provides an average $accu_y$ of 97.56%, $prec_n$ of 97.56%, $reca_l$ of 97.56%, F_{score} of 97.56%, and MCC of 95.12%. Besides, on 1000 epochs, the PIODL-WPDCWM system gets an average $accu_y$ of 97.51%, $prec_n$ of 97.52%, $reca_l$ of 97.51%, F_{score} of 97.51%, and MCC of 95.03%. Moreover, on 1500 epochs, the PIODL-WPDCWM approach attains an average $accu_y$ of 97.66%, $prec_n$ of 97.66%, $reca_l$ of 97.66%, F_{score} of 97.66%, and MCC of 95.32%. Also, on 2500 epochs, the PIODL-WPDCWM model delivers an average $accu_y$ of 97.73%, $prec_n$ of 97.73%, $reca_l$ of 97.73%, F_{score} of 97.73%, and MCC of 95.46%. At last, on 3000 epochs, the PIODL-WPDCWM approach attains an average $accu_y$ of 97.68%, $prec_n$ of 97.69%, $reca_l$ of 97.68%, F_{score} of 97.68%, and MCC of 95.37%.

Table 2.
Phishing website recognition of the PIODL-WPDCWM model under dissimilar epochs.

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	MCC
Epoch - 500					
Legitimate	97.48	97.63	97.48	97.56	95.12
Phishing	97.64	97.48	97.64	97.56	95.12
Average	97.56	97.56	97.56	97.56	95.12
Epoch - 1000					
Legitimate	96.74	98.25	96.74	97.49	95.03
Phishing	98.28	96.79	98.28	97.53	95.03
Average	97.51	97.52	97.51	97.51	95.03
Epoch - 1500					
Legitimate	97.49	97.82	97.49	97.66	95.32
Phishing	97.83	97.50	97.83	97.66	95.32
Average	97.66	97.66	97.66	97.66	95.32
Epoch - 2000					
Legitimate	96.77	98.27	96.77	97.51	95.08
Phishing	98.29	96.82	98.29	97.55	95.08
Average	97.53	97.54	97.53	97.53	95.08
Epoch - 2500					
Legitimate	97.26	98.18	97.26	97.72	95.46
Phishing	98.20	97.28	98.20	97.74	95.46
Average	97.73	97.73	97.73	97.73	95.46
Epoch - 3000					
Legitimate	97.14	98.20	97.14	97.67	95.37
Phishing	98.22	97.17	98.22	97.69	95.37
Average	97.68	97.69	97.68	97.68	95.37

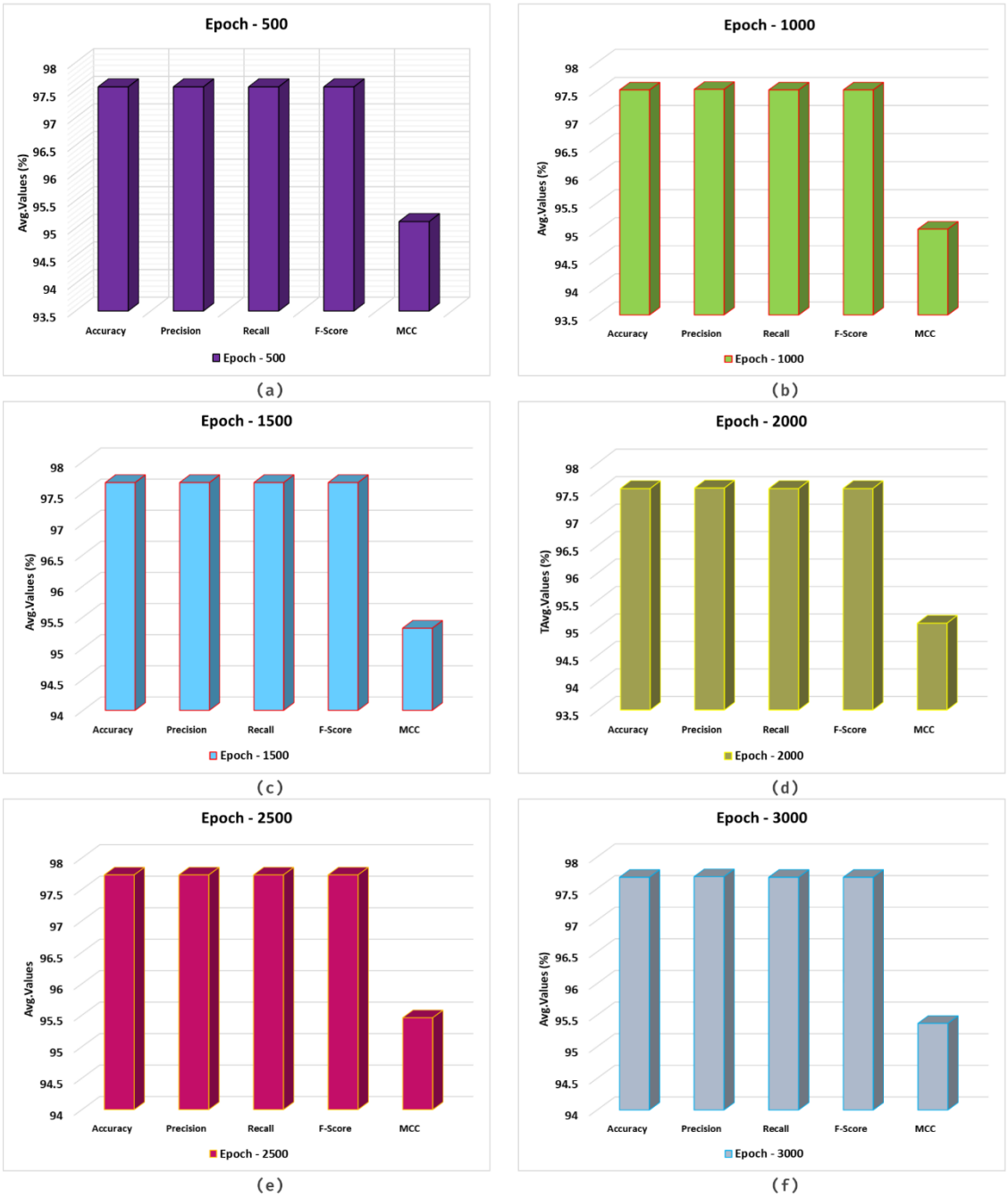

Figure 6.
Average outcome of PIODL-WPDCWM model (a-f), Epochs 500-3000.

In Figure 7, the training (TRA) $accu_y$ and validation (VAL) $accu_y$ performances of the PIODL-WPDCWM technique under epoch 2500 are depicted. The $accu_y$ values are evaluated through a range of 0-25 epochs. The outcome underscored that the values of TRA and VAL $accu_y$ shows an increasing trend, indicating the capability of the PIODL-WPDCWM algorithm through enhanced performance across numerous repetitions. Furthermore, the TRA and VAL $accu_y$ values remain close through the epochs, notifying of the lesser overfitting and displaying the greater result of the PIODL-WPDCWM method, which guarantees reliable predictions on unnoticed samples.

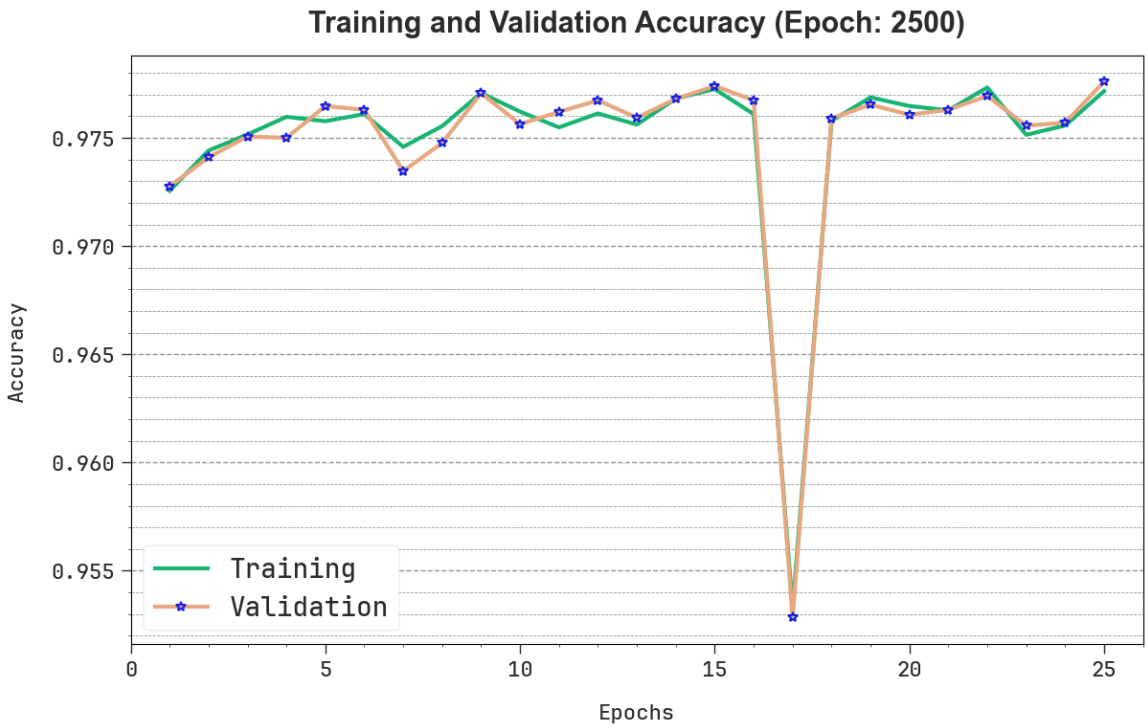

Figure 7.
$Accu_y$ curve of PIODL-WPDCWM technique under Epoch 2500.

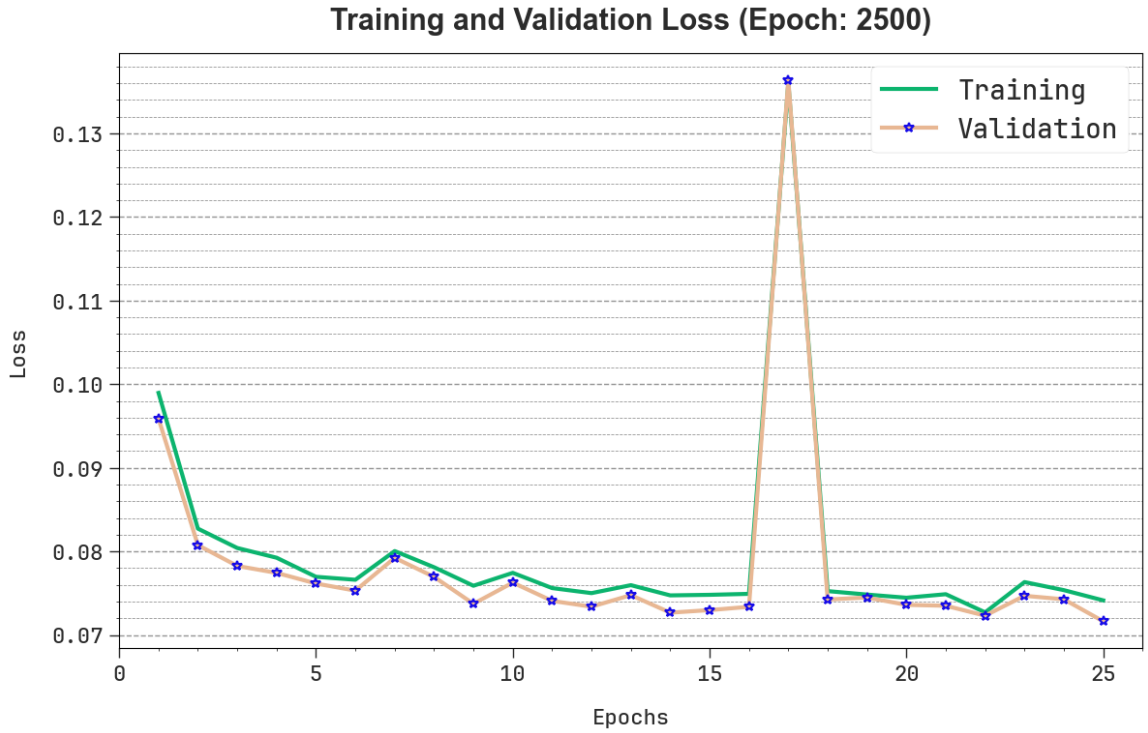

Figure 8.
Loss curve of PIODL-WPDCWM technique at Epoch 2500.

In Figure 8, the TRA loss (TRALOS) and VAL loss (VALLOS) curves of the PIODL-WPDCWM technique under epoch 2500 are shown. The values of loss are estimated across a range of 0-25 epochs. It is showcased that the values of TRALOS and VALLOS represent a diminishing trend, which notified the proficiency of the PIODL-WPDCWM approach in corresponding a trade-off between generalized and data fitting.

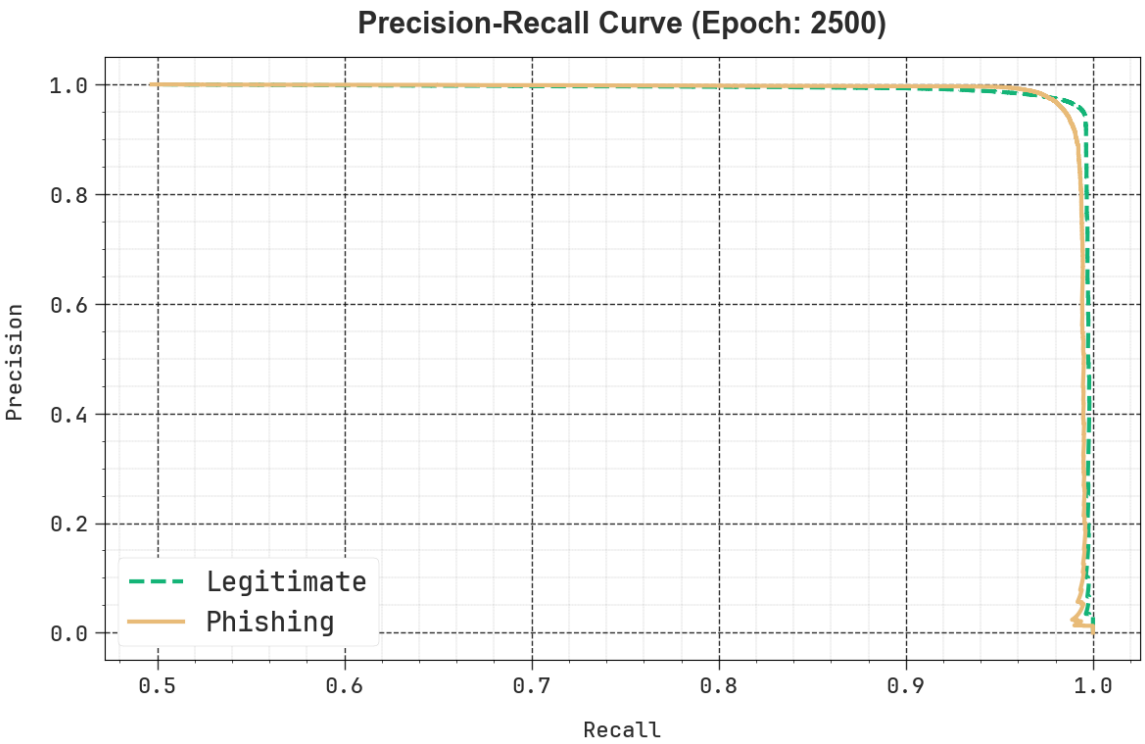

Figure 9.
PR curve of PIODL-WPDCWM system over Epoch 2500.

In Figure 9, the PR outcome investigation of the PIODL-WPDCWM model under epoch 2500 provides clarification into its outcome by scheming Precision against Recall for 2 classes. The outcome illustrates that the PIODL-WPDCWM technique constantly achieves higher PR values over distinct class labels, which notified its proficiency to preserve a substantial portion of true positive predictions among all the positive predictions while similarly capturing a large proportion of actual positives.

In Figure 10, the ROC outcome of the PIODL-WPDCWM method under epoch 2500 is examined. The performances indicate that the PIODL-WPDCWM technique attains enhanced ROC analysis across each class, which represents substantial proficiency in discerning the classes. This consistent tendency of higher ROC curve outcomes through several classes implies the skillful outcome of the PIODL-WPDCWM system in predicting classes, which indicates the robust nature of the classification method.

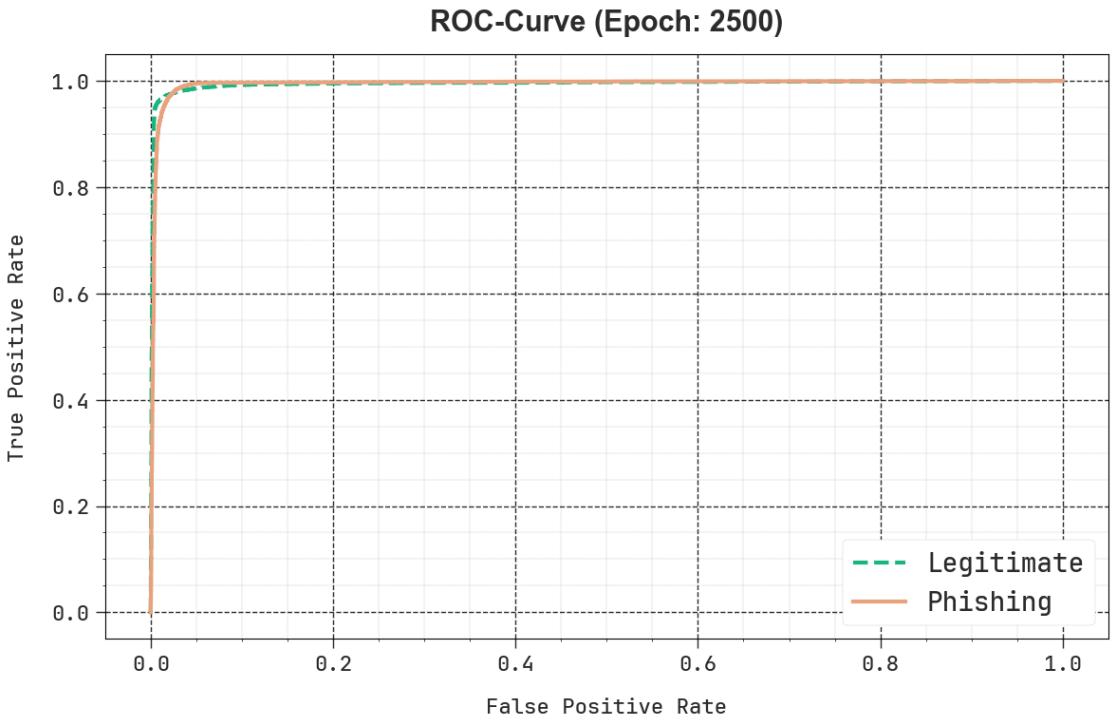

Figure 10.
ROC curve of PIODL-WPDCWM technique under Epoch 2500.

Table 3 and Figure 11 demonstrate the comparison investigation of the PIODL-WPDCWM system with other systems [25-27]. The table outcomes implied that the RNN, LSTM-LSTM, CNN-RNN, RoFBET, ABET, and Ensemble bagging approaches have reported the least solutions. In the meantime, DNN+Adam and ForestPA-PWDM systems have gained closer solutions. In addition, the PIODL-WPDCWM methodology reported greater performance with maximal $prec_n$, $reca_l$, $accu_y$, and F_{score} of 97.73%, 97.73%, 97.73%, and 97.73%, correspondingly.

Table 3.
Comparative analysis of PIODL-WPDCWM model with existing frameworks.

Framework	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}
RNN Algorithm	91.41	91.98	93.35	91.49
LSTM-LSTM	92.58	93.24	93.21	92.84
CNN-RNN	93.83	92.52	93.01	94.77
RoFBET Model	92.50	92.98	93.12	91.60
ABET Model	93.11	93.01	94.04	95.38
Ensemble bagging	92.01	95.72	96.84	96.46
DNN+Adam	95.81	94.36	95.34	91.36
ForestPA-PWDM	94.94	93.25	94.76	91.86
PIODL-WPDCWM	97.73	97.73	97.73	97.73

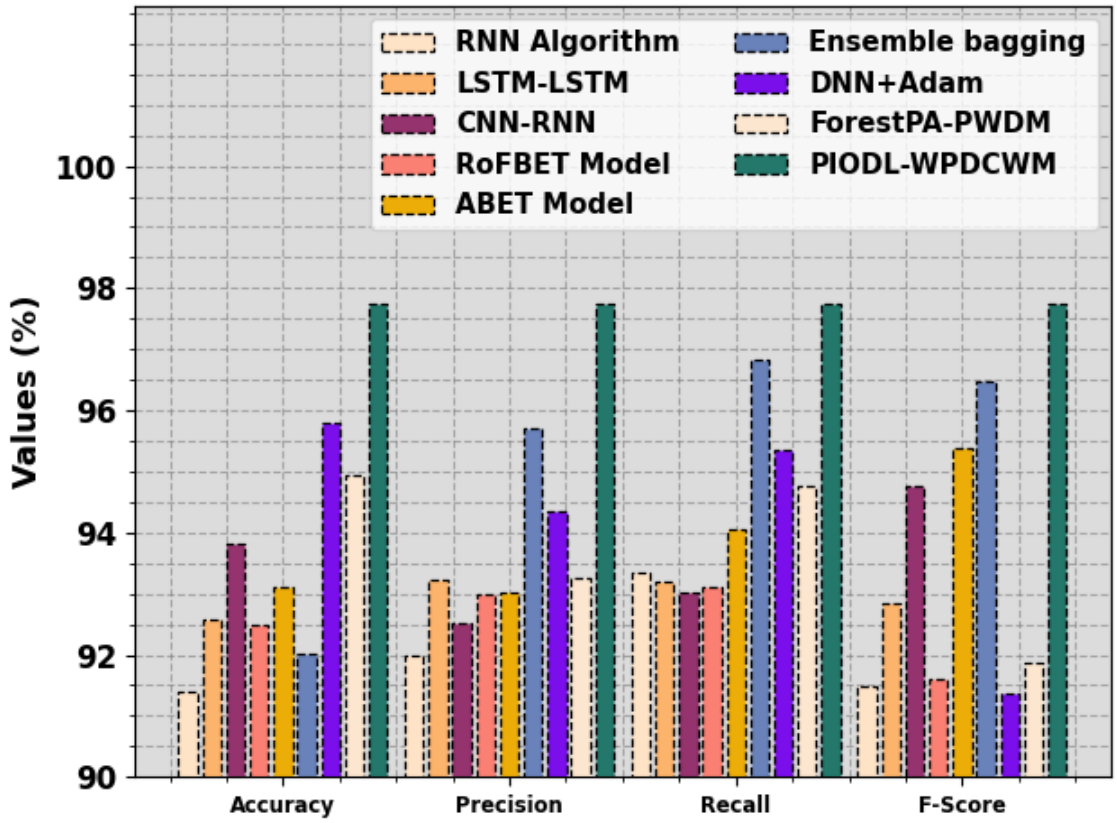

Figure 11.
Comparative analysis of PIODL-WPDCWM model with existing frameworks.

In Table 4 and Figure 12, the comparison outcomes of the PIODL-WPDCWM model are specified in terms of execution time (ET). The experimental outcomes showed that the PIODL-WPDCWM system gets the optimum solution. Based on ET, the PIODL-WPDCWM methodology gains a lower ET of 5.27 sec, whereas the RNN, LSTM-LSTM, CNN-RNN, RoFBET, ABET, Ensemble bagging, DNN+Adam, and ForestPA-PWDM systems obtain higher ET values of 8.86 sec, 9.05 sec, 11.77 sec, 11.02 sec, 7.92 sec, 12.79 sec, 10.46 sec, and 11.07 sec, correspondingly.

Table 4.
ET outcome of PIODL-WPDCWM technique with other approaches.

Framework	Execution Time(sec)
RNN Algorithm	8.86
LSTM-LSTM	9.25
CNN-RNN	11.77
RoFBET Model	11.02
ABET Model	7.92
Ensemble bagging	12.79
DNN+Adam	10.46
ForestPA-PWDM	11.07
PIODL-WPDCWM	5.27

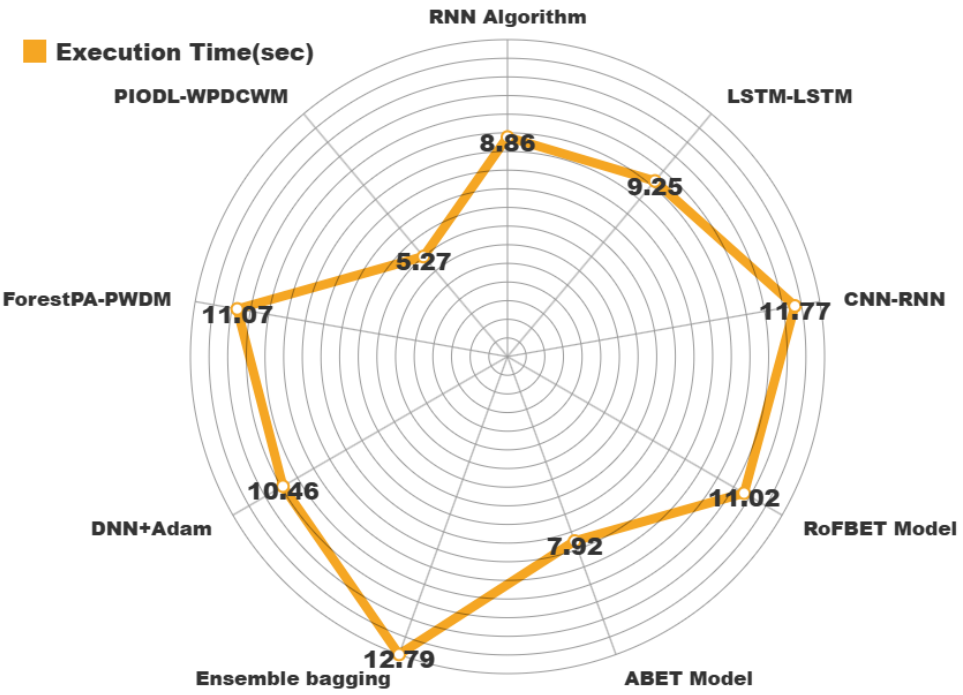

Figure 12.
ET outcome of PIODL-WPDCWM approach with other systems.

5. Conclusion

In this article, we have developed a new PIODL-WPDCWM algorithm. The objective of the PIODL-WPDCWM system lies in securing web mining activities and defending users from phishing attacks on websites. It encompasses four major stages: z-score normalization, BBOA-based feature subset selection, website phishing detection, and the PIO-based parameter tuning process. Primarily, the presented PIODL-WPDCWM technique involves z-score normalization to ensure that input features are standardized to a common scale. For the feature selection process, the BBOA is employed to classify the most relevant and informative features from the data. Additionally, the S-LSTM-AE approach is deployed for the detection and classification of website phishing. Lastly, the PIO algorithm can be utilized for the hyperparameter tuning model of the S-LSTM-AE method. To certify the enhanced performance of the PIODL-WPDCWM system, a wide range of simulation studies has been conducted, and the attained outcomes establish the enhancement of the PIODL-WPDCWM technique over other existing models.

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the article [23, 24].

References

[1] A. Subasi, E. Molah, F. Almkallawi, and T. J. Chaudhery, "Intelligent phishing website detection using random forest classifier," in *2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, 2017: IEEE, pp. 1-5.

[2] M. Kaytan and D. Hanbay, "Effective classification of phishing web pages based on new rules by using extreme learning machines," *Computer Science*, vol. 2, no. 1, pp. 15-36, 2017.

[3] R. M. Mohammad, F. Thabtah, and L. McCluskey, "Intelligent rule-based phishing websites classification," *IET Information Security*, vol. 8, no. 3, pp. 153-160, 2014. https://doi.org/10.1049/iet-ifs.2013.0109

[4] D. Aksu, Z. Turgut, S. Üstebay, and M. A. Aydin, "Phishing analysis of websites using classification techniques," in *International Telecommunications Conference: Proceedings of the ITelCon 2017, Istanbul*, 2019: Springer, pp. 251-258.

[5] D. R. Ibrahim and A. H. Hadi, "Phishing websites prediction using classification techniques," in *2017 International Conference on New Trends in Computing Sciences (ICTCS)*, 2017: IEEE, pp. 133-137.

[6] A. J. Park, R. N. Quadari, and H. H. Tsang, "Phishing website detection framework through web scraping and data mining," in *2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2017: IEEE, pp. 680-684.

[7] J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran, and B. Bindhumadhava, "Phishing website classification and detection using machine learning," in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, 2020: IEEE, pp. 1-6.

[8] M. Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7913-7921, 2010. https://doi.org/10.1016/j.eswa.2010.04.020

[9] G. Varshney, M. Misra, and P. K. Atrey, "A survey and classification of web phishing detection schemes," *Security and Communication Networks*, vol. 9, no. 18, pp. 6266-6284, 2016. https://doi.org/10.1002/sec.1513

[10] V. B. Bollikonda and K. Kiran, "Unveiling the Hidden: Exploring Challenges in Dark Web Investigation Using Measurement Sensors," *Journal of Cybersecurity & Information Management*, vol. 15, no. 1, pp. 166-166, 2025.

[11] O. K. Sahingoz, E. BUBE, and E. Kugu, "Dephides: Deep learning based phishing detection system," *IEEE Access*, vol. 12, pp. 8052-8070, 2024. https://doi.org/10.1109/ACCESS.2024.3352629

[12] P. P. Kumar, T. Jaya, and V. Rajendran, "SI-BBA–A novel phishing website detection based on Swarm intelligence with deep learning," *Materials Today: Proceedings*, vol. 80, pp. 3129-3139, 2023. https://doi.org/10.1016/j.matpr.2023.03.104

[13] F. S. Alsubaei, A. A. Almazroi, and N. Ayub, "Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics," *IEEE Access*, vol. 12, pp. 8373-8389, 2024. https://doi.org/10.1109/ACCESS.2024.XXXXXXX

[14] S. Pillai and A. Sharma, "Hybrid unsupervised web-attack detection and classification–A deep learning approach," *Computer Standards & Interfaces*, vol. 86, p. 103738, 2023. https://doi.org/10.1016/j.csi.2023.103738

[15] S. Asiri, Y. Xiao, S. Alzahrani, and T. Li, "PhishingRTDS: A real-time detection system for phishing attacks using a Deep Learning model," *Computers & Security*, vol. 141, p. 103843, 2024. https://doi.org/10.1016/j.cose.2024.103843

[16] M. A. Alohali *et al.*, "Metaheuristics with deep learning driven phishing detection for sustainable and secure environment," *Sustainable Energy Technologies and Assessments*, vol. 56, p. 103114, 2023. https://doi.org/10.1016/j.seta.2023.103114

[17] E. Zhu, K. Cheng, Z. Zhang, and H. Wang, "PDHF: Effective phishing detection model combining optimal artificial and automatic deep features," *Computers & Security*, vol. 136, p. 103561, 2024. https://doi.org/10.1016/j.cose.2024.103561

[18] E. A. Aldakheel, M. Zakariah, G. A. Gashgari, F. A. Almarshad, and A. I. Alzahrani, "A deep learning-based innovative technique for phishing detection in modern security with uniform resource locators," *Sensors*, vol. 23, no. 9, p. 4403, 2023. https://doi.org/10.3390/s23094403

[19] A. Al-Mekhlafi, S. Klawitter, and F. Klawonn, "Standardization with zlog values improves exploratory data analysis and machine learning for laboratory data," *Journal of Laboratory Medicine*, vol. 48, no. 5, pp. 215-222, 2024. https://doi.org/10.1515/jlm-2024-0060

[20] N. Eini, S. Janizadeh, S. M. Bateni, C. Jun, E. Heggy, and M. Kirs, "Predicting equilibrium scour depth around non-circular bridge piers with shallow foundations using hybrid explainable machine learning methods," *Results in Engineering*, vol. 24, p. 103492, 2024. https://doi.org/10.1016/j.rineng.2024.103492

[21] W. Du, J. Zhang, G. Meng, and H. Zhang, "Aero-engine fault detection with an LSTM auto-encoder combined with a self-attention mechanism," *Machines*, vol. 12, no. 12, p. 879, 2024. https://doi.org/10.3390/machines12120879

[22] Y. Zhao, C. Zhao, and L. Zhao, "A scalable multi-FPGA platform for hybrid intelligent optimization algorithms," *Electronics*, vol. 13, no. 17, p. 3504, 2024. https://doi.org/10.3390/electronics13173504

[23] V. Nadar, "PhiUSIIL phishing URL Dataset. Kaggle," Retrieved: https://www.kaggle.com/datasets/ndarvind/phiusiil-phishing-url-dataset, 2023.

[24] A. Prasad and S. Chandra, "PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning," *Computers & Security*, vol. 136, p. 103545, 2024. https://doi.org/10.1016/j.cose.2023.103545

[25] P. Yang, G. Zhao, and P. Zeng, "Phishing website detection based on multidimensional features driven by deep learning," *IEEE Access*, vol. 7, pp. 15196-15209, 2019. https://doi.org/10.1109/ACCESS.2019.2893560

[26] Y. A. Alsariera, V. E. Adeyemo, A. O. Balogun, and A. K. Alazzawi, "Ai meta-learners and extra-trees algorithm for the detection of phishing websites," *IEEE Access*, vol. 8, pp. 142532-142542, 2020. https://doi.org/10.1109/ACCESS.2020.3019183

[27] S. Alnemari and M. Alshammari, "Detecting phishing domains using machine learning," *Applied Sciences*, vol. 13, no. 8, p. 4649, 2023. https://doi.org/10.3390/app13084649