Computers, Materials & Continua
DOI: 10.32604/cmc.2023.034752
Article

Tech Science Press

Optimal Hybrid Deep Learning Enabled Attack Detection and Classification in IoT Environment

Fahad F. Alruwaili*

College of Computing and Information Technology, Shaqra University, Sharqa, Saudi Arabia
*Corresponding Author: Fahad F. Alruwaili. Email: alruwaili@su.edu.sa
Received: 26 July 2022; Accepted: 20 October 2022

Abstract: The Internet of Things (IoT) paradigm enables end users to access networking services amongst diverse kinds of electronic devices. IoT security mechanism is a technology that concentrates on safeguarding the devices and networks connected in the IoT environment. In recent years, False Data Injection Attacks (FDIAs) have gained considerable interest in the IoT environment. Cybercriminals compromise the devices connected to the network and inject the data. Such attacks on the IoT environment can result in a considerable loss and interrupt normal activities among the IoT network devices. The FDI attacks have been effectively overcome so far by conventional threat detection techniques. The current research article develops a Hybrid Deep Learning to Combat Sophisticated False Data Injection Attacks detection (HDL-FDIAD) for the IoT environment. The presented HDL-FDIAD model majorly recognizes the presence of FDI attacks in the IoT environment. The HDL-FDIAD model exploits the Equilibrium Optimizer-based Feature Selection (EO-FS) technique to select the optimal subset of the features. Moreover, the Long Short Term Memory with Recurrent Neural Network (LSTM-RNN) model is also utilized for the purpose of classification. At last, the Bayesian Optimization (BO) algorithm is employed as a hyperparameter optimizer in this study. To validate the enhanced performance of the HDL-FDIAD model, a wide range of simulations was conducted, and the results were investigated in detail. A comparative study was conducted between the proposed model and the existing models. The outcomes revealed that the proposed HDL-FDIAD model is superior to other models.

Keywords: False data injection attacks; hyperparameter optimizer; deep learning; feature selection; IoT; security

1 Introduction

The rapid progression of the Internet of Things (IoT) phenomenon in industrial sectors has increased the susceptibility of crucial network structures to severe cyber-attacks. The Industrial IoT (IIoT) environment helps resolve several intractable problems in the industry by providing real-time response systems and permitting the self-controlling systems to function separately [1]. To ensure an

effective roll-out of the IIoT applications, it is important to investigate the security problems in detail and overcome them. To be specific, it is highly complex to detect stealthy assaults like False Data Injection Attacks (FDIA) on Predictive Maintenance (PdM) mechanisms because of the nature of the attack [2]. In False Data Injection Attack (FDIA), an attacker stealthily negotiates the dimensions from the IoT sensors. Likewise, the manipulated sensor dimensions evade the sensor's fundamental 'faulty data' recognition system and proliferate to hide the resultant values of the sensors. 'False Data Injection' (FDI) attack is a type of major assault that can affect these mechanisms. In general, the FDI attacks cause severe issues in industrial structures. It corrupts the sensor dimensions to deceive the assaulted industrial platform [3].

The FDI assaults are applied by intruding on the data processing programs, sensors, and sensor communication structures. These attacks tend not to showcase their effects on the PdM mechanism [4]. However, the attack proliferates from the sensor to the Machine Learning (ML) part of the PdM mechanism and misleads the network by predicting the deferred asset failure or maintaining the interval. This mechanism tends to incur heavy losses in terms of human lives, and at times, it also results in the unintended failure of dangerous applications [5]. With the help of the latest DL approaches, the engine can forecast its future demands, execute adjustments, and save about 15% of fuel usage. But, the susceptibility of the sensor attacks towards these ML-related and IoT engines is considered a crucial challenge [6]. The existing sensor attack recognition solutions in the Cyber-Physical Systems (CPS) and IoT fields are inadequate to address this problem. This is attributed to the fact that whenever such traditional systems are positioned separately among the millions of sensors, it suffers from scalability issues and source overhead since several IoT networks face constraint in terms of energy [7].

The FDIAs tend to harm the external elements, resulting in enormous economic loss and life-threatening cases [8]. Thus, it becomes necessary to detect and prevent the FDIAs in any serious structure [9]. Several prevailing solutions are either theoretical or only implement the methods in cyberspace, like Intrusion Prevention Systems (IPS) that are generally utilized for protecting traditional computer networks. The existing methods lack specific security properties and cannot handle critical infrastructure, high rate of events, the requirement for real-time detection and interaction requirement, a pro-active defense, and a complicated cyber and physical interface [10]. In this background, the current research work attempts to overcome the issues with the help of ML approaches to detect injection attacks.

1.1 Existing FDI Detection Approaches

Aboelwafa et al. [11] proposed a new approach for FDI attack recognition with the help of Auto Encoders (AEs). It exploited the sensor information about time and space, and the proposed method excelled in classifying the falsified data. In addition, the falsified data was also cleaned with the help of the Denoising Autoencoders (DAEs). The performance was estimated to demonstrate the proposed approach's achievement in identifying the FDI attacks. It also considerably demonstrated a Support Vector Machine (SVM)-based method to achieve a similar goal. Alromih et al. [12] examined a Randomized Watermarking Filtering Scheme (RWFS) for IoT applications, offering an en-route filter to remove the injected data at an initial communication phase. The injected data were filtered based on a watermark applied in the original information and embedded directly from arbitrary places throughout the packet payload. This mechanism utilized the Homomorphic Encryption approaches to conceal the reported measurement in several adversaries.

In literature [13], a Hybrid GSW (Gentry, Sahai, and Waters) and DM (Ducas and Micciancio)-related Fully Homomorphic Encryption (HGSW–DM–FHE) approach was presented to control

the FDIA in privacy-preserving data aggregation in the fog computing environment. The presented HGSW–DM–FHE method was found to be extremely fault-tolerant, and the data aggregation procedure in another device did not impact even in the case of the failure of fog devices. Moudoud et al. [14] introduced a hierarchical structure to secure the 5G-enabled IoT networks and a security method to forecast and recognize FDIA and DDoS attacks. The presented security approach was developed based on the Markov stochastic procedure. The method tracked every network device's performance and utilised a range-based behaviour-sifting policy. Wang et al. [15] examined a DL-related Locational Detection (DLLD) structure to find the particular places of FDIA on a real-time basis. The DLLD structure was developed by combining the Convolutional Neural Network (CNN) with a typical Bad Data Detector (BDD). The BDD was utilized to remove the minimum quality data. The modified CNN was utilized for multi-label classification to capture the inconsistency and co-occurrence dependencies from the power flow measurement because of potential attacks.

1.2 Paper Contribution

The current research article develops a Hybrid Deep Learning to Combat Sophisticated False Data Injection Attacks detection (HDL-FDIAD) in the IoT environment. The presented HDL-FDIAD model exploits the Equilibrium Optimizer-based Feature Selection (EO-FS) technique to select the optimal subset of features. Moreover, the Long Short-Term memory with Recurrent Neural Network (LSTM-RNN) model is utilized for classification. At last, the Bayesian Optimization (BO) algorithm is employed as a hyperparameter optimizer in this study. To validate the enhanced performance of the proposed HDL-FDIAD model, a wide range of simulations was conducted, and the results were investigated under different measures.

2 Materials and Methods

The current research article proposes a novel HDL-FDIAD model to determine the FDI attacks in the IoT environment. The HDL-FDIAD model exploits the EO-FS technique to select the optimal subset of features. Moreover, the BO with LSTM-RNN model is also utilized for classification. Fig. 1 illustrates the block diagram of the proposed HDL-FDIAD approach.

2.1 Feature Subset Selection Process

The HDL-FDIAD model exploits the EO-FS technique to select the optimal subset of features. EO is a dynamic mass balance approach that functions to control the volume of the data [16]. An arithmetical expression is applied in this stage to characterize the mass balance and describe the focus of the non-reactive components in a dynamic controlled environment. Further, this expression functions with different strategies with source and sink variations. The whole theoretical description of the EO phase is described herewith. An arbitrary population is initialized through uniform distribution of the numbers based on the particle amount and dimension in the searching area, as given below.

$$C_i^{initial} = C_{\min} + rand_i\,(C_{\max} - C_{\min}) \quad i = 1, 2, \ldots, n \tag{1}$$

In Eq. (1), $C_i^{initial}$ denotes the vector of the initial concentration of the $i-th$ particle, C_{min} and C_{max} indicate the lower limit and upper limit, respectively, $rand_i$ denotes a uniformly-distributed value that lies in the range of 0 and 1, and n describes the population size. In order to define the equilibrium state (i.e., global optimal), a pool of four optimal candidates is chosen to identify the encompassing

alternative particles by corresponding to the arithmetical mean of the four particles. A particle is gathered by processing a pooling vector as given below.

$$\vec{C}_{eq.pool} = \left\{\vec{C}_{eq(1)}, \vec{C}_{eq(2)}, \vec{C}_{eq(3)}, \vec{C}_{eq(4)}, \vec{C}_{eq(ave)}\right\} \tag{2}$$

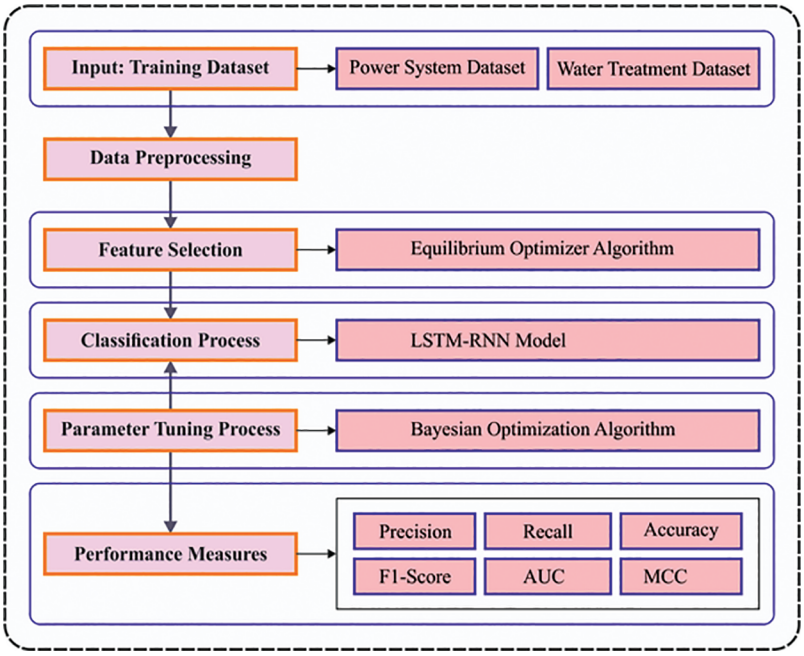

Figure 1: Block diagram of the HDL-FDIAD approach

During the evolution process, the initialized particle upgrades the concentration from the primary generation based on $\vec{C}_{eq(1)}$. In the following generation, the upgraded value is denoted by $\vec{C}_{eq(ave)}$. Subsequently, each particle with a solution candidate is upgraded as per the conclusion of the evolution process. The exponential term F demonstrates that the EO technique accomplishes a suitable balance between intensification and diversification. λ is an arbitrary value that lies in the range of 0 and 1 to control the turn-over rate to a realistic control volume.

$$\vec{F} = e^{-\vec{\lambda}(t-t_0)} \tag{3}$$

In Eq. (3), t denotes the iterative count (*Iter*).

$$t = \left(1 - \frac{Iter}{\text{Max\_}iter}\right)\left(a_2 \frac{Iter}{\text{Max\_}iter}\right) \tag{4}$$

In Eq. (4), $Iter = current\ iteration$, $\text{Max\_}iter =$ maximum iteration and the variable a_2 are applied to manage the exploitation ability of *EO*. To ensure convergence and improve the global value along with the local searching ability of the approach, the following equation is applied.

$$\vec{t}_0 = \frac{1}{\vec{\lambda}} ln\left(-a_1 sign\left(\vec{r} - 0.5\right)\left[1 - e^{-\vec{\lambda} t}\right]\right) + t \tag{5}$$

In this expression, a_1 and a_2 are applied to control the global and local searching abilities of the EO method. The $sign\left(\vec{r}-0.5\right)$ corresponds to the value nearby the exploration and exploitation paths. In *EO*, the values of a_1 and a_2 are chosen as two and one. The term is modified using the following expression by substituting Eq. (5) in Eq. (3).

$$\vec{F}=a_1 sign\left(\vec{r}-0.5\right)\left[e^{-\vec{\lambda}t}-1\right] \tag{6}$$

The generation rate in the EO approach is applied as a time function to improve the exploitation phase. The first-order exponential decay procedure from the multi-purpose generative method is defined herewith.

$$\vec{G}=\vec{G}_0 e^{-\vec{k}(t-t_0)} \tag{7}$$

In Eq. (7), $G_0=$ primary value and $k=$ decay variable. At last, the generation rate is considered as $k=\lambda$.

$$\vec{G}=\vec{G}_0 e^{-\vec{\lambda}(t-t_0)}=\vec{G}_0\vec{F}_0 \tag{8}$$

Now, G_0 is evaluated by using Eq. (9):

$$\vec{G}_0=G\vec{C}P\left(\vec{C}_{eq}-\vec{\lambda}\vec{C}\right) \tag{9}$$

$$G\vec{C}P=\begin{cases}0.5r_1, & r_2\geq 0\\ 0, & r_2<0\end{cases} \tag{10}$$

Here r_1, r_2 correspond to two arbitrary integers that lie in the range of 0 and 1. *GCP* indicates the control generation rate. Using the above-mentioned equation, the last-upgraded concentration (particle) equation is given below.

$$\vec{C}=\vec{C}_{eq}+\left(\vec{C}-\vec{C}_{eq}\right)\vec{F}+\frac{\vec{G}}{\vec{\lambda}V}\left(1-\vec{F}\right) \tag{11}$$

The upgraded equation has an equilibrium concentration, a global search and a local search to accomplish the exact solutions. The fitness function of the EO-FS method assumes the classification accuracy and the number of selected features. It increases the classification accuracy and reduces the set size of the selected features. Thus, the subsequent fitness function is utilized to evaluate the individual solutions, as displayed in Eq. (12).

$$Fitness=\alpha * ErrorRate+(1-\alpha)*\frac{\#SF}{\#All\_F} \tag{12}$$

Here, ErrorRate refers to the classification error rate calculated with the selected features' help.

2.2 FDI Detection and Classification Process

In this stage, the LSTM-RNN model is utilized for the purpose of classification. Generally, a Feedforward Neural Network (FFNN) can be defined below [17].

$$Y=F(X,\theta) \tag{13}$$

$X=\{x_1, x_2,\ldots,x_n\}$ refers to an input set

$Y = \{y_1,\ y_2, \ldots, y_m\}$ stands for an output set

F stands for an FFNN module.

θ refers to a parameter set of the module.

In the classification module, Y represents a set of classes. CNN is a kind of FFNN and is employed to perform semantic segmentation, image classification and the target recognition process. Unlike other NNs, the CNN mechanism contains convolution and pooling layers. The convolution layer aims to extract the local features of the input dataset. Fig. 2 demonstrates the framework of the LSTM method.

$$Y_F = Conv\,(X,\ \theta_{CONV}) \tag{14}$$

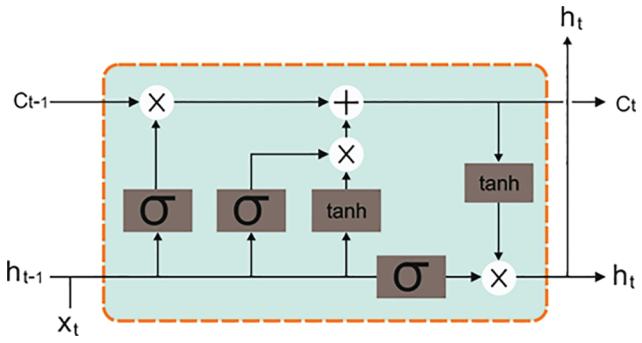

Figure 2: Structure of the LSTM approach

Conv. is a convolution layer, whereas Y_F is a feature subset extracted by the convolution layer from X. θ_{C0NV} is a parameter set in the convolution layer. The aim of the pooling layer is to compress the local feature, thus highlighting the feature.

$$Y_{CF} = Poo1\ (Y_{Fr}\theta_{Pool}) \tag{15}$$

Pool is a pooling layer, Whereas Y_{CF} represents a set of compressed features. Here, the CNN and the pooling layer from Y_F are combined and presented. θ_{Pool} is a parameter set in the pooling layer. In a classification module, CNN contains an FC layer and a *Softmax* layer and both are incorporated along with the front-end of the RNN layer to form a CRNN mechanism. $Y = F\,(X, \theta)$ categorizes the features. Here, $Y = F\,(X,\ \theta)$ of CNN is $F\,(\mathrm{X}, \theta)$ of CNN, as given below.

$$Y = Soft\max\left(FC\left(Pool\left(Conv\,(X, \theta_{CONV}), \theta_{pool}\right), \theta_{FC}\right)\right) \tag{16}$$

The fully-connected layer is an FC layer, whereas Softmax denotes one *Softmax* layer. RNN is an alternative version of the FFNN model and is mainly employed for datasets with a sequential architecture, such as speech recognition, machine translation and so on. LSTM-RNN is a widely-applied RNN method that can resolve the gradient vanishing problems with memory cells to store long-term data. As a classification method, the LSTM-RNN approach contains the *Softmax* layer and the FC layer.

$y = F\,(\mathrm{X}, \theta)$ of LSTM-RNN is given by:

$$y = Soft\max\,(FC\,(LSTM\,(\mathrm{x}, \theta_{LSTM}), \theta_{FC})) \tag{17}$$

LSTM is one LSTM layer.

2.3 Hyperparameter Tuning Process

At last, the BO algorithm is employed as a hyperparameter optimizer in this study. It is a sequential method used for the optimization of the black-box function (x) parameters. The presented method shows effectiveness for the method configured initially, and now it becomes a common solution [18]. The BO approach integrates the previous output to estimate a response surface function $\hat{f}(x)$ and applies $\hat{f}(x)$ to select the following configuration, x_n. Further, it also estimates $f(X_n)$ through a true black-box function that estimates the subsequent output through the estimated performance, $f(x_n)$. The procedure is repeated sequentially until the ending condition is satisfied. In order to build the response surface of $\hat{f}(x)$, the three most common selections are used for ML problems, such as the RF regressor, Tree-structured Parzen Estimator and the Gaussian process. The algorithm utilizes the acquisition function that offers a trade-off between the exploitation and exploration phases. In this study, the black-box function (x) characterizes the performance i.e., predictive error or accuracy of the DNN using a configuration model x which is extremely non-convex. Further, $f(x)$ is estimated in an arbitrary point x. However, the individual estimation takes a significant number of times since the evaluation of $f(x)$ involves the training process of the DNN procedure.

Algorithm l: Bayesian Optimization

Inputs: BO algorithm $\hat{f}$, parameter space $\mathcal{X}$, black-box function f

$\mathcal{H} \leftarrow \emptyset$

For $n = 1, 2, \ldots$ do

Choose $x_n \in \arg\max_{x \in \mathcal{X}} \hat{f}(x;\ \mathcal{H})$

Assess $y_n \leftarrow f(x_n)$

Upgrade $\mathcal{H} \leftarrow \mathcal{H} \cup (x_n,\ y_n)$

Check for the present conditions

End for

3 Results and Discussion

In this section, the FDI attack detection performance of the proposed HDL-FDIAD model was validated using two datasets, namely, power system dataset and water treatment dataset. The first power system dataset holds 22,714 samples under normal class and 9,582 samples under FDIA class. Similarly, the water treatment dataset includes 395,298 samples under normal and 54,621 samples under FDIA class. The HDL-FDIAD model selected a set of 128 features and 84 features from the databases under study. Table 1 illustrates the details of both datasets.

Table 1: Dataset details

Class	Number of samples (Features = 128)	
	Power system	Water treatment
Normal	22714	395298
False Data Injection Attacks (FDIAs)	9582	54621
Total	32296	449919

Table 2 and Fig. 3 show the results offered by the HDL-FDIAD model and other existing models on power dataset [19]. The experimental outcomes confirm that the proposed HDL-FDIAD model gained effectual outcomes on both the class labels. With respect to $prec_n$, the HDL-FDIAD model identified the normal class samples with a maximum $prec_n$ of 91.92%, whereas the Naïve Bayes (NB), SVM, AdaBoost, k-Nearest Neighbor (KNN), Random Forest (RF), Logistic Regression (LR) and the Decision Tree (DT) models obtained the least $prec_n$ values such as 82.44%, 84.08%, 82.31%, 85.03%, 89.35%, 83.93% and 83.08% respectively. Also, in relation to $prec_n$, the proposed HDL-FDIAD model identified the FDIA class samples with a maximum $prec_n$ of 96.71%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and DT models gained the least $prec_n$ values such as 80.06%, 81.87%, 84.94%, 81.80%, 94.12%, 83.61% and 85.88% correspondingly. Moreover, with regard to $reca_l$, the HDL-FDIAD method identified the normal class samples with a maximum $reca_l$ of 93.36%. However, the NB, SVM, AdaBoost, KNN, RF, LR and the DT models reached the least $reca_l$ values such as 82.90%, 85.49%, 88.63%, 84.56%, 90.41%, 83.93% and 82% correspondingly. Furthermore, with respect to $reca_l$, the HDL-FDIAD approach identified the FDIA class samples with a maximum $reca_l$ of 97.23%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and DT models achieved the least $reca_l$ values such as 87.59%, 83.26%, 85.44%, 81.38%, 93.49%, 85.74% and 83.94% correspondingly.

Table 2: Precision and recall analyses results of the HDL-FDIAD approach and other existing methodologies on the power dataset

Methods	Precision			Recall		
	Normal	FDIA	Average	Normal	FDIA	Average
HDL-FDIAD	91.92	96.71	94.32	93.36	97.23	95.30
Naïve Bayes	82.44	80.06	81.25	82.90	87.59	85.25
Support vector machine	84.08	81.87	82.98	85.49	83.26	84.38
AdaBoost	82.31	84.94	83.63	88.63	85.44	87.04
K-Nearest neighbor	85.03	81.80	83.42	84.56	81.38	82.97
Random forest	89.35	94.12	91.74	90.41	93.49	91.95
Logistic regression	83.93	83.61	83.77	83.93	85.74	84.84
Decision tree	83.08	85.88	84.48	82.00	83.94	82.97

Table 3 portrays the results of the proposed HDL-FDIAD model and other existing models on the power dataset. The experimental outcomes infer that the proposed HDL-FDIAD method attained the effectual outcomes on both the class labels. In terms of $accu_y$, the proposed HDL-FDIAD model identified the normal class samples with a maximum $accu_y$ of 95.94%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and DT models reached the least $accu_y$ values such as 88.17%, 88.66%, 84.08%, 86.99%, 92.62%, 81.55% and 84.39% correspondingly. In addition, with regards to $accu_y$, the HDL-FDIAD method identified the FDIA class samples with a maximum $accu_y$ of 93.72%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model gained the least $accu_y$ values, such as 82.41%, 81.04%, 81.92%, 83.27%, 91.41%, 87.46% and 85.73% correspondingly. In addition to these, with respect to $F1_{score}$, the HDL-FDIAD method identified the normal class samples with a maximum $F1_{score}$ of 91.08%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model attained the least $F1_{score}$ values, such as 82.25%, 86.14%, 88.28%, 81.1%, 83.11%, 83.84% and 84.26% correspondingly. Furthermore, with respect to $F1_{score}$, the HDL-FDIAD method classified the FDIA class samples with

a maximum $F1_{score}$ of 89.67%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model accomplished the least $F1_{score}$ values such as 80.71%, 80.99%, 82.49%, 84.25%, 82.42%, 86.26% and 83.13% correspondingly.

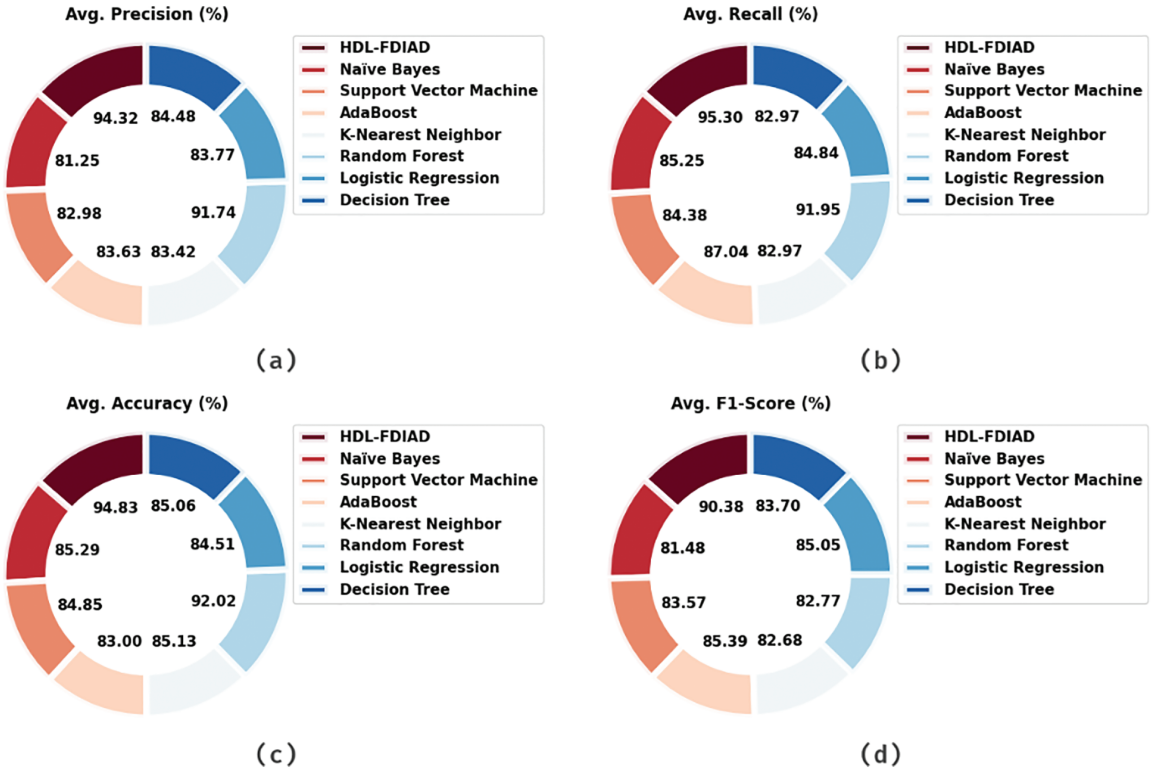

Figure 3: Average analysis results of the HDL-FDIAD approach on power dataset (a) $\boldsymbol{Prec_n}$, (b) $\boldsymbol{Reca_l}$, (c) $\boldsymbol{Accu_y}$, and (d) $\boldsymbol{F}1_{score}$

Table 3: Accuracy and F1-score analyses results of the HDL-FDIAD approach and other existing methodologies on power dataset

Methods	Accuracy			F1-Score		
	Normal	FDIA	Average	Normal	FDIA	Average
HDL-FDIAD	95.94	93.72	94.83	91.08	89.67	90.38
Naïve Bayes	88.17	82.41	85.29	82.25	80.71	81.48
Support vector machine	88.66	81.04	84.85	86.14	80.99	83.57
AdaBoost	84.08	81.92	83.00	88.28	82.49	85.39
K-Nearest neighbor	86.99	83.27	85.13	81.1	84.25	82.68
Random forest	92.62	91.41	92.02	83.11	82.42	82.77
Logistic regression	81.55	87.46	84.51	83.84	86.26	85.05
Decision tree	84.39	85.73	85.06	84.26	83.13	83.70

Both Training Accuracy (TA) and Validation Accuracy (VA) values, acquired by the proposed HDL-FDIAD method on Power Dataset, are shown in Fig. 4. The experimental outcomes infer that the HDL-FDIAD method achieved the maximal TA and VA values, whereas the VA values were higher than the TA values.

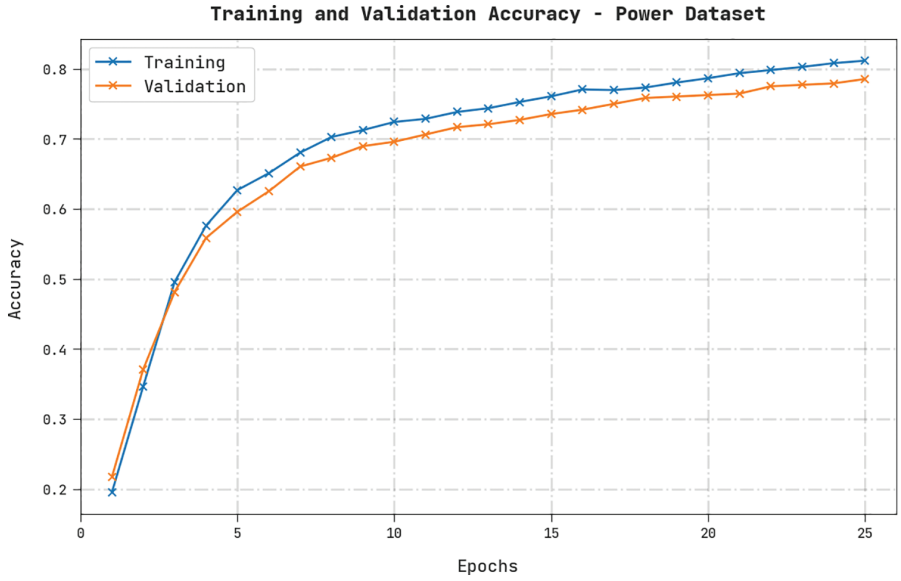

Figure 4: TA and VA analyses results of the HDL-FDIAD approach on the power dataset

Both Training Loss (TL) and Validation Loss (VL) values, achieved by the HDL-FDIAD approach on Power Dataset, are exhibited in Fig. 5. The experimental outcomes denote that the HDL-FDIAD algorithm established the least TL and VL values while the VL values were lesser than the TL values.

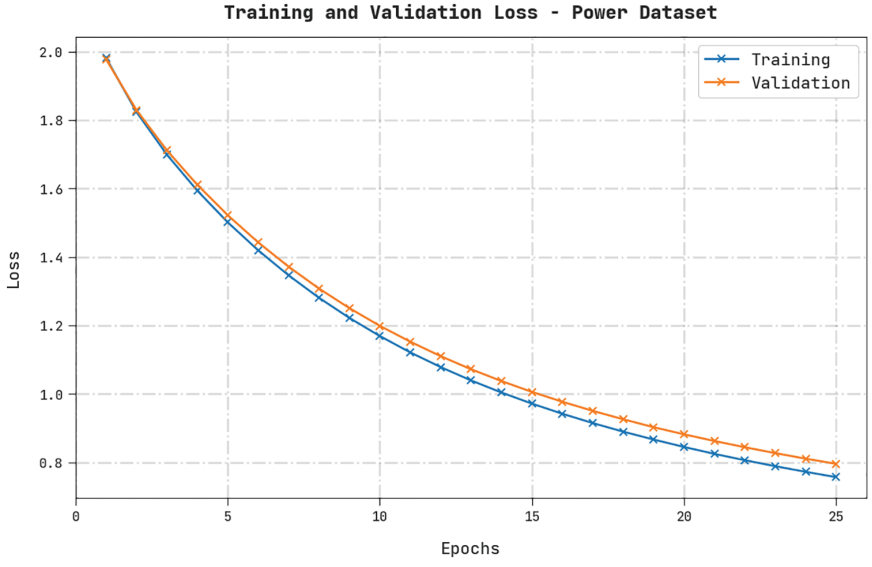

Figure 5: TL and VL analyses results of the HDL-FDIAD approach on power dataset

A clear precision-recall analysis was conducted on the HDL-FDIAD method using the Power Dataset and the results are displayed in Fig. 6. The figure denotes that the HDL-FDIAD method produced enhanced precision-recall values under all the classes.

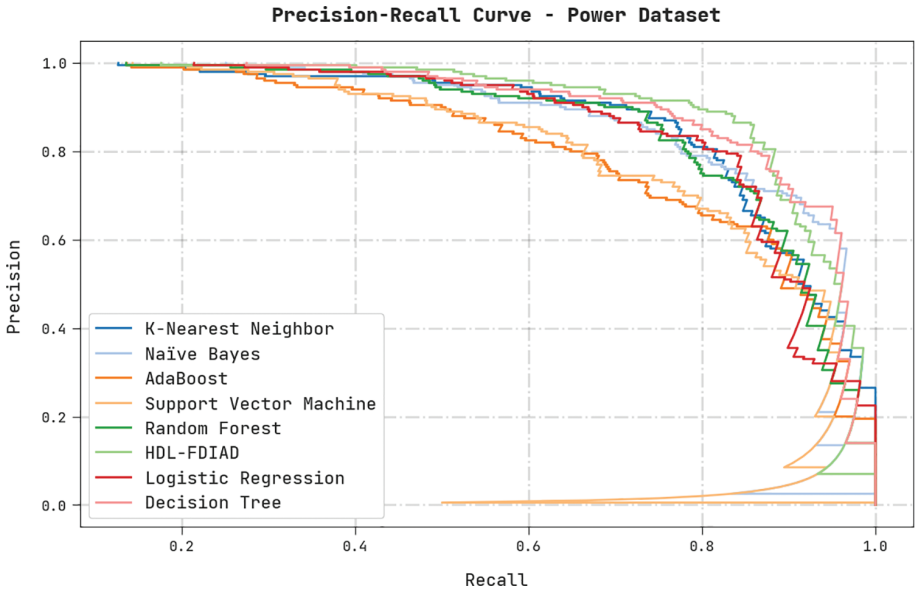

Figure 6: Precision-recall analysis results of the HDL-FDIAD approach on power dataset

A brief ROC analysis was conducted on the HDL-FDIAD methodology using the Power Dataset, and the results are shown in Fig. 7. The results signify that the HDL-FDIAD approach established its ability in categorizing the Power dataset under distinct classes.

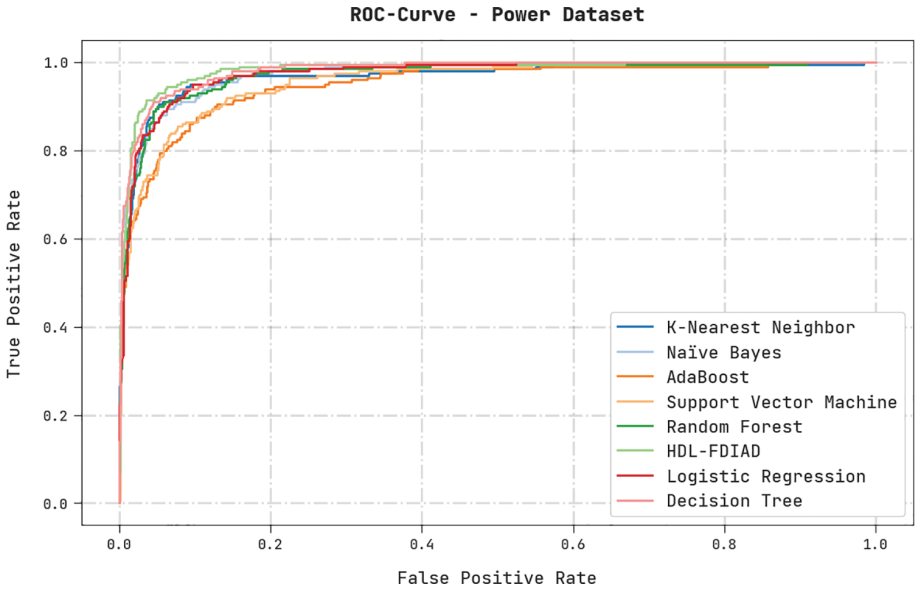

Figure 7: ROC analysis results of the HDL-FDIAD approach on power dataset

Table 4 and Fig. 8 show the results rendered by the proposed HDL-FDIAD model and other existing models on the Water Treatment dataset. The experimental outcomes confirm that the proposed HDL-FDIAD model reached the effectual outcomes on both the class labels. With respect to $prec_n$, the HDL-FDIAD model identified the normal class samples with a maximum $prec_n$ of 99.12%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model attained the least $prec_n$ values such as 98.89%, 96.39%, 96.87%, 95.56%, 94.54%, 95.11% and 98.20% correspondingly. Additionally, in terms of $prec_n$, the HDL-FDIAD model identified the FDIA class samples with a maximum $prec_n$ of 98.56%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model gained the least $prec_n$ values, such as 96.36%, 94.28%, 98.69%, 96.62%, 97.06%, 95.96% and 97.18% correspondingly. Furthermore, with respect to $reca_l$, the proposed HDL-FDIAD model identified the normal class samples with a maximum $reca_l$ of 98.51%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model obtained the least $reca_l$ values, such as 95.99%, 97.45%, 94.06%, 95.32%, 96.59%, 94.49% and 98.36% respectively. Additionally, with respect to $reca_l$, the proposed HDL-FDIAD model identified the FDIA class samples with a maximum $reca_l$ of 99.09%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model obtained the least $reca_l$ values, such as 95.26%, 97.92%, 96.61%, 94.77%, 97.02%, 96.63% and 95.29% respectively.

Table 4: Precision and recall analysis results of the HDL-FDIAD approach and other existing methodologies on water treatment dataset

Methods	Precision			Recall		
	Normal	FDIA	Average	Normal	FDIA	Average
HDL-FDIAD	99.12	98.56	98.84	98.51	99.09	98.80
Naïve Bayes	98.89	96.36	97.63	95.99	95.26	95.63
Support vector machine	96.39	94.28	95.34	97.45	97.92	97.69
AdaBoost	96.87	98.69	97.78	94.06	96.61	95.34
K-Nearest neighbor	95.56	96.62	96.09	95.32	94.77	95.05
Random forest	94.54	97.06	95.80	96.59	97.02	96.81
Logistic regression	95.11	95.96	95.54	94.49	96.63	95.56
Decision tree	98.20	97.18	97.69	98.36	95.29	96.83

Table 5 displays the results attained by the HDL-FDIAD method and other existing models on Water Treatment dataset. The experimental outcomes confirm that the HDL-FDIAD model obtained the effectual outcomes on both the class labels. With respect to $accu_y$, the HDL-FDIAD technique identified the normal class samples with a maximum $accu_y$ of 98.82%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT algorithm attained the least $accu_y$ values, such as 96.24%, 95.35%, 95.10%, 94.24%, 95.21%, 94.99% and 95.88% correspondingly. Moreover, with respect to $accu_y$, the proposed HDL-FDIAD approach identified the FDIA class samples with a maximum $accu_y$ of 98.56%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model obtained the least $accu_y$ values, such as 94.15%, 97.82%, 97.96%, 98.44%, 97.95%, 97% and 94.62% correspondingly. Additionally, with respect to $F1_{score}$, the HDL-FDIAD model classified the normal class samples with a maximum $F1_{score}$ of 98.34%, whereas the NB, SVM, AdaBoost, KNN, RF, LR and the DT model accomplished the least $F1_{score}$ values, such as 94.32%, 96.34%, 97.94%, 97.16%, 95.59%, 95.59% and 99% correspondingly. Further, with respect to $F1_{score}$, the HDL-FDIAD technique identified the FDIA

class samples with a maximum $F1_{score}$ of 98.15%. In contrast, the NB, SVM, AdaBoost, KNN, RF, LR and the DT model attained the least $F1_{score}$ values such as 96.90%, 94.54%, 94.69%, 96.74%, 98.36%, 94.29% and 94.07% correspondingly.

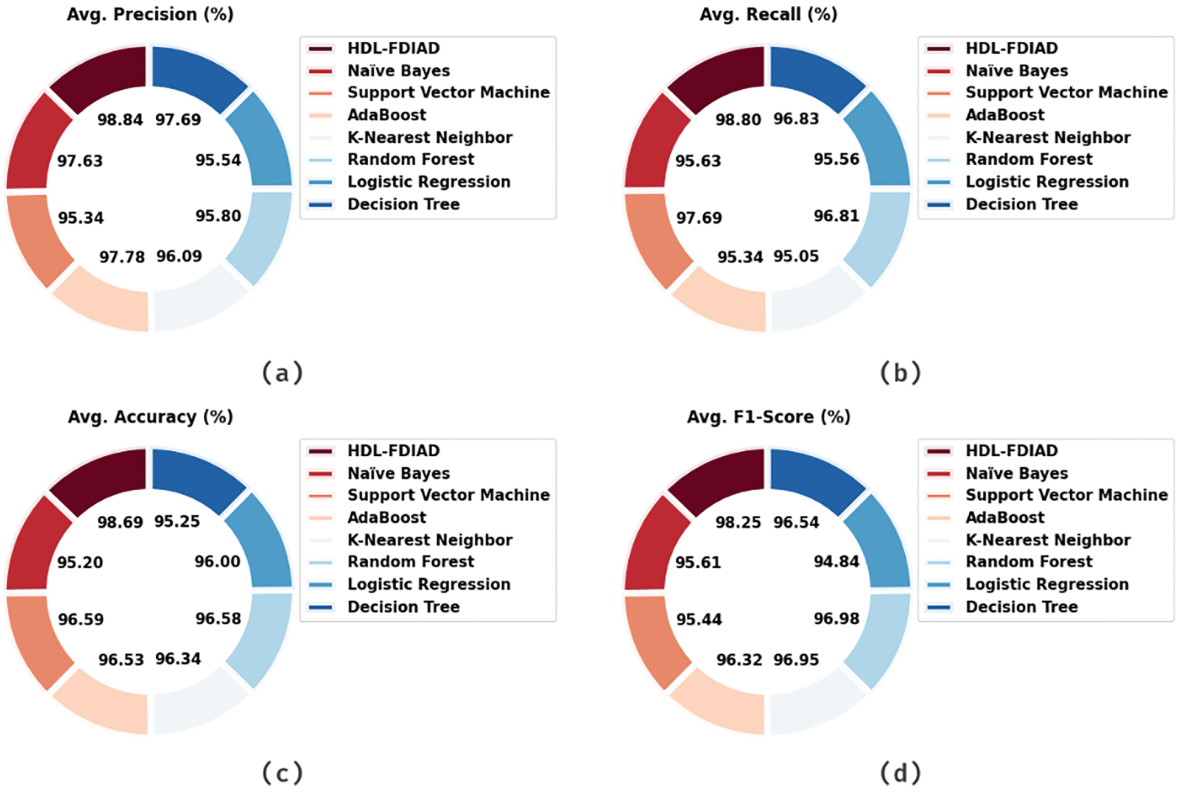

Figure 8: Average analysis results of the HDL-FDIAD approach on water treatment dataset (a) $Prec_n$, (b) $Reca_l$, (c) $Accu_y$, and (d) $F1_{score}$

Table 5: Accuracy and F1-score analyses results of the HDL-FDIAD approach and other existing methodologies on water treatment dataset

Methods	Accuracy			F1-Score		
	Normal	FDIA	Average	Normal	FDIA	Average
HDL-FDIAD	98.82	98.56	98.69	98.34	98.15	98.25
Naïve Bayes	96.24	94.15	95.20	94.32	96.90	95.61
Support vector machine	95.35	97.82	96.59	96.34	94.54	95.44
AdaBoost	95.10	97.96	96.53	97.94	94.69	96.32
K-Nearest neighbor	94.24	98.44	96.34	97.16	96.74	96.95
Random forest	95.21	97.95	96.58	95.59	98.36	96.98
Logistic regression	94.99	97.00	96.00	95.39	94.29	94.84
Decision tree	95.88	94.62	95.25	99.00	94.07	96.54

Both TA and VA values, obtained by the HDL-FDIAD method on Water Treatment Dataset, are demonstrated in Fig. 9. The experimental outcomes denote that the HDL-FDIAD technique achieved the maximal TA and VA values. In contrast, the VA values were higher than the TA values.

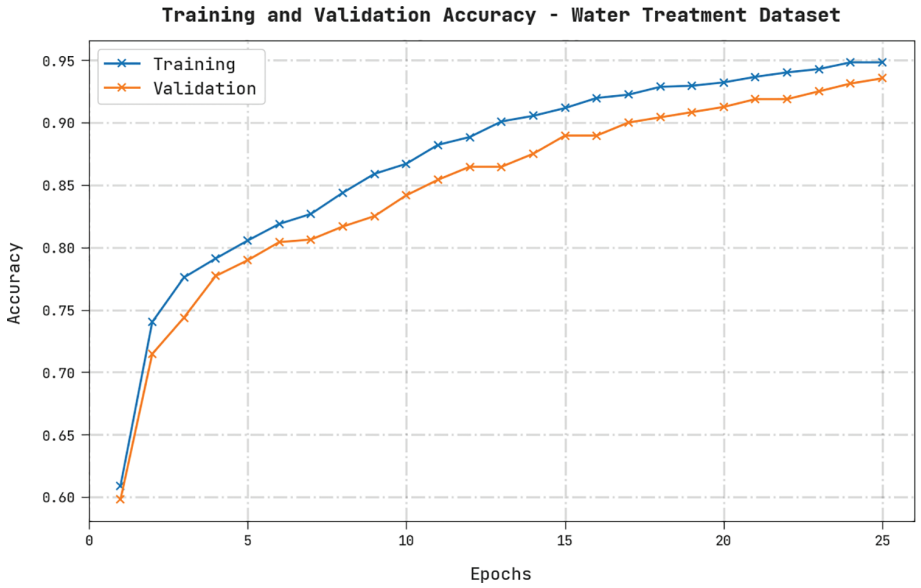

Figure 9: TA and VA analyses results of the HDL-FDIAD approach on water treatment dataset

Both TL and VL values, achieved by the HDL-FDIAD approach on Water Treatment Dataset, are established in Fig. 10. The experimental outcomes imply that the proposed HDL-FDIAD algorithm exhibited the least TL and VL values. In contrast, the VL values were lesser than the TL values.

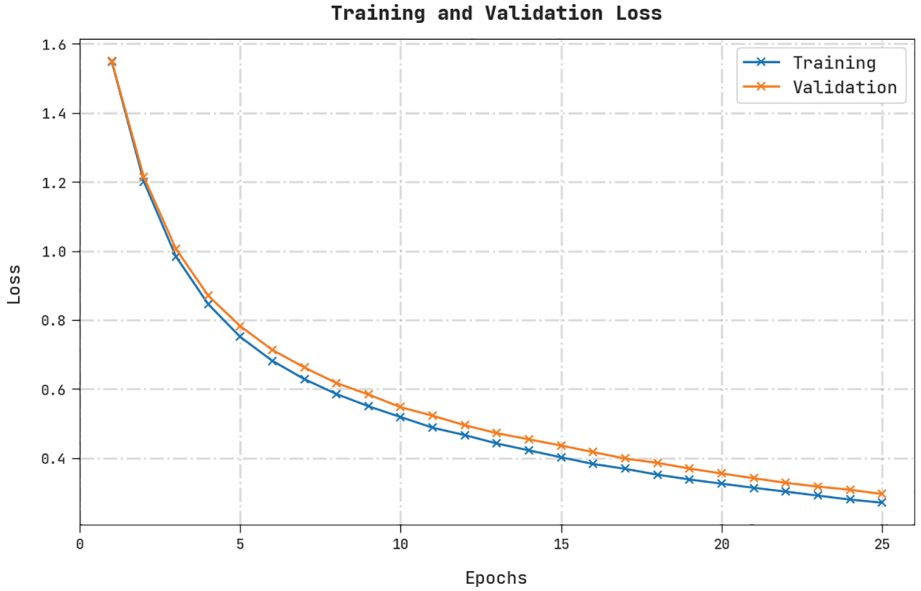

Figure 10: TL and VL analyses results of the HDL-FDIAD approach on water treatment dataset

A clear precision-recall analysis was conducted on the HDL-FDIAD method using the Water Treatment Dataset, and the results are portrayed in Fig. 11. The figure denotes that the HDL-FDIAD methodology achieved enhanced precision-recall values under all the classes.

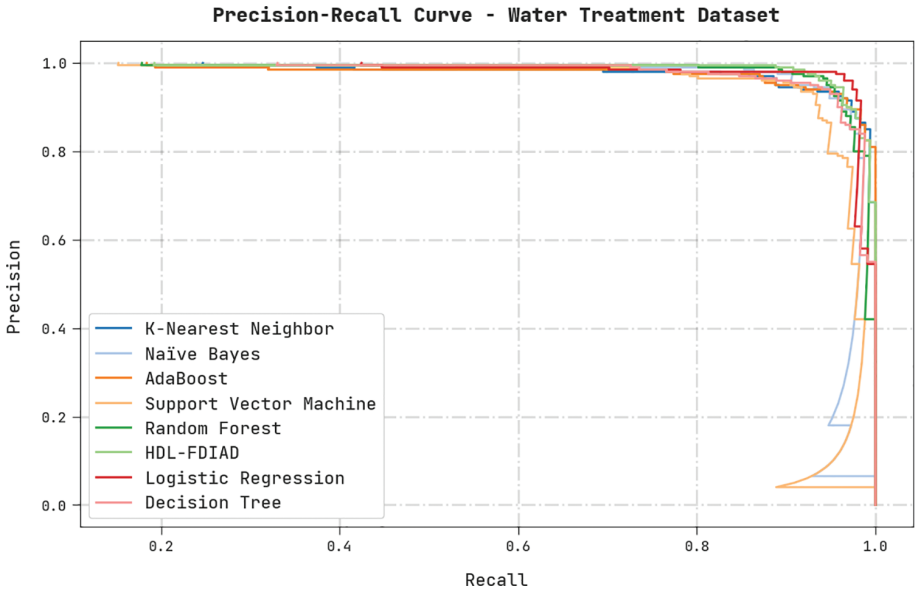

Figure 11: Precision-recall analysis results of the HDL-FDIAD approach on water treatment dataset

A brief ROC analysis was conducted on the HDL-FDIAD method using the Water Treatment Dataset, and the results are shown in Fig. 12. The results indicate that the HDL-FDIAD method established its ability to categorise the Water Treatment dataset under distinct classes.

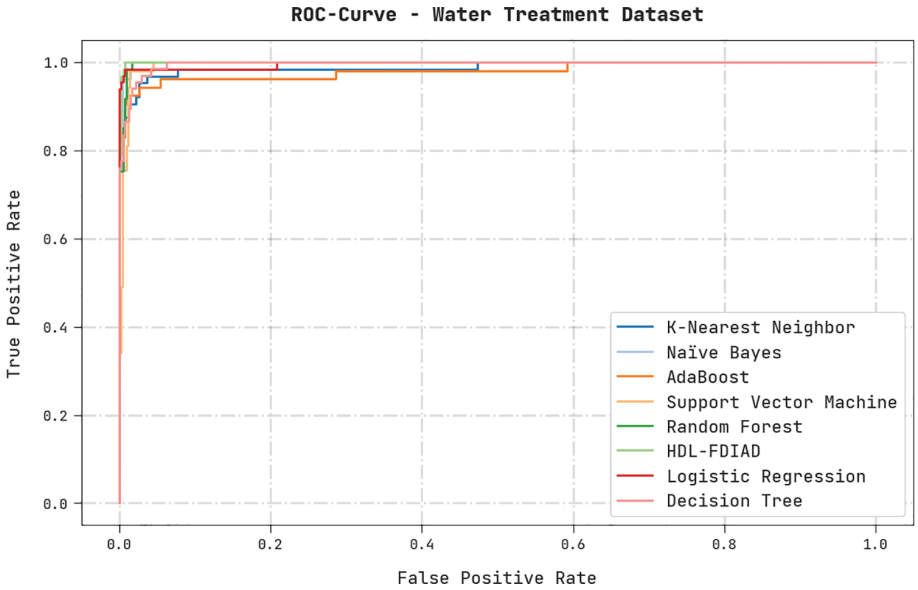

Figure 12: ROC analysis results of the HDL-FDIAD approach on water treatment dataset

4 Conclusion

The current research article has developed a novel HDL-FDIAD model to determine FDI attacks in the IoT environment. The HDL-FDIAD model exploits the EO-FS technique to select the optimal subset of features. Moreover, the LSTM-RNN model is utilized for the purpose of classification. At last, the BO algorithm is employed as a hyperparameter optimizer in this study. To validate the enhanced performance of the HDL-FDIAD model, a wide range of simulations was conducted, and the results were investigated in detail. The comparative study outcomes confirmed that the proposed HDL-FDIAD model is superior to other techniques. Thus, the HDL-FDIAD technique can be exploited to identify the FDI attacks in the IoT environment. In the future, the HDL-FDIAD model can be extended to cloud and fog computing environments too.

Funding Statement: The author received no specific funding for this study.

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

References

[1] Y. Zhang, J. Zhao, D. Zheng, K. Deng, F. Ren *et al.,* "Privacy-preserving data aggregation against false data injection attacks in fog computing," *Sensors*, vol. 18, no. 8, pp. 2659, 2018.

[2] C. Yang, L. Feng, H. Zhang, S. He and Z. Shi, "A novel data fusion algorithm to combat false data injection attacks in networked radar systems," *IEEE Transactions on Signal and Information Processing Over Network*, vol. 4, no. 1, pp. 125–136, 2018.

[3] N. N. Tran, H. R. Pota, Q. N. Tran and J. Hu, "Designing constraint-based false data-injection attacks against the unbalanced distribution smart grids," *IEEE Internet Things Journal*, vol. 8, no. 11, pp. 9422–9435, 2021.

[4] X. Wang, X. Luo, Y. Zhang and X. Guan, "Detection and isolation of false data injection attacks in smart grids via nonlinear interval observer," *IEEE Internet Things Journal*, vol. 6, no. 4, pp. 6498–6512, 2019.

[5] A. Chattopadhyay and U. Mitra, "Security against false data-injection attack in cyber-physical systems," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 2, pp. 1015–1027, 2020.

[6] D. Huang, X. Shi and W. A. Zhang, "False data injection attack detection for industrial control systems based on both time-and frequency-domain analysis of sensor data," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 585–595, 2021.

[7] J. Tian, B. Wang, T. Li, F. Shang, K. Cao *et al.,* "TOTAL: Optimal protection strategy against perfect and imperfect false data injection attacks on power grid cyber–physical systems," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 1001–1015, 2021.

[8] X. Wang, X. Luo, M. Zhang, Z. Jiang and X. Guan, "Detection and isolation of false data injection attacks in smart grid via unknown input interval observer," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3214–3229, 2020.

[9] R. Liu, H. M. Mustafa, Z. Nie and A. K. Srivastava, "Reachability-based false data injection attacks and defence mechanisms for cyberpower system," *Energies*, vol. 15, no. 5, pp. 1754, 2022.

[10] V. P. Srinivasan, K. Balasubadra, K. Saravanan, V. S. Arjun and S. Malarkodi, "Multi label deep learning classification approach for false data injection attacks in smart grid," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 6, pp. 2168–2187, 2021.

[11] M. M. N. Aboelwafa, K. G. Seddik, M. H. Eldefrawy, Y. Gadallah and M. Gidlund, "A machine-learning-based technique for false data injection attacks detection in industrial IoT," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8462–8471, 2020.

[12] A. Alromih, M. A. Rodhaan and Y. Tian, "A randomized watermarking technique for detecting malicious data injection attacks in heterogeneous wireless sensor networks for internet of things applications," *Sensors*, vol. 18, no. 12, pp. 4346, 2018.
[13] A. Amuthan and R. Sendhil, "Hybrid GSW and DM based fully homomorphic encryption scheme for handling false data injection attacks under privacy preserving data aggregation in fog computing," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5217–5231, 2020.
[14] H. Moudoud, L. Khoukhi and S. Cherkaoui, "Prediction and detection of FDIA and DDoS attacks in 5G enabled IoT," *IEEE Network*, vol. 35, no. 2, pp. 194–201, 2021.
[15] S. Wang, S. Bi and Y. J. A. Zhang, "Locational detection of the false data injection attack in a smart grid: A multilabel classification approach," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8218–8227, 2020.
[16] D. T. A. Hamied, A. M. Shaheen, W. A. Salem, W. I. Gabr and R. A. El-sehiemy, "Equilibrium optimizer based multi dimensions operation of hybrid AC/DC grids," *Alexandria Engineering Journal*, vol. 59, no. 6, pp. 4787–4803, 2020.
[17] B. B. Sahoo, R. Jha, A. Singh and D. Kumar, "Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting," *Acta Geophysica*, vol. 67, no. 5, pp. 1471–1481, 2019.
[18] Y. Zhang, D. W. Apley and W. Chen, "Bayesian optimization for materials design with mixed quantitative and qualitative variables," *Scientific Reports*, vol. 10, no. 1, pp. 4924, 2020.
[19] A. Kumar, N. Saxena, S. Jung and B. J. Choi, "Improving detection of false data injection attacks using machine learning with feature selection and oversampling," *Energies*, vol. 15, no. 1, pp. 212, 2021.