Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.054966

ARTICLE

A Secure Framework for WSN-IoT Using Deep Learning for Enhanced Intrusion Detection

Chandraumakantham Om Kumar[1,*], Sudhakaran Gajendran[2], Suguna Marappan[1], Mohammed Zakariah[3] and Abdulaziz S. Almazyad[4]

[1]School of Computer Science & Engineering, Vellore Institute of Technology, Chennai Campus, Chennai, 600127, India

[2]School of Electronics & Engineering, Vellore Institute of Technology, Chennai Campus, Chennai, 600127, India

[3]Department of Computer Sciences and Engineering, College of Applied Science, King Saud University, Riyadh, 11543, Saudi Arabia

[4]Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, 11543, Saudi Arabia

*Corresponding Author: Chandraumakantham Om Kumar. Email: omkumar.cu@vit.ac.in

Received: 12 June 2024 Accepted: 31 July 2024 Published: 15 October 2024

ABSTRACT

The security of the wireless sensor network-Internet of Things (WSN-IoT) network is more challenging due to its randomness and self-organized nature. Intrusion detection is one of the key methodologies utilized to ensure the security of the network. Conventional intrusion detection mechanisms have issues such as higher misclassification rates, increased model complexity, insignificant feature extraction, increased training time, increased run time complexity, computation overhead, failure to identify new attacks, increased energy consumption, and a variety of other factors that limit the performance of the intrusion system model. In this research a security framework for WSN-IoT, through a deep learning technique is introduced using Modified Fuzzy-Adaptive DenseNet (MF_AdaDenseNet) and is benchmarked with datasets like NSL-KDD, UNSWNB15, CIDDS-001, Edge IIoT, Bot IoT. In this, the optimal feature selection using Capturing Dingo Optimization (CDO) is devised to acquire relevant features by removing redundant features. The proposed MF_AdaDenseNet intrusion detection model offers significant benefits by utilizing optimal feature selection with the CDO algorithm. This results in enhanced Detection Capacity with minimal computation complexity, as well as a reduction in False Alarm Rate (FAR) due to the consideration of classification error in the fitness estimation. As a result, the combined CDO-based feature selection and MF_AdaDenseNet intrusion detection mechanism outperform other state-of-the-art techniques, achieving maximal Detection Capacity, precision, recall, and F-Measure of 99.46%, 99.54%, 99.91%, and 99.68%, respectively, along with minimal FAR and Mean Absolute Error (MAE) of 0.9% and 0.11.

KEYWORDS

Deep learning; intrusion detection; fuzzy rules; feature selection; false alarm rate; accuracy; wireless sensor networks

1 Introduction

WSN is an integral part of the IoT infrastructure in the process of connecting everyday things. Both technologies place a significant emphasis on the safety and security of information sharing in the network [1,2] The proper functioning of any WSN and IoT application depends on the encryption, authentication, and freshness of data coming from sensor nodes. Depending on the type of application, such as military or critical infrastructure monitoring, the security level for WSNs varies [3]. WSN offers the highest degree of security. To ensure integrity, authenticity, and secrecy for IoT, it is necessary to integrate several security rules and procedures [4]. The installation of security in IoT devices is more difficult as a result of all these needs [5].

To minimize traffic load, data processing in WSN is restricted to data gathering. In contrast, data processing in IoT comprises data analytics techniques to convert information into knowledge across all IoT applications [6]. WSN does not require a broadband service because all nodes are connected via communication networks [7]. Information can be gathered from every node in the network and sent to the destination node, which then transfers the information back to the server [8]. Hence, an essential component of IoT is an Internet connection. IoT applications must also have location-aware services, dynamic network configuration, interoperability, compliance with laws and regulations, and other essential features [9].

High availability is required for many WSN applications, and it is important to deal with Denial of Service (DoS) attacks. Even though research on DoS attack detection has gained popularity recently, WSN-IoT still faces significant challenges in this area. Intrusion Detection Systems (IDS) serve as a subsequent protection line, observing suspicious activities and producing alerts when detected. IDS are implemented alongside security procedures such as authentication, security systems, and encryption to strengthen security against cyber-attacks. They differentiate between malicious and benign activities using a range of benign traffic patterns and precise attack-specific rules. Mining techniques are used to describe and deploy IDSs with robust behavior and higher accuracy than traditional IDSs, which may struggle against modern, sophisticated cyberattacks.

The IDS aids in identifying attacks in the network to maintain cyber-security [10]. IDS have benefited from the power that artificial intelligence (AI) has given computers and other machines to learn from a dataset with little assistance from humans [11]. An effective intrusion detection system was developed using both machine learning (ML) and deep learning (DL), two subfields of artificial intelligence [12]. Machine learning systems use features that have been explicitly retrieved for the classification and detection of network traffic [13]. In comparison to machine learning, deep learning may strengthen and improve the detection accuracy of the model. The deep learning system, with its neural network, can extract characteristics from the dataset and then perform classification and detection [14].

To align with the existing IDS taxonomy, the proposed method fits within the hybrid IDS category, incorporating elements of signature-based, behavior-based, and anomaly-based detection to provide a comprehensive security solution. Signature-based IDS uses predefined signatures of known threats, behavior-based IDS identifies deviations from normal behavior patterns, and hybrid IDS combines multiple approaches to leverage their strengths and mitigate their weaknesses. This comprehensive approach ensures robust detection and prevention of cyber threats in WSN-IoT environments. Hence, in this research, intrusion detection based on a deep learning model is attempted to accomplish a minimal false alarming rate, high detection accuracy, and better precision. Also, it promotes energy-efficient information sharing in the WSN-IoT environment.

Initially, the user's IP address is checked against a blacklist. Next, a modified fuzzy concept detects intrusions. Finally, adaptive DenseNet-based intrusion detection enhances security. Also, the additional optimization-based feature selection technique chooses the optimal best features from the dataset by removing the redundant information. The Dingo Optimization Algorithm (DOA) is a metaheuristic inspired by the foraging behavior of dingoes, aiming to solve complex optimization problems. It simulates the hunting strategies of dingoes, balancing exploration (discovering new solutions) and exploitation (refining existing solutions) within a population of candidate solutions. DOA employs mechanisms like crossover and mutation to generate new solutions, which are evaluated based on a fitness function. It has been successfully applied across diverse domains for its ability to efficiently navigate complex solution spaces and find near-optimal solutions, offering a robust approach to optimization challenges. Choosing the DOA was based on its proven effectiveness in solving complex optimization problems by mimicking the foraging behavior of dingoes. DOA balances exploration and exploitation well, making it suitable for optimizing the feature selection process in the proposed intrusion detection system. Its ability to handle high-dimensional datasets and find near-optimal solutions, as well as its versatility across various domains, were key factors in selecting DOA to enhance the efficiency and effectiveness of the intrusion detection model in WSN-IoT environments.

The proposed intrusion detection mechanism utilizes threefold intrusion detection mechanisms to reduce the computation overhead and enhance the efficacy of detection. At first, the IP address of the user is checked against the blacklist table. Secondly, a modified fuzzy concept is utilized for detecting the intrusion, and finally, the adaptive DenseNet-based intrusion detection for enhancing the security of the WSN-IoT network. Also, the additional optimization-based feature selection technique chooses the optimal best features from the dataset by removing the redundant information.

The major contributions of the research are:

- **Proposed Hybrid optimization for optimal feature selection:** The hunting behavior of the dingo in dingo optimization and the higher capture capability of the Gannet in Gannet optimization will be hybridized to form the novel Capturing Dingo Optimization (CDO) algorithm. The proposed CDO algorithm is utilized to select the optimal features from the pre-processed input data to enhance the accuracy of attack detection with minimal computational complexity.

- **Proposed Modified Fuzzy and Adaptive DenseNet (MF_AdaDenseNet) based Intrusion Detection:** An intrusion detection mechanism is devised in the CH for detecting the genuinity of the incoming data packets. In this, fuzzy rules are incorporated with the Adaptive DenseNet to enhance the model's detection accuracy.

Problem Statement and Motivation

The secure data transmission in the WSN-IoT using the IDS in the CH identifies the malicious activity in the network once it tries to enter the network. Besides, the network lifetime and the energy efficiency of the network are enhanced through the optimal CH selection. The application domains of the secure and energy-efficient WSN-IoT are industrial environmental monitoring [15], aquaculture, home automation, smart grid deployment, and so on. Several methods were devised for secure energy-efficient information sharing in the WSN-IoT environment. Still, the low detection rate and false alarm limit the performance of the model. Some of the challenges faced by the existing methods are:

- The secure information sharing in WSN-IoT using the authentication mechanism requires additional nodes and computation overhead [16]. Hence, the IDS is considered a secure mechanism for information sharing with limited resources [17].

• The physical architecture of the WSN-IoT makes it easy for an attacker to replicate a sensor within the network to steal data. The detection of the attacker in the sensor nodes is essential to identify the risk at the earlier stage.

• The intrusion detection using the deep learning mechanism has the probability of over-fitting issues that limit the generalization capability of the classifier and reduce the detection accuracy [18].

The organization of the proposed IDS is as follows: Section 2 details the related works with the problem statement, and Section 3 presents the proposed methodology. Section 4 elaborates on the result and discussion of the IDS with ablation study. Finally, Section 5 concludes the research with its future scope.

2 Related Works

The prior methods related to intrusion detection in WSN and IoT are detailed in this section. The intrusion detection with an energy-efficient machine learning mechanism was designed by [17] through optimal cluster head selection. In this optimal CH selection, machine learning-based intrusion detection was devised to identify the risk in the sensor nodes. The devised method failed to analyze the performance of the intrusion detection mechanism and analyze the security of the model. An extreme learning-based method with a multi-kernel function was introduced by [18] for the detection of intrusion in the network. The multi-kernel parameter settings enhance the detection accuracy and solve the issue concerning the low detection rate. The higher energy consumption of the node was a challenging task that limited the performance. The intrusion detection in WSN using the machine learning technique was designed by [19] with hybrid optimization criteria. The features in the dataset are transformed into the normal value format through the data normalization technique. In addition, the complexity overhead was reduced through the feature selection criteria. The analysis of the method was devised using the UNSW-NB15 and NSL-KDD datasets for measuring the performance. The method acquired better performance in detecting the intrusion; still, the over-fitting issues limit the model. Halbouni et al. [20] designed a network intrusion detection mechanism with a hybrid deep learning mechanism for extracting the temporal and spatial features of the incoming data. In this, the enhanced performance was accomplished through the inclusion of the dropout and batch normalization process. The experimental assessment depicts the effectiveness of the introduced model. However, the detection rate and the false alarming rate of the model were higher.

Attack detection using the lightweight mechanism was designed by [21] using the light Gradient Boosting Machine mechanism with under-sampling and over-sampling algorithms to solve the issues concerning underfitting and overfitting. Thus, the detection accuracy of the method was enhanced through the balanced data. In addition, the computation overhead was reduced through the feature selection criteria. The assessment depicts improved performance based on various measures, such as flooding, grey holes, black holes, and normal attacks, and accomplished superior performance. However, the detection accuracy of the method was degraded by removing enormous significant features.

A machine learning-based attack detection using the XGBoost was designed by [22] through feature selection based on adaptive synthetic sampling and Boruta techniques. The analysis of the designed model was employed through various assessment measures, and superior outcomes were accomplished using the NSL-KDD dataset. Still, the over-fitting issues exist due to the non-capability of the model to handle large amounts of data. Intrusion detection using the k-nearest neighbor (KNN) was designed by [23,24] for the IoT environment. In this, attribute selection using optimization was devised along with the principal component analysis (PCA). The analysis of the model is based on

various assessment measures to depict its superiority and enhanced accuracy. Still, the requirement of hand-crafted feature extraction adds a computation burden to the model. A Random Forest based intrusion detection mechanism devised by [25] utilized entropy-based feature selection. In addition, SMOTE-based data balancing and normalization were employed during the data pre-processing stage. The analysis of the model portrayed the superior outcome based on assessment measures; still, the overfitting issues limit its applicability in handling large datasets. Similarly, intrusion detection based on ensemble learning with Pearson's correlation coefficient (PCC), isolation forest (IF), and Matthews correlation coefficient (MCC) [26,27] using a machine learning model like K-NN classifier illustrates its superior outcomes. Still, overfitting issues and the need for additional feature extraction approaches limit its potential in real-time processing.

Some of the IoT-based attack detection systems, Kandhro et al. [28] developed a Generative Adversarial Network (GAN) based intrusion detection system to detect real-time cyber threats. In this, the discriminate model of the GAN was incorporated with hashing transformation to acquire local information and identify the attack. The assessment of the devised technique accomplished superior performance using six datasets: NSL-KDD, CIDDS-001, UNSWNB15, CIDDS-2018, BoT-IoT, and Edge-IIoT. Nevertheless, the model's computational overhead increases due to the neglect of the significant attribute extraction. IDS based on deep learning was designed by [29] using the lightweight ensemble learning criteria. In this, the numeric tabular data was converted into an image to classify the attacks. A hybrid optimization was utilized to train the classifier to minimize data loss during information learning. The validation of the method was evaluated through the estimation of MCC. A semi-supervised learning technique was devised by [30] through the attention-based feature fusion criteria. Besides, the manifold regularization and dense cross-layer connections were included in the ladder network to enhance the learning and avoid degradation issues. Thus, the learning rate of the model was enhanced, and the data dependencies were minimized. The assessment depicts the low false alarm rate for heterogeneous data learning. The unstable outcome of the model degrades the performance of attack detection. The description of the related works is depicted in Table 1.

Table 1: Review of related works

Authors	Techniques used	Advantages	Limitations
Srividya et al. [17]	Optimal-machine learning	The attack detection in the trusted path based on the attributes of the nodes accomplished an enhanced detection rate.	Failed to analyze the accuracy of attack detection.
Zhang et al. [18]	Extreme machine learning	Detected the intrusion patterns with higher accuracy.	Consumes enormous energy.
Hemanand et al. [19]	Machine learning	Detected abnormal and normal nodes with improved accuracy to enhance network security.	Failed to consider the energy-efficient node deployment.

(Continued)

Table 1 (continued)

Authors	Techniques used	Advantages	Limitations
Halbouni et al. [20]	Hybrid deep learning	Acquired higher detection rate by incorporating dropout layer and batch normalization.	Degrades its performance while using the imbalanced dataset.
Dener et al. [21]	Machine learning	Accomplished improved performance in detecting the intrusion through data balancing technique.	It requires hand-crafted feature extraction, and the interpretability of the method was lacking.
Amaouche et al. [22]	XGBoost	The feature selection helps to focus on the most relevant features for attack detection and potentially improves performance.	XGBoost was computationally expensive, which might limit its feasibility for resource-constrained devices in vehicles.
Mohy-Eddine et al. [23]	Ensemble machine learning	Combining IF and PCC reportedly reduces computational overhead and prediction time.	Ensemble methods were more complex to implement and understand compared to single classifiers.
Mohy-Eddine et al. [24]	KNN	The model reportedly lowers the FAR, thus generating fewer false positives and reducing unnecessary alerts that can burden security teams.	K-NN classifier has a higher risk of over-fitting.
Amaouche et al. [25]	Random forest	Mutual information-based feature selection helps identify the most relevant features for attack detection, potentially improving model efficiency and interpretability.	The computation burden was higher due to the incorporation of additional SMOTE and feature selection.

(Continued)

Table 1 (continued)

Authors	Techniques used	Advantages	Limitations
Mohy-Eddine et al. [26]	KNN	Feature selection significantly reduces prediction time, making the model more suitable for real-time intrusion detection in IoT environments.	Higher computational complexity.
Mohy-Eddine et al. [27]	Ensemble machine learning	The use of MCC suggests good performance on imbalanced datasets.	The use of multiple feature selection and classifiers makes the computation burden to the model.
Kandhro et al. [28]	Deep learning	The accuracy of the devised method was superior compared to other state-of-the-art techniques.	Not capable of identifying all kinds of attacks.
Okey et al. [29]	Deep learning	Accomplished improved detection accuracy and validated the performance using Matthew's Correlation Coefficient.	Failed to consider the significant attributes that enhance the detection accuracy and minimize the computation overhead.
Long et al. [30]	Semi-supervised ladder network	The generalization capability of the model was improved through the ladder network in the decoding process to enhance the detection accuracy.	The outcome of the method was unreliable and stable due to the unlabeled learning criteria.
Nguyen et al. [31]	Genetic convolutional neural network	The three layer feature selection mechanism have contributed significantly in improving the detecting performance.	The outcome is relativey high due to combination of using CNN and BG as classifies.

3 Methodology

The WSN-assisted IoT network is made up of numerous sensor devices that are connected via a radio communication link operating within the radio range. The network has a large number of randomly deployed sensor nodes to collect various physical information to realize tasks such as intelligent perception, efficient control, and decision-making. Besides, the nature of network deployment leads to the attack more easily. Hence, the intruder in the network needs to be identified using an efficient IDS. Several intrusion detection methods were devised by previous researchers; still, accurate intrusion detection with an efficient network model is a challenging task. Hence, in this research, an intrusion detection mechanism is proposed. The block diagram for the proposed MF_AdaDenseNet-based IDS is depicted in Fig. 1.

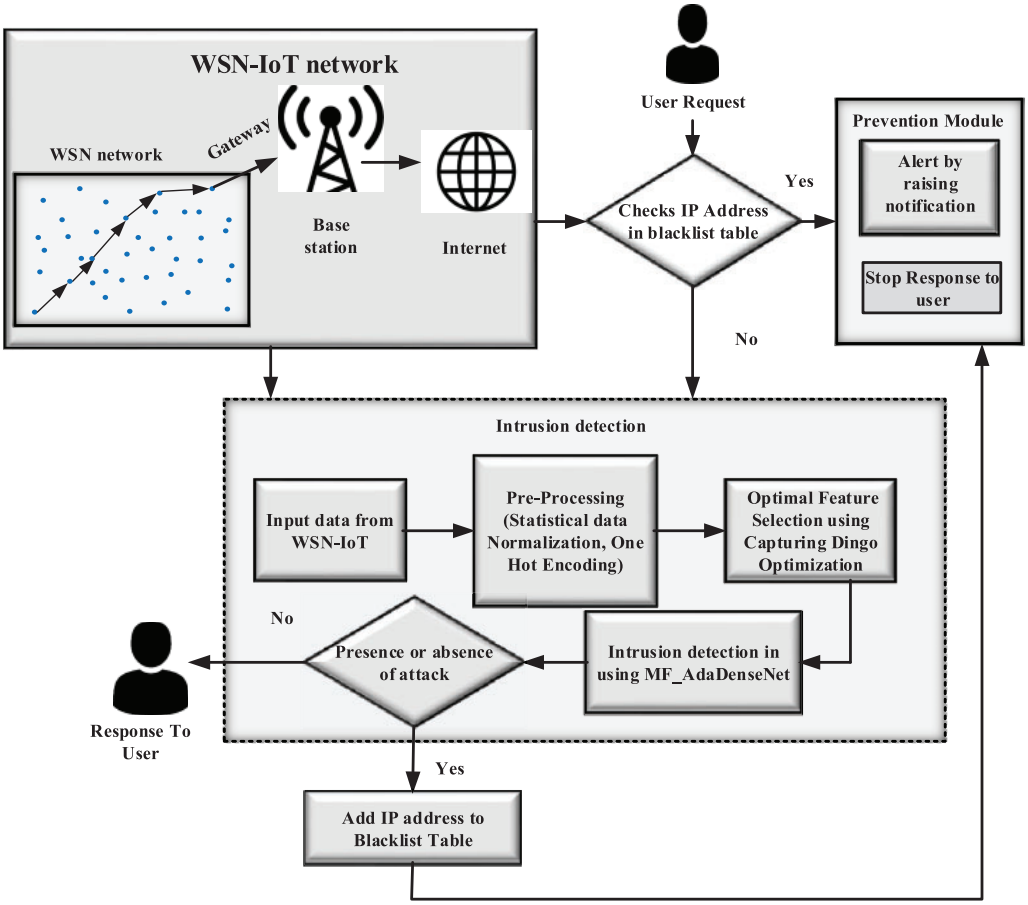

Figure 1: Block diagram of proposed MF_AdaDenseNet based IDS

The WSN-IoT network consists of wireless sensor nodes and an IoT gateway. The sensor nodes collect data from the environment and forward it to the gateway. The gateway then sends the data to the base station for processing. The system receives data from the WSN-IoT network. The data undergoes pre-processing steps like normalization and statistical data capturing to prepare it for further analysis. Dingo optimization, a specific technique, is used for optimal feature selection. The pre-processed data is then fed into the Detection System. This module likely uses fuzzy logic to classify the data into different attack categories (intrusion, no attack, etc.). Membership functions that define the degree of

membership for data points in categories. Fuzzy rules that map combinations of fuzzy data attributes. A defuzzification method to convert the fuzzy output into a crisp attack classification. Data points with a "mixed" attack possibility from the fuzzy module might be fed into this module for further analysis. AdaDenseNet can potentially learn complex patterns in the data to provide a more confident intrusion classification for these ambiguous cases. If the system detects an intrusion, it might add the corresponding IP address to a blacklist table to prevent future attacks from that source. The system might also generate a notification or raise an alert to the user about a potential intrusion. The initial step in handling user requests involves checking the IP address against a blacklist table to determine if it's blocked due to prior intrusion attempts. If the IP address is found in the blacklist, the request is automatically dropped to prevent unauthorized access. For requests not on the blacklist, they undergo pre-processing, which includes statistical analysis and one-hot encoding to normalize the data. Optimal feature selection is then performed using the capturing dingo optimization algorithm, refining the dataset for efficient processing. The core task of intrusion detection and classification is executed through the MF_AdaDenseNet module, designed to leverage adaptive learning and dense neural networks for the accurate detection of intrusion patterns. This multi-stage approach ensures robust security measures while efficiently handling incoming requests for further analysis and action.

3.1 Data Acquisition

The proposed intrusion detection method utilizes the data gathered by various datasets like NSL-KDD [32], CIDDS-001 [33], UNSWNB15 [34], CIDDS-2018 [35], BoT-IoT [36], and Edge-IIoT [37].

NSL-KDD: The dataset described in Table 2 comprises 125,967 instances with 42 attributes. There are four various types of attacks, and benign user details are recorded.

Table 2: Fitness value analysis

Optimization	Iteration for best fitness
Proposed (CDO)	**20**
DOX	30
Gannet optimization	29

CIDDS-001: The dataset comprises four various attacks along with the benign user. There are 14 various attributes present in the data, with 16 million instances. The dataset was gathered from the OpenStack environment, which had internal servers (backup, mail, file, and web) and external servers (file synchronization and web server) in 2017.

UNSWNB15: The dataset comprises nine various types of attacks along with the normal user. The dataset was created by the Australian Centre for Cyber Security in 2015. The dataset comprises 49 attributes. The sources utilized for gathering the data are Microsoft Security Bulletins (MSB), Common Vulnerabilities and Exposures (CVE), and Symantec Corporation BID (BulletinID). There are 2,57,673 instances available, including both the benign and attacker.

CIC IDS-2018: The CIC IDS 2018 dataset on AWS is an extensive dataset for intrusion detection systems, encompassing diverse contemporary network attack scenarios. It contains detailed logs and network traffic data, enabling the development and assessment of machine learning models for cybersecurity. This dataset aids researchers and practitioners in creating and benchmarking intrusion detection systems, thereby advancing network security research.

Edge-IIoT: The Edge-IIoT dataset, also known as the Edge Industrial IoT dataset, is a collection of data specifically gathered from sensors, devices, and equipment deployed at the edge of a network in industrial settings. The dataset comprises 46 various attributes with normal and attack classes.

BoT-IoT: The BoT-IoT dataset is a comprehensive collection of data designed specifically for cybersecurity research and analysis in the context of IoT devices. Network traffic data from a variety of IoT devices, such as smart TVs, IP cameras, home routers, and other connected devices, is included in the dataset. The dataset comprises 63 various attributes. The data includes normal and malicious activities, simulating different types of attacks that IoT devices might encounter in real-world scenarios.

3.2 Intrusion Detection Using Blacklist Table

The IP address of the incoming user request is checked initially in the blacklist table for the detection of its availability in the blacklist. If the IP address occurs in the blacklist table, then the response for the request is dropped. Otherwise, the user request is fed into the proposed intrusion detection module MF_AdaDenseNet to detect the genuineness of the requested user.

3.3 Intrusion Detection Using MF_AdaDenseNet

The user request, whose IP address is not in the blacklist table, is passed into the proposed MF_AdaDenseNet module for intrusion detection. Three various steps utilized for the detection of intrusion are pre-processing, optimal feature selection, and intrusion detection using MF_AdaDenseNet to responsd to the user request. The response can be either giving information access or dropping the request, depending on the outcome of the proposed MF_AdaDenseNet method.

3.3.1 Pre-Processing

The intrusion detection datasets have features such as packet size, duration of connections, and other scales. Normalization makes all the features scale into a similar range. Further, the one-hot encoding technique assists in making the features within the 0 to 1 range to make the execution simpler. The data pre-processing is devised through two two-fold processes to obtain the normalized outcome. The reason behind the pre-processing is to normalize the data to obtain equal weights for all the features in the dataset.

***Statistical Normalization*:** The transformation of data from any distribution format into the standard normal distribution is devised through statistical data normalization. Here, the unit variance and zero mean are considered for the standard data normalization [19]. The formulation for the statistical normalization is expressed as,

$$SN_i = \frac{OV_i - M}{SD} \tag{1}$$

where, the statistical normalization is defined as SN_i, the original values are defined as OV_i, the standard deviation is defined as SD, and the mean is defined as M. Then, the expression for estimating the M and SD are formulated as,

$$M_i = \frac{1}{p}\sum_{i=1}^{p} OV_i.SD \tag{2}$$

$$SD = \sqrt{\frac{1}{p}\sum_{i=1}^{p}(OV_i - M)} \tag{3}$$

where, the total number of attributes is indicated as p. In the statistical normalization process, the outcome is in the range of $[-3, 3]$, which is not normalized in the range of $[0, 1]$ due to the higher value of p as per the central limit theorem. However, the utilization of statistical normalization is due to its applicability to large datasets, which accomplishes quicker execution and offers consistent data. Thus, the second step of pre-processing is devised based on the one-hot encoding process.

***One hot encoding*:** It is common practice to employ one-hot encoding [21] when label encoding is inadequate, and there is no ordinal link between two nominal categorical variables. For a categorical random variable A with K dissimilar values $(a_1, a_2, \ldots, a_k)$. Every component of a vector b that represents a specific value a_i is zero, with the exception of the i^{th} component, which is encoded as 1.

For example, if A takes values from the set $W = e, f, h$, and $a_1 = e, a_2 = f, a_3 = h$. A one–hot encoding for a is $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$.

3.3.2 Optimal Feature Selection Using Capturing Dingo Optimization

From the normalized attributes, the most informative features are selected optimally using the proposed Capturing Dingo Optimization (CDO) algorithm. The CDO algorithm combines the dingo's hunting style and the gannet's capture behavior to determine the best global solution for selecting the ideal feature.

Motivation behind the CDO Algorithm

Dingo optimization (DOX) is designed to solve optimization issues by considering the social behaviour of the hunting strategy. Dingoes live in a pack of 12 to 15 members in a group with a leader to control the pack. The leader may be a male or female member who is responsible for choosing the hunting and sleeping places and the decision-making process. The leader is assigned by considering the strongest and dominant member of the pack and is responsible for organizing and maintaining the discipline of the pack. The second-best member of the Dingo pack is the advisor, who is the intermediate member of the pack and the leader. The absence of the leader is replaced by the advisor, Dingo, who maintains the discipline of the pack. The third hierarchical place goes to the subordinate Dingo, who is responsible for scouting the territory and alerting the pack if it finds any attack from the intruder. The communication between the members of the pack is devised through various sound intensities. The DOX follows diversification, surrounding, and intensification phases of the hunting strategy to avoid the local optimal solution trapping. Besides, the DOX is less complex in terms of computation and is flexible to use for solving various optimization issues concerning real-world solutions. Here, the prey (target solution) surrounding capability of the DOX is further enhanced by integrating the capturability behaviour of the Gannet. A gannet is a carnivorous bird that has high capturability in catching prey by performing various diving strategies. Thus, the escaping capability of the prey is minimized through the hybridization of DOX with the capturability behavior of the Gannet. Hence, the global best solution with balanced diversification and intensification is acquired by the proposed CDO algorithm with a fast convergence rate. In 0–100 iterations of fitness, the values proposed by the CDO algorithm attain the best fitness value at the 20th iteration. In contrast, in other algorithms, DOX attains the 30th iteration, and gannet attains the 29th iteration. This analysis describes that the proposed CDO leads to producing a less complex model due to faster convergence

speed. By selecting the best fitness value, CDO chooses optimal features, which leads to increasing training speed. Table 2 defines the best fitness value attained for proposed and existing optimization algorithms using iterations.

Mathematical Modelling

The CDO algorithm's mathematical structure includes three distinct phases centered around the target, intensification and diversification. The notations utilized by the CDO with the description are presented in Table 3.

Table 3: Notations with descriptions

Notation	Description
$\overrightarrow{K_c}$	The spacing between the search agent and the solution
$\overrightarrow{X}$	Coefficient vector
$\overrightarrow{Y}$	Coefficient vector
$\overrightarrow{S}_s$	Position of the target
$\overrightarrow{S}$	Position of the search agent
$\overrightarrow{v}$	The value gets degraded from 3 to 0 towards the course of iteration linearly
$\overrightarrow{u}_2$	[0, 1] range random vector
$\overrightarrow{u}_1$	[0, 1] range random vector
T	Iteration
T^{Max}	Maximal number of iterations
$\overrightarrow{D}_L$	Intensity of leader
$\overrightarrow{D}_A$	Intensity of advisor
$\overrightarrow{D}_R$	Intensity of remaining members of pack
FF_L	Fitness of leader
FF_A	Fitness of advisor
FF_R	Fitness of remaining members of pack
$\overrightarrow{K}_L$	The spacing between the leader and the solution
$\overrightarrow{K}_A$	The spacing between the advisor and the solution
$\overrightarrow{K}_R$	The spacing between the remaining members of pack and the solution
$\overrightarrow{S}_L$	Position of the leader
$\overrightarrow{S}_A$	Position of the advisor
$\overrightarrow{S}_R$	Position of the remaining members of pack
$\overrightarrow{X}_1$	Coefficient vector of leader
$\overrightarrow{X}_2$	Coefficient vector of advisor
$\overrightarrow{X}_3$	Coefficient vector of remaining members of pack
$FF\,(q)$	Fitness function
$FF\,(q)_{ratio}$	Ratio of optimal features selected from the dataset
m	U and V shaped diving behaviour's probability of Gannet search agent

The pseudo-code for the CDO algorithm is presented in Algorithm 1.

Algorithm 1: Pseudo-code for CDO algorithm

```
Input: The candidates in the search space
Output: Best solution
Initialize the search agents
Initialize the parameters
While (τ < T^max)
{
For each search agent:
    {
      Estimate the feasibility by evaluating the fitness using Eq. (5)
      Update the position of the search agent using Eq. (19)
      Update the position of the best two search agents using Eqs. (23) and (24)
      Re-estimate the fitness using Eq. (5)
    }
    End for
    τ = τ + +
    Check the termination criteria
}
End while
```

***Initialization*:** The search agents and the target are located randomly in the search space with the maximal iteration T^{Max}. The feasibility of the position of the search agent is evaluated based on the fitness function.

***Fitness Estimation*:** The goal of the fitness estimation is to select the best attributes from the dataset for the reduction of redundant features. Besides, the intrusion detection error rate is also minimized through the optimal best feature selection criteria. Let us consider the number of features in the dataset is indicated as E and the optimal best-selected features be indicated as G. The selection of attributes from the dataset is formulated as,

$$FF\,(q) = 1 - \frac{F_p + F_N}{N + P}, \quad (N + P) \in E \tag{4}$$

where, the detection of the normal nodes correctly by the classifier is indicated as T_p based on the selected features. The detection of normal nodes that are misclassified by the classifier is indicated as F_p. The detection of intruders correctly by the classifier is indicated as T_N, and the detection of intruders that are misclassified as normal nodes are indicated as F_N. The combination of $T_p + F_N = P$ and $F_P + T_N = N$. The role of the search agents in the search space is to identify the optimal best features that minimize the classification error rate as per the Eq. (5). Then, the ratio of selected attributes is formulated as,

$$FF\,(q)_{ratio} = \frac{G}{E}, \quad G \in E \tag{5}$$

The optimal best features selected from the dataset based on $FF\,(q)$ enhances the detection accuracy by selecting the features with minimal classification error rate.

***Surrounding the target*:** Encircling the target in order to seize it is the leader's pack of search agents. The target's location was previously unknown; thus, the search agents identified it by searching for it

in the search space. The mathematical formula for modelling the search agents surrounding the target is expressed as,

$$\overrightarrow{K_c} = \left| \overrightarrow{X} . \overrightarrow{S}_s (q) - \overrightarrow{S} (\tau) \right| \tag{6}$$

$$\left[\overrightarrow{S} (\tau + 1) \right]_{DOX} = \overrightarrow{S}_s (\tau) - \overrightarrow{Y} . \overrightarrow{K} (c) \tag{7}$$

$$\overrightarrow{X} = 2 . \overrightarrow{u}_1 \tag{8}$$

$$\overrightarrow{Y} = 2 \overrightarrow{v} . \overrightarrow{u}_2 - \overrightarrow{v} \tag{9}$$

$$v = 3 - \left(T * \left(\frac{3}{T^{\max}} \right) \right) \tag{10}$$

Here, the surrounding capability of the search agent is further enhanced by hybridizing the capturability behaviour of the Gannet search agents in the search space. It is expressed as,

$$[S (\tau + 1)]_{Gannet} = \begin{cases} S (\tau) + i_1 + i_2, \, m \geq 0.5 \\ S (\tau) + j_1 + j_2, \, m < 0.5 \end{cases} \tag{11}$$

In this, the solution acquired by the Gannet search agents utilizes the same probability for both the U and V-shaped dives to capture the target, which is indicated as m. The diving behaviour of the Gannet search agents with high capturability never let the target escape from capture. Then, the formulation for the parameters is expressed as,

$$i_2 = B * (S (\tau) - S_r (\tau)) \tag{12}$$

$$j_2 = C * (S (\tau) - S_c (\tau)) \tag{13}$$

$$B = (2 * d_1 - 1) * I \tag{14}$$

$$C = (2 * d_2 - 1) * J \tag{15}$$

where, the randomly chosen search agent is indicated as $S_r (\tau)$, the random number with the range $[0, 1]$ is indicated as d_1 and d_2, and $S_c (\tau)$ refers to the average solution accomplished by the candidates in the search space is expressed as,

$$S_c (\tau) = \frac{1}{S} \sum_{x=1}^{S} S_x (\tau) \tag{16}$$

Here, xth candidate in the search space is indicated as $S_m (\tau)$, the range of i_1 is $[-I, I]$, and the range of j_1 is $[-J, J]$.

Then, the novel hybridized equation acquired by the CGO algorithm is formulated as,

$$\overrightarrow{S} (\tau + 1) = 0.5 \left[\overrightarrow{S} (\tau + 1)_{DOX} \right] + 0.5 \left[\overrightarrow{S} (\tau + 1)_{Gannet} \right] \tag{17}$$

$$\overrightarrow{S} (\tau + 1) = \left[\begin{cases} 0.5 [S (\tau) + i_1 + i_2] + 0.5 \left[\overrightarrow{S}_s (\tau) - \overrightarrow{Y} . \overrightarrow{K} (c) \right], \, m \geq 0.5 \\ 0.5 [S (\tau) + j_1 + j_2] + 0.5 \left[\overrightarrow{S}_s (\tau) - \overrightarrow{Y} . \overrightarrow{K} (c) \right], \, m < 0.5 \end{cases} \right] \tag{18}$$

Using the above mentioned update function presented in Eq. (18), the target is surrounded by the CGO search agents.

***Capturing the target*:** After surrounding the target, the search agents capture the target by following the commands of the leader. Here, to solve the optimization issues, the position of the first two search agents is considered, and the remaining members of the pack update the position based on the first two best positions. The mathematical formulation for the position updating is expressed as,

$$\overrightarrow{K}_L = \left| \overrightarrow{X}_1 \cdot \overrightarrow{S}_L - \overrightarrow{S} \right| \tag{19}$$

$$\overrightarrow{K}_A = \left| \overrightarrow{X}_2 \cdot \overrightarrow{S}_A - \overrightarrow{S} \right| \tag{20}$$

$$\overrightarrow{K}_R = \left| \overrightarrow{X}_3 \cdot \overrightarrow{S}_R - \overrightarrow{S} \right| \tag{21}$$

$$\overrightarrow{S}_1 = \left| \overrightarrow{S}_L - \overrightarrow{Y} \cdot \overrightarrow{K}_L \right| \tag{22}$$

$$\overrightarrow{S}_2 = \left| \overrightarrow{S}_A - \overrightarrow{Y} \cdot \overrightarrow{K}_A \right| \tag{23}$$

$$\overrightarrow{S}_3 = \left| \overrightarrow{S}_R - \overrightarrow{Y} \cdot \overrightarrow{K}_R \right| \tag{24}$$

Here, the intensities of the search agents based on the fitness function are expressed as,

$$D_L = \log\left(\frac{1}{FF_L - (1E - 100)} + 1 \right) \tag{25}$$

$$D_A = \log\left(\frac{1}{FF_A - (1E - 100)} + 1 \right) \tag{26}$$

$$D_R = \log\left(\frac{1}{FF_R - (1E - 100)} + 1 \right) \tag{27}$$

Thus, the search agents update their position and estimate the location of the target to capture.

***Intensification*:** In the intensification phase, the search agents capture the target by decreasing the $\overrightarrow{v}$ linearly from the value 3 to 0 throughout the iteration. After capturing the target, when the search agent is within the range [1, 1], it indicates that the location of the search agent is between the location of the targets and the current position. Thus, the search agents update the position and capture the target. When the value is less than '1', the search agents capture the target and identify the solution.

***Diversification*:** The coefficient vectors $\overrightarrow{X}$ and $\overrightarrow{Y}$ are responsible for the diversification phase, wherein the range of $\overrightarrow{X}$ is from [0, 3] that helps to diversify more area in the search space and avoids the solution trapping at a local optimal solution. The value of $\overrightarrow{Y} < -1$ that indicates that the target is moving away from the search agent. When the value of $\overrightarrow{Y} > 1$, the search agents are closer to the target. Thus, the diversification phase is extended further by adjusting the coefficient vectors to obtain the global best solution.

***Re-evaluating the fitness*:** After updating the solution by the search agents in the intensification phase, the feasibility of the updated solution is evaluated based on the $FF\,(q)$.

***Termination*:** The acquisition of the global best solution or the attainment of T^{Max} stops the iteration and terminates the algorithm. Thus, the more informative features are selected from the dataset using the CDO algorithm to enhance detection accuracy with minimal computation.

3.3.3 Proposed MF_AdaDenseNet for Intrusion Detection

The proposed MF_AdaDenseNet for Intrusion Detection is designed by integrating the modified fuzzy (MF) concept with the Adaptive DenseNet (AdaDenseNet) for detecting intrusion in the network. Here, the incorporation of Fuzzy logic allows for the representation of uncertainty in decision-making, which enhances the detection accuracy. The DenseNet utilizes fewer parameters to process intrusion detection using feature reuse criteria. Also, the self-attention mechanism enables the model to capture long-range dependencies in the data by assigning different weights to features. Thus, by integrating these mechanisms, an enhanced outcome is derived for the proposed intrusion detection mechanism. The intrusion detection based on MF_AdaDenseNet uses two-fold detection criteria. Initially, the MF is utilized for the detection of intrusion in three categories: normal, intruder, and mixed (either intruder or normal). Here, the request of the user who was identified as an intruder by the MF is dropped, and a response is denied. The user request identified as normal is responsible for accessing the information. The user request identified as mixed is fed into the AdaDenseNet to detect the intrusion. Thus, by using the MF concept, the complexity of the network is reduced.

Modified Fuzzy-Based Intrusion Detection

MF-based intrusion detection considers the packet size, byte rate, packet rate, and average packet for detecting the intrusion under three various categories [38]. Modified Fuzzy technique is described in Algorithm 2.

Algorithm 2: Modified Fuzzy technique

Inputs:

- Packet Size
- Byte Rate
- Packet Rate
- Average Packet Size

Outputs:

- Category (Normal, Suspicious, Intrusion)

Steps:

1. **Define Membership Functions:**
   - Define triangular membership functions for each variable and linguistic term (Small, Medium, Large) based on the specified ranges.
2. **Fuzzification:**
   - For a given network traffic sample, calculate the degree of membership for each input variable in each linguistic term using the defined membership functions.
3. **Rule Evaluation:**
   - Define fuzzy rules that map combinations of input variables to output categories (Normal, Suspicious, Intrusion).
   - For each rule:
     - Use the "min" operator to combine the degree of membership of each input variable in the rule's antecedent (e.g., if (Packet Size is Small) AND (Byte Rate is Low), then min (degree of membership for Small in Packet Size, degree of membership for Low in Byte Rate) becomes the activation degree of the rule).

(Continued)

Algorithm 2 (continued)

4. **Aggregation:**
   - For each output category, combine the activation degrees of all rules that map to that category using the "max" operator. This represents the overall fuzzy membership for each category for the given traffic sample.
5. **Defuzzification:**
   - Choose a defuzzification method like the centroid method. This method calculates the weighted average of the membership function values in the aggregated fuzzy set for each category. The weight for each point is its membership degree.
   - The category with the highest centroid value becomes the final output (Normal, Suspicious, or Intrusion).

Example Calculation

Consider the following inputs to be available:

- Packet Size: 1200 bytes
- Byte Rate: 180 bytes/sec
- Packet Rate: 25 packets/sec
- Average Packet Size: 1400 bytes

1. **Fuzzify Inputs:**
   - Packet Size: Small (0), Medium (0.6), Large (0.4)
   - Byte Rate: Low (0), Medium (0.7), High (0.2)
   - Packet Rate: Low (0), Medium (1), High (0)
   - Average Packet Size: Small (0), Medium (0.5), Large (0.3)
2. **Evaluate Rules:**
   - Rule 1: min (0, 0, 0, 0) = 0 (Category is Normal)
   - Rule 2: min (0.6, 0.7, 1, 0.5) = 0.5 (Category is Suspicious)
   - Rule 3: min (0.4, 0.2, 0, 0.3) = 0 (Category is Intrusion)
3. **Aggregate Results:**
   - Normal: 0
   - Suspicious: 0.5
   - Intrusion: 0
4. **Defuzzify Output:**
   - The output category is determined based on the highest membership value. Here, the input data is "Suspicious" with a membership value of 0.5.

The defuzzification output can be trained on AdaDenseNet to detect intrusion in WSN-IoT networks.

AdaDenseNet-Based Intrusion Detection

The attack possibility identified as mixed is fed into the AdaDenseNet to detect the intrusion. The proposed AdaDenseNet is designed by incorporating a self-attention module within the traditional DenseNet to enhance the accuracy of intrusion detection. The architecture of AdaDenseNet is depicted in Fig. 2, and the corresponding configuration is presented in Table 4.

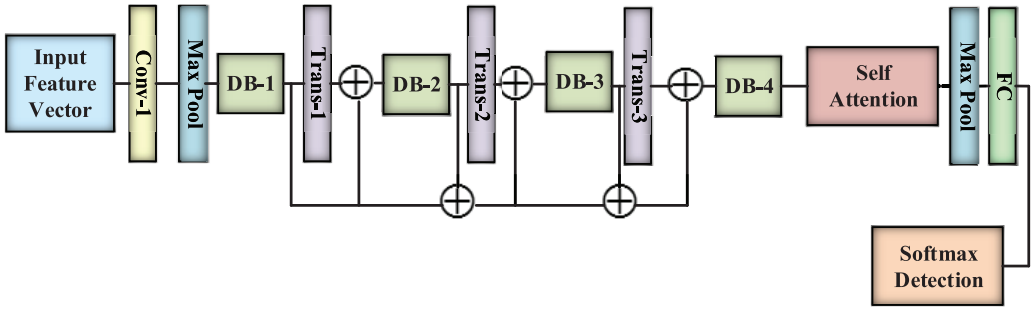

Figure 2: Proposed AdaDenseNet for intrusion detection

Table 4: Configuration of proposed AdaDenseNet

Layer	Details
Convolutional layer-1	Filter size = (64), Strides = 2
Maxpool layer	Filter size = (3), Strides = 2
DenseBlock1 (DB-1)	Contains 6 dense blocks, each block made of conv with filter size = (128) and conv with filter size = (32)
DenseBlock2 (DB-2)	Contains 12 dense blocks, each block made of conv with filter size = (128) and conv with filter size = (32)
DenseBlock3 (DB-3)	Contains 24 dense blocks, each block made of conv with filter size = (128) and conv with filter size = (32)
DenseBlock4 (DB-4)	Contains 16 dense blocks, each block made of conv with filter size = (128) and conv with filter size = (32)
Self-Attention	tanh activation function
Transition layers	Each layer consists of an average pooling layer with filter size = (2) and strides = 2
Global average pooling layer	Filter size 32 and dense layer
Detection	Softmax as the activation function for intrusion detection

Self-Attention

The efficiency of the intrusion detection mechanism is enhanced by providing importance to the prominent features through the self-attention mechanism during the information learning phase. The tan h activation function is utilized in the proposed AdaDenseNet for estimating self-attention and is formulated as,

$$r_t = \tan \mathrm{h}\,(W.k_t) \tag{28}$$

where, the weights are indicated as W and the input feature is indicated as k_t.

Then, the context information $\alpha_{t,i}$ is estimated by comparing the randomly initialized trainable matrix r and the allocation coefficient r_t. It is expressed as,

$$\alpha_{t,i} = \frac{\exp\left(score\left(r_t, r\right)\right)}{\sum_{t=1}^{T} \exp\left(score\left(r_t, r\right)\right)} \tag{29}$$

The feature vector accomplished after self-attention is expressed as,

$$f = \sum_{t=1}^{T} \alpha_t . k_t \tag{30}$$

Thus, intrusion detection using the self-attention features pays more attention to the significant features that enhance the detection accuracy.

3.3.4 Request Drop or Response to the User

The outcome of the proposed AdaDenseNet is that the user is either an intruder or a normal, genuine user. For the intruder request, the response is dropped by alerting the detection of intrusion, and the IP address of the corresponding user request is added to the blacklist table. The response to access the information is provided for the normal genuine user.

4 Results Discussion

The implementation of the proposed MF_AdaDenseNet is performed in PYTHON programming language in Windows 10 OS PC with 8 GB RAM. Here, the analysis is devised with various K-Fold values like 2, 4, 6, 8, and 10 and Epoch 10, 20, 30, 40, and 60. The better outcomes are obtained with the highest K-fold value of 10 and epoch 60. The enhanced result arrived with the highest K-fold value, and the epoch value is presented in this section for various assessment measures. Proposed approach split all the dataset 80% for training and 20% for testing. From five datasets the essential features like 'service_pop3', 'spkts', 'service_ftp', 'dwin', 'dwin', 'trans_depth', 'spkts', 'is_ftp_login', 'ct_state_ttl', 'tcprtt', 'dinpkt', 'service_ftp-data', 'ct_dst_src_ltm', 'swin', 'dwin', 'service_smtp', 'tcprtt', 'ackdat', 'sttl', 'swin', 'synack', 'dinpkt', 'service_ssl', 'service_http', 'sttl', 'su_attempted', 'service_bgp', 'service_vmnet', 'su_attempted', 'service_iso_tsap', 'service_domain', 'num_root', 'service_auth', 'service_eco_i', 'service_klogin', 'service_ssh', 'service_netstat', 'service_eco_i', 'service_netstat', 'srv_diff_host_rate', 'protocol_type_udp', 'protocol_type_icmp', 'srv_serror_rate', 'serror_rate', 'service_link', 'service_pop_2', 'service_ntp_u' 'service_private', 'count', 'service_domain_u', 'service_netbios_dgm', 'service_ntp_u', 'service_red_i', 'flag_SH', 'dst_host_srv_count', 'service_nnsp', 'wrong_fragment', 'service_telnet', 'service_netstat', 'service_uucp', 'service_http', 'service_vmnet', 'service_http_443', 'service_ssh', 'dst_host_rerror_rate', 'service_hostnames', 'service_ntp_u', 'service_kshell', 'src_bytes', 'service_link', 'src_bytes', 'service_remote_job', 'same_srv_rate', 'su_attempted', 'service_smtp', 'proto_IGMP ', 'proto_TCP ', 'flows', 'tcp_syn', 'proto_IGMP ', 'proto_UDP ', 'flows', 'tcp_fin' were selected using optimal value attained by proposed CDO algorithm. The evaluation of the intrusion detection methods is analyzed based on accuracy, precision, recall, F-Measure, and False Alarm Rate (FAR) [39].

4.1 Comparison with Convention Intrusion Detection Algorithms

The proposed MF_AdaDenseNet-based intrusion detection method is compared with conventional algorithms like ELETL (Long et al., 2022) [30], CNN-LSTM (Halbouni et al., 2022) [20], RBP-DT (Srividya and Devi, 2022) [17], MK-ELM (Zhang et al., 2020) [18], LightGBM (Dener et

al., 2022) [21] and CSGO-LSVM (Hemanand et al., 2022) [19]. Initially, the UNSWNB15 dataset is utilized to compare the accomplishment of the MF_AdaDenseNet to other intrusion detection mechanisms. The interpretation based on various assessment measures like accuracy, precision, recall, F-Measure, and FAR of intrusion detection methods is depicted in Table 5. The enhanced outcome of the MF_AdaDenseNet is due to the three-fold detection mechanism with optimal features. The conventional CSGO-LSVM method utilizes optimal feature selection criteria; still, the incapability of handling the large dataset using the LSVM limits the performance. Likewise, the overfitting issues prevail in the LightGBM, instability in the decision-making criteria of the RBP-DT, and the wrong decision-making of the ensemble detection of ELETL make the method acquire minimal performance measures compared to the proposed model. The conventional MK-ELM has the capability of reducing the overfitting issues; still, the incapability to solve the nonlinear data approximation limits the performance of the model. The proposed MF_AdaDenseNet acquired an outstanding outcome compared to the conventional method with minimal computation complexity and enhanced detection accuracy due to the CDO optimal feature selection and MF_AdaDenseNet-based intrusion detection criteria.

Table 5: Comparison of methods using UNSWNB15 dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	92.86	93.94	96.36	92.4	95.38	97.03	98.41
Precision	89.74	94.37	96.75	92.54	95.24	97.64	98.28
Recall	90.8	94.58	96.72	92.19	95.89	97.03	99.13
F-Measure	90.27	94.48	96.74	92.37	95.56	97.34	98.68
FAR	3.07	2.85	1.95	3.03	2.52	1.83	1.25
Time complexity	0.008	0.007	0.006	0.009	0.006	0.007	0.005

Here, the analysis of intrusion detection models depicts the superior outcome of the proposed model in terms of all assessment measures. The accuracy measured by the MF_AdaDenseNet is 98.41%, which is 5.64%, 4.54%, 2.08%, 6.11%, 3.08%, and 1.40% superior compared to CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The reason behind the enhanced outcome is the three-fold intrusion detection stage devised by the MF_AdaDenseNet model. The analysis using the CIDDS-001 is presented in Table 6. The analysis of the proposed method accomplished outstanding performance compared to all conventional intrusion detection methods in terms of all assessment measures.

Table 6: Comparison of methods using the CIDDS-001 dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	93.37	94.44	96.56	92.89	95.88	97.53	98.91
Precision	90.25	94.87	96.95	93.04	95.74	98.14	98.78

(Continued)

Table 6 (continued)

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Recall	91.3	95.08	96.93	92.69	96.09	97.53	99.33
F-Measure	90.77	94.97	96.93	92.86	95.92	97.84	98.83
FAR	2.77	2.55	1.96	2.73	2.22	1.23	0.95
Time complexity	0.006	0.01	0.008	0.009	0.008	0.006	0.004

The CIDDS-001 dataset-based analysis of the intrusion detection models portrays the superiority of the MF_AdaDenseNet model. The time complexity evaluated by the proposed model is 0.004 s, which is 33.33%, 60.00%, 50.00%, 55.56%, 50.00%, and 33.33% superior compared to CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The inclusion of the CDO-based optimal feature selection assists in minimizing the computation burden, which assists in minimizing the time complexity of the proposed model. The analysis using the NSL-KDD Dataset is presented in Table 7. The analysis of the proposed method accomplished outstanding performance compared to all conventional intrusion detection methods in terms of all assessment measures.

Table 7: Comparison of methods using NSL-KDD dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	93.44	94.51	97.44	92.98	96.26	97.61	99.3
Precision	90.32	95.25	97.83	93.12	96.12	98.22	99.17
Recall	91.38	95.15	97.81	93.07	96.67	97.92	99.91
F-Measure	90.85	95.2	97.82	93.09	96.39	98.07	99.68
FAR	3.27	2.55	2.03	3.23	2.72	1.23	1.15
Time complexity	0.006	0.007	0.007	0.006	0.008	0.005	0.004

The analysis of the NSL-KDD dataset based on various measures indicates the superiority of the proposed MF_AdaDenseNet model. The FAR derived by the MF_AdaDenseNet is 1.15, which is 64.83%, 54.90%, 43.35%, 64.40%, 57.72%, and 6.50% superior compared to CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The MF_AdaDenseNet method employs a two-stage approach. The initial blacklist check can potentially eliminate known malicious actors, reducing false positives and improving overall accuracy. The analysis using the cicids2018 is presented in Table 8. The proposed model outperforms the existing techniques. Compare to the previous models proposed MF_AdaDenseNet attain high performance results in terms of 98.72% accuracy, 98.51% for precision, 98.62% recall, 98.12% F-Measure, 1.32% FAR, and 0.004 s time complexity by using cicids2018 dataset.

The analysis using Edge-IIoT dataset is presented in Table 9. The analysis of the proposed method accomplished outstanding performance compared to all conventional intrusion detection methods in terms of all assessment measures.

Table 8: Comparison of methods using CICIDS2018 dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	95.5	96.41	97.23	97.4	97.23	96.41	98.72
Precision	90.72	94.2	95.95	93.74	95.95	94.2	98.51
Recall	91.2	94.32	95.9	94.21	95.9	94.32	98.62
F-Measure	92.64	94.2	95.82	94.25	95.82	94.2	98.12
FAR	2.42	2.32	2.14	3.31	2.14	2.32	1.32
Time complexity	0.008	0.00761	0.00854	0.00921	0.00854	0.00761	0.004

Table 9: Comparison of methods using Edge-IIoT dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	93.25	96.56	97.4	93.95	97.24	96.22	97.66
Precision	88.53	95.12	96.91	93.01	96.19	96.19	97.24
Recall	91.18	95.04	96.81	93.01	95.97	97.4	97.51
F-Measure	91.07	95.13	96.96	93.02	96.49	97.01	97.08
FAR	3.32	2.99	1.93	3.01	2.82	1.41	1.21
Time complexity	0.008	0.007	0.009	0.008	0.007	0.006	0.005

The F-Measure evaluated by the proposed MF_AdaDenseNet is 97.08%, which is 6.19%, 2.01%, 0.12%, 4.18%, 0.61%, and 0.07% superior compared to the CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The use of the CDO algorithm for feature selection could lead to a more focused and informative feature set, potentially improving the model's ability to distinguish between legitimate and intrusive requests. Also, the proposed MF_AdaDenseNet might be effective in learning complex patterns from the data, leading to better classification performance. Thus, the superior outcome is derived from the proposed intrusion detection model. The analysis using the BoT-IoT Dataset is presented in Table 10. The analysis of the proposed method accomplished outstanding performance compared to all conventional intrusion detection methods in terms of all assessment measures.

Table 10: Comparison of methods using BoT-IoT dataset

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	92.34	92.93	93.93	95.94	96.93	97.69	99.01
Precision	85.19	87.16	94.16	95.91	91.97	95.09	98.72
Recall	90.1	94.16	96.62	92.14	95.15	96.53	96.9
F-Measure	90.1	94.16	96.62	92.14	95.15	96.93	98.9
FAR	3.38	2.61	1.1	2.85	2.85	1.19	0.9
Time complexity	0.009	0.006	0.007	0.011	0.009	0.007	0.005

The recall evaluated by the proposed MF_AdaDenseNet is 96.9%, which is 7.02%, 2.83%, 0.29%, 4.91%, 1.81%, and 0.38% superior compared to the CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The use of optimal feature selection could lead to a more focused and informative feature set, potentially improving the ability of the proposed model to identify true positives and reduce false negatives. Thus, higher recall and precision are derived by the proposed method.

4.2 Ablation Study

Fig. 3 illustrates the three-stage intrusion detection process design of the proposed method, which demonstrates its superiority. The ablation study is analyzed in terms of accuracy to determine the role of each module in more precisely detecting intrusions. For example, the proposed MF-AdaDenseNet's accuracy is 99.3%, while CDO+MF-DenseNet, CDO+DenseNet, CDO+AdaDenseNet, Proposed without CDO, and Proposed without blacklist checking procedures estimate 88.35%, 85.253%, 92.46%, 94.36%, and 96.21%, respectively. Thus, the proposed strategy achieves a higher result by combining all of the techniques.

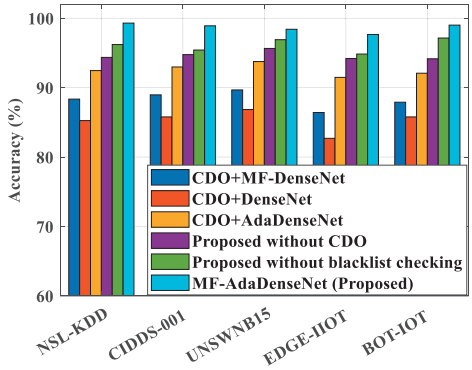

Figure 3: Ablation study

4.3 Accuracy-Loss Analysis

Accuracy-loss analysis is a critical technique used to identify overfitting and overspecialization issues in intrusion detection systems based on training and testing data. In this analysis, the performance of the model is evaluated on both the training and testing datasets. Overfitting occurs when a model learns to fit the training data too closely, capturing noise and irrelevant patterns that do not generalize well to unseen data. Overspecialization, on the other hand, refers to a situation where the model becomes too specific to the training data and fails to generalize to new, unseen data. The Accuracy-Loss analysis for all the five datasets is presented in Fig. 4. Accuracy-loss analysis involves monitoring the model's accuracy on the training and testing datasets during the training process. If the model's accuracy on the training dataset continues to improve while its accuracy on the testing dataset begins to decline or stagnate, this indicates overfitting. Similarly, if the model's accuracy on the training dataset is significantly higher than its accuracy on the testing dataset, it suggests overspecialization.

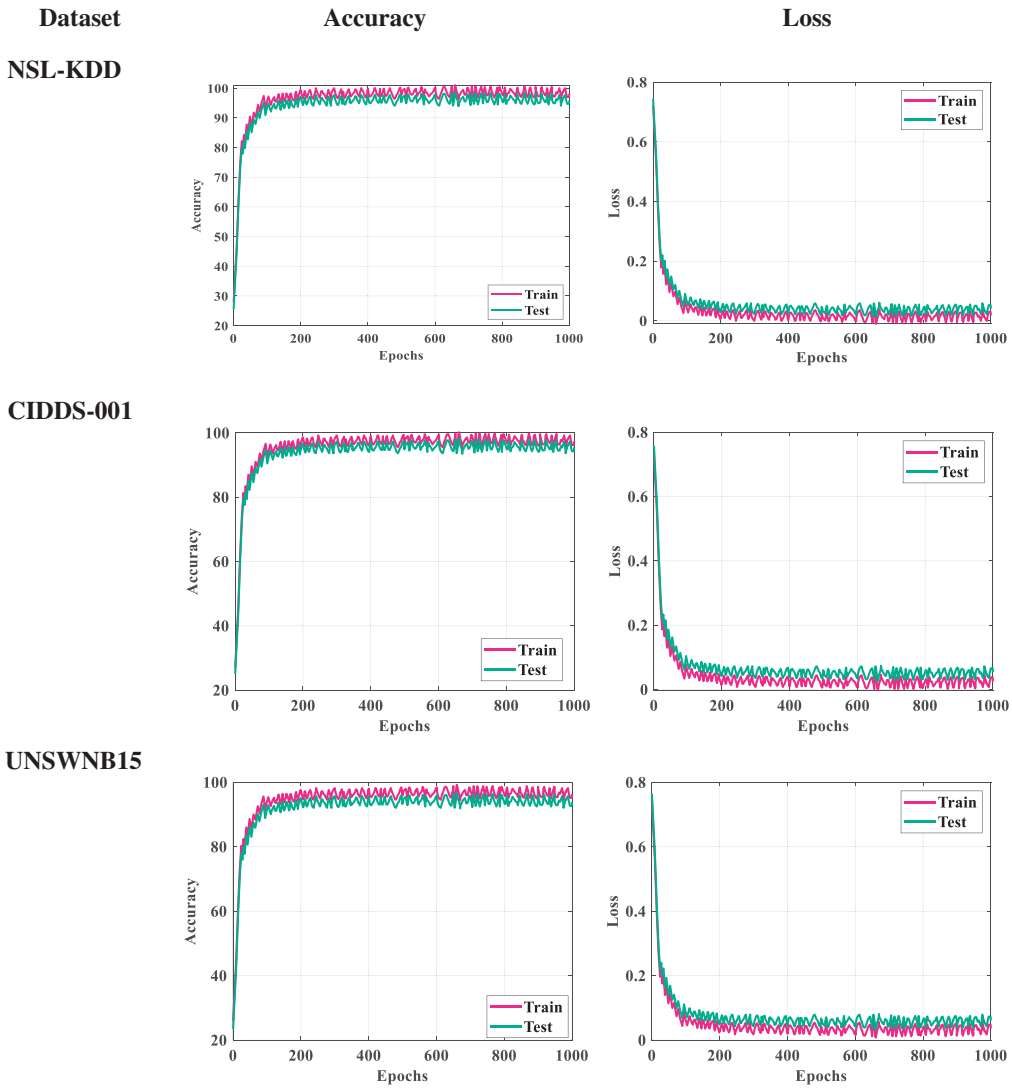

Figure 4: (Continued)

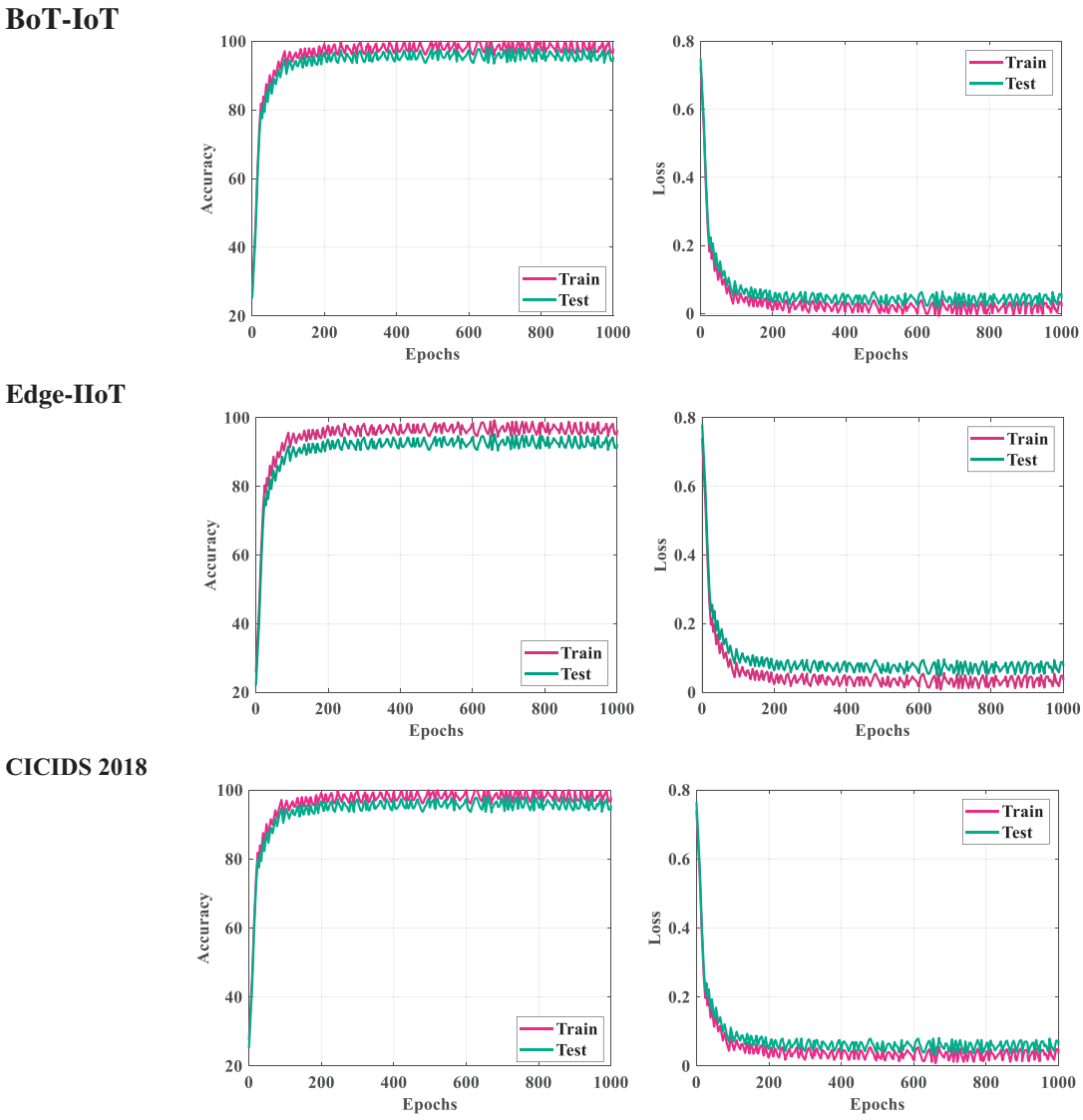

Figure 4: Accuracy-loss analysis

It is observed that the accuracy of the model on the training dataset steadily increases with the number of training iterations, indicating that the model is effectively learning from the training data. However, the accuracy of the testing dataset also shows consistent improvement without any significant decrease or stagnation, which suggests that the model does not fit the training data. Furthermore, the gap between the accuracy of the model on the training and testing datasets remains relatively small throughout the training process, indicating that the model is not over-specialized to the training data and can generalize well to unseen test data.

4.4 Convergence Analysis

The outcome of the proposed method, which is closer to the desired target, depicts the convergence of the optimization. In the proposed intrusion detection mechanism, the optimal best features are

selected using the proposed CDO algorithm that hybrids the capturability behavior of Gannet with the hunting behavior of Dingo. The desired solution of the CDO algorithm depends on the fitness criteria, and the analysis based on the convergence of optimization is depicted in Fig. 5. The convergence rate of the proposed CDO is faster than that of the other conventional DOX and Gannet algorithms. The incorporation of the cupturability behavior of the Gannet search agents helps to capture the target faster, which in turn enhances the convergence rate.

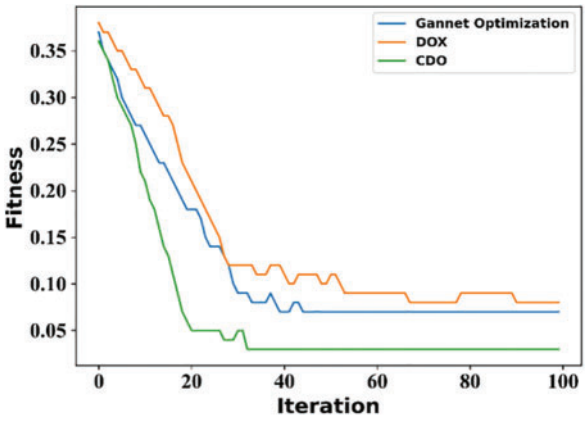

Figure 5: Convergence analysis

4.5 Comparative Discussion

The comparative discussion based on the best outcome of the proposed MF_AdaDenseNet method is depicted in Table 11. The maximal accuracy estimated by the MF_AdaDenseNet is 99.3% using the NSL-KDD dataset, which is 5.90%, 4.82%, 1.87%, 6.36%, 3.06%, 1.70% improved outcomes compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT, and ELETL methods. The maximal precision obtained by MF_AdaDenseNet is 99.17% using the NSL-KDD dataset, which is 8.92%, 3.95%, 1.35%, 6.10%, 3.08%, and 0.96% improved outcome compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT and ELETL methods. The maximal recall obtained by MF_AdaDenseNet is 99.91% using the NSL-KDD dataset, which is 8.54%, 4.76%, 2.11%, 6.85%, 3.24%, and 1.99% improved outcome compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT and ELETL methods. The maximal F-Measure obtained by MF_AdaDenseNet is 99.68% using the NSL-KDD dataset, which is 8.34%, 4.26%, 1.92%, 6.20%, 3.10%, and 1.74% improved outcomes compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT and ELETL methods. The minimal FAR obtained by MF_AdaDenseNet is 0.9% using the NSL-KDD dataset, which is 67.63%, 63.41%, 48.28%, 63.56%, 57.75%, and 26.83% improved outcome compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT and ELETL methods. The minimal Time complexity obtained by MF_AdaDenseNet is 0.004 s using the NSL-KDD dataset, which is 33.33%, 42.86%, 42.86%, 33.33%, 50.00%, and 20.00% improved outcome compared to the conventional CSGO-LSVM, MK-ELM, CNN-LSTM, LightGBM, RBP-DT and ELETL methods.

The comparative discussion depicts the outstanding performance of the MF_AdaDenseNet for all the assessment measures and the three datasets. The proposed CDO algorithm chooses the optimal best features through the balanced diversification and intensification criteria. Besides, the consideration of minimal parameters in the CDO minimizes the computation complexity of feature

selection criteria. The incorporation of the capturability of the Gannet search agent within the Dingo search agent's target surrounding capability eliminates the target escaping capability, which enhances the convergence rate. Also, the consideration of classification error in estimating the fitness of the CDO algorithm minimizes the FAR in intrusion detection. Thus, the CDO-based optimal feature selection helps the MF_AdaDenseNet model enhance detection accuracy. In addition, the proposed MF_AdaDenseNet module with the fuzzy concept and deep learning criteria detects the intrusion more accurately with minimal computation complexity. Thus, the proposed model outperformed other state of art techniques.

Table 11: Comparative discussion

Metrics	CSGO-LSVM	MK-ELM	CNN-LSTM	LightGBM	RBP-DT	ELETL	Proposed
Accuracy	93.44	94.51	97.44	92.98	96.26	97.61	99.3
Precision	90.32	95.25	97.83	93.12	96.12	98.22	99.17
Recall	91.38	95.15	97.81	93.07	96.67	97.92	99.91
F-Measure	90.85	95.2	97.82	93.09	96.39	98.07	99.68
FAR	3.38	2.61	1.1	2.85	2.85	1.19	0.9
Time complexity	0.006	0.007	0.007	0.006	0.008	0.005	0.004

***Big O Notation*:** The computational complexity of the proposed MF_AdaDenseNet-based intrusion detection model is defined as $O(N * \log N) * (M * P * H * W * K_h * K_w)$. In this, N refers to the population size of the CDO algorithm and M refers to the number of feature maps in the current layer. Then, P refers to the number of feature maps from the previous layer. H, W, K_h, and K_w signifies the height of the input, width of input, height of kernel, and width of input kernel, respectively.

Practical Implications

The practical implications of the proposed CDO-based feature selection and three-fold intrusion detection model are:

Enhanced Security: With numerous sensor devices communicating wirelessly, security becomes paramount. The deployment of a robust IDS helps identify and mitigate potential threats.

Efficient Resource Utilization: Traditional IDS methods may consume considerable energy, which is a critical concern in resource-constrained IoT networks. The proposed mechanism aims for energy efficiency by employing optimized feature selection techniques.

Timely Response to Intrusions: By checking incoming requests against a blacklist of known malicious IP addresses, the system can promptly block potential intruders. This proactive approach helps in preventing unauthorized access and potential damage to the network.

Advanced Intrusion Detection Techniques: The use of statistical and one-hot encoding techniques, along with feature selection algorithms like the CDO algorithm, contributes to the effectiveness of intrusion detection. These techniques enable the system to accurately identify and classify intrusions while minimizing false positives.

Scalability and Adaptability: The proposed MF_AdaDenseNet module offers scalability and adaptability, which are crucial for handling the dynamic nature of IoT networks. It can efficiently

process incoming requests and adapt to changing network conditions, ensuring continuous protection against intrusions.

5 Conclusion and Future Work

In this research, a deep learning-based intrusion detection model is introduced to enhance the detection accuracy of WSN-IoT with minimal FAR. The introduced intrusion detection model is a three-fold process, wherein the IP address of the user request is checked in the blacklist table for detecting the intrusion initially. Secondly, MF-based intrusion detection is employed based on fuzzy rules, and finally, AdaDenseNet-based intrusion detection is devised to detect network intrusion. Besides, the optimal feature selection technique named CDO is introduced by hybridizing the capturability of the Gannet optimization and the hunting behavior of the chimp optimization to enhance the feature selection criteria with fast convergence. The combined CDO-based feature selection and MF_AdaDenseNet-based intrusion detection mechanism enhanced the detection accuracy of the model. The outcome of the devised method accomplished the maximal accuracy, precision, recall, F-Measure of 99.46%, 99.54%, 99.91%, and 99.68%, and minimal FAR of 0.9%. However, the FAR needs to be further reduced, which is higher for the proposed MF_AdaDenseNet. Pre-processing incoming requests through statistical and one-hot encoding techniques adds overhead to the system, potentially increasing latency and resource consumption, especially in real-time applications where timely responses are crucial. The IP Black listing approach may impact performance when dealing with legitimate devices that start acting maliciously or when attackers impersonate legitimate IP addresses. To address this concern, we plan to develop a whitelist mechanism in the future that will enable the reuse of IP addresses after verifying that their malicious behavior has ceased, improving the system's flexibility and overall performance. Also, the proposed MF_AdaDenseNet module may face scalability issues as the network grows in size or complexity. In the future, a hybrid deep learning model will be considered for detecting the intrusion of the network. Also, the hyper-parameter tuning of the hybrid deep learning model will be devised to enhance the accuracy of the detection rate with minimal FAR. The complexity of the method will be analyzed in the future to depict the superiority of the model and its applicability in real-world processing application domains by dynamic assigning of IP addresses.

Acknowledgement: The authors extend their appreciation to King Saud University for funding the publication of this research work.

Funding Statement: Authors extend their appreciation to King Saud University for funding the publication of this research through the Researchers Supporting Project number (RSPD2024R809), King Saud University, Riyadh, Saudi Arabia.

Author Contributions: Conceptualization, Chandraumakantham Om Kumar, Suguna Marappan, Mohammed Zakariah, and Sudhakaran Gajendran; methodology, Suguna Marappan, Chandraumakantham Om Kumar, Abdulaziz S. Almazyad, and Sudhakaran Gajendran; software, Suguna Marappan, and Sudhakaran Gajendran; validation, Mohammed Zakariah, Abdulaziz S. Almazyad, and Suguna Marappan; formal analysis, Chandraumakantham Om Kumar, Suguna Marappan, and Sudhakaran Gajendran; investigation, Suguna Marappan, Chandraumakantham Om Kumar, Mohammed Zakariah, Abdulaziz S. Almazyad, and Sudhakaran Gajendran; resources, Suguna

Marappan, and Sudhakaran Gajendran; data curation, Chandraumakantham Om Kumar, and Sudhakaran Gajendran; writing—original draft preparation, Chandraumakantham Om Kumar, and Sudhakaran Gajendran; writing—review and editing, Sudhakaran Gajendran, Mohammed Zakariah, and Chandraumakantham Om Kumar; visualization, Sudhakaran Gajendran, Abdulaziz S. Almazyad, and Chandraumakantham Om Kumar; supervision, Sudhakaran Gajendran, and Chandraumakantham Om Kumar; project administration, Sudhakaran Gajendran, Abdulaziz S. Almazyad, and Chandraumakantham Om Kumar. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this research work are available for free: NSLKDD- https://ieee-dataport.org/documents/nsl-kdd-0, CIDDS-001-; https://www.kaggle.com/dsv/4061966, UNSWNB15-; https://research.unsw.edu.au/projects/unsw-nb15-dataset, CIDDS-2018-; https://www.unb.ca/cic/datasets/ids-2018.html, BoT-IoT-; https://research.unsw.edu.au/projects/bot-iot-dataset, and Edge-IIoT https://ieee-dataport.org/documents/edge-iiotset-new-comprehensive-realistic-cyber-security-dataset-iot-and-iiot-applications (accessed on 30 July 2024).

Ethics Approval: Not applicable.

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

References

[1] D. A. J. Rajan and E. R. Naganathan, "Trust based anonymous intrusion detection for cloud assisted WSN-IOT," *Global Trans. Proc.*, vol. 3, no. 1, pp. 104–108, 2022. doi: 10.1016/j.gltp.2022.04.022.

[2] K. Albulayhi, Q. Abu Al-Haija, S. A. Alsuhibany, A. A. Jillepalli, A. A. Jillepalli and F. T. Sheldon, "IoT intrusion detection using machine learning with a novel high performing feature selection method," *Appl. Sci.*, vol. 12, no. 10, 2022, Art. no. 5015. doi: 10.3390/app12105015.

[3] V. Choudhary, A. Srivastava, A. Kumar, and S. Taruna, "Comparative analysis of security issues and trends in IoT and WSN," *SAMRIDDHI: J. Phys. Sci., Eng. Tech.*, vol. 14, no. 2, pp. 216–222, 2022. doi: 10.18090/samriddhi.v14i02.16.

[4] A. Arshad, Z. Mohd Hanapi, S. Subramaniam, and R. Latip, "A survey of Sybil attack countermeasures in IoT-based wireless sensor networks," *PeerJ. Comput. Sci.*, vol. 7, no. 2, 2021, Art. no. e673. doi: 10.7717/peerj-cs.673.

[5] A. Anand and S. K. Mishra, "An energy aware routing to optimize route selection in cluster based wireless sensor-IoT network (EACW)," *Int. J. Trend Sci. Res. Develop.*, vol. 6, no. 7, pp. 239–246, 2022.

[6] A. Zrelli and T. Ezzedine, "A new approach of WSN deployment, K-coverage and connectivity in border area," *Wirel. Pers. Commun.*, vol. 121, no. 4, pp. 3365–3381, 2021. doi: 10.1007/s11277-021-08881-7.

[7] B. A. Begum and S. V. Nandury, "Data aggregation protocols for WSN and IoT applications-A comprehensive survey," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 35, no. 2, pp. 651–681, 2023. doi: 10.1016/j.jksuci.2023.01.008.

[8] L. Sasirega, "Trust establishment for detecting aggressor nodes and improving route stability in WSN-IoT," *Turk. J. Comput. Math. Educ.*, vol. 12, no. 11, pp. 6012–6020, 2021.

[9] C. Iwendi, P. K. Maddikunta, T. R. Gadekallu, K. Lakshmanna, A. K. Bashir and M. J. Piran, "A metaheuristic optimization approach for energy efficiency in the IoT networks," *Softw.: Pract. Exp.*, vol. 51, no. 12, pp. 2558–2571, 2021.

[10] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, and T. -H. Kim, "Machine and deep learning solutions for intrusion detection and prevention in IoTs: A survey," *IEEE Access*, vol. 10, pp. 121173–121192, 2022.

[11] V. Kathiresan, S. Karthik, P. Divya, and D. P. Rajan, "A comparative study of diverse intrusion detection methods using machine learning techniques," in *2022 Int. Conf. Comput. Commun. Inf. (ICCCI)*, Coimbatore, India, 2022.
[12] S. T. Mehedi, A. Anwar, Z. Rahman, K. Ahmed, and R. Islam, "Dependable intrusion detection system for IoT: A deep transfer learning based approach," *IEEE Trans. Ind. Inform.*, vol. 19, no. 1, pp. 1006–1017, Jan. 2023. doi: 10.1109/TII.2022.3164770.
[13] Z. Wu, H. Zhang, P. Wang, and Z. Sun, "RTIDS: A robust transformer-based approach for intrusion detection system," *IEEE Access*, vol. 10, pp. 64375–64387, 2022.
[14] S. Tsimenidis, T. Lagkas, and K. Rantos, "Deep learning in IoT intrusion detection," *J. Netw. Syst. Manag.*, vol. 30, no. 1, pp. 1–40, 2022. doi: 10.1007/s10922-021-09621-9.
[15] K. Hamza, Y. Himeur, and A. I. Awad, "Deep transfer learning for intrusion detection in industrial control networks: A comprehensive review," *J. Netw. Comput. Appl.*, vol. 220, no. 10, 2023, Art. no. 103760. doi: 10.1016/j.jnca.2023.103760.
[16] A. F. Raslan, A. F. Ali, A. Darwish, and H. M. El-Sherbiny, "An improved sunflower optimization algorithm for cluster head selection in the Internet of Things," *IEEE Access*, vol. 9, pp. 156171–156186, 2021.
[17] P. Srividya and L. N. Devi, "An optimal cluster & trusted path for routing formation and classification of intrusion using the machine learning classification approach in WSN," *Global Trans. Proc.*, vol. 3, no. 1, pp. 317–325, 2022. doi: 10.1016/j.gltp.2022.03.018.
[18] W. Zhang, D. Han, K. -C. Li, and F. I. Massetto, "Wireless sensor network intrusion detection system based on MK-ELM," *Soft Comput.*, vol. 24, no. 16, pp. 12361–12374, 2020. doi: 10.1007/s00500-020-04678-1.
[19] D. Hemanand, G. V. Reddy, S. S. Babu, K. R. Balmuri, T. Chitra and S. Gopalakrishnan, "An intelligent intrusion detection and classification system using CSGO-LSVM model for wireless sensor networks (WSNs)," *Int. J. Intell. Syst. Appl. Eng.*, vol. 10, no. 3, pp. 285–293, 2022.
[20] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022.
[21] M. Dener, S. Al, and A. Orman, "STLGBM-DDS: An efficient data balanced DoS detection system for wireless sensor networks on big data environment," *IEEE Access*, vol. 10, pp. 92931–92945, 2022.
[22] S. Amaouche, A. Guezzaz, S. Benkirane, and M. Azrour, "IDS-XGbFS: A smart intrusion detection system using XGboostwith recent feature selection for VANET safety," *Cluster Comput.*, vol. 27, no. 3, pp. 3521–3535, 2024. doi: 10.1007/s10586-023-04157-w.
[23] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An effective intrusion detection approach based on ensemble learning for IIoT edge computing," *J. Comput. Virol. Hacking Tech.*, vol. 19, no. 4, pp. 469–481, 2023. doi: 10.1007/s11416-022-00456-9.
[24] M. Mohy-eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection," *Multimed. Tools Appl.*, vol. 82, pp. 23615–23633, 2023. doi: 10.1007/s11042-023-14795-2.
[25] S. Amaouche *et al.*, "FSCB-IDS: Feature selection and minority class balancing for attacks detection in VANETS," *Appl. Sci.*, vol. 13, no. 13, p. 7488, 2023. doi: 10.3390/app13137488.
[26] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An intrusion detection model using election-based feature selection and K-NN," *Microprocess. Microsyst.*, 2023, Art. no. 104966. doi: 10.1016/j.micpro.2023.104966.
[27] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrour, and Y. Farhaoui, "An ensemble learning based intrusion detection model for industrial IoT security," *Big Data Min. Anal.*, vol. 6, no. 3, pp. 273–287, 2023. doi: 10.26599/BDMA.2022.9020032.
[28] I. A. Kandhro *et al.*, "Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures," *IEEE Access*, vol. 11, pp. 9136–9148, 2023.

[29] O. D. Okey, D. C. Melgarejo, M. Saadi, R. L. Rosa, J. H. Kleinschmidt and D. Z. Rodríguez, "Transfer learning approach to IDS on cloud IoT devices using optimized CNN," *IEEE Access*, vol. 11, pp. 1023–1038, 2023.

[30] J. Long, W. Liang, K. -C. Li, Y. Wei, and M. D. Marino, "A regularized cross-layer ladder network for intrusion detection in industrial Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 19, no. 2, pp. 1747–1755, Feb. 2023. doi: 10.1109/TII.2022.3204034.

[31] M. T. Nguyen and K. Kim, "Genetic convolutional neural network for intrusion detection systems," *Future Gener. Comput. Syst.*, vol. 113, no. 11, pp. 418–427, 2020. doi: 10.1016/j.future.2020.07.042.

[32] R. Z. Zhao, "NSL-KDD," *IEEE Dataport*, Feb. 2, 2022. doi: 10.21227/8rpg-qt98.

[33] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, "Flow-based benchmark data sets for intrusion detection," in *Proc. 16th Eur. Conf. Cyber Warfare Secur.*, 2017, pp. 361–369.

[34] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *2015 Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Canberra, ACT, Australia, 2015.

[35] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *4th Int. Conf. Inf. Syst. Secur. Privacy (ICISSP)*, Portugal, Jan. 2018. doi: 10.5220/0006639801080116.

[36] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Gener. Comput. Syst.*, vol. 100, no. 7, pp. 779–796, 2019. doi: 10.1016/j.future.2019.05.041.

[37] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.

[38] T. Cho, S. Nam, and M. Shahzad, "GAFS: Genetic algorithm-based filtering scheme for improving detection power in sensor networks," *Int. J. Res.-Granthaalayah*, vol. 3, no. 12, pp. 100–116, 2015. doi: 10.29121/granthaalayah.v3.i12.2015.2894.

[39] D. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv:2010.16061*.