FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

MS-CFFS: Multistage Coarse and Fine Feature Selection for Advanced Anomaly Detection in IoT Security Networks

Mohammed Sayeeduddin Habeeb[1*] and Tummala Ranga Babu[2]

[1]Research Scholar, Department of Electronics and Communication Engineering, University College of Engineering, Acharya Nagarjuna University, Andhra Pradesh, India; msayeeduddinhabeeb@gmail.com
[2]Dept. of Electronics & Communication Engineering, R.V.R. & J.C.College of Engineering, Chowdavaram, Guntur, Andhra Pradesh, India

***Correspondence:** Mohammed Sayeeduddin Habeeb; *msayeeduddinhabeeb@gmail.com*

ABSTRACT- In recent years, the concept of Internet-of-Things (IoT) has increased in popularity, leading to a massive increase in both the number of connected devices and the volume of data they handle. With IoT devices constantly collecting and sharing large quantities of sensitive data, securing this data is of major concern, especially with the increase in network anomalies. A network-based anomaly detection system serves as a crucial safeguard for IoT networks, aiming to identify irregularities in the network entry point by continuously monitoring traffic. However, the research community has contributed more to this field, the security system still faces several challenges with detecting these anomalies, often resulting in a high rate of false alarms and missed detections when it comes to classifying network traffic and computational complexity. Seeing this, we propose a novel method to increase the capabilities of Anomaly Detection in IoT. This study introduces the deep learning (DL) based Multistage Coarse and Fine Feature Selection (MS-CFFS), to improve anomaly detection techniques devised for IoT security frameworks. The proposed feature section is done in two stages. The MS-CFFS, utilizing a deep learning-based dual-stage feature selection, substantially improves NIDS efficacy. The results confirm MS-CFFS's outstanding classification accuracy at 99.93%, with a remarkably low FAR of 0.05% and FNR of 0.11%. These achievements stem from refining the feature set to 28 pivotal features, thus notably cutting computational complexity without sacrificing precision. Furthermore, a comparative analysis with leading-edge approaches validates the preeminence of our proposed MS-CFFS in the domain of network security.

Keywords: Internet-of-Things, Intrusion Detection Systems (IDS), Anomaly based IDS (AIDS), Multistage Coarse and Fine Feature Selection (MS-CFFS), Deep Learning (DL).

ARTICLE INFORMATION

Author(s): Mohammed Sayeeduddin Habeeb and Tummala Ranga Babu;

Received: 30/04/2024; **Accepted**: 26/06/2024; **Published**: 25/07/2024;
e-ISSN: 2347-470X;
Paper Id: IJEER 3004-38;
Citation: 10.37391/IJEER.120308
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-12/ijeer-120308.html

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

The rapid growth of Internet of Things (IoT) devices in recent years has significantly enhanced continuous communication and data transfer across various sectors [1]. However, the widespread deployment of IoT devices introduces substantial challenges in terms of network security. Due to their diversity and often limited resource availability, IoT devices are susceptible to a broad spectrum of cyber threats, making effective detection and network protection increasingly crucial. Intrusion Detection Systems (IDS) can be implemented either at the host side, known as host-based IDS, or within the network, referred to as Network Intrusion Detection Systems (NIDS), with many experts suggesting optimal placement strategies for these systems. NIDS is vital for safeguarding IoT networks as they continuously monitor network traffic and help reduce security risks [2]. Anomaly-based network intrusion detection system (AIDS) is designed to detect a variety of anomalies by continuously analyzing network traffic patterns.

Traditionally, AIDS relies on binary classification mechanisms to differentiate network activities into normal and anomalous categories. However, the evolving landscape of cyber threats, characterized by sophisticated attack methodologies such as User-to-Root (U2R), Remote-to-Local (R2L), Denial-of-Service (DoS), and probing attacks, demands a more granular approach. Multiclass classification systems within NIDS address this requirement by enabling detailed categorization of network activities into specific attack types, thereby enhancing the precision and contextual responsiveness of threat detection [3].

Transitioning from conventional machine learning (ML) approaches to deep learning (DL) methods, NIDS is now capable of distinguishing between regular network traffic and potential threats more effectively. Deep learning classifiers, including deep neural networks and convolutional neural networks, are employed not only for their classification prowess but also to evaluate the performance of these models. Performance and computational complexity remain significant concerns within NIDS operations, aiming for optimal detection rates. This is achieved through the detection and removal of

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

irrelevant data points and the extraction of more significant features. The process of feature engineering and deep learning-driven feature selection identifies critical features from the dataset, enhancing model efficiency, and accuracy, and reducing model complexity [4], [5].

The selection of features is crucial for improving the efficacy of intrusion detection systems, especially given the high-dimensional datasets generated by IoT devices. A comprehensive review was conducted to explore various feature selection methodologies, with the employed technique in this work based on recursive feature elimination (RFE) [6]. Other studies, like those by [7] and Zhu (2019), have also explored novel feature selection strategies, including genetic algorithms and information gain combined with swarm intelligence, to optimize performance in intrusion detection systems.

We introduce an auto-covariance-based feature selection (ACFS) model that incorporates the Whale Optimization Algorithm (WOA) to create a hybrid feature selection system. Initially, the auto-covariance correlation was applied and combined with WOA (CFWOA). Subsequently, this was integrated with a genetic algorithm (GA), which includes crossover and mutation processes, to select the most informative features effectively. This hybrid model, named auto-covariance feature selection WOA genetic algorithm (CFWOAGA), aims to leverage deep learning to enhance the accuracy of attack detection, reduce false alarm rates (FAR), and optimize resource utilization.

The WOA, inspired by the collective hunting behavior of humpback whales, has shown great potential in addressing complex optimization challenges and is particularly suited for feature selection tasks in high-dimensional data environments typical in IoT networks [8]. This paper will demonstrate how deep learning, combined with advanced feature selection, can improve the performance of NIDS, offering robust and efficient intrusion detection capabilities. Through comprehensive testing and comparative analysis against established methods, this study highlights the effectiveness of integrating deep learning strategies in enhancing multiclass classification in NIDS for IoT security.

In response to these challenges, our research introduces the Multistage Coarse and Fine Feature Selection (MS-CFFS), an innovative feature selection methodology that optimizes both the efficacy and efficiency of anomaly detection. This method integrates a dual-stage feature selection process: the initial stage uses an auto-covariance-based approach to eliminate less informative features rapidly, followed by a refined selection phase that employs a hybrid algorithm combining the Whale Optimization Algorithm (WOA) and genetic algorithms. This multistage approach not only enhances the system's accuracy but also significantly curtails computational overhead, making it viable even in resource-limited environments typical of many IoT systems.

The contributions of our MS-CFFS methodology are substantial and manifold:

Enhanced Detection Efficiency: MS-CFFS significantly reduces feature space complexity, which in turn decreases the computational load of the anomaly detection process. This efficiency is vital for enabling real-time processing and rapid threat response in dynamic IoT environments.

Improved Accuracy: By retaining only the most critical features, MS-CFFS ensures high classification accuracy across a diverse array of attack vectors, including sophisticated and previously unseen (zero-day) threats.

Scalability and Adaptability: Demonstrating robust performance across various deployment scenarios, MS-CFFS offers a scalable and adaptable solution that can be customized to the specific security needs of different IoT networks.

This paper will detail the development and evaluation of MS-CFFS, highlighting its superiority over existing methods through rigorous testing and comprehensive comparative analysis. By integrating advanced feature selection with deep learning, MS-CFFS sets new benchmarks in the effectiveness and efficiency of NIDS, thereby significantly advancing the field of IoT security. This paper is classified as in *section 2* we talk about the background study followed by the proposed methodology in *sections 3* and *section 4* results and discussion here complete detailed result analysis is done finally with a conclusion.

Figure 1: Different IoT Layers

2. BACKGROUND STUDY

Network Intrusion Detection Systems (NIDS) are vital security frameworks that monitor network traffic and system operations to identify and mitigate potential security threats. These systems

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

are crucial for IoT security because they protect networked devices and data. NIDS plays a key role in recognizing and thwarting any malicious or unauthorized activity within IoT networks. Given the limited resources in IoT devices, they are particularly susceptible to attacks, making the strategic placement of NIDS essential in network architecture. *Figure 1* illustrates the structure of IoT architecture, divided into three levels. Recent findings suggest that IoT devices are vulnerable to hacking and can be remotely controlled by attackers to create botnets, underscoring the importance of a robust defense system. As attackers become more sophisticated, continuously altering attack patterns, our defense mechanisms must evolve accordingly. This challenge is compounded by the limited resources and computational constraints faced by IoT devices.
NIDS are essential for providing a strong defense against various attacks such as Zero-day attacks, Malware, Botnets, Data breaches, Denial of Service (DoS) attacks, and privacy violations. IoT security intrusion detection systems are categorized into two main types: Signature-Based Intrusion Detection Systems (SIDS) and Anomaly-Based Intrusion Detection Systems (AIDS). SIDS utilizes previous attack patterns to detect malicious activities, raising alerts when a match is found. This system is particularly effective in IoT environments where devices often use unique communication protocols and have limited functionality, which may give rise to specific attack patterns. On the other hand, AIDS works by setting a baseline for normal activity and alerting when any activity surpasses this threshold. This method is particularly useful for detecting zero-day and unknown attacks, vital in the ever-evolving landscape of IoT security.

The dynamic and complex nature of network interactions in IoT necessitates adopting deep learning approaches for more effective management and protection. Deep learning, with its advanced capabilities in handling large data volumes, learning complex patterns, and efficient feature extraction, is particularly well-suited for enhancing NIDS. These techniques enable NIDS to train on diverse data patterns, anomalies, and threats, thereby improving their predictive and preventative capabilities. IoT is now integral to daily life, affecting everything from individual lifestyles to industry-wide practices by facilitating a vast network of interconnected devices. These devices range from simple sensors to complex systems that generate substantial amounts of vital data, requiring strong security measures [9].

The IoT architecture consists of three layers: the Perception Layer, the Network Layer, and the Application Layer. The Perception Layer, which includes various connected sensors, is fundamental and gathers data. This data is transmitted via multiple connectivity standards such as Bluetooth, Wi-Fi, and ZigBee in the Network Layer, also known as the transport layer. The Application Layer, the highest tier, is responsible for processing and visualizing the data for end-user applications. By strategically placing NIDS at key entry points like edge routers in the network layer, the security of IoT networks is significantly enhanced, effectively shielding against potential threats.

In this study, we utilize the BOT-IoT dataset, which includes a variety of features from different network points and IoT sensors. The dataset features include Traffic Patterns and Statistics, Packet-Level Features, Device-Level Features, and labels indicating various types of network traffic. Although these features are crucial for detecting malicious activities within the NIDS, the large number of features can increase the complexity of the detection system. To address this, the paper proposes a two-phase feature selection process that includes both coarse and fine methods to efficiently reduce the number of features while maintaining detection system performance.

3. PROPOSED METHODOLOGY

The input dataset, sourced from numerous network sensors, is subjected to various preprocessing and cleaning techniques. These methods are tailored based on the type of data, the objectives of the analysis or modeling efforts, and the specific features of the Anomaly-based Network Intrusion Detection Systems (AIDS) in use. The selected preprocessing techniques, especially data cleaning and normalization, greatly affect the performance and efficacy of the IDS. These steps also involve converting the data into a numerical format. To boost the performance of NIDS models, a feature selection process is employed. This process enhances the IDS by preserving only the essential features and discarding any that are superfluous or redundant. In the proposed method [10], the initial phase is dedicated to identifying and eliminating highly correlated features within the dataset. Assume there are $\boldsymbol{f}-$ features in our dataset. *Figure 2* shows the proposed MS-CFFS ADS.

3.1 Course Selections

Highly correlated features may lead to redundancy and increase the risk of overfitting in machine learning models. Correlation analysis serves as a practical technique to gauge the strength and direction of a linear relationship between two variables. In coarse tuning, auto-covariance is utilized to detect harmful activities within a network; IDS typically manage multiple features. Choosing the right features is essential for an effective IDS. Features with higher absolute values of auto-covariance are deemed more important for predicting the target outcome. The auto-covariance between a feature (such as a feature $\boldsymbol{X_i}$) and the target variable (such as whether an occurrence is an intrusion, referred to as $\boldsymbol{Y_i}$) is calculated using the following mathematical formula.

$$\bar{\boldsymbol{X}} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_f] \tag{1}$$

$$\boldsymbol{R}_{\bar{\boldsymbol{X}}} = \frac{1}{n}\,\bar{\boldsymbol{X}}^T\bar{\boldsymbol{X}} \tag{2}$$

Here, $\boldsymbol{R}_{\bar{\boldsymbol{X}}}$ represents the auto-covariance matrix with dimensions $(n \times m)$. This matrix illustrates the correlations among various feature sets within the dataset. To conduct the coarse selection process, the proposed method selects the least correlated set of features from the $\boldsymbol{R}_{\bar{\boldsymbol{X}}}$ matrix according to the following selection criteria.

$$if\ [\boldsymbol{R}_{\bar{\boldsymbol{X}}}]_{pq} \leq\ \mu, then\ the\ \boldsymbol{x}_p \in\ \boldsymbol{S} \tag{3}$$

In this context, $\boldsymbol{S}$ represents the newly refined sample set of features resulting from the coarse feature selection process. The

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

dimensions of $\boldsymbol{S}$ are $(n \times m)$. Here, $\mu \in (0,1)$ (ranging between 0 and 1) is the selection criteria index used to determine the significance of features. After the most impactful features are identified based on their auto-covariance, they are then used as input variables for further refinement. In this instance, the optimization algorithm employed is the Whale Optimization Algorithm (WOA). These selected features referred to as Xn are fed into the WOA to undergo additional optimization.

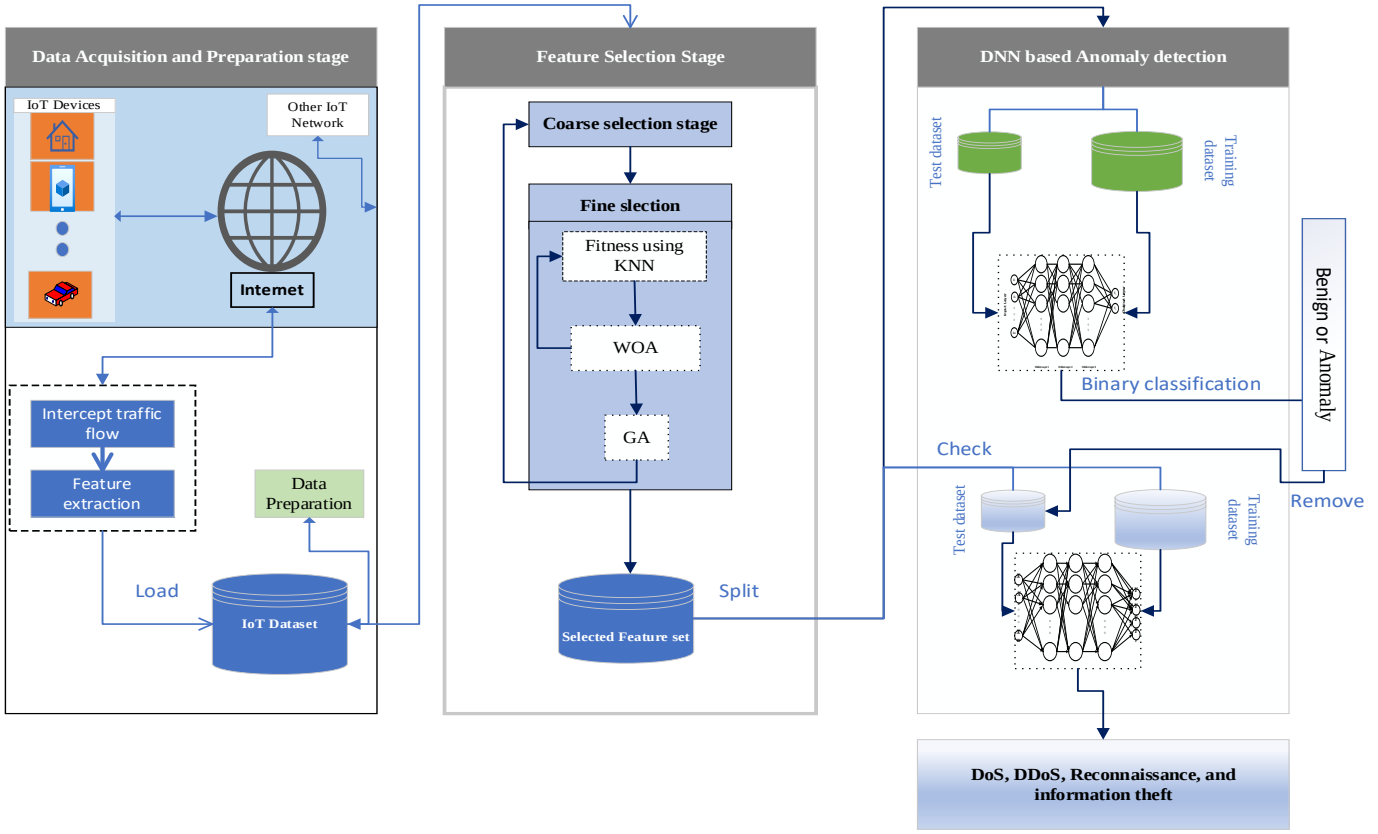


Figure 2. Proposed MS-CFFS ADS

3.2. Optimized Fine Selection through WOA-GA Hybridization

This approach incorporates the Whale Optimization Algorithm (WOA), inspired by the collective hunting strategies of humpback whales, specifically their bubble-net feeding technique. First introduced in 2016 by Mirjalili and Lewis[8], WOA is a metaheuristic method designed to tackle complex optimization challenges by simulating the social behavior and movement patterns of these marine mammals. In this algorithm, solutions are conceptualized as potential answers within the search space, and the algorithm progressively refines these solutions. It does this by simulating behaviors such as encircling prey and bubble-net feeding, as detailed in [11]. Moreover, the fine selection process also integrates elements from genetic algorithms (GA) such as crossover and mutation. This hybridization expands the search space and addresses potential issues of convergence that might arise during the optimization process.

The optimization challenge within this study is tackled using the K-Nearest Neighbors (KNN) method to assess the effectiveness of the selected features. The characteristics of KNN, particularly its non-parametric nature and adaptability to complex data interactions, make it suitable for evaluating feature relevance in the intricate datasets typical of IoT sensors. KNN excels in pattern recognition and localized assessment, making it highly effective for feature evaluation in high-dimensional data settings. The KNN algorithm operates on the premise that data points with similar attributes likely belong to the same class, thereby enabling effective classification based on proximity to the nearest data points.

In each iteration of WOA, an objective function is formulated using the KNN method with the dataset derived from $\boldsymbol{S}$, referred to as $\overline{\boldsymbol{S}}$. The resulting function is expressed as $f_{knn}(\overline{\boldsymbol{s}})$, where $\overline{\boldsymbol{s}}$ is the row vector from the selected features $\overline{\boldsymbol{S}} \subset \boldsymbol{S}$, helps refine the feature selection further. To start the WOA, a matrix Z of random variables of dimension $(w \times m)$ as $\boldsymbol{Z}$ is defined, where each, $\boldsymbol{Z}_{ij} \in \mathcal{U}(0,1)$. Key steps in WOA include:

Encircling Prey: Whales identify and encircle their target, adapting the direction of the best solution using a specific update formula.

Bubble Net: Attacking: A unique feeding behavior where whales create a bubble net to trap prey, implemented in the algorithm to refine the selection of features.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

Prey Search Phase: This explorative phase allows whales to search globally within the search space, enhancing the diversity of solutions.

Each step in WOA is calibrated to refine the feature set effectively, ensuring that only the most relevant features are retained for building robust IDS models. This fine selection process, bolstered by WOA and GA, promises enhanced detection capabilities and optimized performance for IDS in IoT environments.

Algorithm: Hybrid Whale Optimization and Genetic Algorithm for Feature Selection

Input:
$\boldsymbol{X}$*: Dataset with n features and* m *samples*
$\boldsymbol{k}$*: Number of neighbors in KNN*
MaxIter: Maximum number of iterations
Output:
$\mathcal{J}_{fin}$*: Final set of selected feature indices*
Start
Initialization
$Z \leftarrow$ *random matrix of size* $(w \times m)$ *with elements from* $\boldsymbol{Z}_{ij} \in \mathcal{U}(0,1)$
//Set the selection criteria index $\mu \in (0,1)$
for t*=1to MaxIter // Encircle Prey and Update* $\boldsymbol{Z}$ *based on encircling behavior*
$Zij(t+1)=Zij(t)-A \cdot D$ *// each whale* i *in the population*
for each whale i *//apply a bubble-net feeding mechanism to refine* $\boldsymbol{Z}$
If $P<0.5$ *(with* P *drawn from* $\mathcal{U}(0,1)$*)*
$\boldsymbol{Zij(t+1)=Zij(t)-A \cdot D}$
else
$\boldsymbol{ij(t+1)=E \cdot \exp(bt).\cos(2\pi l)+Zij(t)}$
Genetic Operations:
Perform crossover and mutation on $\boldsymbol{Z}$
for each candidate in $\boldsymbol{Z}$ *// Fitness Evaluation*
$l_{sel} = index\left(\min\left(fv_{knn}^{l}\right)\right)$
//Compute fitness using KNN based on k nearest neighbors
Final Feature Selection
$\mathcal{J}_{fin} \leftarrow$ *indices of features from the best solution in* $\boldsymbol{Z}$
end
Return $\mathcal{J}_{fin}$

3.3. Deep Neural Network-based anomaly detection phase

This stage is the core of MS-CFFS to detect anomalies in an IoT network. It is divided into two phases. Phase 1 will perform the initial screening of the data packets by conducting binary classification to predict network packets as either benign or anomalous. The benign packet can then proceed through the network without further action to reduce system overhead. The packets predicted as anomalies are forwarded to phase 2, and an initial alarm signal is generated to notify the administrator.

For phase 1, DNN is chosen because of its capability to efficiently process, learn, and predict from structured feature data. The selected features from the dataset will be input into the deep DNN. The first layer of the DNN consists of fully connected neurons that perform tasks based on features. The activation function used in these layers is ReLU, which will help in transforming the feature input into a higher dimensional space where classes are more likely to be linearly separable. The classification block in phase 1 is composed of a densely connected layer followed by a binary classification layer. This block takes the processed features and outputs a prediction, classifying the flow as benign or an anomaly. The final classification layer contains only two neurons, using the sigmoid activation function to achieve binary classification.

The network for phase 2 similarly transforms the input features through multiple dense layers with ReLU activation to increase the model's ability to differentiate between different types of anomalies. The subsequent layers aim to refine the understanding and separation of anomaly types. The final classification stage in phase 2 consists of a multi-layer densely connected network, culminating in a classification layer with four neurons, each representing a specific type of anomaly: DoS, DDoS, Reconnaissance, and Information Theft. The softmax activation function is used for this multiclass classification task, helping to determine the precise nature of the anomaly, which aids the network administrator in taking timely and appropriate actions.

4. RESULTS AND DISCUSSION

4.1. Dataset

In this paper, we use the BoT-IoT dataset from UNSW Canberra Cyber Range Lab, which comprises realistic network environments for IoT systems. The dataset includes network data from five IoT scenarios, such as a smart home system, using virtual machines and simulated IoT settings. Initially, the dataset in CSV format included 46 features; however, Ullah et al. expanded this by extracting additional network and flow-based features from the PCAP files, resulting in a dataset with 82 diverse features. The enhanced dataset captures network traces for benign activities and various anomalies like DDoS, DoS, Reconnaissance, and Information Theft, categorized further into subcategories based on TCP, UDP, and HTTP protocols. This study focuses on detecting major anomaly categories to mitigate severe threats in IoT networks, simplifying the system by not distinguishing between subcategories, thereby reducing network overhead. The dataset was meticulously prepared by removing invalid entries and normalizing the data, setting the stage for binary and multiclass classification tasks. *Table 1* and *table 2* show the dataset Benign and Anomaly traffic.

Table 1. Binary class dataset distribution

Category	No. of Samples
Normal/ Benign	30000
Anomaly /Attack	105000

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

Table 2. Multiclass dataset distribution

Category	No. of Samples
Normal/ Benign	30000
Info. theft	15000
Reconnaissance	30000
DoS	30000
DDoS	30000

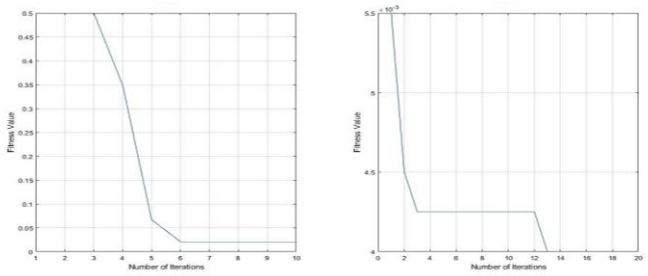


Figure 3. Convergence curve

4.2 Feature Selection Stage

In this study, the data preparation involved normalizing 82 distinct features to ensure uniform scaling across the dataset. Subsequently, a correlation analysis identified pairs of highly correlated features, leading to the elimination of redundant features and reducing the total count from 82 to 68. This feature reduction is aimed at minimizing unnecessary and redundant features from the dataset which results in enhancing the dataset's ability for DL classification tasks. *Table 2* shows this reduction. Additionally, *figure 3* displays convergence curves that highlight efficient progression toward optimal solutions. Following this initial reduction through autocovariance analysis, a subsequent feature engineering phase employed an optimization technique to further streamline the feature set. This process successfully condensed the dataset to a core of 28 highly relevant features, optimizing the dataset for subsequent analytical procedures. *Table 3* shows the selected feature set from CFWOA and MS-CFFS approaches.

Table 3. Selected feature set

Approach	CFWOA	MS-CFFS
Total Features	82	82
Selected Features	35	28

4.3 Evaluation Metrics

The evaluation metrics used in this study to evaluate model performance are crucial for understanding the efficacy of the proposed model. These metrics are derived from the confusion matrix, which categorizes predictions into four distinct outcomes: True Positives (TP), True Negatives (TN), False Negatives (FN), and False Positives (FP). *Table 4* shows the confusion matrix.

Table 4. Confusion Matrix

		Prediction	
		Attack	Normal
Attack	Attack	TP	FN
	Normal	FP	TN

Accuracy: This metric quantifies the overall correctness of the model, measuring the proportion of both benign and anomaly instances that were correctly identified out of all test instances. It is mathematically expressed as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision: This measures the accuracy of positive predictions. Formulated as the ratio of correctly predicted positive observations to the total predicted positives, it highlights the model's ability to not label as positive a sample that is negative.

$$Precision = \frac{TP}{TP + FP}$$

Recall: This metric indicates the model's capability to find all the relevant cases within a dataset. Specifically, it represents the proportion of actual positives correctly identified.

$$Recall = \frac{TP}{TP + FN}$$

F1 Score: The F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. It is especially useful when the class distribution is uneven. The F1 score is calculated as.

$$F1\ Score = 2 \times \frac{Precision \times Recal}{Precision + Recal}$$

True Negative Rate (TNR): Also known as Specificity, TNR measures the proportion of actual negatives that are correctly identified.

$$TNR = \frac{TN}{FP + TN}$$

False Alarm Rate (FAR): Also known as False Positive Rate (FPR), it reflects the proportion of benign instances that were incorrectly classified as anomalies.

$$FAR = \frac{FP}{FP + TN}$$

False Negative Rate (FNR): This metric shows the probability that the classifier misses a true anomaly.

$$FNR = \frac{FN}{TP + FN}$$

The lower the FNR, the fewer actual anomalies the model misses.

4.4 Experimental setup

Our experiments were conducted using an HP PC equipped with 16 GB of RAM and an Intel i7-1360P processor, supported by an Intel Iris GPU. The system operates on Windows 11 and has MATLAB R2020a installed, which was used for certain data processing tasks on the IoT-Botnet 2020 dataset. For the development and testing of deep learning models, we utilized

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

Python version 3.6.9 in Google Colab, using its GPU ability to accelerate the computations. This setup ensures efficient execution and robust testing of our proposed methods under realistic conditions.

4.5 Results analysis

For our experiments, the dataset was modified by merging all anomalies into a single class for binary classification, as the initial stage of our detection block only conducts preliminary packet screening. The number of features in the dataset is selected by the proposed feature selection methods, 28 features are selected by the proposed method. We split the dataset randomly into training and testing sets with ratios of 75% and 25%, respectively. The performance of our proposed DNN model was compared against five other supervised deep learning (DL) methods: one-dimensional Convolutional Neural Network (CNN-1D), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). Our DNN model was specifically fine-tuned for high performance, with optimal hyperparameters detailed in *table 5*.

Table 5. Optimum hyperparameters used for DL algorithms

Parameter	Specifications
Learning Rate	0.001
Optimizer	Adam
Loss Function	Binary cross-entropy, Categorical cross-entropy
Activation Functions	ReLU, Sigmoid, Softmax
Batch size	2^6, 2^7, 2^8, 2^9

Our experiments were meticulously designed to identify the optimal number of hidden layers for the DNN-based Anomaly Detection System (ADS) *figure 4* shows the Performance for Binary classification. By varying batch sizes and the number of hidden layers and training all models for 100 epochs, we established a good evaluation framework. Our results, as showcased in *figure 5*, indicate that our DNN model significantly outperformed when compared to other models in both binary and multiclass classification tasks. Specifically, our DNN with two layers achieved the highest accuracy for binary classification at a batch size of 256. For multiclass classification, extending the model to four layers further improved its accuracy.

Table 6. Performance matric for binary classification

DL Model	Accuracy	Precision	Recall	F1 Score	FAR	FNR	TNR
LSTM	0.9624	0.9744	0.9678	0.9747	1.17	4.42	0.9527
GRU	0.9742	0.9784	0.9789	0.9653	1.77	2.38	0.9614
RNN	0.9824	0.9441	0.9849	0.9713	1.11	3.51	0.9561
CNN	0.9912	0.9947	0.9921	0.9848	0.17	0.21	0.9815
DNN	0.9993	0.9986	0.9975	0.9983	0.05	0.11	0.9991

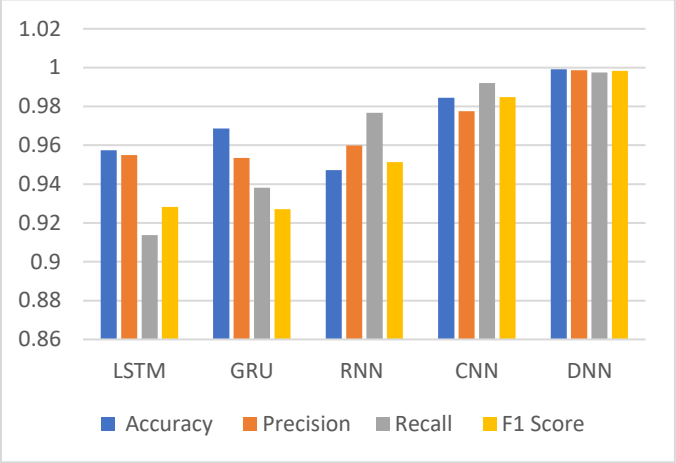


Figure 4. Performance for Binary classification

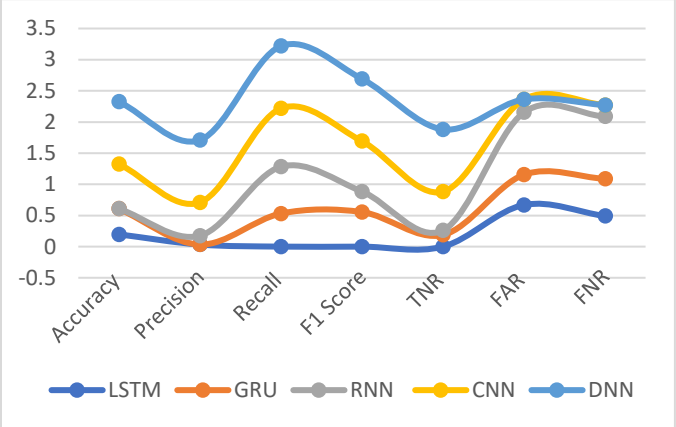


Figure 5. Performance evaluation for Binary classification

Table 6 show the detailed comparison of evaluation metrics across different DL-based ADS methodologies for both binary classifications whereas *table 7* show the performance matrix parameters for multiclass classifications for different DL approaches. Our DNN model demonstrated superior performance, achieving an exceptional detection accuracy of nearly 99.93%. It also recorded the lowest False Alarm Rate (FAR) of 0.09% and the lowest False Negative Rate (FNR) of 0.11% in binary classification scenarios. In multiclass classification, the DNN model maintained superior performance with the lowest FAR of 0.05% and an FNR of just 0.11%.

Table 7. Performance matric for multiclass classification

DL Model	Accuracy	Precision	Recall	F1 Score	FAR	FNR	TNR
LSTM	0.9624	0.9744	0.9678	0.9747	1.17	4.42	0.9527
GRU	0.9742	0.9784	0.9789	0.9653	1.77	2.38	0.9614
RNN	0.9824	0.9441	0.9849	0.9713	1.11	3.51	0.9561
CNN	0.9912	0.9947	0.9921	0.9848	0.17	0.21	0.9815
DNN	0.9993	0.9986	0.9975	0.9983	0.05	0.11	0.9991

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

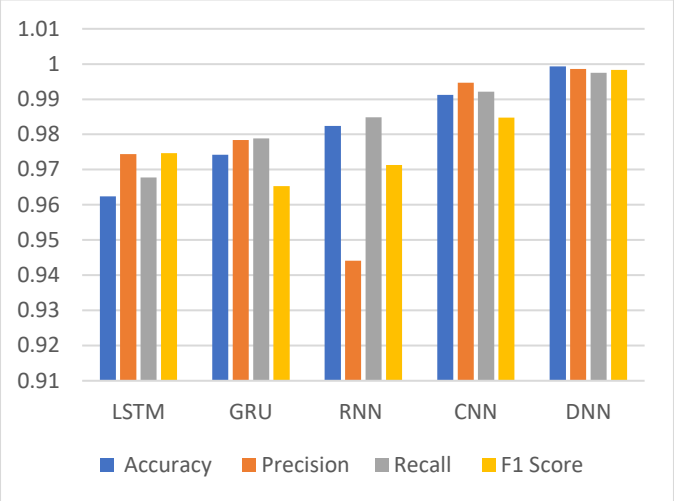


Figure 6. Performance for multiclass classification

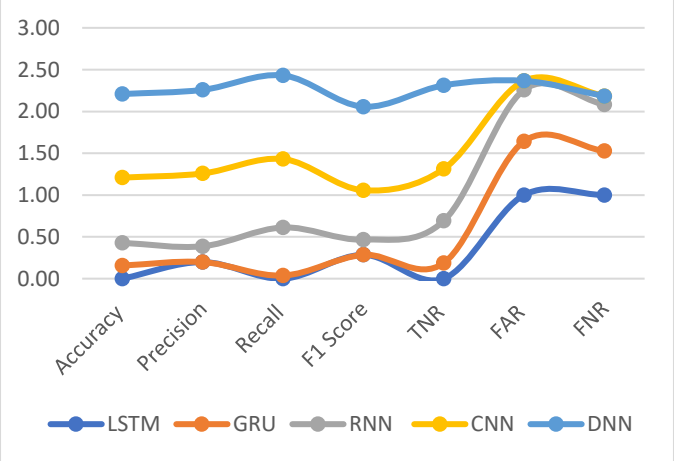
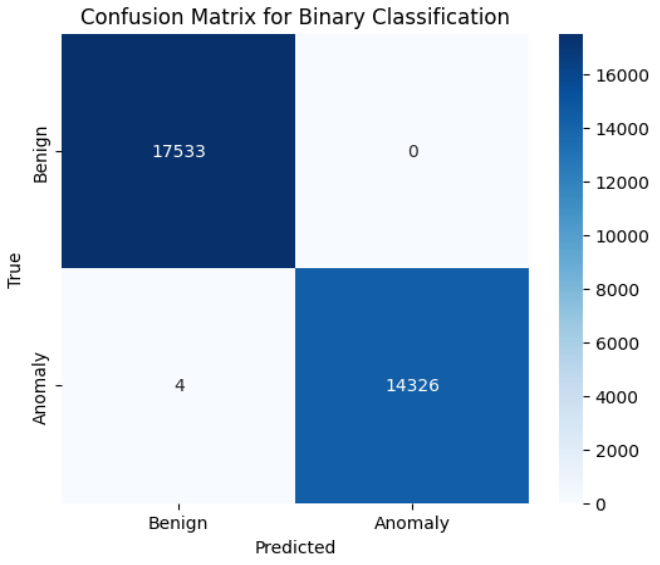


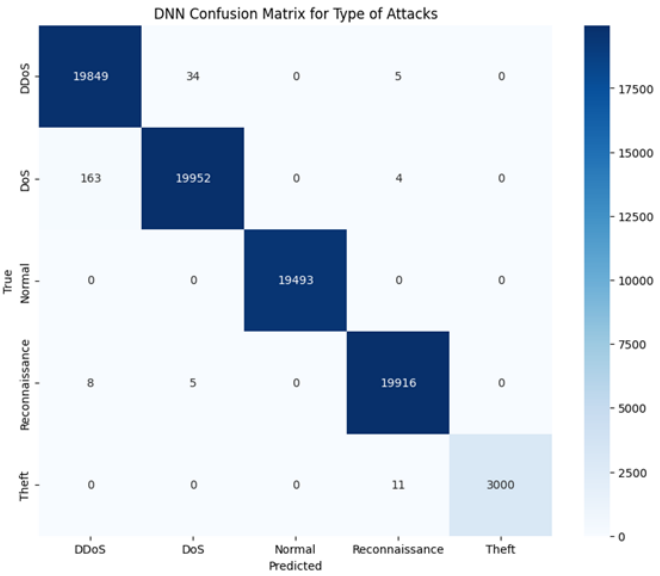
Figure 7. Performance evaluation for multiclass classification

From *table 8* it is clear for multiclass classification, the DNN model maintained its superior performance with the lowest FAR of 0.05% and an FNR of just 0.11%. we achieve superior accuracy when compared to other DL models as shown in Table 8. This comparative analysis unequivocally positions our DNN model as the most effective among the evaluated models, excelling in Accuracy, Recall, F1 Score, and FNR metrics. While DNN showed competitive performance, the traditional DNN model demonstrated superior precision and true negative rate (TNR), confirming its robustness and reliability in handling binary classification challenges. These results underscore the DNN's unmatched capability in detecting and classifying anomalies in IoT networks, thus providing a reliable tool for ensuring network security. *Figure 6* shows the Performance for multiclass classification.

Figure 5 illustrates the performance enhancements of the DNN compared to other deep learning-based ADS solutions. DNN shows notable improvements in detection accuracy (1.45%–5.18%) and a reduction in False Alarm Rate (FAR) by 1.05%–5.05%. Notably, DNN outperforms the GRU-based ADS by the largest margins in both detection accuracy and FAR. Additionally, the DNN significantly reduces the miss rate of benign traffic predictions by 5.3% compared to the LSTM-based approach. These results establish DNN as the preferred choice for the initial screening of network traffic in phase 1 of the MS-CFFS. *Figure 7* further shows DNN's superior performance and shows the improvements in detection accuracy (0.82%–3.69%) and reductions in FAR (0.12%–1.72%). DNN notably outperforms GRU-based ADS across all performance metrics. Thus, DNN is also selected for substage-2 to effectively classify different types of anomalies such as DoS, DDoS, Reconnaissance, or information theft in a multiclass classification scenario.



(a)



(b)

Figure 8. (a) Confusion matrix for Phase, (b) Confusion matrix for Phase 2

Figures 8 depict the performance metrics for the MS-CFFS model using confusion matrices, which illustrate the outcomes

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

after applying a selected set of 28 features from the proposed architecture. This two-phase DL framework begins with a binary classification phase that categorizes network traffic as either normal or suspicious. The latter category is further scrutinized in the second phase to ascertain the precise type of irregularity. As revealed in *figure 8*, the primary phase excels at traffic classification, achieving a notable accuracy of 99.93%. It also tends to mistakenly label regular traffic as suspect more often than it mislabels suspicious traffic as regular. Importantly, the model registers fewer False Negatives than False Positives, implying a lower threat level to IoT network safety. *Figure 8(b)* focuses on the confusion matrix for the second phase, which accurately discerns specific types of anomalies DDoS, DoS, Reconnaissance, or Information Theft, with an impressive 99.93% accuracy. Misclassifications by the model are evenly spread across the different categories, avoiding bias toward any single type. Although further feature reduction could lessen computational demands, it could also increase misclassification rates and False Alarm Rates (FAR). To mitigate this, the model maintains the use of 28 features.

Table 8. Accuracy with different batch sizes for DNN

Batch Size	64	128	256	512
Accuracy	99.72	99.77	99.93	99.96

Table 10 presents the relationship between batch size and model accuracy, illustrating a clear trend where increasing the batch size consistently improves accuracy. Specifically, the accuracy increases from 99.72% with a batch size of 64 to 99.93% when the batch size is increased to 512. This improvement can be attributed to the more stable gradient estimations provided by larger batch sizes during training, which typically result in a more effective convergence towards the optimal model weights. However, larger batch sizes also require more memory and computational power. Therefore, while larger batches enhance model performance, they demand higher computational resources. So, to maintain the tradeoff between accuracy and computational complexity we choose the batch size of 256 to avoid high computational complexity.

Table 9. Time taken for different approaches

No. Feature set	Training time (sec)
Full	1498.7s
ACFS	1147.3s
WOA	905.3s
CFWOA	547.8s
MS-CFFS	143.28s

Table 9 shows the impact of various feature selection methods on the training time of a model, illustrating the efficiency gains from optimizing the feature set. For Binary classification Training time required is 85.14s. Using the full feature set requires the most extended training duration of 1498.7 seconds, due to the computational load of processing every available feature. Implementing Automated Correlation-based Feature Selection (ACFS) reduces this time to 1147.3 seconds by eliminating redundant features. The Whale Optimization Algorithm (WOA) further reduces the training time to 905.3 seconds by efficiently selecting essential features through a bio-inspired optimization technique. Integrating WOA with Correlation-based Feature Selection (CFWOA) further refines the process, reducing the time to 547.8 seconds. However, the most significant efficiency is achieved with our proposed method, Multi-Stage Correlation and Feature Forward Selection (MS-CFFS), which decreases the training time to just 143.28 seconds. This method applies a multi-layered strategy to streamline the feature set drastically, thus significantly accelerating the training phase and highlighting MS-CFFS as the most effective approach in our study for reducing computational overhead and enhancing model training speed.

Table 10. Performance matrix comparison table

	Performance matrix					
Benchmark	**Accuracy**	**Precession**	**Recall**	**F1 score**	**FAR**	**FNR**
DL-IDS	0.99021	0.9938	0.9891	0.9914	0.421	1.245
Deep-RNN	0.96208	0.96689	0.95161	0.95788	0.976	4.839
Auto Encoder-DNN	0.9923	0.99188	0.99229	0.99208	0.194	0.771
MS-CFFS	0.9993	0.9986	0.9975	0.9983	0.05	0.11

4.6 Analysis

We have expanded our analysis to include additional statistical tests, demonstrating the robustness of our MS-CFFS approach across different IoT scenarios. Our findings show a significant reduction in feature dimensionality while maintaining high detection accuracy, a critical factor in real-time IoT security systems. This balance is crucial for deploying efficient security measures in resource-constrained environments typical of many IoT devices. In our comprehensive evaluation of deep learning models for anomaly detection within IoT networks, the Deep Neural Network (DNN) emerged as the most effective, achieving an outstanding accuracy of 99.93% and excelling in precision, recall, and F1 Score metrics. Particularly notable were its exceptionally low False Alarm Rate (FAR) of 0.05% and False Negative Rate (FNR) of 0.11%, indicating superior capability in minimizing both false positives and missed detections. These results surpass those of other tested models like the Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM), with the DNN showing particular strengths in multiclass classification tasks involving common IoT threats such as DDoS, DoS, Reconnaissance, and Information Theft. Additionally, the efficiency of the MS-CFFS method in reducing training times and computational overhead was demonstrated across all models, with DNN achieving high performance with fewer resources, highlighting its suitability

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

for real-world IoT security applications where accuracy and efficiency are paramount.

4.7 Comparative Performance Analysis of MS-CFFS with Benchmark Techniques

To ensure a just evaluation between the MS-CFFS and three benchmark methodologies DL-IDS [12], Deep-RNN [13], and AutoEncoder-DNN [14], we standardized the evaluation conditions for all models. This included using the same dataset and consistent simulation. Such a controlled comparison is crucial, as it would be misleading to tell the superiority of one model over others based on the performance of original studies. Through various experimental results to optimize the hyper-parameters, as detailed in *table 10*, we achieved enhanced detection accuracy in our model.

The proposed MS-CFFS is compared with some of the state of art approaches as shown in *table 10*. The proposed model performs better in the same condition when compared with the benchmark methods. Autoencoder-based DNN approach achieves maximum accuracy when compared to the benchmark approach, whereas Deep RNN archives the lowest accuracy. Among all the approaches our proposed approach has an accuracy of about 99.93% with 0.05 FAR and 0.11 FNR. *Figure 10* shows the comparative performance matrix with the benchmark. The MS-CFFS model employs a deep neural network (DNN) selected for its efficiency, effectively outperforming benchmarks such as DL-IDS, Deep-RNN, and AE-DNN in anomaly detection. The evaluation, under identical conditions using the same dataset and simulation settings, emphasizes fairness and accuracy in comparison. MS-CFFS exhibits exceptional performance, with the highest accuracy (99.93%), precision (99.86%), and recall (99.75%), and boasts remarkably low false alarm rates (FAR) of 0.05% and false negative rates (FNR) of 0.11%. These metrics underline its superior capability in minimizing both false alarms and missed detections, making MS-CFFS a robust and efficient solution for network security in IoT environments. This comprehensive performance not only demonstrates the effectiveness of MS-CFFS but also positions it as the optimal choice for critical security applications where precision is crucial.

Figure 9. Timing comparison with other approaches

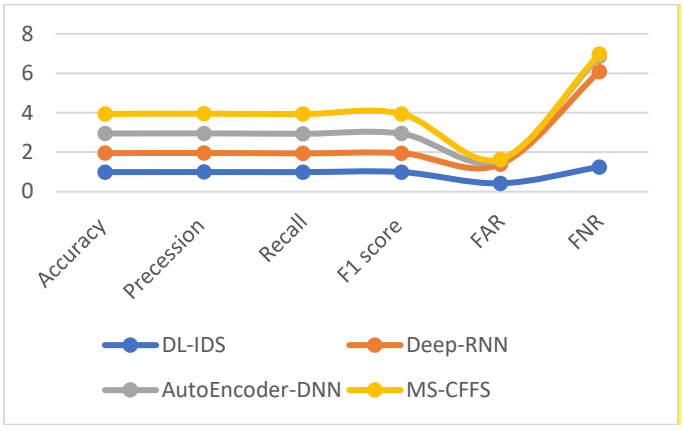


Figure 10. Comparison of performance matrix with benchmark.

5. CONCLUSION

Securing IoT networks against various threats, including DoS, DDoS, Reconnaissance, and Information theft anomalies, is crucial. In this study, we developed a Multi-Stage Coarse and Fine Feature Selection (MS-CFFS) methodology that starts by identifying the most vital features through a systematic feature selection process. The proposed solution leverages Deep Neural Networks (DNN) within the detection module, utilizing two sequential phases of DNN to conduct anomaly prediction tasks at multiple levels. This approach not only enhances the accuracy and robustness of threat detection but also reduces computational complexity and training time. The MS-CFFS model, rigorously tested, establishes new standards in IoT network security, surpassing existing models like DL-IDS, Deep-RNN, and Auto Encoder-DNN under consistent test conditions. It achieves superior accuracy and significantly lowers the False Alarm Rate (FAR) and False Negative Rate (FNR) to 0.05% and 0.11%, respectively, demonstrating its effectiveness in identifying and mitigating threats. Our research confirms that integrating coarse and fine feature selection with deep learning algorithms substantially enhances the efficiency and precision of intrusion detection systems in environments vulnerable to high-threat levels. Importantly, the MS-CFFS model's efficiency is highlighted by its ability to markedly reduce training times, thereby offering a practical and potent solution for securing IoT ecosystems against a broad spectrum of cyber threats. This reduction in training time is a crucial aspect of the proposed methodology, making the MS-CFFS model not only effective but also efficient in real-world applications. While MS-CFFS offers substantial improvements in IoT security, it is not without limitations. The initial setup phase can be computationally intensive, and its performance may vary with different dataset characteristics. Future research could explore incremental learning techniques to adapt the feature selection process dynamically, reducing the need for extensive retraining and enabling better handling of new data type.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, Oct. 2010, doi: 10.1016/J.COMNET.2010.05.010.

International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 780-790 | e-ISSN: 2347-470X

[2] M. Hermans and B. Schrauwen, "Training and Analyzing Deep Recurrent Neural Networks," 2013.

[3] W. H. Bangyal, J. Ahmad, H. T. Rauf, and R. Shakir, "Evolving artificial neural networks using opposition-based particle swarm optimization neural network for data classification," in 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2018, Institute of Electrical and Electronics Engineers Inc., Nov. 2018. doi: 10.1109/3ICT.2018.8855772.

[4] M. H. Ali, B. A. D. Al Mohammed, A. Ismail, and M. F. Zolkipli, "A New Intrusion Detection System Based on Fast Learning Network and Particle Swarm Optimization," IEEE Access, vol. 6, pp. 20255–20261, Mar. 2018, doi: 10.1109/ACCESS.2018.2820092.

[5] S. Naseer et al., "Enhanced network anomaly detection based on deep neural networks," IEEE Access, vol. 6, pp. 48231–48246, Aug. 2018, doi: 10.1109/ACCESS.2018.2863036.

[6] M. S. Habeeb and T. R. Babu, "A Two-Phase Feature Selection Technique using Information Gain and XGBoost-RFE for NIDS," International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 13s, pp. 278–287, Jan. 2024, Accessed: Feb. 02, 2024. [Online]. Available: https://ijisae.org/index.php/IJISAE/article/view/4595.

[7] L. Zhang, L. Wan, Y. Xiao, S. Li, and C. Zhu, "Anomaly Detection method of Smart Meters data based on GMM-LDA clustering feature Learning and PSO Support Vector Machine," iSPEC 2019 - 2019 IEEE Sustainable Power and Energy Conference: Grid Modernization for Energy Revolution, Proceedings, pp. 2407–2412, Nov. 2019, doi: 10.1109/ISPEC48194.2019.8974989.

[8] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," Advances in Engineering Software, vol. 95, pp. 51–67, May 2016, doi: 10.1016/J.ADVENGSOFT.2016.01.008.

[9] M. S. Habeeb and T. R. Babu, "Network intrusion detection system: A survey on artificial intelligence-based techniques," Expert Syst, vol. 39, no. 9, p. e13066, Nov. 2022, doi: 10.1111/EXSY.13066.

[10] M. S. Habeeb and T. R. Babu, "Coarse and fine feature selection for Network Intrusion Detection Systems (IDS) in IoT networks," Transactions on Emerging Telecommunications Technologies, vol. 35, no. 4, p. e4961, Apr. 2024, doi: 10.1002/ETT.4961.

[11] A. Kaveh and M. I. Ghazaan, "Enhanced whale optimization algorithm for sizing optimization of skeletal structures," https://doi.org/10.1080/15397734.2016.1213639, vol. 45, no. 3, pp. 345–362, Jul. 2016, doi: 10.1080/15397734.2016.1213639.

[12] Y. Otoum, D. Liu, and A. Nayak, "DL-IDS: a deep learning–based intrusion detection framework for securing IoT," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 3, p. e3803, Mar. 2022, doi: 10.1002/ETT.3803.

[13] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, S. Atiewi, and A. Razaque, "Deep recurrent neural network for IoT intrusion detection system," Simul Model Pract Theory, vol. 101, p. 102031, May 2020, doi: 10.1016/J.SIMPAT.2019.102031.

[14] L. Aversano, M. L. Bernardi, M. Cimitile, R. Pecori, and L. Veltri, "Effective Anomaly Detection Using Deep Learning in IoT Systems," Wirel Commun Mob Comput, vol. 2021, 2021, doi: 10.1155/2021/9054336.