ISSN: 2093-5374 / E-ISSN: 2093-5382

Smart Memory Augmented Neural Network for Anomaly-Based Intrusion Detection System in IoT

Veena Potdar[1*], Dr. Mohan Govindasa Kabadi[2], and Dr.N. Dayanand Lal[3]

[1*]Department of Computer Science and Engineering, Dr. Ambedkar Institute of Technology Bangalore, India. veenapotdar@gmail.com; vpotdar@gitam.in, https://orcid.org/0000-0003-3006-688X

[2]Department of Computer Science & Engineering, GITAM University, Bengaluru, India. mkabadi@gitam.edu, https://orcid.org/0000-0002-9975-1773

[3]Department of Computer Science & Engineering, GITAM University, Bengaluru, India. dnarayan@gitam.edu, https://orcid.org/0000-0003-3485-9481

Received: September 22, 2025; Revised: November 14, 2025; Accepted: December 17, 2025; Published: March 31, 2026

Abstract

The Internet of Things (IoT) has enormously developed and is utilized in diverse applications such as healthcare, transportation, military, and agriculture. However, this increasing trend and proliferation of smart objects also make them highly susceptible to malicious attacks. Thus, an anomaly-based Intrusion Detection System (IDS) is employed to prevent attacks by classifying network behavior as normal or anomalous. However, existing IDSs fail to preserve long-term dependencies, and redundant features in the network traffic led to miscalculation. To address these issues, a Smart Memory Augmented Neural Network (SMANN) is developed to observe and remember long-term dependencies during detection by incorporating a memory augmentation framework into the Long Short-Term Memory (LSTM). Furthermore, feature selection is performed using the proposed Fast Flying Particle Swarm Optimization (FFPSO) for selecting highly relevant features by avoiding the problem of oscillation. The separation between the anomalous and non-anomalous patterns of data is ensured by using the fast AnoGAN (f-AnoGAN). To confirm the efficiency of FFPSO-SMANN, the UGR-16, UNSW, NSL-KDD, and CICIDS 2018 dataset are used for assessing and classification analysis. The FFPSO-SMANN is analyzed based on accuracy, precision, recall, and F1-score. The accuracy of FFPSO-SMANN is 99.95% on the NSL-KDD dataset, which is superior to existing methods.

Keywords: Anomaly Based Intrusion Detection System, Feature Selection, Internet of Things, Fast Flying Particle Swarm Optimization, Smart Memory Augmented Neural Network.

1 Introduction

IoT represents the extensive utilization of intelligent objects equipped with different modules such as processors, captors, and activators for communication (Douiba et al., 2023; Priyanka et al., 2023). An important operation of IoT devices, where the devices are controlled and communicate through the Internet (Lu et al., 2021; Sugitha et al., 2022). The collected data are used to develop intelligent

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 17, number: 1 (March - 2026), pp. 101-119. DOI: *10.58346/JOWUA.2026.I1.007*

*Corresponding author: Department of Computer Science and Engineering, Dr. Ambedkar Institute of Technology, Bangalore, India.

decision-making algorithms and efficiently maintain IoT settings (Rahaim et al., 2024; Saba et al., 2022). IoT supports various applications and services such as industrial monitoring, smart healthcare, environmental protection, smart homes, and public security (Ma et al., 2023; Nandanwar & Katarya, 2024). IoT devices are susceptible to attackers and noncriminals because of insufficient security measures (Jayasankar et al., 2024; Ayesh, 2024). Anomalies in network performance led to malicious activities including intrusion attempts, system errors, configuration errors, and Distributed Denial of Service (DDoS) attacks (Antonius et al., 2023).

An Intrusion Detection System (IDS) identifies network traffic attacks and alerts the network administrator with the source IP, destination IP, and other attack types (Keserwani et al., 2021; Biju & Wilfred Franklin, 2024; Prabu & Sudhakar, 2024; Keserwani et al., 2021). The utilization of signature-based IDS is effective in detecting known network attacks (Lazzarini et al., 2023). In anomaly-based IDS, variations in traffic patterns are analyzed to evaluate incoming traffic and compute anomaly probability, even for unidentified attacks (Rohini et al., 2023; Otoum & Nayak, 2021). It also has the capability to detect zero-day attacks that cannot be identified using a signature-based system (Alghanam et al., 2023). In this research, a memory-augmented Smart Memory Augmented Neural Network (SMANN) is developed along with Fast Flying Particle Swarm Optimization (FFPSO)-based feature optimization for effective intrusion classification (Veerasamy et al., 2023; Prabu & Sudhakar, 2022.

The main contributions of this research are summarized as follows:

- Highly informative features are selected by FFPSO by avoiding oscillation during the search process using a single guide vector approach and Adaption of Search Dimension (ASD) strategy.
- F-AnoGAN effectively separates anomalous and non-anomalous patterns from the features selected by FFPSO.
- The incorporated memory augmentation in SMANN effectively observes and remembers long-term dependencies.
- To analyze the generalization of FFPSO-SMANN, it is evaluated using four datasets: UGR-16, NSL-KDD, UNSW, and CICIDS 2018.

The remaining sections of this research are arranged as follows: related works are discussed in Section 2. the proposed FFPSO-SMANN method is presented in Section 3, experimental results are detailed in Section 4, and conclusion is provided in Section 5.

2 Related Work

To enhance the capacity for attack mitigation, several AI-based and feature selection-based approaches have been developed in recent years.

2.1. Recent Work on Optimization Approaches for Choosing Features

Zareh Farkhady et al., (2025) developed the Binary Chimp Optimization algorithm (BCHO) to select features, where a new local search operator was introduced to avoid the issue of local optima. (Ammannamma & Chakravarthy, 2025), developed the modified Gazelle Optimization algorithm (Mod-GO) together with the Conventional Ghost-net-based Squeeze-Excited Deep-Scale Capsule Network (CGN_SEDSCapsNet). However, this classifier required an additional optimizer, the Enhanced Artificial Humming Bird (EAH) algorithm, for parameter optimization. Anchuri et al., (2025) implemented the Niche-Strategy-based Gorilla Troops Optimization (NSGTO) for feature selection, while LSTM was used for classification. Makhadmeh et al., (2025) proposed an improved version of the

Marine Predators Algorithm with Crossover (MPAC) operator to identify highly relevant features. Furthermore, hybrid optimization combining the Bald Eagle Search (BES) and Butterfly Optimization Algorithm (BOA) was developed by Biju & Franklin, (2025), and the Improved Spotted Hyena Optimization (ISHO) and Honey Badger Algorithm (HBA) were developed by Saikam & Ch, (2024). In (Biju & Franklin, 2025), BOA and BES were employed to acquire both packet- and flow-level features for enhancing classification using a multi-head attention-based bidirectional gated recurrent unit (MHA-BiGRU). Moreover, the effective ensemble of Improved Spotted Hyena Optimization (ISHO) and Honey Badger Algorithm (HBA) (Saikam & Ch, 2024) addressed data imbalance and over-fitting risk. Classification was ensured by the Squeeze-and-Excitation (SE)-Deep Residual Network 152 (SE-ResNet152), which eliminated features with lesser importance.

2.2. Recent Work on Statistical and Transform Approaches for Choosing Features

Thockchom et al., (2023) developed an ensemble learning-based method for IDS, which used logistic regression, decision tree, and Gaussian naive Bayes as base classifiers, while stochastic gradient descent served as the meta-classifier. Nkongolo et al., (2021) and Bacha et al., (2024) presented Principal Component Analysis (PCA) and Kernel PCA (KPCA) to reduce the feature dimensional. Similarly, Bhavsar et al., (2023) used the Pearson-Correlation Coefficient (PCC), and Ahmed et al., (2025) introduced the Pearson Correlation Matrix with Random Forest (PCM-RF) for selecting significant features. The PLC-Convolutional Neural Network (PCC-CNN) (Bhavsar et al., 2023) integrated the essential features acquired from linear-based extractions and achieved effective prediction even under imbalanced attack samples. In (Ahmed et al., 2025), RF was used to rank the features, demonstrating the capability to identify fraudulent network information, while PCM was utilized due to its flexibility in adapting to current variations in network patterns. Conversely, (Xue et al., 2025) developed the Hybrid Auto encoder (HAE)-Hybrid Res Net-LSTM (HRL) for both feature selection and classification. In HAE, the CNN-Gate Recursive Unit (GRU) was used for feature selection to reduce redundancy and enhance classification performance.

2.3. Recent Work on Classification Approaches

Some of the deep learning classifiers are detailed as follows: (Parameswari et al., 2024) implemented the Rat Swarm Hunter-Prey Optimization (RSHPO) to tune the parameters of a Deep Max-out Network (DMN). Several Long Short-Term Memory (LSTM)-based approaches have recently been developed, including the Conventional Bidirectional LSTM (ConvBiLSTM) by (Sheeba & Shaji, 2025), CNN-LSTM by (Thaljaoui, 2025), LSTM-Deep Auto-encoder (LSTM-DAE) by (Kunang et al., 2024), and ENS_CLSTM by (Mo et al., 2025). The hybrid Binary Dwarf Mongoose Optimization was used to obtain highly relevant characteristics (Sheeba & Shaji, 2025); Bayesian optimization was applied to tune the hyper-parameters of CNN-LSTM (Thaljaoui, 2025); in (Kunang et al., 2024), unsupervised learning approaches such as Denouncing Auto Encoder (DAE) and stacked models were used for feature extraction; and in (Mo et al., 2025), category characteristics of traffic were learned and extracted using a sliding window approach.

Shajari et al., (2022) presented Tensor-based Online Anomaly Identification (TOAI) over large network operation data. Larriva-Novo et al., (2021) presented the Multi-Layer Perceptron Neural Network (MLPNN) to enhance intrusion classification. A few statistical and conventional IDS approaches are as follows: (Vishwakarma & Kesswani, 2023; Vishwakarma & Kesswani, 2022) performed a two-phase IDS that included data categorization based on type and classification using various Naive Bayes classifier, along with unsupervised elliptic envelope-based classification.

(Pérez-Bueno et al., 2022) developed Probabilistic Principal Component Analysis (PPCA) to identify anomalies in network security. Furthermore, the Conditional Tabular Generative Adversarial Network (CTGAN) was used to generate synthetic information for minority classes to address data imbalance issue and enhance detection performance in (Wang et al., 2024).

3 Anomaly-Based IDS Using FFPSO-SMANN

This research proposes the FFPSO-SMANN for performing effective anomaly and intrusion classification in Io T networks. The important steps of this research are as follows: In the first step, data are obtained and the required prepossessing is performed to convert the raw data-set into a suitable form for the Artificial Intelligence (AI) approach. In the next step, the appropriate features are selected using FFPSO, and anomalous traffic is differentiated from normal traffic using F-AnoGAN. Finally, SMANN is developed to perform classification in the last step. The FFPSO-SMANN-based anomaly and intrusion classification is illustrated in figure 1.

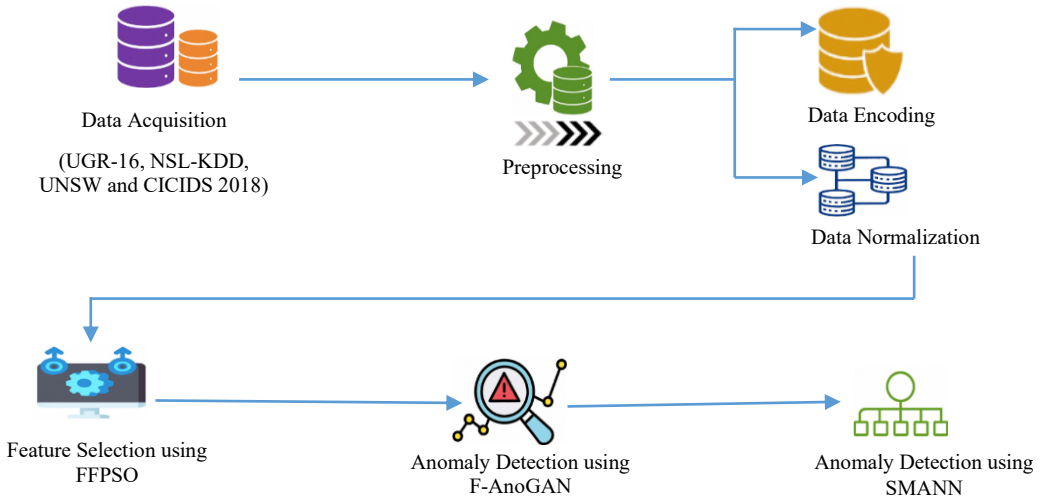


Figure 1: FFPSO-SMANN-based anomaly-based IDS

3.1. Data Acquisition

The developed research is analyzed using the University of Granada 16 (UGR-16) data-set (Maciá-Fernández et al., 2018), NSL-KDD (Faiaz et al., 2024), UNSW (More et al., 2024), and CICIDS 2018 (Protić & Stanković, 2023).

3.1.1. UGR-16 Dataset

This data-set was collected from a Tier-3 Internet Service Provider (ISP), where traffic is produced by web and email servers, company host, virtualization environments, and recursive Domain Name System (DNS) servers. In this network, a group of sensors are installed in the edge routers; therefore, egress and ingress network traffic flows are observed. The data-set contains Net-flow traces related to more than 16,000 million links connected for over four months in 2016.

3.1.2. NSL-KDD Dataset

NSL-KDD is an extended version of an original KDD Cup 1999 data-set generated for IDS evaluation. NSL-KDD provides a standardized and flexible benchmark for IDS evaluation and includes network traffic with 43 features.

3.1.3. UNSW-NB15 Dataset

The UNSW-NB15 data-set, generated by the Australian Cybersquatting Center Labs in 2016, integrates real network traffic with diverse modern network attacks. This data-set includes 49 features that help evaluate IDS performance in real-world network scenarios.

3.1.4. CIC-IDS-2018 Dataset

This data-set is generated from real network communication data and remains insufficient in IDS research. Each attack type is denoted by distinct characteristic with 79 features, offering meaningful patterns of network behavior and probable security threats.

3.2. Dataset Preprocessing

The raw data-set is not directly suitable for AI-based prediction. The important prepossessing operations include data encoding and data normalization.

3.2.1. Data Encoding

One-Hot Encoding is applied to categorical features such as protocol type, service, and flag to convert them into binary vectors compatible with the model.

3.2.2. Data Normalization

Normalization is performed to avoid dimensional influence among features and ensure comparability between the data and the methods used here is Min–Max normalization, expressed in equation (1), which maps features to the $[0, 1]$ range.

$$I' = \frac{I - min(I)}{max(I) - min(I)} \tag{1}$$

Where I and I' denote the input and preprocessed output, respectively, and $max(I)$ and $min(I)$ are the maximum and minimum input.

3.3. Feature Selection Using FFPSO

FFPSO uses the personal and global bests to determine a single guide vector for velocity updates. A quantization function, defined in equation (2), is applied on P_i^t and G^t for each particle i to discover single guide vector. This quantization generates a matrix New_guides^i for Q. The interval is identical in every dimension for any successive member pairs in New_guides^i.

$$New\_guides^i = \begin{cases} lb_{i,d}, & q = 1 \\ lb_{i,d} + (q-1) \times \left(\frac{|ub_{i,d} - lb_{i,d}|}{(Q-1)}\right), & 2 \le q \le Q-1 \\ ub_{i,d}, & q = Q \end{cases} \tag{2}$$

Where Q is an odd integer $Q \geq 3$, $d = 1, 2, 3, \ldots D$, $lb_{i,d} = \min(P_{i,d}^t, G_d^t)$, and$ub_{i,d} = \max(P_{i,d}^t, G_d^t)$. Each row in New_guides^i is a potential candidate to become a single guide in the velocity updates of particle for successive iterations. Next, the members are incorporated in New_guides^i, Q velocity vectors are computed to New_vel^i using equation (3) for every dimension d utilizing single guiding factor.

$$New\_vel_{q,d}^i = w^t \times V_{i,d}^t + c_1 + rand_d \times (New\_guides_{q,d}^i - X_{q,d}^i), \text{ for } q = 1,2, \ldots Q \quad (3)$$

The search agent portion is updated using the dynamic Adaptation of Search Dimension (ASD), which generates new agents according to the Search Dimension-Ratio (SDR), as expressed in equation (4).

$$SDR = \frac{P}{D} \quad (4)$$

Where P represent number of altered decision variables out of D (dimension). The initial, lower and, upper value limits are fixed to 0.35, $1/D$, and 0.75, correspondingly. Equation (5) dynamically adapts SDR in each iteration.

$$SDR^{t+1} = \begin{cases} \frac{SDR^t}{\lambda}, & if\ G^t\ enhancs \\ \lambda \times SDR^t, & if\ G^t\ does\ not\ enhance \end{cases}, 0 < \lambda < 1 \quad (5)$$

Where, the rate of adaptation to discover a new value of SDR is controlled by restriction, represented by lambda (λ). Equation (5) denotes that the SDR is improved by dividing it by λ when the global best solution is enhanced by the current iteration. Otherwise, the SDR is decreased by multiplying it with λ.

For the new location of particle i, probable candidates are calculated based on equation (6) by each row of New_{vel}^i according to SDR.

$$New\_pos_{q,d}^i = \begin{cases} X_{i,d}^t + New\_vel_{q,d}^i, & if r_{3,d} \leq SDR^t \\ G_d^t, & if r_{3,d} > SDR^t \end{cases} \quad (6)$$

The parameter r_3 is a vector of random numbers D in the range of $[0, 1]$, and the probable candidate matrix is denoted as New_vel^i, which is obtained from the first condition of equation (6). Next, the finest candidate with an index is denoted as $best\_ind$, which is selected from New_pos^i.

Next, the velocity and position are updated using equations (7) and (8), respectively.

$$V_i^{t+1} = New\_vel_{best\_ind}^i \quad (7)$$

$$X_i^{t+1} = New\_pos_{best\_ind}^i \quad (8)$$

Equation (2) allows FFPSO to discover appropriate single guides for every particle and update its velocity. According to equation (6), particles fly over only a portion of search space dimensions. equation (5) computes the rate at which the features are emphasized.

The FFPSO chooses the features according to the fitness function shown in equation (9), which improves classification accuracy while reducing the number of features.

$$Fitness = \propto \times (1 - CE) + \beta \times \left(1 - \frac{|S|}{|F|}\right) \quad (9)$$

Where the number of selected features is $|S|$, the total available features are $|F|$, cross entropy is CE, and $\propto$ and β are weight values. The number of particles and maximum iterations considered for FFPSO are 50 and 100, respectively. The iterative process continues until it reaches the maximum iterations.

FFPSO returns 21 features for NSL-KDD, 24 for UNSW-NB15, 28 for CICIDS-2018, and 9 for UGR-16 from a total of 43, 49, 79, and 14 features, respectively.

Anomaly Detection Using F-AnoGAN

F-AnoGAN, based on GAN, is an effective unsupervised anomaly detection approach. During training, G processes noise input vectors z sampled from the latent space Z and maps them to the data space X, while D evaluates similarity between the disseminations of input x and generated data $G(z)$. The discriminator D distinguishes between original input x and reconstructed information $G(E(x))$. Figure 2 shows the F-AnoGAN architecture. The loss function of F-AnoGAN is depicted in equation (10).

$$L = \frac{1}{n}\left\|x - G(E(x))\right\|_2 + \frac{k}{n_d}\left\|f(x) - f(G(E(x)))\right\|_2 \quad (10)$$

Where, dimension of latent feature depiction is n_d and weighting factor is denoted as k.

Initially, the encoder E transforms the test information into the hidden space, and the generator G transforms the hidden depiction into the feature space. The computation of the anomaly score from equation (11) is used to detect anomalies by calculating the variation between the restored information $G(E(x))$ and the original input x.

$$A(x) = A_R(x) + k.A_D(x) \quad (11)$$

Where, $A_R(x) = \frac{1}{n}\left\|x - G(E(x))\right\|_2$ and $A_D(x) = \frac{1}{n_d}\left\|f(x) - f(G(E(x)))\right\|_2$.

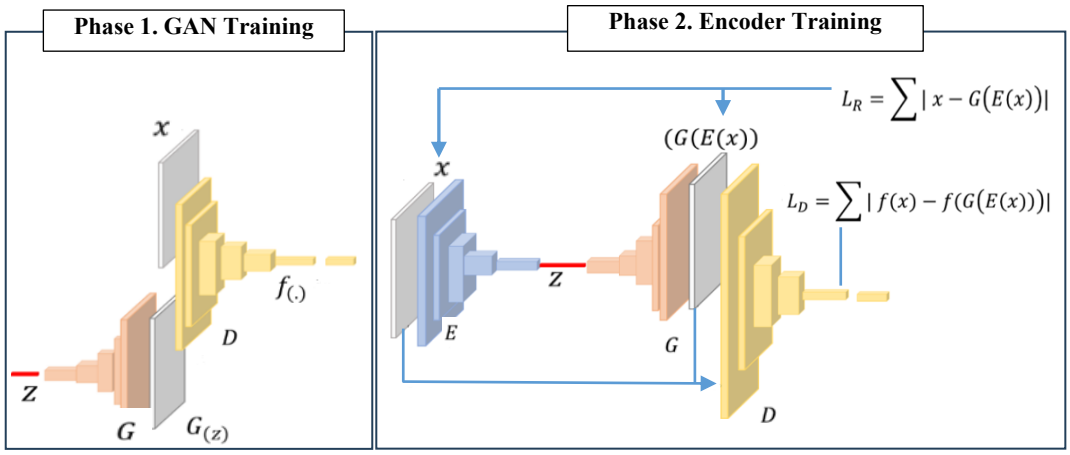


Figure 2: Architecture of F-AnoGAN used in anomaly detection

The steps processed in F-AnoGAN to detect anomalies from normal traffic using the selected features from FFPSO are stated as follows:

- **Step 1:** First, F-AnoGAN is designed with three primary sub-networks: discriminator (D), generator (G), and encoder (E), where it receives the selected features from FFPSO.
- **Step 2:** The generator receives noise input and produces synthetic samples, while discriminator is updated to distinguish between real and generated samples.
- **Step 3:** In the second phase, the encoder learns to map each input sample to its respective latent representation $\hat{z} = E(x)$, and the generator rebuilds it as $\hat{x} = G(E(x))$.
- **Step 4:** For F-AnoGAN, the loss function is computed using equation (10). This dual-component loss ensures that the reconstruction retains fine-grained information with the input.

- **Step 5:** For a test sample x, the encoder generates a latent representation $\hat{z} = E(x)$, which is passed via the generator to recreate $\hat{x} = G(\hat{z})$.
- **Step 6:** The anomaly score is computed and evaluated using a predefined threshold. A higher anomaly score represents anomalous behavior.

3.4. Classification Using SMANN

In the initial phase of LSTM, gate value for the input is computed based on the previous time step and hidden state. This computation obtains a value between 0 and 1, which affects the capacity of the memory cell. Input gate values are computed using equations (12) and (13).

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

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{13}$$

Where the input gate and candidate cell state activators at time t are denoted as i_t and $\tilde{C}_t$, respectively; the weight matrices for the input gate and candidate state are W_i and W_c, respectively; hidden state from the previous time step $t-1$ is denoted as h_{t-1}; the input vector is x_t; and the bias terms for the input gate and candidate cell state are b_i and b_c, respectively. The sigmoid activation is represented as σ, and the hyperbolic tangent activation is denoted as $\tanh$.

In the second step, the forget value is measured by considering both hidden state from the previous time step and the current input, obtaining a value between 0 to 1. The computation of how much data is preserved in the memory cells is computed using equation (14).

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{14}$$

Where the forget gate activation at time t is depicted as f_t, and weight matrix and bias of the forget gate are denoted as W_f and b_f, respectively.

In the third step, the memory cell is updated by combining the output of the input gate and the current input. The update process is defined in equation (15), which dynamically alters the information based on current inputs and past information provided by the previous memory cell state.

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \tag{15}$$

Where the current and previous memory states are denoted as C_t and C_{t-1}, respectively.

In the fourth step, the output gate is defined based on current input, hidden state from the previous time step, and the updated memory cell value. The data from the memory cells are calculated using equation (16), which defines the mechanism by which the output gate controls the appropriate data release for consequent steps in the network.

$$o_t = \sigma(W_0[h_{t-1}, x_t] + b_0) \tag{16}$$

Where the activation of output gate at time t is o_t; the weight and bias of the output gate are denoted as W_0 and b_0, respectively.

$$h_t = o_t \tanh(C_t) \tag{17}$$

The hyperparameters considered in SMANN are given in table 1.

In this research, optimized features are selected using FFPSO, where the issue of oscillation is avoided by using the single-guide vector approach and ASD strategy.

Table 1: Hyperparameters of SMANN

Parameters	Value
Batch size	64
Optimizer	Adam
Initial Learning rate	0.001
Epoch	100
Loss	Categorical cross entropy
Activation	Softmax

Experimental Results

In this section, an analysis is conducted to evaluate the efficiency of FFPSO-SMANN. The UGR-16 data-set is considered to evaluate FFPSO-SMANN, where a 20:80 ratio is used for testing and training data during the analysis.

3.5. Experimental Setup and Results

The design of FFPSO-SMANN is implemented using Anaconda Navigator 3.5.2.0 with Python 3.10.12, whereas the system is configured with a Windows 10 operating system, 8GB RAM, and an Intel Core i5 processor.

3.6. Evaluation Metrics

The performance metrics are defined as follows:

- Accuracy, shown in equation (18), is the ratio between correctly categorized instances and the total number of instances.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+TN+FN}} \times 100 \quad (18)$$

- Precision, expressed in equation (19), denotes the number of instances correctly categorized as positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \times 100 \quad (19)$$

Recall, also called the True Positive Rate (TPR), is represented in equation (20). It is obtained from correctly classified attacks divided by the total number of attacks, calculating a classifier's ability a detect all positive samples in the dataset.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \times 100 \quad (20)$$

- F1-Score is the harmonic mean of recall and precision, as represented in equation (21).

$$\text{F1} - \text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (21)$$

3.7. Result Analysis

The result analysis is performed to evaluate the efficiency of FFPSO-SMANN a) for various clustering methods, b) for various feature selection methods using optimization and c) for various classifiers.

3.7.1. Result Analysis Over Anomaly Detection Using F-AnoGAN

The classification outcomes of these models are presented in table 2. The performance of F-AnoGAN across all 4-benchmark data-set outperforms the other generative models such as standard GAN, AC-GAN, and Cycle GAN.

Table 2: Analysis of various GAN models

Models	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	AUC (%)
GAN	**UGR16**	94.13	92.87	93.02	93.07	94.31
	NSL-KDD	92.75	91.91	91.63	91.77	92.85
	UNSW	93.92	92.89	92.7	92.83	93.94
	CICIDS 2018	95.05	94.01	93.98	94.02	95.21
AC-GAN	**UGR16**	95.34	94.25	94.28	94.29	95.45
	NSL-KDD	94.01	93.98	93.92	93.96	94.11
	UNSW	93.85	92.72	92.7	92.74	93.91
	CICIDS 2018	95.12	95.09	94.05	94.08	95.3
CycleGAN	**UGR16**	96.45	96.31	96.4	96.38	96.52
	NSL-KDD	95.15	95.09	95.02	95.07	95.23
	UNSW	94.97	94.89	94.85	94.91	95.04
	CICIDS 2018	96.2	96.17	96.12	96.15	96.35
F-AnoGAN	**UGR16**	99.98	99.98	99.98	99.98	99.99
	NSL-KDD	99.95	99.98	99.96	99.97	99.97
	UNSW	98.98	98.98	98.96	98.97	99.97
	CICIDS 2018	98.95	98.95	99.95	99.45	99.99

3.7.2. Result Analysis for Various Feature Selection Using Optimization Methods

The classification outcomes of different optimization-based feature selection methods are presented in table 3. The proposed FFPSO achieves 99.98 % higher accuracy in the UGR-16 data-set compared to WOA, SSO, ACO, and PSO. The incorporation of the single-guide vector approach and ASD strategy enhances convergence, as shown in figure 3.

Table 3: Analysis of various feature selection using optimization methods

Classifiers	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	AUC (%)
WOA	**UGR16**	92.15	91.78	90.92	91.35	92.88
	NSL-KDD	91.87	90.43	89.75	90.08	91.65
	UNSW	93.14	92.32	91.55	91.93	93.48
	CICIDS 2018	93.68	92.89	92.14	92.51	94.21
SSO	**UGR16**	95.35	94.12	93.78	93.95	95.52
	NSL-KDD	94.92	93.67	93.11	93.38	95.15
	UNSW	96.01	95.21	94.85	95.02	96.36
	CICIDS 2018	96.33	95.87	95.45	95.65	96.98
ACO	**UGR16**	97.45	96.76	96.32	96.54	97.78
	NSL-KDD	96.98	96.12	95.67	95.89	97.32
	UNSW	97.76	97.21	96.87	97.02	98.11
	CICIDS 2018	98.12	97.56	97.14	97.35	98.22
PSO	**UGR16**	98.56	98.45	98.6	98.52	98.90
	NSL-KDD	98.75	98.8	98.7	98.75	98.88
	UNSW	97.25	97.10	97.00	97.05	98.20
	CICIDS 2018	96.95	96.90	97.00	96.95	98.50
FFPSO	**UGR16**	99.98	99.98	99.98	99.98	99.99
	NSL-KDD	99.95	99.98	99.96	99.97	99.97
	UNSW	98.98	98.98	98.96	98.97	99.97
	CICIDS 2018	98.95	98.95	99.95	99.45	99.99

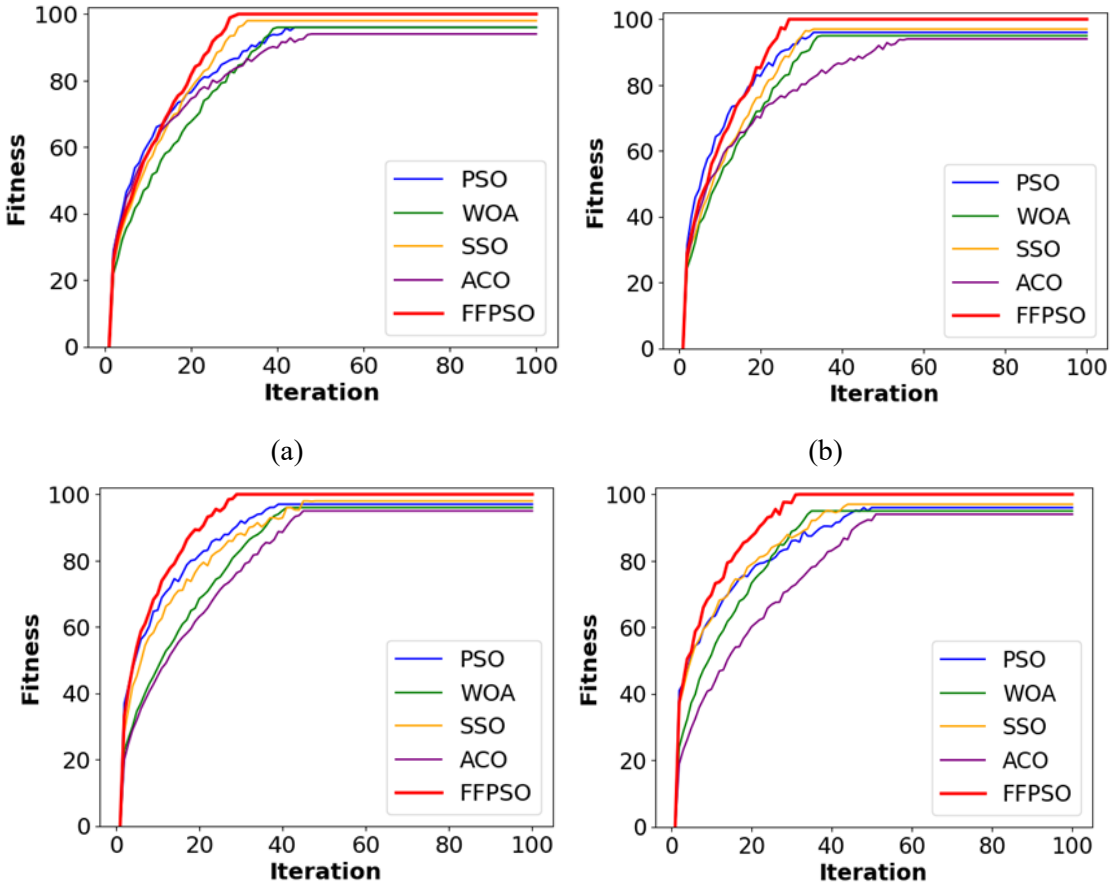
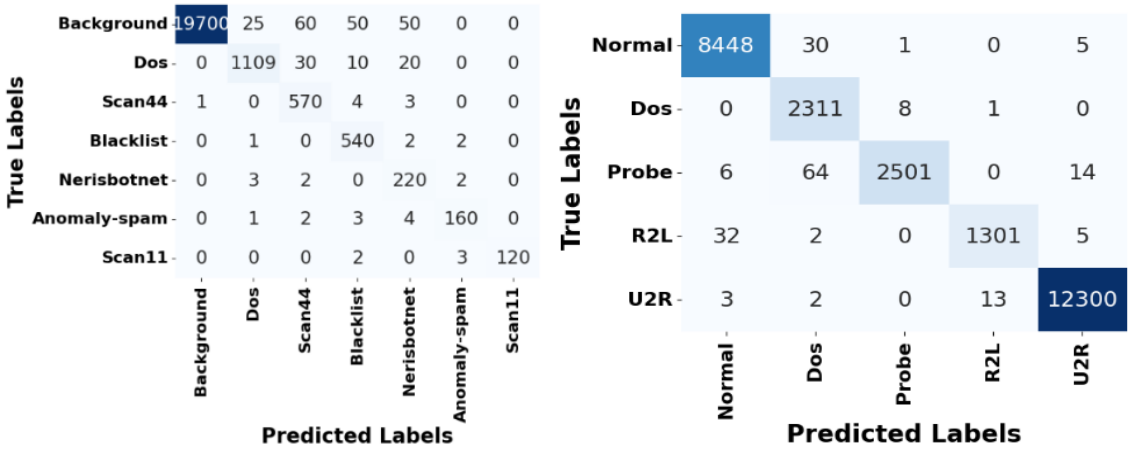
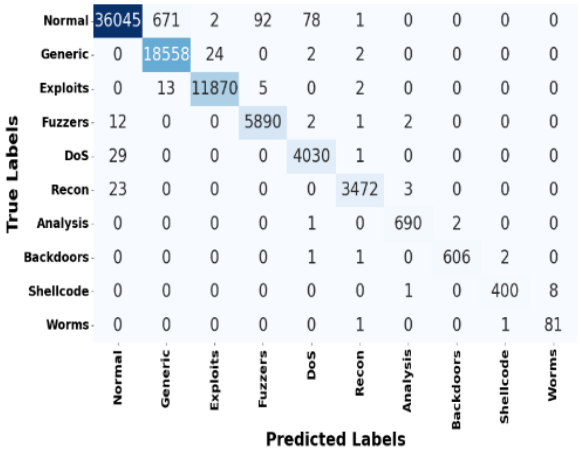


Figure 3: Convergence analysis, UGR16, NSL-KDD, UNSW, and CICIDS 2018 data-set

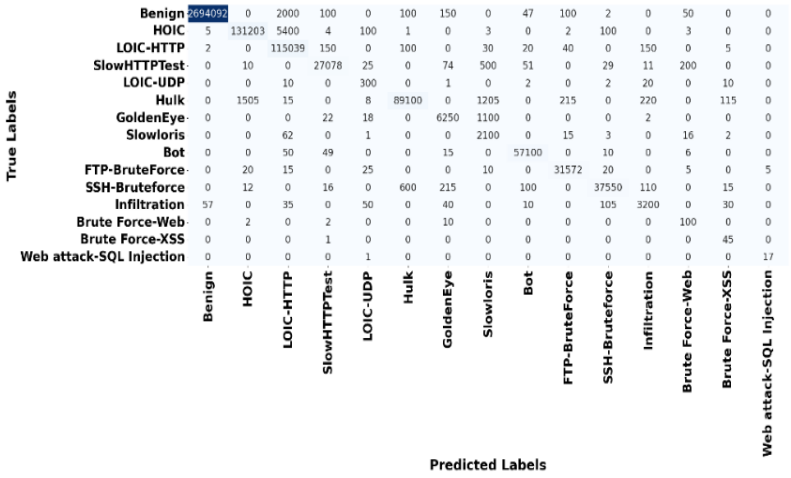
3.7.3. Result Analysis for Various Classifiers

The confusion matrices obtained during intrusion detection for UGR-16, NSL-KDD, UNSW, and CICIDS-2018 are shown in figure 4.





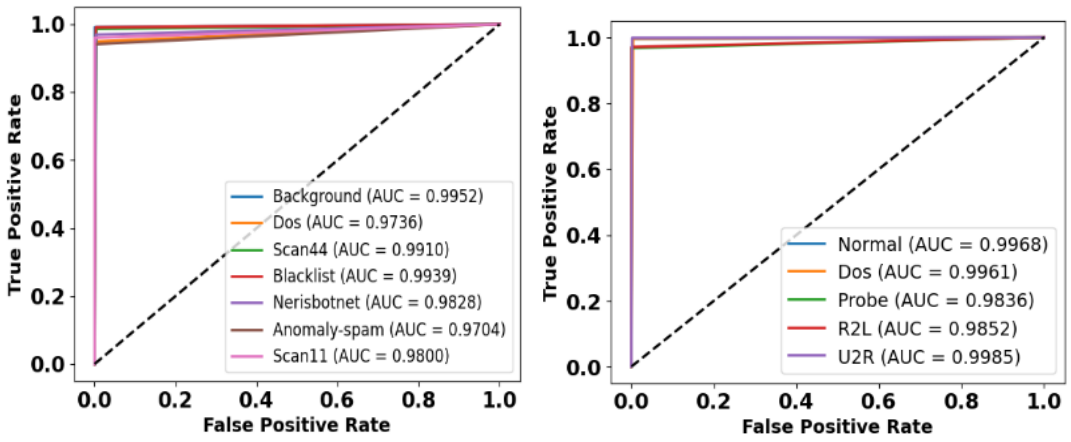
(c)



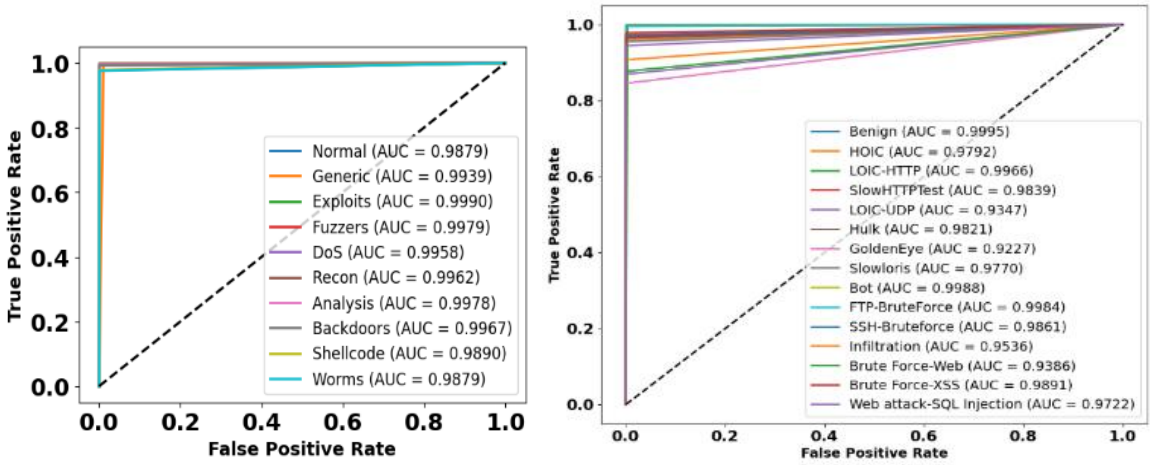
(d)

Figure 4: Confusion matrix, a) UGR-16, b) NSL-KDD, c) UNSW, d) CICIDS 2018

Figure 5 illustrates the Receiver Operating Characteristic (ROC) curves of SMANN for different data-set.



(a) (b)



(c) (d)

Figure 5: ROC curve, a) UGR-16, b) NSL-KDD, c) UNSW, d) CICIDS 2018

The classification outcomes of SMANN and other classifiers, including Vanilla RNN, GRU, and Bi-LSTM, for different data-set are presented in table 4. The developed SMANN evaluated using the UGR-16 data-set achieves the highest accuracy of 94.62% for actual features and 99.98% for selected features, outperforming Vanilla RNN, GRU, and Bi-LSTM.

Table 4: Analysis of various classifiers

Classifiers	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	AUC (%)
Vanilla RNN	**UGR16**	94.12	93.45	92.78	93.11	94.33
	NSL-KDD	93.58	92.23	91.85	92.03	93.72
	UNSW	94.45	93.56	92.78	93.17	94.21
	CICIDS 2018	94.89	93.92	93.35	93.63	95.12
GRU	**UGR16**	96.45	95.56	94.78	95.17	96.21
	NSL-KDD	95.89	94.92	94.35	94.63	96.12
	UNSW	96.75	95.88	95.43	95.65	96.89
	CICIDS 2018	97.12	96.56	96.14	96.35	97.22
Bi-LSTM	**UGR16**	98.12	97.56	97.14	97.35	98.22
	NSL-KDD	97.75	96.88	96.43	96.65	97.89
	UNSW	98.32	97.45	97.12	97.28	98.54
	CICIDS 2018	98.78	98.12	97.85	97.98	98.92
SMANN	**UGR16**	99.98	99.98	99.98	99.98	99.99
	NSL-KDD	99.95	99.98	99.96	99.97	99.97
	UNSW	98.98	98.98	98.96	98.97	99.97
	CICIDS 2018	98.95	98.95	99.95	99.45	99.99

3.8. Complexity and Statistical Analysis

The complexity and statistical analysis of SMANN with its baseline classifiers, including Vanilla RNN, GRU, and Bi-LSTM, are provided in table 5.

Table 5: Complexity and statistical analysis of SMANN

Methods	Dataset	P-Values	Memory usage (MB)	Training time (s)	Inference Time(s)
Vanilla RNN	**UGR16**	0.035	620	720	370
	NSL-KDD	0.034	610	690	355
	UNSW	0.037	590	750	340
	CICIDS 2018	0.036	630	730	325
GRU	**UGR16**	0.030	680	620	285
	NSL-KDD	0.028	675	600	265
	UNSW	0.031	655	630	250
	CICIDS 2018	0.032	690	650	235
Bi-LSTM	**UGR16**	0.027	760	590	195
	NSL-KDD	0.026	750	560	180
	UNSW	0.028	730	570	165
	CICIDS 2018	0.029	770	580	150
SMANN	**UGR16**	0.015	540	430	120
	NSL-KDD	0.014	535	390	105
	UNSW	0.016	520	410	98
	CICIDS 2018	0.017	550	420	110

3.9. Comparative Analysis

The recent related works considered for the comparison of FFPSO-SMANN include CGN_SEDSCapsNet (Ammannamma & Chakravarthy, 2025), MPAC (Makhadmeh et al., 2025), PCC-CNN (Bhavsar et al., 2023), HAE-HRL (Xue et al., 2025), and MLPNN (Larriva-Novo et al., 2021). The comparison for FFPSO-SMANN with the existing methods is provided in table 6.

Table 6: Comparison of FFPSO-SMANN

Classifiers	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
CGN_SEDSCapsNet (Ammannamma & Chakravarthy, 2025)	**UNSW**	98.4	98.5	98.5	98.5
MPAC (Makhadmeh et al., 2025)	**NSL-KDD**	99.58	NA	99.77	NA
	UNSW	98.98	NA	99.76	NA
PCC-CNN (Bhavsar et al., 2023)	**NSL-KDD**	94	95	77	80
HAE-HRL (Xue et al., 2025)	**NSL-KDD**	93.58	NA	NA	NA
	UNSW	85.47	NA	NA	NA
	CICIDS 2018	94.62	NA	NA	NA
MLPNN (Larriva-Novo et al., 2021)	**UGR16**	99.3	NA	NA	NA
	NSL-KDD	99.7	NA	NA	NA
	UNSW	99.2	NA	NA	NA
FFPSO-SMANN	**UGR16**	99.98	99.98	99.98	99.98
	NSL-KDD	99.95	99.98	99.96	99.97
	UNSW	98.98	98.98	98.96	98.97
	CICIDS 2018	98.95	98.95	99.95	99.45

4 Discussion

This section discusses the results analyzed for IDS using the baseline models and existing researches. First, the anomaly detection performance of F-AnoGAN is compared with different GAN models, including standard GAN, AC-GAN, and CycleGAN. Next, the features optimized by FFPSO achieve

better detection accuracy than those obtained using WOA, SSO, ACO, and PSO. A comparison with existing researches confirms that FFPSO-SMANN achieves improved classification performance compared to CGN_SEDSCapsNet (Ammannamma & Chakravarthy, 2025), MPAC (Makhadmeh et al., 2025), PCC-CNN (Bhavsar et al., 2023), HAE-HRL (Xue et al., 2025), and MLPNN (Larriva-Novo et al., 2021).

5 Conclusion

Although IoT is extensively utilized in diverse applications because it simplifies daily life and enhances the quality of service, security, and privacy issues remain significant challenges. In this research, an anomaly-based IDS is developed using FFPSO for feature selection, F-AnoGAN for anomaly detection, and SMANN for classification. FFPSO selects highly relevant features while avoiding the oscillation issue through the single-guide vector approach and ASD strategy. This effective prediction leads to minimized false positives by considering past network behavior. The results demonstrate that FFPSO-SMANN provides enhanced classification compared to CGN_SEDSCapsNet, MPAC, PCC-CNN, HAE-HRL, and MLPNN. The accuracy of FFPSO-SMANN is 99.98% for the UGR-16 data-set, confirming its superior performance over MLPNN.

Author Contributions

Veena Potdar: Visualization; Conceptualization; Formal Analysis; Resources; Project Administration; Investigation.

Mohan Govindasa Kabadi: Methodology; Supervision; Data Curation; Manuscript - Review & Editing; Validation; Manuscript Original Draft.

All authors have read and approved the final manuscript

Declarations

Funding: This research received no external funding.

Conflict of Interest: The authors declare that they have no conflict of interest.

Ethics Approval: I/We declare that the work submitted for publication is original, previously unpublished in English or any other language(s), and not under consideration for publication elsewhere.

Consent for publication: I certify that all the authors have approved the paper for release and agree with its content.

Data Availability: The datasets generated during and/or analyzed during the current study are available in the [University of Granada-16 data-set] repository: https://nesg.ugr.es/nesg-ugr16/

References

[1] Ahmed, N., Ngadi, M. A., Rathore, M. S., & Mahmood, A. (2025). PCM-RF a Hybrid Feature Selection Mechanism for Intrusion Detection System in IoT. *Security and Privacy*, *8*(1), e499. https://doi.org/10.1002/spy2.499

[2] Alghanam, O. A., Almobaideen, W., Saadeh, M., & Adwan, O. (2023). An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning. *Expert Systems with Applications*, *213*, 118745. https://doi.org/10.1016/j.eswa.2022.118745

[3] Ammannamma, T., & Chakravarthy, A. S. N. (2025). A bio-inspired optimal feature with convolutional GhostNet based squeeze excited deep-scale capsule network for intrusion detection. *Computers & Security*, *150*, 104221. https://doi.org/10.1016/j.cose.2024.104221

[4] Anchuri, S., Ganesh, A., & Perugu, P. (2025). NSGTO-LSTM: Niche-strategy-based gorilla troops optimization and long short-term memory network intrusion detection model. *ETRI Journal*, *47*(6), 1049-1060. https://doi.org/10.4218/etrij.2024-0256

[5] Antonius, F., Sekhar, J. C., Rao, V. S., Pradhan, R., Narendran, S., Borda, R. F. C., & Silvera-Arcos, S. (2023). Unleashing the power of Bat optimized CNN-BiLSTM model for advanced network anomaly detection: Enhancing security and performance in IoT environments. *Alexandria Engineering Journal*, *84*, 333-342. https://doi.org/10.1016/j.aej.2023.11.015

[6] Ayesh, A. N. (2024). Enhancing Urban Living in Smart Cities Using the Internet of Things (IOT). *International Academic Journal of Science and Engineering*, *11*(1), 237–246. https://doi.org/10.9756/IAJSE/V11I1/IAJSE1127

[7] Bacha, S., Aljuhani, A., Abdellafou, K. B., Taouali, O., Liouane, N., & Alazab, M. (2024). Anomaly-based intrusion detection system in IoT using kernel extreme learning machine. *Journal of Ambient Intelligence and Humanized Computing*, *15*(1), 231-242. https://doi.org/10.1007/s12652-022-03887-w

[8] Bhavsar, M., Roy, K., Kelly, J., & Olusola, O. (2023). Anomaly-based intrusion detection system for IoT application. *Discover Internet of things*, *3*(1), 5. https://doi.org/10.1007/s43926-023-00034-5

[9] Biju, A., & Franklin, S. W. (2025). Dual Feature-Based Intrusion Detection System for IoT Network Security. *International Journal of Computational Intelligence Systems*, *18*(1), 1-19.

[10] Biju, A., & Wilfred Franklin, S. (2024). Evaluated bird swarm optimization based on deep belief network (EBSO-DBN) classification technique for IOT network intrusion detection. *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, *65*(1), 108-116. https://doi.org/10.1080/00051144.2023.2269646

[11] Douiba, M., Benkirane, S., Guezzaz, A., & Azrour, M. (2023). Anomaly detection model based on gradient boosting and decision tree for IoT environments security. *Journal of Reliable Intelligent Environments*, *9*(4), 421-432.

[12] Faiaz, M. A., Mitra, D., & Das Prangon, R. (2024, December). Intrusion Detection Using Convolutional Neural Network: A Color Mapping Approach on NSL-KDD Dataset. In *Proceedings of the 11th International Conference on Networking, Systems, and Security,* (154-162). https://doi.org/10.1145/3704522.3704541

[13] Jayasankar, T., Kiruba Buri, R., & Maheswaravenkatesh, P. (2024). Intrusion detection system using metaheuristic fireworks optimization based feature selection with deep learning on Internet of Things environment. *Journal of Forecasting*, *43*(2), 415-428. https://doi.org/10.1002/for.3037

[14] Keserwani, P. K., Govil, M. C., Pilli, E. S., & Govil, P. (2021). A smart anomaly-based intrusion detection system for the Internet of Things (IoT) network using GWO–PSO–RF model. *Journal of Reliable Intelligent Environments*, *7*(1), 3-21. https://doi.org/10.1007/s40860-020-00126-x

[15] Kunang, Y. N., Nurmaini, S., Stiawan, D., & Suprapto, B. Y. (2024). An end-to-end intrusion detection system with IoT dataset using deep learning with unsupervised feature extraction. *International Journal of Information Security*, *23*(3), 1619-1648. https://doi.org/10.1007/s10207-023-00807-7

[16] Larriva-Novo, X., Villagrá, V. A., Vega-Barbas, M., Rivera, D., & Sanz Rodrigo, M. (2021). An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets. *Sensors*, *21*(2), 656. https://doi.org/10.3390/s21020656

[17] Lazzarini, R., Tianfield, H., & Charissis, V. (2023). A stacking ensemble of deep learning models for IoT intrusion detection. *Knowledge-Based Systems*, *279*, 110941. https://doi.org/10.1016/j.knosys.2023.110941

[18] Lu, H., Wang, T., Xu, X., & Wang, T. (2021). Cognitive memory-guided autoencoder for effective intrusion detection in internet of things. *IEEE Transactions on Industrial Informatics*, *18*(5), 3358-3366. https://doi.org/10.1109/TII.2021.3102637

[19] Ma, Z., Liu, L., Meng, W., Luo, X., Wang, L., & Li, W. (2023). ADCL: toward an adaptive network intrusion detection system using collaborative learning in IoT networks. *IEEE Internet of Things Journal*, *10*(14), 12521-12536. https://doi.org/10.1109/JIOT.2023.3248259

[20] Maciá-Fernández, G., Camacho, J., Magán-Carrión, R., García-Teodoro, P., & Therón, R. (2018). UGR '16: A new dataset for the evaluation of cyclostationarity-based network IDSs. *Computers & Security*, *73*, 411-424. https://doi.org/10.1016/j.cose.2017.11.004

[21] Makhadmeh, S. N., Fraihat, S., Awad, M., Sanjalawe, Y., Al-Betar, M. A., & Awadallah, M. A. (2025). A crossover-integrated Marine Predator Algorithm for feature selection in intrusion detection systems within IoT environments. *Internet of Things*, *31*, 101536. https://doi.org/10.1016/j.iot.2025.101536

[22] Mo, J., Ke, J., Zhou, H., & Li, X. (2025). Hybrid network intrusion detection system based on sliding window and information entropy in imbalanced dataset. *Applied Intelligence*, *55*(6), 433. https://doi.org/10.1007/s10489-025-06307-6

[23] More, S., Idrissi, M., Mahmoud, H., & Asyhari, A. T. (2024). Enhanced intrusion detection systems performance with UNSW-NB15 data analysis. *Algorithms*, *17*(2), 64. https://doi.org/10.3390/a17020064

[24] Nandanwar, H., & Katarya, R. (2024). TL-BILSTM IoT: transfer learning model for prediction of intrusion detection system in IoT environment. *International Journal of Information Security*, *23*(2), 1251-1277. https://doi.org/10.1007/s10207-023-00787-8

[25] Nkongolo, M., Van Deventer, J. P., & Kasongo, S. M. (2021). Ugransome1819: A novel dataset for anomaly detection and zero-day threats. *Information*, *12*(10), 405. https://doi.org/10.3390/info12100405

[26] Otoum, Y., & Nayak, A. (2021). As-ids: Anomaly and signature based ids for the internet of things. *Journal of Network and Systems Management*, *29*(3), 23. https://doi.org/10.1007/s10922-021-09589-6

[27] Parameswari, A., Ganeshan, R., Ragavi, V., & Shereesha, M. (2024). Hybrid rat swarm hunter prey optimization trained deep learning for network intrusion detection using CNN features. *Computers & Security*, *139*, 103656. https://doi.org/10.1016/j.cose.2023.103656

[28] Pérez-Bueno, F., García, L., Maciá-Fernández, G., & Molina, R. (2022). Leveraging a probabilistic PCA model to understand the multivariate statistical network monitoring framework for network security anomaly detection. *IEEE/ACM Transactions on Networking*, *30*(3), 1217-1229.

[29] Prabu, K., & Sudhakar, P. (2022, December). Design and Implementation of an Automated Control System for Anomaly Detection Using an Enhanced Intrusion Detection System. In *2022 Third International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)* (pp. 1-7). IEEE. https://doi.org/10.1109/ICSTCEE56972.2022.10100003

[30] Prabu, K., & Sudhakar, P. (2024, January). A Comprehensive Survey: Exploring Current Trends and Challenges in Intrusion Detection and Prevention Systems in the Cloud Computing Paradigm. In *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)* (pp. 351-358). IEEE. https://doi.org/10.1109/IDCIoT59759.2024.10467700

[31] Priyanka, J., Poorani, T. R., & Ramya, M. (2023). An Investigation of Fluid Flow Simulation in Bioprinting Inkjet Nozzles Based on Internet of Things. *Indian Journal of Information Sources and Services*, *13*(2), 46–52. https://doi.org/10.51983/ijiss-2023.13.2.3845

[32] Protić, D. D., & Stanković, M. M. (2023). Cybersecurity attacks: which dataset should be used to evaluate an intrusion detection system? *Vojnotehnički glasnik*, *71*(4), 970-995. https://dx.doi.org/10.5937/vojtehg71-46524

[33] Rahaim, L. A. A., Hasan, D. S., & Ali, A. H. (2024). Smart Cars Parking Systems of Big Cities based on the Internet of Things. *Journal of Internet Services and Information Security*, *14*(3), 380-392. https://doi.org/10.58346/JISIS.2024.I3.023

[34] Rohini, G., Gnana Kousalya, C., & Bino, J. (2023). Intrusion detection system with an ensemble learning and feature selection framework for IoT networks. *IETE Journal of Research*, *69*(12), 8859-8875. https://doi.org/10.1080/03772063.2022.2098187

[35] Saba, T., Rehman, A., Sadad, T., Kolivand, H., & Bahaj, S. A. (2022). Anomaly-based intrusion detection system for IoT networks through deep learning model. *Computers and Electrical Engineering*, *99*, 107810. https://doi.org/10.1016/j.compeleceng.2022.107810

[36] Saikam, J., & Ch, K. (2024). An ensemble approach-based intrusion detection system utilizing ISHO-HBA and SE-ResNet152. *International Journal of Information Security*, *23*(2), 1037-1054. https://doi.org/10.1007/s10207-023-00777-w

[37] Shajari, M., Geng, H., Hu, K., & Leon-Garcia, A. (2022). Tensor-based online network anomaly detection and diagnosis. *IEEE Access*, *10*, 85792-85817. https://doi.org/10.1109/ACCESS.2022.3197651

[38] Sheeba, S. M., & Shaji, R. S. (2025). Hybrid-CID: Securing IoT with mongoose optimization. *International Journal of Computational Intelligence Systems*, *18*(1), 1-18.

[39] Sugitha, G., Preethi, B. C., & Kavitha, G. (2022). Intrusion detection framework using stacked auto encoder based deep neural network in IOT network. *Concurrency and Computation: Practice and Experience*, *34*(28), e7401. https://doi.org/10.1002/cpe.7401

[40] Thaljaoui, A. (2025). Intelligent network intrusion detection system using optimized deep CNN-LSTM with UNSW-NB15. *International Journal of Information Technology*, 1-17.

[41] Thockchom, N., Singh, M. M., & Nandi, U. (2023). A novel ensemble learning-based model for network intrusion detection. *Complex & Intelligent Systems*, *9*(5), 5693-5714.

[42] Veerasamy, M., Jaganathan, S. C. B., Dhasarathan, C., Mubarakali, A., Ramasamy, V., Kalpana, R., & Marina, N. (2023). Legendre neural network method for solving nonlinear singular systems. In *Intelligent Technologies for Sensors* (pp. 25-37). Apple Academic Press.

[43] Vishwakarma, M., & Kesswani, N. (2022). DIDS: A Deep Neural Network based real-time Intrusion detection system for IoT. *Decision Analytics Journal*, *5*, 100142. https://doi.org/10.1016/j.dajour.2022.100142

[44] Vishwakarma, M., & Kesswani, N. (2023). A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection. *Decision Analytics Journal*, *7*, 100233. https://doi.org/10.1016/j.dajour.2023.100233

[45] Wang, X., Dai, L., & Yang, G. (2024). A network intrusion detection system based on deep learning in the IoT: X. Wang et al. *The Journal of Supercomputing*, *80*(16), 24520-24558. https://doi.org/10.1007/s11227-024-06345-w

[46] Xue, Y., Kang, C., & Yu, H. (2025). HAE-HRL: A network intrusion detection system utilizing a novel autoencoder and a hybrid enhanced LSTM-CNN-based residual network. *Computers & Security*, *151*, 104328. https://doi.org/10.1016/j.cose.2025.104328

[47] Zareh Farkhady, R., Majidzadeh, K., Masdari, M., & Ghaffari, A. (2025). 3DLBS-BCHO: A three-dimensional deep learning approach based on branch splitter and binary chimp optimization for intrusion detection in IoT. *Cluster Computing*, *28*(2), 83. https://doi.org/10.1007/s10586-024-04768-x

Authors Biography

Veena Potdar received her B.E. degree in Computer Science in the year 1999 from Gogte Institute of Technology, Belgaum. She completed her M. Tech degree in the year 2005 from Dr. Ambedkar Institute of Technology, Bangalore & is currently working as Associate Professor in the department of Computer Science & Engineering in the same college. She has a teaching experience of 24 years. She is a permanent member of Indian Society for Technical Education, Institute of Engineers & Cryptology Research Society of India & nominee member for Computer Society of India. Her areas of interests are cyber security, IoT & security in Databases.

Dr. Mohan Govindasa Kabadi is a distinguished Computer Science Professor and academic leader based in Bangalore, India. He is currently a Professor and DRC Chair at GITAM Deemed University, a role he has held since August 2021. Previously, he was the Professor and Head of Computer Science at Presidency University Bangalore (2017–2021) and served as Dean of Administration and Head of Computer Science at Acharya Institute of Technology (2012–2015). He also held leadership roles as Principal at C. Byregowda Institute of Technology (2009–2012) and Professor and Head at SJCIT (1988–2009). Dr. K G Mohan has extensive experience managing undergraduate, postgraduate, and PhD programs, overseeing academic administration, and leading faculty teams. With decades of expertise in teaching and research, he has significantly contributed to advancing Computer Science Education in India.

Dr.N. Dayanand Lal is an Assistant Professor in the Department of Computer Science and Engineering at GITAM School of Technology, Bengaluru. He holds an M.Tech from BVBCET, Hubli, and earned his Ph.D. from Dr. M.G.R. Educational & Research Institute University. His academic and research interests primarily span cyber security, data-driven systems, and the application of information technology in agriculture. He is actively involved in interdisciplinary research and serves as a co-investigator on a Department of Science and Technology (DST) funded project focused on IT-assisted silk cocoon farming aimed at enhancing the socio-economic conditions of rural communities. He has been contributing to teaching, mentoring, and applied research in emerging areas of computing.