An Intelligent Multi-Head Attention-Driven Temporal Convolutional Architecture for Intrusion Detection in Wireless Sensor Networks

M. Pradeepa
Department of Computer and Information Science, Faculty of Science, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India
pradeepamca87@gmail.com (corresponding author)

R. Ponnusamy
Department of Computer and Information Science, Faculty of Science, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India
povi2006@gmail.com

Received: 23 September 2025 | Revised: 22 October 2025 | Accepted: 27 October 2025

 DOI: https://doi.org/10.48084/etasr.15044

ABSTRACT

Wireless Sensor Networks (WSNs) have diverse uses but are prone to attacks due to their open deployment and low-cost devices. Preventive mechanisms are used to protect WSNs against certain types of attacks. Intrusion Detection Systems (IDSs) are essential as they prevent intruders from causing damage to the network. An IDS can gather information related to attack methods, helping prevent further damage. Deep Learning (DL) methods are widely applied in IDSs because they offer superior performance while processing uneven attacks in WSNs. This study presents an Intrusion Detection Framework for Securing WSNs using a Deep Representation Learning (IDFSWSN-DRL) model, aiming to develop an effective IDS to ensure real-time detection and mitigation of malicious activities. The data preprocessing stage applies robust scaling, label encoding, and data splitting to enhance data quality and support improved model accuracy. The Gazelle Optimization Algorithm (GOA) is employed for Feature Selection (FS). A Temporal Convolutional Network with Multi-Head Attention (TCN-MHA) is used for classification. Finally, the Catch Fish optimization Algorithm (CFA) is used for the hyperparameter tuning process. A comparison study of the IDFSWSN-DRL method illustrated a greater accuracy (99.69%) over recent models on the WSN-DS dataset.

Keywords-wireless sensor networks; multi-head attention; intrusion detection; deep representation learning; attack; feature reduction

I. INTRODUCTION

The WSNs present a modern approach with diverse applications, using numerous compact nodes that communicate through ad-hoc networks and operate under constraints such as limited bandwidth, energy, storage, and processing power [1]. WSNs are used to collect data in inhospitable places and may be involved in major applications [2]. Defensive mechanisms are used to protect WSNs from several types of attacks [3]. Network IDSs can be used to identify security intrusions or attacks and secure WSNs [4]. The development of advanced devices and network technology produces extensive data that progressively reduces IDS detection rates [5]. In addition, IDSs are vital for user authorization, authentication, and addressing suspicious behaviors [6]. IDSs identify malicious unauthorized actions and protect the network, detecting intrusions that are subsequently utilized to prevent insider attacks. Several ML and DL models have been proposed for IDSs in WSNs [7].

This study presents an Intrusion Detection Framework for Securing Wireless Sensor Networks using a Deep Representation Learning (IDFSWSN-DRL) model. The key contributions are as follows:

- The initial pre-processing step ensures high-quality input data, mitigates the impact of outliers, and supports better model generalization.
- The GOA model is utilised for efficient FS, reducing dimensionality and improving classification performance. A novel TCN-MHA classification model is also developed to capture temporal patterns and contextual relevance.

- A bio-inspired metaheuristic, CFA, is applied for tuning to achieve optimal configuration, improving detection accuracy, convergence speed, and overall model robustness. Thus, the integration of CFA enhances adaptability and efficiency.
- The novelty lies in integrating the TCN-MHA with two distinct bio-inspired optimization models within a unified framework. This incorporation enables effective temporal extraction, attention-driven learning, and adaptive optimization.

II. RELATED WORKS ON SECURE WIRELESS SENSOR ENVIRONMENTS

In [8], a two-pronged method was proposed to understand the Restricted Boltzmann Machines (RBMs). In [9], a dynamically stabilized recurrent neural network was enhanced with an intensified sand cat swarm optimizer. This study also utilized the Adaptive Multi-Scale Improved Differential Filter (AMSIDF) and the Wolf-Bird Optimizer Algorithm (WBOA). In [10], a smart hybrid model integrated DL with optimization approaches. The Genetic Sacrificial Whale Optimizer (GSWO) [2] combines the Genetic Algorithm (GA) and WOA techniques. In [11], a Neural Network (NN) method was used to develop the Enhanced Wireless IDS (EW-IDS), using Singular Value Decomposition (SVD) and Principal Component Analysis (PCA) for FS. In [5], a Stacked Convolutional NN and Bidirectional LSTM (SCNN-Bi-LSTM) was presented, using Federated Learning (FL) for optimization. In [12], a smart IDS model used multi-objective PSO-based FS and a smart rule-based multi-class classification approach to detect intrusions with greater precision. Table I summarizes existing studies on intrusion detection in WSNs.

III. ALGORITHM AND SYSTEM DESIGN

This paper presents an IDFSWSN-DRL method that comprises data preprocessing, FS, classification, and the tuning process. Figure 1 depicts the workflow of the proposed IDFSWSN-DRL model.

TABLE I. SUMMARY OF KEY FEATURES AND PERFORMANCE METRICS OF THE EXISTING STUDIES FOR ID IN WSN

Study	Method	Dataset	Results	Limitations and future work
[2]	GSWO, GA, WOA, CatBoost	WSN-DS, WSNBFSF, NSL-KDD, CICIDS2017	Accuracy up to 99.99%, Inference Time 100× Faster	Limited adaptability to growing threats. Dynamic learning may be improved.
[5]	SCNN-Bi-LSTM, FL	WSN-DS, CIC-IDS-2017	Accuracy of ~99.9%	Limited resource efficiency. Future studies should optimize for lightweight deployment.
[8]	RBM, CAO, Fusion model	WSN Data	Improved results	Restricted real-world testing. The dataset may be improved in the future.
[9]	DSRNN-ISCSO-ID-WSN, AMSIDF, WBOA	WSN-DS Dataset	Accuracy of ~29–33%, Precision of ~26–31%.	Limited real-time validation; Enhance scalability and adaptability.
[10]	Smart Hybrid Model	WSN-DS Dataset	Values not specified	Restricted real-world testing. Generalization may be improved.
[11]	EW-IDS, SVD, PCA	IoT and WSN Datasets	Accuracy of 96%	Limited scalability. Future work may include real-time deployment and adaptability.
[12]	multi-objective PSO	KDD'99 Cup, CIDD	Improved Accuracy, Reduced False Positives	Limited adaptability. Detection of emerging attacks may be improved in the future.

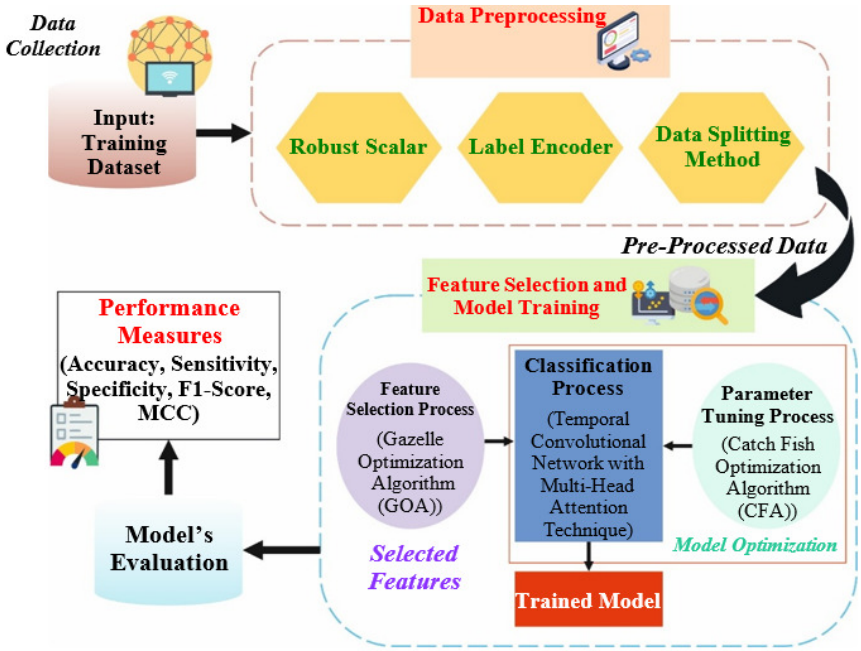

Fig. 1. Workflow of the proposed IDFSWSN-DRL model.

A. Data Preprocessing

The proposed IDFSWSN-DRL technique applies multiple steps for data preprocessing, including robust scaling, label encoding, and data splitting, to transform and structure the data effectively and improve overall accuracy [13]. The data pre-processing stage begins with RobustScaler, which scales features using the median and interquartile range (IQR), making it less sensitive to outliers compared to methods like MinMaxScaler or StandardScaler. The label encoder then converts the input categorical labels into a suitable format for the ML model. Finally, the data is split into training and testing sets to enhance efficiency and accuracy during model development.

B. GOA-Based Feature Reduction Procedure

The FS process is executed using GOA [14]. This method was selected for its robust global search ability, inspired by gazelles' intelligent escape strategies that effectively balance exploration and exploitation. GOA dynamically adapts to intrinsic feature spaces without needing transformation or prior assumptions, unlike conventional methods such as PCA or filter-based methods. This nature-inspired mechanism enables effective convergence towards minimal and relevant feature subsets, improving classification performance and mitigating computational overhead. GOA is a metaheuristic based on the social hierarchy and behavior of wild mountain gazelles. It simulates their cooperative predator escape, guiding the search for optimal solutions in four stages, namely random population initialization, global search, exploration phase, and escaping of the gazelle. In the initial stage, the model randomizes gazelles as search agents, where each candidate position $x_{i,j}$ is generated within upper and lower bounds using a random number as

$$x_{i,j} = rand \times \left(UB_j - LB_j\right) + LB_j \quad (1)$$

where $rand$ is a random number, and UB_j, and LB_j are the upper and lower limits. Each $x_{i,j}$ generates candidates, with the best selected.

In the second stage, if no hunters are closer or if they have not been searched, the model mimics the open gazelle movements applying Brownian Motion (BM). The location g is upgraded as:

$$g_{i+1} = g_i + S \cdot R \times R_B \times (X_i - R_B \times g_i) \quad (2)$$

where S represents the gazelle's movement speed, R_B is a randomly generated vector according to BM, and R is a random number in $(0,1)$.

In the third stage, this model mimics gazelle behavior with two stages: initially using Lévy flight (LF) and later BM, switching movement based on iteration progress.

$$g_{i+1} = g_i + S \cdot \varepsilon \cdot R \times R_L \times (X_i - R_L \times g_i) \quad (3)$$

where ε depicts the dual promising movement directions, capturing values in [-1, 1], and R_L refers to randomly generated vectors. During predator recognition, they transition to BM.

$$g_{i+1} = g_i + S \cdot \varepsilon \cdot CF \times R_B \times (X_i - R_L \times g_i) \quad (4)$$

$$CF = (1 - \frac{iter}{\text{Maxiter}})^{\left(2\frac{iter}{\text{Maxiter}}\right)} \quad (5)$$

CF characterises the cumulative effect of hunters, $iter$ is the present iteration, and $Maxiter$ denotes the maximum iteration count.

The survival rate of the gazelle near hunters is indeterminate, suggesting a potential for effective predation. Let P represent the predator's success rate, and the gazelle's escaping behavior is modelled mathematically based on

$$g_{i+1} = \begin{cases} g_i + CR[LB + R * (UB - LB)] \ (r \le P) \\ g_i + [P(1-r) + r]\left(g_{r_1} - g_{r_2}\right) \text{ (otherwise)} \end{cases} \quad (6)$$

where r is a random number in (0,1). The fitness function selects solutions with high accuracy and minimal feature count.

$$Fitness = \alpha * ErrorRate + (1 - \alpha) * \frac{\#SF}{\#All\_F} \quad (7)$$

$ErrorRate$ is the classifier's error proportion (0–1) using the chosen features. $\#SF$ denotes the selected features, $\#All\_F$ denotes the total features, and α balances accuracy and subset size.

C. Classification Using the TCN-MHA Technique

The proposed technique utilizes the TCN-MHA approach for the classification process [15], chosen for its ability to capture long-range temporal dependencies by utilizing TCN and MHA to focus on the most relevant features in each time step. TCN presents parallelism, stable gradients, and better sequence modeling, while MHA improves interpretability and precision by assigning dynamic importance to input features, compared to conventional RNNs or CNNs. This integration enhances detection accuracy in intrinsic intrusion patterns. TCNs capture long-term dependencies using causal convolutions, which preserve temporal order, and dilated convolutions, which efficiently expand the receptive field. The dilated convolution is defined by

$$F(s) = (X_d^*)f(s) = \sum_{i=0}^{k-1} f\,(i) * X_{s-d*i} \quad (8($$

Here, f is the convolution kernel, $F(s)$ the dilated convolution output, $i = 0,1,\ldots,k-1$ are kernel indices, k the filter length, d is the dilation factor, and $s_{t-d\cdot i}$ refers to past inputs, enabling long-range dependency capture. Residual connections in residual blocks address degradation from increased depth. The final output Y is given by

$$Y = F(x) + x \quad (9)$$

Residual connections ease gradient flow, stabilizing training. MHA dynamically weights key features by normalizing and summing them.

$$Attention(Q,K,V) = soft\max\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (10)$$

where K is the key matrix, A is the input matrix, V is the value matrix, and d_k is the dimension of $Q, K,$ and V. Dot-product attention uses softmax and scales by $\sqrt{d_k}$ for stability. MHA splits input into subspaces, applies self-attention, and concatenates the outputs.

D. CFA-Based Parameter Tuning

Finally, CFA is employed for the hyperparameter tuning process [20]. CFA is inspired by a simple fishing model that operates in two main stages: exploration and exploitation. This model dynamically explores the search space and adjusts based on feedback, unlike grid or random search, enhancing convergence to optimal hyperparameters. Its cooperative search mechanism allows efficient handling of intrinsic, multi-dimensional parameter spaces, resulting in improved model accuracy and stability. The exploration involves both dispersed individual searching and cooperative fishing for enhanced results, while exploitation focuses on coordinated encirclement to leverage collective strength. Each fisherman acts as a searching agent with positions represented by a matrix $Fisher$ of size $N \times d$, where N is the number of agents and d is the dimensionality:

$$Fisher_{i,j} = \left(ub_j - lb_j\right) * r + lb_j \quad (11)$$

where $Fisher_{i,j}$ is the i^{th} fisherman's position in the j^{th} dimension, ub_j and lb_j denote upper and lower limits, and r is an arbitrary number in (0, 1). Each fitness value is calculated by the fitness function $f_{objective}$, which guides the search for optimal solutions.

During exploration, fishermen alternate between independent searching and group encirclement, depending on the capture rate parameter α, which is influenced by fish availability and evaluation counts, as shown in

$$\alpha = (1 - \frac{3 \times EF_S}{2 \times \text{Max}EP_S})^{\frac{3 \times EP_S}{2 \times \text{Max}EF_S}} \quad (12)$$

If a random value p exceeds α, group capture is favored; otherwise, independent search occurs, where the position updates according to

$$Fisher_{i,j}^{T+1} = Fisher_{i,j}^{T} + \left(Fisher_{pos,j}^{T} - Fisher_{i,j}^{T}\right) \times Exp + r_s \times s \times R \quad (13)$$

Here, Exp measures relative fitness differences, r_s is a random number, s is a unit vector, and R scales with the distance between fishermen to enable adaptive, cooperative searching. In exploitation, fishermen cooperate to encircle prey, modeled by a Gaussian distribution centred on the best position, as

$$Fisher_i^{T+1} = G_{best} + GD\left(0, \frac{r_4 \times \sigma \times |(Fisher) - G_{best}|}{3}\right) \quad (14)$$

where GD represents the Gaussian distribution function. σ increases with assessment counts, reaching 0 from 1, and r_4 is a random integer between 1 and 3.

The CFA originates an FF to improve the classification outcome. It defines a progressive number to epitomize the higher result of the candidate's output. The FF examines the reduction of the classifier error rate, as:

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{\text{Number of misclassified instances}}{\text{Overall instances}} \times 100 \quad (15)$$

IV. RESULTS AND DISCUSSION

The experimental analysis of the IDFSWSN-DRL approach was examined using the WSN-DS dataset [17]. Table II describes the dataset. The total features are 18, out of which 13 features are selected. Figure 2 shows the classifier outputs of the IDFSWSN-DRL method on the test dataset, displaying the confusion matrices for all five classes at a 70% TRAPE/30% TESPE.

TABLE II. DATASET DESCRIPTION

Class labels	Sample numbers
Normal	340066
Blackhole	10049
Grayhole	14596
Flooding	3312
Scheduling Attacks	6638
Overall samples	**374661**

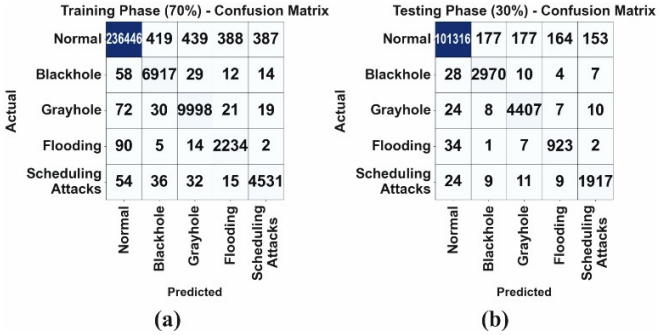

Fig. 2. Confusion matrices of the proposed IDFSWSN-DRL technique.

Table III depicts the experimental outputs of the IDFSWSN-DRL model compared to existing approaches [18-21]. The Binary Chimp Optimization Algorithm with ML-based ID (BCOA-MLID), Red Kite Optimization Algorithm with an Average Ensemble Model for ID (RKOA-AETD), AdaBoost, GB, XGBoost, KNN-AOA, KNN-PSO, LSTM, DNN, GWO-LSTM, and Coyote Optimization Algorithm with Global Search-based Improved Deep NN (COA-GS-IDNN) models attained lower results. However, the IDFSWSN-DRL technique outperformed the existing models with higher $accur_y$, $sensi_y$, $speci_y$, and $F_{measure}$ of 99.69%, 97.88%, 99.65%, and 95.27%, respectively.

TABLE III. COMPARISON OF THE PROPOSED IDFSWSN-DRL TECHNIQUE WITH EXISTING MODELS

Methods	$Accur_y$	$Sens_y$	$Spec_y$	$F_{Measure}$
IDFSWSN-DRL (Proposed)	99.69	97.88	99.65	95.27
BCOA-MLID [18]	99.47	96.31	99.22	94.11
RKOA-AETD [19]	98.99	75.41	96.51	79.58
AdaBoost [19]	96.30	96.56	95.76	90.90
GB [19]	95.08	95.94	94.89	94.02
XGBoost [19]	97.53	96.72	95.05	92.05
KNN-AOA [19]	97.89	96.28	97.13	90.85
KNN-PSO [19]	93.57	96.43	95.63	93.75
LSTM [20]	95.13	96.01	94.95	94.08
DNN [20]	97.58	96.78	95.11	92.12
GWO-LSTM [21]	97.97	96.34	97.18	90.92
COA-GS-IDNN [21]	93.64	96.49	95.68	93.81

V. CONCLUSION

This study aimed to develop an effective IDS for WSNs to enhance networking security by precisely detecting and mitigating malicious activities in real time. The method comprises data preprocessing, GOA-based FS, TCN-MHA-based classification, and CFA-based tuning. The proposed IDFSWSN-DRL approach was evaluated on the WSN-DS dataset, highlighting a superior accuracy of 99.69% and outperforming existing models. Limitations include restricted generalizability due to the specific dataset and scenarios tested, as well as potential computational complexity with massive datasets. Future work could explore more scalable algorithms, integrate real-time data processing, and validate the approach across diverse domains for improving applicability and robustness. Moreover, investigating automated feature selection methods may improve efficiency.

REFERENCES

[1] V. Gowdhaman and R. Dhanapal, "An intrusion detection system for wireless sensor networks using deep neural network," *Soft Computing*, vol. 26, no. 23, pp. 13059–13067, Dec. 2022, https://doi.org/10.1007/s00500-021-06473-y.

[2] T. M. Nguyen, H. H. P. Vo, and M. Yoo, "Enhancing Intrusion Detection in Wireless Sensor Networks Using a GSWO-CatBoost Approach," *Sensors*, vol. 24, no. 11, May 2024, https://doi.org/10.3390/s24113339.

[3] D. K. Madhuri, "A New Level Intrusion Detection System for Node Level Drop Attacks in Wireless Sensor Network," *Journal of Algebraic Statistics*, vol. 13, no. 1, pp. 159–168, May 2022.

[4] S. Pundir, M. Wazid, D. P. Singh, A. K. Das, J. J. P. C. Rodrigues, and Y. Park, "Intrusion Detection Protocols in Wireless Sensor Networks Integrated to Internet of Things Deployment: Survey and Future Challenges," *IEEE Access*, vol. 8, pp. 3343–3363, 2020, https://doi.org/10.1109/ACCESS.2019.2962829.

[5] S. M. S. Bukhari *et al.*, "Secure and privacy-preserving intrusion detection in wireless sensor networks: Federated learning with SCNN-Bi-LSTM for enhanced reliability," *Ad Hoc Networks*, vol. 155, Mar. 2024, Art. no. 103407, https://doi.org/10.1016/j.adhoc.2024.103407.

[6] N. M. Alruhaily and D. M. Ibrahim, "A Multi-layer Machine Learning-based Intrusion Detection System for Wireless Sensor Networks," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 4, 2021, https://doi.org/10.14569/IJACSA.2021.0120437.

[7] R. Zhang and X. Xiao, "Intrusion Detection in Wireless Sensor Networks with an Improved NSA Based on Space Division," *Journal of Sensors*, vol. 2019, no. 1, 2019, Art. no. 5451263, https://doi.org/10.1155/2019/5451263.

[8] J. Srivastava and J. Prakash, "Deep learning-enabled energy optimization and intrusion detection for wireless sensor networks," *OPSEARCH*, vol. 62, no. 1, pp. 368–405, Mar. 2025, https://doi.org/10.1007/s12597-024-00791-z.

[9] A. Punitha, P. Ramani, P. Ezhilarasi, and S. Sridhar, "Dynamically stabilized recurrent neural network optimized with intensified sand cat swarm optimization for intrusion detection in wireless sensor network," *Computers & Security*, vol. 148, Jan. 2025, Art. no. 104094, https://doi.org/10.1016/j.cose.2024.104094.

[10] K. P. Sharma *et al.*, "Hybrid Convolutional Neural Network for Robust Attack Detection in Wireless Sensor Networks," *Internet Technology Letters*, vol. 8, no. 6, 2025, Art. no. e650, https://doi.org/10.1002/itl2.650.

[11] B. Mopuru and Y. Pachipala, "Advancing IoT Security: Integrative Machine Learning Models for Enhanced Intrusion Detection in Wireless Sensor Networks," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 14840–14847, Aug. 2024, https://doi.org/10.48084/etasr.7641.

[12] S. Subramani and M. Selvi, "Multi-objective PSO based feature selection for intrusion detection in IoT based wireless sensor networks," *Optik*, vol. 273, Feb. 2023, Art. no. 170419, https://doi.org/10.1016/j.ijleo.2022.170419.

[13] S. Abbas *et al.*, "Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks," *PeerJ Computer Science*, vol. 10, Jan. 2024, Art. no. e1793, https://doi.org/10.7717/peerj-cs.1793.

[14] G. Li, H. Ge, Y. Jiang, Y. Zhang, and X. Jin, "Non-destructive detection of early wheat germination via deep learning-optimized terahertz imaging," *Plant Methods*, vol. 21, no. 1, May 2025, Art. no. 75, https://doi.org/10.1186/s13007-025-01393-6.

[15] W. Fu *et al.*, "Physics-Informed Temporal Attention Dynamic Convolution Network for Oil and Gas Well Production Forecasting." Social Science Research Network, June 20, 2025, https://doi.org/10.2139/ssrn.5312750.

[16] S. A. Alnefaie, A. Alkuhayli, A. M. Al-Shaalan, S. A. Alnefaie, A. Alkuhayli, and A. M. Al-Shaalan, "Optimizing Load Frequency Control of Multi-Area Power Renewable and Thermal Systems Using Advanced Proportional–Integral–Derivative Controllers and Catch Fish Algorithm," *Fractal and Fractional*, vol. 9, no. 6, May 2025, https://doi.org/10.3390/fractalfract9060355.

[17] "WSN-DS." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/bassamkasasbeh1/wsnds.

[18] M. Aljebreen *et al.*, "Binary Chimp Optimization Algorithm with ML Based Intrusion Detection for Secure IoT-Assisted Wireless Sensor Networks," *Sensors*, vol. 23, no. 8, Apr. 2023, https://doi.org/10.3390/s23084073.

[19] F. F. Alruwaili, M. M. Asiri, F. S. Alrayes, S. S. Aljameel, A. S. Salama, and A. M. Hilal, "Red Kite Optimization Algorithm With Average Ensemble Model for Intrusion Detection for Secure IoT," *IEEE Access*, vol. 11, pp. 131749–131758, 2023, https://doi.org/10.1109/ACCESS.2023.3335124.

[20] H. Zhang, D. Upadhyay, M. Zaman, A. Jain, and S. Sampalli, "SC-MLIDS: Fusion-based Machine Learning Framework for Intrusion Detection in Wireless Sensor Networks," *Ad Hoc Networks*, vol. 175, Aug. 2025, Art. no. 103871, https://doi.org/10.1016/j.adhoc.2025.103871.

[21] K. Sedhuramalingam and N. Saravanakumar, "A novel optimal deep learning approach for designing intrusion detection system in wireless sensor networks," *Egyptian Informatics Journal*, vol. 27, Sept. 2024, Art. no. 100522, https://doi.org/10.1016/j.eij.2024.100522.