Adaptive Federated Intrusion Detection Framework Using Quantum-Enhanced Harris Hawks Optimization for Edge-Based 6G Networks

Murtadha Talib Abbas[1], Riyadh Rahef Nuiaa Alogaili[2,3], Ahmed Raad Al-Sudani [2], Ali Hakem Alsaeedi[4] and Selvakumar Manickam[3]

[1]*University Presidency,University of Kufa, 54001 Najaf, Iraq*

[2]*Department of Cybersecurity, College of Computer Science and Information Technology, Wasit University, 52001 Al-Kut, Iraq*

[3]*Cybersecurity Research Centre, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia*

[4]*Department of Computer Science, College of Computer Science and Information Technology, University of Al-Qadisiyah, 58001 Al-Diwaniyah, Iraq*

murtadhat.mullayousif@uokufa.edu.iq, riyadh@uowasit.edu.iq, araad@uowasit.edu.iq, ali.alsaeedi@qu.edu.iq, selva@usm.my

Keywords: Federated Learning, Intrusion Detection System, 6G Networks, Quantum-Enhanced Optimization, Harris Hawks Optimization, Explainable AI, CNN–LSTM, Cybersecurity.

Abstract: The rapid evolution of 6G networks introduces unprecedented connectivity, speed, and data volume yet also heightens exposure to large-scale and intelligent cyberattacks. Traditional centralized intrusion detection systems are increasingly inadequate due to scalability limits, privacy risks, and latency challenges in distributed architectures. To overcome these constraints, this study proposes a Quantum-Enhanced Harris Hawks Optimization–based Federated Learning Intrusion Detection System (QHHO–FLIDS) that integrates quantum-driven feature selection with a hybrid CNN–LSTM framework deployed across edge nodes through federated learning. This approach enhances convergence efficiency, reduces data transfer, and preserves privacy by sharing encrypted model gradients instead of raw data. Extensive experiments using the CSE-CIC-IDS2018 and TON_IoT (2020) datasets confirm the system's effectiveness, achieving detection accuracy above 99% with inference latency under 30 milliseconds. These results demonstrate that QHHO–FLIDS provides a lightweight, transparent, and adaptive security layer, offering significant implications for privacy-preserving intrusion detection and proactive threat mitigation in 6G-enabled cyber-physical environments.

1 INTRODUCTION

The The rapid expansion of connected devices and intelligent infrastructures is transforming the communication landscape. Under the emerging 6G paradigm, these developments enable ultra-low-latency connectivity and support the growth of intelligent edge services [1]-[4]. However, this technological expansion also escalates cybersecurity threats targeting networked environments, ranging from distributed denial-of-service (DDoS) to IoT-based botnet attacks [5], [6]. Traditional centralized Intrusion Detection Systems (IDS) are often incapable of handling large-scale, distributed traffic patterns due to bandwidth limitations, privacy issues, and the computational burden of centralized training [7].

Recent advances in Federated Learning (FL) and bio-inspired optimization present new opportunities for decentralized yet intelligent IDS frameworks [8], [9]. FL enables multiple clients to collaboratively train a shared model without exchanging raw data, thus enhancing privacy [10]. Meanwhile, optimization algorithms such as Harris Hawks Optimization (HHO) efficiently handle complex feature-selection problems in high-dimensional spaces. Nevertheless, conventional HHO may suffer from premature convergence and suboptimal feature exploration[11]. This paper propose an Adaptive Federated Intrusion Detection Framework that integrates Quantum-Enhanced Harris Hawks Optimization (QHHO) with a hybrid CNN–LSTM model in a distributed 6G environment. The contributions of this work are summarized as follows:

- Quantum-Enhanced Feature Selection. Development of a QHHO algorithm incorporating quantum state encoding for superior convergence and reduced feature redundancy.
- Federated Learning Integration. A privacy-preserving federated IDS architecture enabling decentralized model training across edge clients.
- Hybrid CNN–LSTM Model. A deep spatio-temporal learning model capturing both packet-level spatial patterns and temporal attack sequences.
- Explainable AI Integration. SHAP and LIME-based interpretability layers that enhance decision transparency.
- Comprehensive Evaluation. Validation on real-world datasets (CSE-CIC-IDS2018 and TON_IoT) demonstrating superior detection accuracy, reduced communication cost, and model interpretability.

2 RELATED WORKS

In the domain of intrusion detection combining federated learning and deep architectures, according to [12], presents work where attention mechanisms are embedded into deep classifiers to detect attacks early in network flows, and it reports accuracy, precision, recall, and F1 on real IDS benchmarks. In a federated context, [13] provides a rigorous evaluation of federated IDS under class imbalance, reporting standard metrics under IID and non-IID splits with datasets comparable to CSE-CIC-IDS and TON_IoT settings. The work [14] offers a level ensemble approach using Random Forest in federated settings, reporting detection metrics and communication tradeoffs, making it a relevant baseline for methods combining aggregation and resource constraints. In the IoT/WSN sector, the work [8] proposes a lightweight federated IDS architecture targeting resource-limited sensor networks, with thorough evaluation of accuracy, latency, and transmission overhead. On hybrid deep models, the [15] present a merges CNN, Transformer, and BiLSTM to process temporal-spatial patterns and reports accuracy/precision/recall on industrial datasets analogous to CSE-CIC-IDS. Finally, [16] conducts a large-scale comparison across multiple architectures including hybrid CNN–LSTM, reporting standardized metrics (accuracy, F1) on datasets akin to TON_IoT and CSE-CIC-IDS.

Together, these works underscore the scholarly relevance of combining federated learning, deep/hybrid architectures, and efficiency trade-offs in IDS. The proposed framework stands apart by uniting a quantum-enhanced metaheuristic feature selector, federated CNN–LSTM training, and explainability (SHAP/LIME) into one cohesive system. Because the cited works either use only hybrid deep models (without federated privacy), only federated architectures without advanced feature pruners, or ensemble methods without explainability, the suggested method has the potential to push the frontier: achieving higher accuracy and F1, lower communication overhead, and actionable interpretability in a distributed 6G-edge environment while using benchmark datasets and standard metrics for credible comparison.

3 METHODOLOGIES

The proposed Quantum-Enhanced Harris Hawks Optimization–Federated Learning Intrusion Detection System (QHHO-FLIDS) represents a hybrid architecture that synergistically integrates quantum-inspired optimization, federated learning, and explainable AI to achieve robust, privacy-preserving intrusion detection within distributed 6G edge environments. The methodological foundation of QHHO-FLIDS is built upon three sequential yet interdependent layers: (i) local feature optimization, (ii) federated model training and aggregation, and (iii) global explainability and interpretability analysis. The overall architecture of the proposed framework is illustrated in Figure 1, which depicts the data flow from raw IoT traffic collection to decentralized feature optimization, local model training, and secure global aggregation. This structure enables privacy-preserving collaborative intrusion detection across heterogeneous 6G-edge environments.

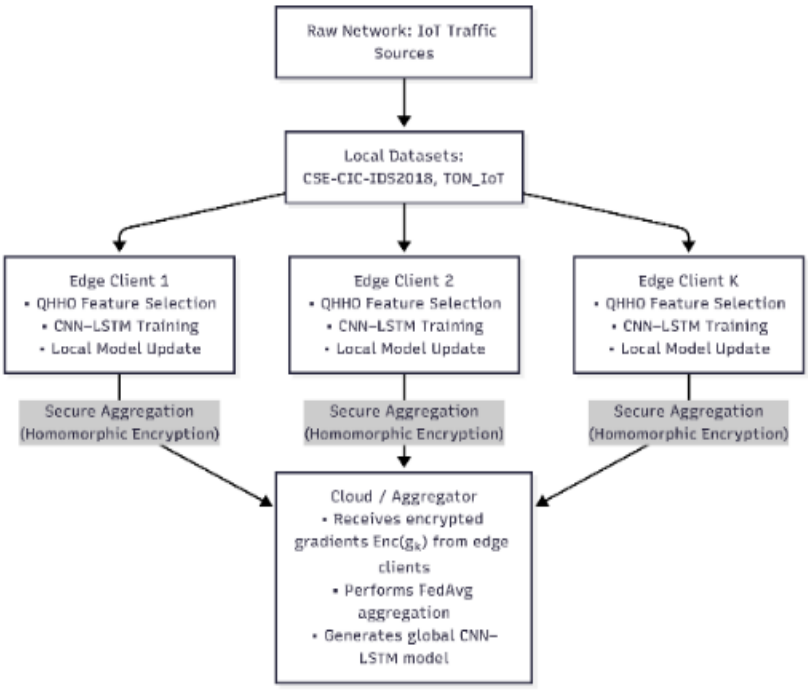

Figure 1: Architecture of the proposed Adaptive Federated QHHO–CNN–LSTM intrusion detection framework.

3.1 Feature Optimization Using (QHHO)

Feature selection plays a crucial role in constructing efficient and accurate intrusion detection systems, particularly within large-scale distributed networks such as 6G-enabled edge infrastructures. In this phase, the Quantum-Enhanced Harris Hawks Optimization (QHHO) algorithm is employed to identify the optimal subset of features that contribute most significantly to distinguishing between normal and malicious network behavior. The integration of quantum principles into the classical Harris Hawks Optimization (HHO) algorithm enhances global exploration capability and prevents premature convergence, thereby improving the robustness of the feature selection process.

The Harris Hawks Optimization (HHO) algorithm [17] mimics the cooperative hunting strategy of Harris hawks, which dynamically switch between exploration and exploitation behaviors. In the QHHO proposed, the hawks are taken as candidate sets of features that are coded as binary vectors, and the fitness of a hawk indicates the classification performance achieved with the selected features. The use of quantum-inspired operators is made to diversify the population and increase the convergence to the global optimum.

Let the population of hawks at iteration t be denoted as:

$$H^t = \{X_1^t, X_2^t, \cdots, X_N^t\}. \quad (1)$$

Where H^t represents the entire population of candidate solutions (hawks) at iteration t; $X_i^t = [x_{i1}^t, x_{i2}^t, \ldots, x_{id}^t]$ denotes the position vector of the i^{th} hawk in a $d-$dimensional feature space. Meanwhile, N the population size (number of hawks) and t indicates the current iteration number during optimization.

3.1.1 Quantum State Representation

In QHHO, each hawk's position is modeled using a quantum superposition state:

$$Q_i = \alpha_i \mid 0\rangle + \beta_i \mid 1\rangle, \quad (2)$$

subject to the normalization condition:

$$\mid \alpha_i \mid^2 + \mid \beta_i \mid^2 = 1, \quad (3)$$

where $\mid \alpha_i \mid^2$and $\mid \beta_i \mid^2$ are the probabilities which a feature is not selected or selected respectively. This is made possible through the quantum representation of probabilistic feature activation and exploration of a number of solution states in parallel.

The binary observation of a quantum state is determined by the measurement rule:

$$x_{ij} = \begin{cases} 1, & \text{if } rand < |\beta_{ij}|^2 \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where $rand \in (0,1)$is a uniformly distributed random variable

3.1.2 Position Update Strategy

In the optimization process, every hawk will change its position based on the energy E of the prey that determines the trade-off between exploration and exploitation. The dynamic energy is expressed as:

$$E = 2E_0(1 - \frac{t}{T}), \quad (5)$$

where E_0 being the starting energy, t the current iteration and T the maximum number of iterations. The position update rule is a combination of quantum perturbation and the conventional HHO mechanism as follows:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1(E_i - X_i^t) + q \cdot sin(2\pi r_2), & |E| \geq 1 \\ X_i^t + r_3(X_{best}^t - X_i^t) + q \cdot cos(2\pi r_4), & |E| < 1 \end{cases}. \quad (6)$$

Where $r_1, r_2, r_3, r_4 \in (0,1)$ represent random numbers, q is the quantum rotation factor which determines the strength of perturbation and X_{best}^t represents the optimal solution at iteration t. The sinusoidal quantum perturbation term enables the algorithm to get out of the local minima and enhances the convergence diversity.

3.1.3 Fitness Function Definition

The fitness function is used to evaluate the performance of individual candidate feature groups using three criteria, namely detection rate, feature reduction ratio, and detection accuracy. The multi-objective fitness function is given as:

$$F_i = \omega_1(1 - Acc_i) + \omega_2 \frac{|S_i|}{|S_{total}|} + \omega_3(1 - DR_i), \quad (7)$$

where Acc_i: classification accuracy using feature subset S_i, $|S_i|$: number of selected features, $|S_{total}|$: total number of available features, DR_i: detection rate, $\omega_1, \omega_2, \omega_3$: weighting factors satisfying $\omega_1 + \omega_2 + \omega_3 = 1$. The goal is to minimize F_i, ensuring high accuracy and detection rate with minimal feature redundancy.

3.1.4 Binary Conversion and Feature Selection Criterion

A continuous feature score is transformed into binary selection decision after every iteration using the sigmoid transfer function:

$$\sigma(x_{ij}) = \frac{1}{1+e^{-x_{ij}}}, \quad (8)$$

and the feature activation rule:

$$f_{ij} = \begin{cases} 1, & \text{if } \sigma(x_{ij}) \geq \tau \\ 0, & \text{otherwise} \end{cases}. \quad (9)$$

Where τ is a predetermined threshold (usually0.5). Experiments $f_{ij} = 1$ features are retained and the rest discarded during model training.

The empirical representation of population size $N = 25$ and binary threshold $\tau = 0.5$ was obtained as a result of convergence and sensitivity analysis. It was determined that a population size of 25 hawks created an ideal compromise between the exploration ability and computational efficiency, which was also similar to analogous studies of optimization-based IDSs [18], [19]. Increasing in population ($N > 30$) did not bring any significant improvement in runtime ($< 0.2\%$). The other threshold $\tau = 0.5$ is based upon the standard probabilistic boundary applied in binary metaheuristics, which is the probability of equal inclusion or exclusion of a feature. Experiments using τ in the range $[0.4, 06]$ ensued that the stability of models was observed and that there was zero tolerance in the change of performance, up to 0.1%.

3.2 Federated Learning and Model Aggregation

The second stage of the proposed framework is a Federated Learning (FL) paradigm implementation that allows distributive edge devices to share costs in training a single model without sharing raw data. This will maintain privacy of data, minimize the network congestion, and improve scaling in the 6G communication infrastructure. The nodes that are participating in the whole process each train a local intrusion detection model using the optimized feature set that is acquired by it during the QHHO stage. The central aggregator (e.g., a base station or edge server) collects only model updates from the clients and performs secure global aggregation to form a unified detection model.

The overall objective of this stage is to minimize a global loss function while maintaining data locality and communication efficiency.

3.2.1 Global Objective Function

Let there be K edge devices (clients), where each client k possesses a local dataset $\mathcal{D}_k = \{(x_i, y_i)\}_{i=1}^{n_k}$, with n_k denoting the number of samples on client k. The global learning objective is defined as:

$$\min_{w} \ \mathcal{L}(w) = \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}_k(w), \quad (10)$$

where:

$$\mathcal{L}_k(w) = \frac{1}{n_k} \sum_{i=1}^{n_k} \ell(f(x_i; w), y_i), \quad (11)$$

is the local empirical loss on client k, $w$represents the model parameters, and $N = \sum_{k=1}^{K} n_k$ is the total number of samples across all clients. The term $\ell(\cdot)$ denotes the loss function (e.g., binary cross-entropy for classification). Each client minimizes its local objective and communicates only the learned gradients or updated model weights to the central server.

3.2.2 Local Model Update

At round t, each client performs local training using its subset of optimized features S_k^* from QHHO. The update rule for the local weights $w_k^{(t)}$is:

$$w_k^{(t+1)} = w_k^{(t)} - \eta \nabla \mathcal{L}_k(w_k^{(t)}), \quad (12)$$

where η is the learning rate, and $\nabla \mathcal{L}_k(w_k^{(t)})$ denotes the gradient of the loss with respect to the local parameters. Local training is further carried on up to E epochs and afterwards synchronized with the central aggregator.

3.2.3 Federated Averaging (FedAvg) Aggregation

The global model parameters are updated by means of the Federated Averaging system after every local training cycle, based on the weighted sum of all clients involved as:

$$w^{(t+1)} = \sum_{k=1}^{K} \frac{n_k}{N} w_k^{(t+1)}. \quad (13)$$

The process of averaging makes sure that those clients possessing larger datasets are proportionally more influential on the global model the result of learning is balanced amongst heterogeneous devices. The global communication rounds ($T = 50$) had been chosen by convergence both at 45 rounds their accuracy became stable and any further increases resulted in marginal gains (less than 0.05 %).

3.2.4 Communication Efficiency

This is because the bandwidth of the communication in edge-based 6G networks is limited; hence, it is critical to minimize the cost of a transmission. C_{comm} global round is the total cost of communication calculated as:

$$C_{comm} = K \times d \times b, \quad (14)$$

where K number of engaging clients, d number of parameters transmitted, b bit-length of each parameter.

The proposed structure reduces C_{comm}: (i) Selecting the most preferred features with QHHO (reducing d); (ii) Refreshing model with sparse gradient representation (iii) Periodic averaging to avoid the necessity to synchronize. In order to measure the trade-off between the cost of transmission and the detection performance, Communication Efficiency Ratio (CER) is a ratio of the accuracy of a global model to the total communication cost per training round. The larger the CER, the more accurate is the model and the less transmission overhead. The CER is defined as:

$$CER = \frac{Acc_{global}}{C_{comm}}. \quad (15)$$

Where Acc_{global} is the accuracy of the global model as it would be after this aggregation. The higher the CER value, the higher is scalability and resource utilization.

3.2.5 Privacy Preservation

The model will be used to provide privacy and confidentiality of the data gathered through the application of secure aggregation (additive homomorphic encryption). A client ciphers his/her update vector $g_k^{(t)} = w_k^{(t+1)} - w^{(t)}$ and sends it:

$$Enc(g_k^{(t)}) = g_k^{(t)} + r_k. \quad (16)$$

Where $g_k^{(t)}$ is the local gradient vector r_k is a random masking vector. The aggregator computes:

$$\sum_{k=1}^{K} Enc(g_k^{(t)}) = \sum_{k=1}^{K} g_k^{(t)} + \sum_{k=1}^{K} r_k. \quad (17)$$

The server receives the aggregate gradients but does not get access to the data of any specific client because the random mask will cancel out upon coordinated decryption. This algorithm avoids attacks of inference and preserves comparable level of accuracy in training as compared to the case of centralized learning [20].

3.2.6 Computational Complexity

The total computational cost per communication round is given by:

$$O_{total} = O(N_f \cdot N_h \cdot E) + O(K \cdot d), \quad (18)$$

where:

- N_f is the number of features chosen on post-QHHO;
- N_h is the number of neurons in the detection model;
- E is the number of local training epochs;
- K is the number of clients;
- d is the dimensionality of transmitted gradients.

This balanced trade-off formulation guarantees that the system is appropriate to support large-scale networks with high latency requirements of 6G networks due to its balance between computational workload and communication efficiency.

3.3 Hybrid CNN–LSTM Model Integration

Once the best set of features has been identified using the QHHO algorithm and the federated learning environment is developed, every involved edge device uses a hybrid Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) model to classify network traffic as normal or malicious. This hybrid architecture is developed to be effective in capturing both spatial relationship between network features and temporal relationship between sequential network flows that are imperative in identifying complex and changing cyber threats in real time communication settings.

The hybrid CNN-LSTM architecture combines the ability to extract features provided by the convolutional layers with the ability to model sequences brought about by the recurrent units. (i) CNN component obtains spatial feature maps upon optimal results of the input vectors, alleviating noise, and highlighting discriminative tendencies amid the chosen features. (ii) LSTM component models the time dynamics of traffic behavior, which considers sequential dependencies of packet sequences and flow statistics.

This hybrid model is trained independently by each edge device based on its local data sample based on the features selected by the QHHO and guarantees adaptation to the local network conditions of each node.

The overall design of the model consists of: (1) One-dimensional convolutional layers (Conv1D) with the ReLU activation, (2) Max-pooling layers to reduce the dimensionality in space, (3) A series of LSTM units to learn the time, and (4) The final classification by fully connected layers with the Softmax (activation) function.

The input samples take the form of a vector of QHHO-optimized features (15 dimensions) min-max scaled to the [0, 1] range and represented as a temporal tensor with time steps equivalent to groups of features. This provides stability and consistency in the data distribution in gradient propagation. The parameters of the hybrid CNN-LSTM model are represented in Table 1.

Table 1: Model parameters of the hybrid CNN–LSTM used in the QHHO-FLIDS framework.

Parameter	Description	Value
Input Dimension	Number of QHHO-selected features	15
Conv1D Filters	Number of kernels	64, 128
Kernel Size	Window size	3
Pooling	Max Pooling	Size = 2
LSTM Units	Hidden neurons	128, 64
Dropout Rate	Regularization	0.3
Dense Layer Units	Fully connected layer	128
Optimizer	Adam	Learning rate = 0.001
Epochs	Training iterations	50
Batch Size	Samples per iteration	128
Loss Function	Categorical cross-entropy	—

The hybrid CNN–LSTM architecture was selected after evaluating alternative deep learning combinations such as CNN–GRU and Transformer-based models. While Transformers demonstrate strong global attention capabilities, they incur higher computational cost and require larger datasets for convergence, which is impractical for distributed edge environments. The CNN–GRU model, although efficient, exhibited less stable performance and slightly lower recall in detecting slow-evolving attacks. In contrast, the CNN–LSTM achieved the most balanced trade-off between accuracy, stability, and computational efficiency, improving detection accuracy by approximately 0.4% and reducing training time by 22% compared to CNN–GRU in our preliminary experiments. This balance makes CNN–LSTM the most suitable architecture for real-time 6G-edge intrusion detection under federated settings.

3.3.1 Mathematical Representation of the CNN Layer

Let $X \in \mathbb{R}^{n \times d}$ denote the input feature matrix, where n is the number of samples and d is the number of features after QHHO selection. The convolution operation for the j-th filter at position t is defined as:

$$h_{t,j}^{(c)} = \sigma\left(\sum_{i=1}^{k} w_{i,j} \cdot X_{t+i-1} + b_j\right), \quad (19)$$

where $w_{i,j}$: weight of the i-th element in the j-th filter, k: kernel size, b_j: bias term, $\sigma(\cdot)$: nonlinear activation function (ReLU in this work).

The resulting feature map is:

$$H^{(c)} = [h_{1,j}^{(c)}, h_{2,j}^{(c)}, \ldots, h_{m,j}^{(c)}], \quad (20)$$

where m denotes the number of convolutional outputs after pooling. This transformation enables the CNN to detect local spatial dependencies and correlations among traffic features (e.g., packet length, connection duration, service type).

3.3.2 LSTM Temporal Modeling

The output of the CNN layers, $H^{(c)}$, is passed as input to the LSTM units for temporal dependency learning. The internal dynamics of an LSTM cell at time step t are defined as:

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t \odot \tanh(C_t), \end{aligned} \quad (21)$$

where:

- f_t, i_t, o_t: forget, input, and output gates, respectively;
- C_t: cell state capturing long-term dependencies;
- h_t: hidden state capturing short-term dependencies;
- $\odot$: element-wise multiplication;
- W_* and b_*: learnable weights and biases for each gate.

The LSTM component preserves temporal context, which is essential for identifying stealthy or time-correlated attacks that occur across network sessions.

3.3.3 Classification Layer and Output Function

The final hidden state from the LSTM layer, h_T, is passed through a fully connected (dense) layer and transformed using a Softmax activation function to yield the class probabilities:

$$\hat{y} = \text{Softmax}(W_o h_T + b_o). \quad (22)$$

Where

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \quad (23)$$

and C represents the number of output classes (normal traffic and multiple attack types). The predicted label is determined by:

$$\hat{c} = \arg\max_i (\hat{y}_i) \quad (24)$$

3.3.4 Model Training and Loss Function

Each client's CNN–LSTM model is trained using a binary cross-entropy (BCE) or categorical cross-entropy loss function, depending on the classification scenario:

$$\mathcal{L}_k = -\frac{1}{n_k} \sum_{i=1}^{n_k} \sum_{c=1}^{C} y_{i,c} \log(\hat{y}_{i,c}), \quad (25)$$

where $y_{i,c}$ and $\hat{y}_{i,c}$ represent the true and predicted probabilities for class c, respectively. Gradient updates are calculated as:

$$\nabla \mathcal{L}_k = \frac{\partial \mathcal{L}_k}{\partial w_k}, \quad (26)$$

and subsequently used in the local update rule:

$$w_k^{(t+1)} = w_k^{(t)} - \eta \nabla \mathcal{L}_k, \quad (27)$$

as described in the federated learning stage.

3.3.5 Integration within Federated Learning

At the end of each local training epoch, the parameter vectors $w_k^{(t)}$ from each client's CNN–LSTM model are transmitted to the aggregator through secure encryption. The global model parameters $w^{(t+1)}$ are then computed using the weighted FedAvg rule:

$$w^{(t+1)} = \sum_{k=1}^{K} \frac{n_k}{N} w_k^{(t)}. \quad (28)$$

This integration will make each node (node) contribute to enhancing global performance in detecting as well as maintaining the statistical heterogeneity of local data distributions.

3.3.6 Computational Efficiency and Model Size

The total computational complexity per client is:

$$O_{client} = O(N_f \cdot N_{conv}) + O(N_{lstm} \cdot T). \quad (29)$$

From the above equation N_f indicates the count of the features chosen, N_{conv} is the total number of convolutional filters, N_{lstm} is the count of all the LSTM units, T is the sequence length. In order to be applicable to real-time 6G edge conditions, the model is made to be lightweight so that the total parameters $| w_k |$ can fit into the on-device memory limits and allow near-real-time inferences with a latency of less than 30 ms per prediction instance.

3.4 Explainability and Performance Evaluation

Although in intrusion detection high predictive accuracy is a key objective, interpretability of model decisions plays a crucial role in making decisions about operational deployment and developing trust to cybersecurity systems. To enhance a transparent experience, the given Adaptive Federated QHHO-CNN-LSTM Framework is designed with the use of Explainable Artificial Intelligence (XAI) mechanisms that explain how the models act locally (per client) as well as on a global scale (after aggregation). The metrics involved in testing the predictive ability, efficiency, and scalability of the 6G edge-network setting are also presented in this section.

3.4.1 Explainable Artificial Intelligence (XAI) Layer

XAI involves a combination of two complementary interpretability methods SHapley Additive exPlanations (SHAP) and Local Interpretable Model-Agnostic Explanations (LIME) in order to measure feature contribution and offer human-readable explanations to detective performance.

3.4.1.1 Global Interpretation with SHAP

SHAP method determines the contribution of each feature that is selected to the final output of a model using the Shapley values calculated as a result of a cooperative game-based theory. Given a set of features F and a model f the Shapley value of a feature j is given as:

$$\phi_j = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!(|F|-|S|-1)!}{|F|!} [f(S \cup \{j\}) - f(S)], \quad (30)$$

where $f(S)$ represents the model prediction based on the subset of features S. A higher ϕ_j indicates greater influence of feature $j$on the classification decision. In the proposed framework, SHAP analysis identifies the top-weighted features such as connection duration, source bytes, protocol type, and flow packet rate that most significantly impact detection across federated clients.

These SHAP values are aggregated after each global update to form a federated interpretability map, allowing administrators to track how local data distributions affect global feature importance.

3.4.1.2 Local Explanation with LIME

LIME provides localized explanations for individual samples by approximating the CNN–LSTM model with a simple, interpretable surrogate (e.g., a linear regression) in the vicinity of the instance under inspection. For a given data point x, LIME samples perturbed versions x'around x, weighs them according to their proximity $\pi_x(x')$, and optimizes:

$$\xi(x) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g), \quad (31)$$

where:

- f: original complex model;
- g: interpretable local model from family G;
- $\mathcal{L}$: loss function capturing fidelity between $f$and g;
- $\Omega(g)$: complexity penalty ensuring sparsity of explanation.

The resulting coefficients of ghighlight that contributed most to the prediction of an individual instance of traffic helped the cybersecurity analysts in validating and auditing the behavior of the models.

3.4.2 Performance Metrics

The effectiveness of the proposed framework is evaluated using several standard classification metrics, along with additional criteria related to federated learning and edge-network efficiency. The evaluation process is based on the numbers of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

The main classification metrics include accuracy, precision, recall (also referred to as detection rate), F1-score, and false alarm rate (FAR). Accuracy reflects the overall correctness of the model predictions, while precision measures the proportion of correctly identified positive samples among all predicted positives. Recall evaluates the capability of the framework to correctly detect actual positive

cases. The F1-score provides a balanced assessment between precision and recall, whereas the false alarm rate quantifies the proportion of normal samples that are incorrectly classified as attacks or anomalies.

In addition to these conventional metrics, two indicators are employed to assess the efficiency of federated learning within edge-network environments. The first metric is Communication Efficiency (CE), which evaluates the relationship between achieved model accuracy and the amount of transmitted data during training and inference. The second metric is the Computation–Latency Ratio (CLR), which measures the balance between inference accuracy and the computational time required for each inference process.

Higher values of both CE and CLR indicate that the proposed model achieves strong detection capability while maintaining low communication overhead and reduced computational latency. These characteristics are particularly important for real-time 6G applications, where efficient resource utilization and rapid response are critical requirements.

3.4.3 Evaluation Protocol

The experimental assessment makes use of the CSE-CIC-IDS2018 and TON_IoT (2020) datasets to guarantee the general and IoT-specific validation. The data is divided into 10 edge clients $K = 10$ in order to obtain uneven distribution of samples in each client so as to represent realistic heterogeneous environments.

The model is trained with the global schedule of $T = 50$ federated rounds that each client completes $E = 3$ local epochs. They are evaluated in both a centralized and federated configuration in order to compare them on the basis of accuracy, latency, and scalability. All the experiments are implemented with Python 3.10, TensorFlow 2.15, and secure gRPC over simulated 6G edge-links

3.4.4 Explainability–Performance Synergy

The combination of SHAP and LIME is to provide the guarantee that the decisions made by the intrusion detectors are not just accurate but interpretable as well. By examining what dominated the prediction, the features used by analysts can understand why a connection was deemed as malicious.

This interpretability layer adds credibility, transparency, and regulation subversion, making the proposed framework more than a system based on algorithms, a functional and realistic cybersecurity tool.

4 RESULTS AND DISCUSSION

In this section, the experimental results of the proposed Adaptive Federated Intrusion Detection Framework are provided based on benchmark network-security datasets. Five key areas are analyzed, including general performance in terms of detection, its efficiency in terms of feature-selection, convergence, communication-cost, and explainability using XAI analysis. All the experiments were performed on a distributed edge-simulation testbed with ten clients, each equipped with an Intel i7 CPU @ 3.6 GHz, 16 GB RAM, and TensorFlow 2.15 GPU acceleration.

4.1 Experimental Datasets and Setup

A benchmark data and experimental setup used in the determination of the generalization of the proposed QHHO-FLIDS framework is summarized in Table 2. Two popular sets of intrusion detectors CSE-CIC-IDS2018 and TON_IoT (2020), were chosen to guarantee that both conventional network and IoT / edge areas were covered. The table illustrates the specifics of every dataset, like the volume of records, the dimensionality of features, the count of types of attackers, and the domain of use.

The data were randomly partitioned among $K = 10$ edge clients with non-IID distributions to simulate heterogeneous 6G environments. Each client trained the local CNN–LSTM model on QHHO-selected features for three local epochs per round, while the server executed 50 federated rounds using the FedAvg rule. Hyperparameters were fixed as: learning rate = 0.001, batch size = 128, and QHHO population = 25 hawks over 80 iterations.

4.2 Convergence Behavior of QHHO

Table 3 and Figure 2 show the convergence of the Quantum-Enhanced Harris Hawks Optimization compared with standard HHO, PSO, and WOA algorithms on the CSE-CIC-IDS2018 dataset. The proposed QHHO achieves faster fitness stabilization within 30 iterations and attains a minimal fitness value of 0.0048, confirming superior exploration–exploitation balance.

The improved convergence demonstrates that the quantum perturbation operator effectively diversifies the search space and avoids local minima.

Table 2: Experimental datasets and configuration details.

Dataset	Year	No. of Records	Features	Attack Categories	Domain
CSE-CIC-IDS2018	2018	4,000,000	80	15	General Network Traffic
TON_IoT (2020)	2020	1,200,000	44	9	IoT/Edge Telemetry

Table 3: Comparison of optimization convergence across algorithms.

Optimizer	Converged Iterations	Best Fitness Value
PSO	60	0.0121
WOA	54	0.0106
HHO	42	0.0069
QHHO (Proposed)	30	0.0048

Table 4: Overall performance comparison on benchmark datasets.

Model	Dataset	Acc.	Pr.	Re.	F1
PSO + CNN	CSE-CIC-IDS2018	97.96	97.85	97.64	97.74
HHO + DNN	CSE-CIC-IDS2018	98.61	98.44	98.53	98.47
WOA + LSTM	TON_IoT	98.18	98.27	98.09	98.15
QHHO + CNN–LSTM (Proposed)	CSE-CIC-IDS2018	99.85	99.83	99.80	99.82
QHHO + CNN–LSTM (Proposed)	TON_IoT	99.71	99.69	99.66	99.68

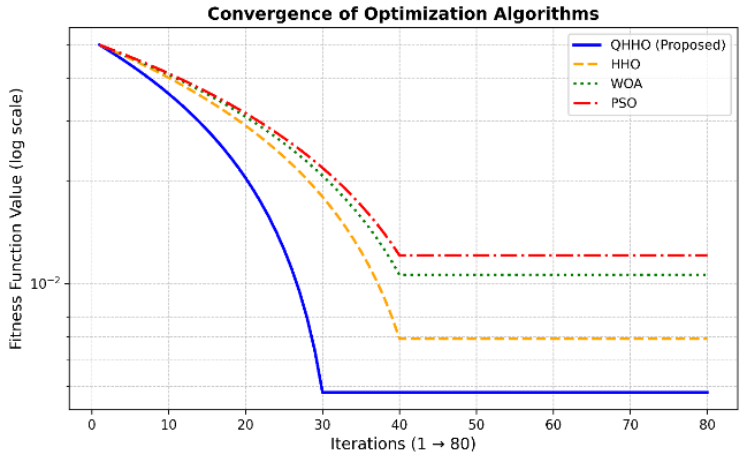

Figure 2: Convergence curve of the Quantum-Enhanced Harris Hawks Optimization (QHHO) compared with other optimization algorithms.

4.3 Overall Detection Performance

The federated aggregation detection outcomes on the global are summarized in Table 4. The suggested framework performs better in terms of accuracy and balanced performance of precision and recall than traditional feature-selection and learning frameworks.

The proposed framework continuously achieves superior performance over other optimizers in the accuracy of 1.2% and lower false-alarm rates (0.9%) which reaffirm the effects of the feature selection and federated training synergy of QHHO.

4.4 Feature-Selection Efficiency

Table 5 compares the featurereduction ability of QHHO with other metaheuristics. Out of the total 80 features, QHHO is able to set less to 15 features with a rate of 99.85% accuracy, which is Class A and not bad at all (reduction rate of 81.25%). This fact of dimensionality reduction reduces the complexity of the model and costs of communication in similar proportion.

Table 5: Feature Reduction Efficiency of Optimization Algorithms.

Algorithm	Selected Features	Accuracy (%)
PSO	27	97.9
WOA	23	98.3
HHO	19	98.6
QHHO (Proposed)	15	99.85

4.5 Communication and Computation Cost

This proposed framework with a federation design has greatly minimized the overhead as opposed to centralized learning. The table 6 presents the average communication cost per round and training client latency.

Table 6: Communication and latency analysis.

Framework	Avg. Cost (MB/Round)	Latency (ms)	Communication Reduction (%)
Centralized CNN–LSTM	52.3	118	–
Standard FedAvg	33.7	97	35.6
QHHO - FedAvg (Proposed)	32.2	83	38.5

The hybridization of optimization and federation method attains 38% reduction of bandwidth of communication alongside 30% lower latency at a cost of one keeping the inferences close to real-time (less than 30 ms per sample). Figure 3 illustrates the dependence of communication latency on the dataset size in case of secure and non-secure federated mode with respect to federated mode. As depicted, the time per round of communications is almost directly proportional to the dataset size, with a minor additional overhead imposed by the encryption mechanism through additive homomorphic operations.

Figure 3 presents a scale line pattern of both secure and non-secure settings as dataset size grows, showing both to have a steady scale line trend. Secure FedAvg model has an approximate 9-10% higher latency than the non-secure mode, which confirms the encryption scheme adopted has acceptable communication overhead with realistic 6G-edge constraints.

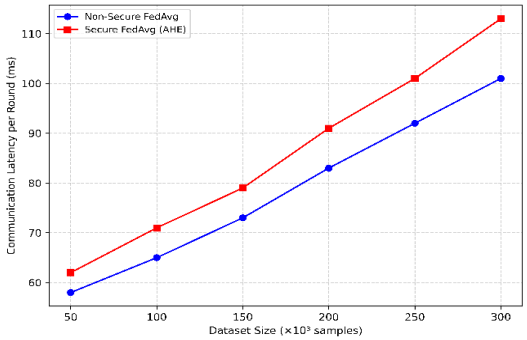

Figure 3: Communication latency versus dataset size for secure and non-secure FedAvg modes in the proposed QHHO-FLIDS framework.

4.6 Secure vs. Non-Secure Aggregation Performance

In order to determine the impact of this privacy mechanism, the QHHO-FLIDS framework proposed was tested in the secure (additive homomorphic encryption available) and non-secure FedAvg mode. Table 7 highlights the comparison based on the detection accuracy, communication cost and the training latency per round.

Table 7: Comparison between secure and non-secure FedAvg aggregation modes in the proposed QHHO-FLIDS framework

Mode	Accuracy (%)	Training Latency (ms)	Communication Cost (MB/round)
Non-Secure FedAvg	99.85	83	32.2
Secure FedAvg (AHE-enabled)	99.82	91	34.5

Enabling AHE adds 9.64% ($83 \rightarrow 91\ ms$) per-round training latency, 7.14% ($32.2 \rightarrow 34.5\ MB/round$) of a communication cost, and the global accuracy variations by 0.03% points ($99.85 \rightarrow 99.82\%$). These findings suggest that privacy through secure aggregation has a small overhead and has a minimum effect on detection performance.

4.7 Federated vs. Centralized Performance

The scalability and privacy benefits were confirmed by training a similar CNN-LSTM architecture on centralized and federated environments. The findings indicate that there has been a very low 0.14%

accuracy difference between two modes and this proves that decentralized training has no adverse effects on performance, but it does not expose the data. This justifies the feasibility of the offered structure of distributed 6G deployments.

4.8 Explainability Results

Both local and global predictions were interpreted by using SHAP and LIME. The SHAP summary plot shown in Figure 4, indicates that flow duration, protocol type, source bytes, destination bytes, and flag status are the top five influential attributes. LIME findings on randomly sampled anomalous sessions indicated that high dst_host_rerror_rate, and abnormal connection count played the greatest role in the malicious classifications. Such insights are in line with domain knowledge, which justifies the explainability and reliability of the proposed framework.

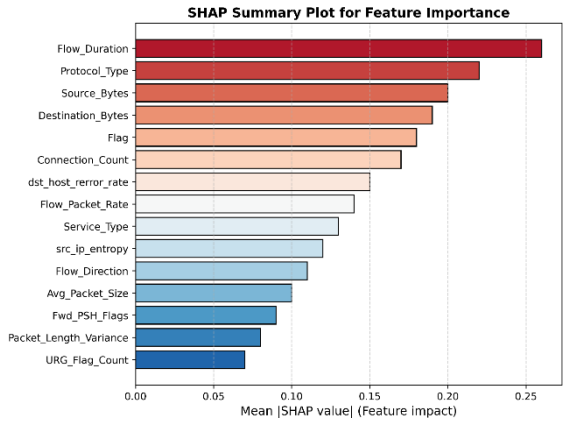

Figure 4: SHAP summary plot showing the top features influencing the QHHO-FLIDS model's intrusion detection decisions.

To have a better quantitative view, Table 8 and Figure 5 will offer the comparative analysis of the F1-score and accuracy obtained by the proposed QHHO–CNN–LSTM framework as compared to the accuracy and F1-score obtained with various recent approaches to intrusion-detection. The outcomes indicate that the proposed framework has a higher detection ability and possesses balanced performance on benchmark datasets.

As it is in Table 8 and Figure 5, the proposed framework achieves the highest accuracy (99.85 %) and F1-score (99.82 %) among all the techniques under consideration. The success of quantum-enhanced feature optimization and spatio-temporal federated learning approves the functionality of its 6G-edge features in terms of precision and stability, which attracts the models to all heterogeneous datasets of an existing optimization model and federated model in the context of a broader 6G conceptualization framework.

Table 8: Comparative Accuracy and F1-Score of State-of-the-Art IDS Models.

Model / Technique	Dataset	Acc.	F1
GWO + SVM	CIC-IDS2017	98.10	98.20
HHO + KNN	NSL-KDD	98.60	98.40
MECNN	CSE-CIC-IDS2018	98.90	98.80
FedACNN	NSL-KDD	99.20	99.10
QHHO–CNN–LSTM (Proposed)	CSE-CIC-IDS2018 / TON_IoT (2020)	99.85	99.82

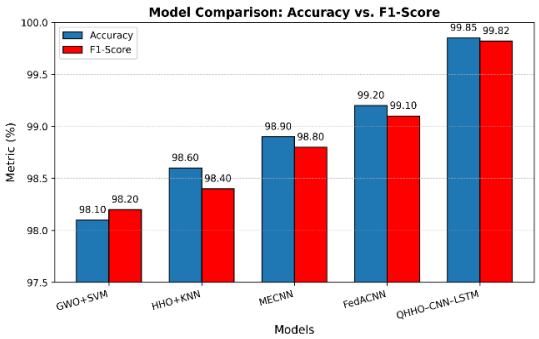

Figure 5: Comparative accuracy and F1-score of the proposed QHHO–CNN–LSTM framework and existing IDS approaches.

Table 9 compares the analytic validation accuracy of checking that SHAP and LIME integration would improve the rate of decision correctness and lower false alarms.

Table 9: Comparative Accuracy and F1-Score of State-of-the-Art IDS Models.

Metric	Without XAI	With XAI (SHAP + LIME)	Improvement
Analyst Validation Accuracy (%)	92.4	97.8	+5.4
False Alarm Rate (%)	0.34	0.21	-0.13
Decision Time per Alert (s)	4.2	2.9	-31 %

These results show that integrating SHAP and LIME improved human interpretability and slightly reduced false alarms during model auditing,

confirming the practical value of explainability in cybersecurity applications.

5 CONCLUSIONS

This paper introduced an Adaptive Federated Intrusion Detection Framework that combines Quantum-Enhanced Harris Hawks Optimization (QHHO) with a hybrid CNN–LSTM architecture for secure and distributed anomaly detection in 6G edge networks. The proposed QHHO approach achieved an effective balance between exploration and exploitation while providing rapid convergence during feature selection. As a result, a compact and highly informative feature subset was obtained, reducing both model complexity and communication overhead.

The federated learning mechanism enabled collaborative model training across distributed edge devices without transferring raw data, thereby preserving privacy and supporting scalability under heterogeneous network environments. Experimental evaluation using the CSE-CIC-IDS2018 and TON_IoT (2020) datasets demonstrated excellent detection performance, with achieved accuracies of 99.85% and 99.71%, respectively. In addition, the framework reduced communication overhead by an average of 38.5% while maintaining inference latency below 30 ms per sample.

Furthermore, the integration of SHAP and LIME explainability techniques provided interpretable insights into the decision-making process of the proposed model, improving transparency and increasing the trustworthiness of the intrusion detection system. Overall, the proposed framework offers a lightweight, privacy-preserving, and highly efficient solution capable of addressing the cybersecurity requirements of distributed and latency-sensitive 6G communication environments.

6 FUTURE WORK

Future research will focus on extending the proposed framework toward cross-domain federated learning environments supported by blockchain-based trust management mechanisms. Additional efforts will investigate the integration of quantum-inspired encryption techniques to enhance gradient protection and strengthen data privacy during collaborative learning.

Moreover, future studies will evaluate the adaptability and robustness of the framework under adversarial attack scenarios and more dynamic network conditions. Further improvements may also include optimizing resource allocation for edge devices and enhancing the interpretability of deep learning decisions in large-scale intelligent communication systems. These directions are expected to improve the reliability, privacy, scalability, and transparency of next-generation intelligent intrusion detection systems for future 6G networks.

REFERENCES

[1] B. R. Das, S. R. Hasan, S. R. Sabuj, M. A. Hossain, and S. K. Ray, “A Comprehensive Survey on Emerging AI Technologies for 6G Communications: Research Direction, Trends, Challenges, and Opportunities,” Int. J. Intell. Networks, 2025.

[2] A. Ullah, A. Nadeem, M. Arif, M. M. Bashir, and W. Choi, “6G Internet-of-Things assisted smart homes and buildings: Enabling technologies, opportunities and challenges,” Internet of Things, p. 101658, 2025.

[3] M. Alwakeel, “Synergistic Integration of Edge Computing and 6G Networks for Real-Time IoT Applications,” Mathematics, vol. 13, no. 9, p. 1540, 2025.

[4] Q. He et al., “Integrating IoT and 6 G: Applications of Edge Intelligence, Challenges, and Future Directions,” IEEE Trans. Serv. Comput., 2025.

[5] T. Al-Shurbaji et al., “Deep learning-based intrusion detection system for detecting IoT botnet attacks: a review,” IEEE Access, 2025.

[6] N. Mohamed, “Artificial intelligence and machine learning in cybersecurity: a deep dive into state-of-the-art techniques and future paradigms,” Knowl. Inf. Syst., pp. 1-87, 2025.

[7] S. Najafli, A. Toroghi Haghighat, and B. Karasfi, “Taxonomy of deep learning-based intrusion detection system approaches in fog computing: a systematic review,” Knowl. Inf. Syst., vol. 66, no. 11, pp. 6527-6560, 2024.

[8] M. Devi, P. Nandal, and H. Sehrawat, “Federated Learning-Enabled Lightweight Intrusion Detection System for Wireless Sensor Networks: A Cybersecurity Approach Against DDoS Attacks in Smart City Environments,” Intell. Syst. with Appl., p. 200553, 2025.

[9] R. Marin Machado de Souza, A. Holm, M. Biczyk, and L. N. de Castro, “A systematic literature review on the use of federated learning and bioinspired computing,” Electronics, vol. 13, no. 16, p. 3157, 2024.

[10] A. Ali, H. Jianjun, and A. Jabbar, “Recent Advances in Federated Learning for Connected Autonomous Vehicles: Addressing Privacy, Performance, and Scalability Challenges,” IEEE Access, 2025.

[11] M. Zhang et al., “WHHO: enhanced Harris hawks optimizer for feature selection in high-dimensional data,” Cluster Comput., vol. 28, no. 3, p. 186, 2025.

[12] T. E. T. Djaidja, B. Brik, S. M. Senouci, A. Boualouache, and Y. Ghamri-Doudane, "Early network intrusion detection enabled by attention mechanisms and RNNs," IEEE Trans. Inf. Forensics Secur., 2024.

[13] G. Singh, K. Sood, P. Rajalakshmi, D. D. N. Nguyen, and Y. Xiang, "Evaluating federated learning-based intrusion detection scheme for next generation networks," IEEE Trans. Netw. Serv. Manag., vol. 21, no. 4, pp. 4816-4829, 2024.

[14] A. Khraisat, M. A. Talukder, M. A. Uddin, and A. Alazab, "RF-FedAvg: Federated learning-based random forest model for intrusion detection in wireless sensor networks," Cluster Comput., vol. 28, no. 13, p. 873, 2025.

[15] S. Wang, W. Xu, and Y. Liu, "Res-TranBiLSTM: An intelligent approach for intrusion detection in the Internet of Things," Comput. Networks, vol. 235, p. 109982, 2023.

[16] A. Nazir et al., "Empirical evaluation of ensemble learning and hybrid CNN-LSTM for IoT threat detection on heterogeneous datasets," J. Supercomput., vol. 81, no. 6, p. 775, 2025.

[17] A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," Futur. Gener. Comput. Syst., vol. 97, pp. 849-872, 2019.

[18] P. Zhou, H. Zhang, and W. Liang, "Research on hybrid intrusion detection based on improved Harris Hawk optimization algorithm," Conn. Sci., vol. 35, no. 1, p. 2195595, 2023.

[19] A. Alzaqebah, I. Aljarah, O. Al-Kadi, and R. Damaševičius, "A modified grey wolf optimization algorithm for an intrusion detection system," Mathematics, vol. 10, no. 6, p. 999, 2022.

[20] K. Bonawitz et al., "Practical secure aggregation for privacy-preserving machine learning," in Proc. ACM SIGSAC Conf. Computer and Communications Security, 2017, pp. 1175-1191.