SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I2P118

Volume 13 Issue 2, 239-253, February 2026

Original Article

Multihop Communication Latency Prediction in Wireless Sensor Networks using Dimensionality Reduction and Recurrent Neural Network Architecture

K. Vinayakan[1], A. Srinivasa Reddy[2], M. Vasuki[3]

[1]*Department of Computer Science, Khadir Mohideen College*
(Affiliated to Bharathidasan University), Adirampattinam, Tamil Nadu, India.
[2]*Department of CSE (Data Science), CVR College of Engineering, Pocharam Road,*
Vastunagar, Mangalpalli (Village Ibrahimpatnam Mandal), Telangana, India.
[3]*Department of Mathematics, Srinivasan College of Arts and Science, Perambalur, Tamil Nadu, India.*

[1]*Corresponding Author : k.vinayakan@gmail.com*

Received: 23 November 2025 Revised: 24 December 2025 Accepted: 25 January 2026 Published: 20 February 2026

Abstract *- The prediction of Multihop Latency (MHL) in Wireless Sensor Networks (WSNs) faces diverse challenges and is substantially addressed by the unintermittent growth of technologies such as Machine Learning (ML), edge computing, security mechanisms, and hybrid modelling models. Effectively handling the difficulty is crucial for attaining the full potential of prediction methods in diverse settings of the Internet of Things (IoT). Factors, namely transmission delays, node congestion, hop count, and energy constraints, are considered in MHL prediction, and also, ML is widely utilized for forecasting and alleviating latency in dynamic network environments. Recently, Deep Learning (DL) has gained popularity in network routing and is also applied to model the Multihop (MH) communication latency prediction in WSNs. This study presents a Multihop Communication Latency Prediction Using Dimensionality Reduction and Recurrent Neural Network (MCLP-DRRNN) technique in WSNs. The aim is to develop an efficient model for accurately predicting the latency of MH communication in WSNs. Initially, the min-max scaling-based data pre-processing is employed. Furthermore, the walrus optimization algorithm (WOA) technique is used for the Feature Selection (FS) process. Moreover, the Minimal Gated Unit (MGU) technique is employed for classification. Finally, the Group Theory Optimization Algorithm (GTOA) technique is implemented for tuning. The comparison analysis of the MCLP-DRRNN model revealed a superior accuracy value of 99.33% compared to existing techniques under the WSN MH dataset.*

Keywords *- Multihop communication, Latency, Wireless Sensor Networks, Minimal Gated Unit, Group Theory Optimisation Algorithm.*

1. Introduction

Wireless communication technology offers significant benefits, including higher productivity and lower installation costs, in industrial control systems. Various wireless technologies are now being introduced to the plant and workshop floors, handling different types of tasks [1]. One such method, an MH wireless network, offers high adaptability but also presents challenges in terms of reliability. These networks are applied in new fields within industrial automation, and dedicated algorithms, such as randomized MH routing protocols, have currently proven effective for such uses [2]. In creating control systems that utilize wireless mesh networks, a key feature is the requirement to conceal the system's complexity through the proper simplification of the networks and their nodes. A straightforward yet crucial method of achieving this is to view the network as a communication channel with changing delays over time [3]. WSNs consist of numerous nodes (ranging from a few tens to thousands), possessing features such as self-organization, random deployment, and low cost.

Sensor Nodes (SNs) are scattered randomly in the observation region and sense signals in their surrounding areas using inbuilt sensors, then transmit the gathered information to the Base Station (BS) [4]. They are widely utilized in areas such as environmental monitoring, military applications, forest fire detection, industry, precision agriculture, and more, gaining the attention of scholars recently [5]. SNs use limited power, making it hard to replace them. To conserve and extend the lifespan, routing protocols must be more Energy-Efficient (EE) [6]. Figure 1 highlights the WSN MH communication.

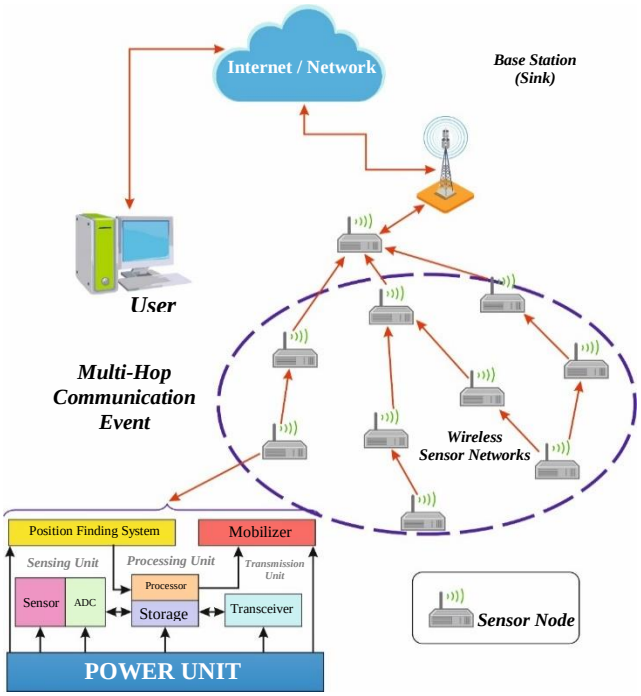


Fig. 1 MH communication in WSNs

It is significant to reduce energy consumption. Unlike single-hop WSNs that depend on direct transmission to a centralized station, intermediate nodes are used by MH for wide reach-out and reliability [7]. MHL prediction in WSNs also enables transmission paths to be formed, thus giving efficiency during transmission [8]. The analysis highlights that anticipating end-to-end latency and data packet size is substantial in enhancing achievement, and the optimization is crucial, where operational efficacy is significant [9]. This retains significance in applications where precise and reliable data transfer is crucial.

Also, extrapolation and interpolation are used to avert bottlenecks and ensure a smoother data flow. ML efficiently handles issues in WSN and also enables SN to make informed decisions locally, thereby reducing the need for massive data transfer. Moreover, energy is conserved by these local decision-making models, thus improving the overall networking function [10].

ML assists WSN in attaining enhanced intelligence and self-monitoring across various fields [11]. It becomes crucial to ensure reliable and timely data transfer for maintaining system performance. Though WSN is cost-efficient and flexible, it encounters unanticipated transfer delays, which can be reduced by precise latency prediction and improved system receptiveness. DL also facilitates dynamic and data-driven prediction approaches. Still, it becomes crucial to reduce complexity due to high-dimensional input data, and integrating dimensionality reduction with RNN presents a promising solution to model and anticipate time-varying transfer latencies effectively.

This study presents a Multihop Communication Latency Prediction Using Dimensionality Reduction and Recurrent Neural Network (MCLP-DRRNN) technique in WSNs. The aim is to present a method for accurately predicting the latency of MH communication in WSNs. Initially, the min-max scaling is used for data pre-processing. Also, the Walrus Optimization Algorithm (WOA) technique is utilized for the Feature Selection (FS) process. Additionally, the minimal gated unit (MGU) technique is employed for classification. Finally, the Group Theory Optimization Algorithm (GTOA) technique is used for tuning. The validation of the MCLP-DRRNN model is conducted using the WSN MH dataset. The key contributions are:

- Initially, min-max is used for scaling to standardize the input data, thus confirming compatibility across diverse feature ranges. The efficiency and accuracy are also improved by giving a normalized data structure to converge and learn with ease.
- The WOA is utilized for detecting and retaining the most appropriate features, and the accuracy is improved by focusing on critical inputs while also mitigating complexity. The overall effectiveness and robustness are also improved by optimizing the feature subset.
- The MGU is a simple RNN model employed for performing classification and reducing computational overhead related to conventional gated units while also mitigating robust temporal dependency. Thus, the efficiency is improved.
- The GTOA methodology is applied for tuning the learning parameters, thus allowing them to converge and stabilize. The overall predictive accuracy and reliability are also enhanced by optimizing these parameters. This contributes to a more efficient and effective learning process for latency prediction.

2. Prior Research on MH Communication Latency Prediction in WSN

Prince et al. [12] introduced an EE Mega-Cluster-based Routing Protocol (EEMCRP) by utilizing fixed clustering. Pushpa et al. [13] introduced a method by incorporating Deep Reinforcement Learning (DRL) and Graph Neural Networks (GNN). The GNN approach enhances spatial awareness by extracting relational dependencies across the SN. Elmonser et al. [14] integrated the Heterogeneous Dynamic MH (HDM) approach with the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol for WSNs. Alsaeedi et al. [15] presented a scalable SDWSN method. These strategies are developed to effectively utilize SDWSN's network resources and precisely forecast network states. In [16], an enhanced Region-based Energy-Efficient Routing Protocol (REERP) is presented for WSN. It relies on incremental cluster growth where new nodes join existing clusters. The remaining energy levels of the nodes determine the selection of a new Cluster Head (CH) within each group. Next, set up the MH communication in every network region, and lastly, employ

energy hole reduction techniques. All these strategies boost the network's life. Cherappa et al. [17] optimized CH Selection (CHS) in WSN using the Adaptive Sailfish Optimization Algorithm (ASFO) method, combined with K-medoids, to improve routing efficiency through an E Cross-layer-based Expedient Routing Protocol (E-CERP), thereby enhancing network lifetime and performance.

In [18], a Taylor-based Gravitational Search Algorithm (TBGSA) is proposed that integrates the Taylor series with a GSA to determine the optimal hops for MH routing. At first, SNs are classified as clusters or groups, and competent nodes can access the CH. Then, actions are switched between numerous nodes like an MH. Primarily, the optimal CH is selected through the Artificial Bee Colony (ABC) model, and data is transferred through MH routing. Priyadarshi et al. [19] developed a modular Artificial Intelligence (AI)-based routing framework for WSNs that integrates Reinforcement Learning (RL), Genetic Algorithms (GA), and Particle Swarm Optimization (PSO) to optimize routing decisions dynamically. He et al. [20] presented an EE clustering and Multihop routing protocol for underwater WSN (UWSNs) by utilizing a novel Hierarchical Chimp Optimization Algorithm (HChOA) model. Natarajan and Manickavelu [21] improved routing in Internet of Underwater WSNs (IUWSNs) by utilizing a Fractional Wave Elk Herd Optimizer (FWEHO) method. The approach integrates Deep Long Short-Term Memory (DLSTM) for energy prediction, tuned with WEHO models for optimizing CHS. Mahalakshmi et al. [22] developed a protocol by integrating Adaptive Shark Smell and Salp Swam Optimizer, namely ASSO and SSO, for CHS and optimal route selection. A Modified Elman Recurrent Neural Network (MERNN) is also used for intrusion detection. Alanazi et al. [23] proposed an advanced system for WSNs in 6G environments by incorporating supervised learning and RL methods. Jalalinejad et al. [24] introduced a protocol for Energy Harvesting WSNs (EH-WSNs) that incorporates both centralized and decentralized clustering based on energy availability and harvested energy. Juwaied and Jackowska-Strumillo [25] presented the DL–Hybrid EE Distributed (DL-HEED) methodology by incorporating GNN into the clustering process of WSNs. Ghamry and Shukry [26] proposed a multi-objective intelligent clustering routing scheme for WSNs in the IoT using DRL.

Although existing studies are effective in the latency prediction process, they face challenges in terms of real-time adaptability, scalability, and EE under dynamic conditions. Techniques such as EEMCRP, DRL-GNN, HDM-LEACH, and SDWSN-DRL exhibit limitations in flexibility, large-scale deployment, spatial awareness in high-dimensional state spaces, and overhead issues due to frequent network updates. Region-based and ABC-TBGSA methods optimize CHS but are limited in their ability to handle heterogeneity and make real-time energy predictions. The research gap is in incorporating energy prediction, dynamic clustering, and adaptive MH routing under constrained resources with minimal control overhead. Existing AI-based and metaheuristic methods often treat routing and clustering separately, increasing complexity and reducing efficiency in EH-aware, large-scale WSNs. Comparative study on existing MH communication latency prediction in WSN in Table 1.

Table 1. A comparison summary of diverse models

References	Year	Objective	Approaches	Performance Measures
Prince et al. [12]	2025	To address hotspot issues for wide-area coverage.	-	Network life of 34.5%, 23.5%, 14.5% and 5.5%.
Pushpa et al. [13]	2025	To enable adaptive real-time sensor node adjustments.	GNN and DRL	Coverage ratio of 96.4%, EE of 95.8%.
Elmonser et al. [14]	2024	To develop a model for smart cities.	HDM, LEACH	-
Alsaeedi et al. [15]	2024	To manage controller-bound traffic using SDN and optimize flow rules.	DRL	Accuracy of 85%.
Dogra et al. [16]	2023	The primary objective is to decrease energy deployment for data to be transferred from the sender to the BS.	-	-
Cherappa et al. [17]	2023	To optimize CHS for balanced energy use, reduced distance, and lower latency in WSNs.	ASFO, E-CERP	Achieves PDR 100% and Packet delay of 0.05s.
Balan et al. [18]	2023	The primary intention is to employ a hybrid optimization model for generating a WSN MH routing protocol with lesser energy consumption.	MH and ABC	The network lifetime is 13.2%, 21.9%, and 29.2%.
Priyadarshi et al. [19]	2025	To develop a hybrid AI-based model for WSNs that adapts dynamically to network changes.	RL, GA, PSO	Packet Delivery Ratio, Latency, EE

He et al. [20]	2024	To present a technique that increases EE in UWSNs and prolongs the networking life.	HChOA	Lifespan and Energy Usage
Natarajan and Manickavelu [21]	2025	To propose a routing approach for energy prediction and CHS.	FWEHO, DLSTM	Energy, Throughput, Delay, Distance
Mahalakshmi, Ramalingam, and Manikandan [22]	2024	To develop a secure and EE routing protocol for fault data prediction.	ASSO, SSO, MERNN	Dispersion Value of 0.8072, Packet Delivery Rate of 98%, Average Delay of 160 Ms, Network Lifetime of 3200 Rounds, and Accuracy of 99.2%
Alanazi et al. [23]	2025	To present an advanced system using ML and 6G infrastructure.	Supervised Learning and RL	Network Lifetime 1320 Rounds, Energy 0.52 J, PDR 93.4%, Delay 108 ms
Jalalinejad et al. [24]	2024	To develop a model to improve EE and network stability under diverse scenarios.	EH-WSNs	95% Alive Nodes, Balanced Energy, 23-55% Overhead Reduction, High Delivery, Low Loss, 1320 Rounds Lifetime
Juwaied and Jackowska-Strumillo [25]	2025	For choosing CH using GNN in WSNs.	DL-HEED, GNN	60% Improved Lifetime, Enhanced EE, Scalable and Adaptive Clustering
Ghamry and Shukry [26]	2024	To optimize IoT-based WSN routing with unequal clustering for energy balance and longevity.	DRL	High EE, Increased Delivered Packets, Lower Delay, More Alive Nodes, Extended Network Lifespan

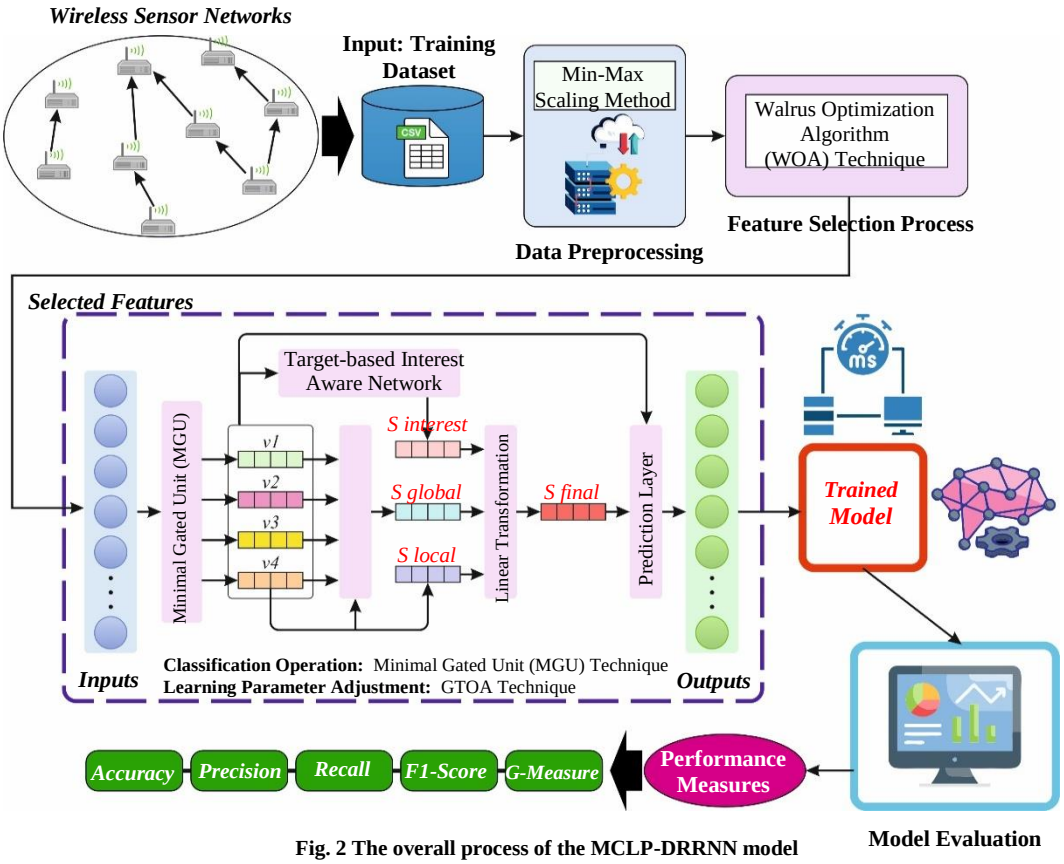


Fig. 2 The overall process of the MCLP-DRRNN model

3. Methodological Framework

This article develops an MCLP-DRRNN model in WSNs. The study aims to create an effective technique for precisely predicting the latency of MH communication in WSN. Figure 2 illustrates the overall process involved in the MCLP-DRRNN model.

3.1. Min-Max Scaling-based Data Pre-Processing Method

Initially, the min-max scaling is executed to normalize the input data into a compatible format [27]. This model is employed for its simplicity and efficiency in transforming features to a uniform range, typically in the range of [0, 1]. This technique also preserves the original distribution shape of the data without centring it around the mean, unlike other normalization models, namely score standardization, which is beneficial for models sensitive to data range, like neural networks. Moreover, this scaling method prevents features with larger magnitudes from dominating the learning process, thereby enhancing convergence speed and stability. Furthermore, this model is considered appropriate for real-time or resource-constrained environments like WSNs. The model is also chosen for its compatibility with subsequent optimization and learning algorithms.

The scaling model is normalized based on the training dataset to avert data leakage, as shown in Equation 21. The same parameters are later applied to ensure it remains unbiased and unseen during training.

$$S' = \frac{S - S_{\min}}{S_{\max} - S_{\min}} \tag{1}$$

Whereas, S, $S_{\max}$ and $S_{\min}$ portray new, maximum, and minimum feature values, and S' depicts the standard value.

3.2. Dimensionality Reduction using the WOA Approach

The WOA model is utilized for FS to recollect and detect the most appropriate features [28]. This effectively balances global and local search processes, also reproducing the walrus's social behaviour. This technique is also considered for its robust capabilities in exploration and exploitation. This model averts premature convergence and better escapes local optima and also enhances performance while mitigating dimensionality, thus accurately detecting the most relevant features. Furthermore, WOA is computationally efficient and easy to implement, making it appropriate for handling high-dimensional data in WSNs. Its adaptability and robustness make it an ideal choice over conventional methods, such as gas or PSO, for feature selection tasks.

The WOA is an original metaheuristic model stimulated by walruses' behaviours. These behaviours are derived from the primary signal obtained by the walruses, that is, a danger signal or a security signal. A metaheuristic model is derived from the behaviours of walrus in roosting, foraging, migrating, and breeding. At first, the optimizer model initiates with random solutions:

$$X = LB + rand(UB - LB) \tag{2}$$

Whereas UB and LB represent the upper and lower bounds, and $rand$ denotes a random function in the range $[0,1]$.

The location of Walruses is upgraded in every iteration:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,d} \end{bmatrix} \tag{3}$$

Whereas n denotes the population size and d depicts the size of the decision variables. The fitness values are well-defined as:

$$F = \begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,d} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,d} \\ \cdots & \cdots & \cdots & \cdots \\ f_{n,1} & f_{n,2} & \cdots & f_{n,d} \end{bmatrix} \tag{4}$$

90% of walrus populations are adults, and the remaining 10% are youngsters.

3.2.1. Danger Signal and Safety Signal

In a dangerous situation, a danger signal should be sent between walruses, which is described with the following equation:

$$danger\_signal = A \times R \tag{5}$$

$$A = 2 \times \alpha \tag{6}$$

$$R = 2 \times r_1 - 1 \tag{7}$$

Whereas, A and R represent factors of danger. α begins with one and in every iteration reduces to 0 with the iteration count t. T means maximal iteration counts. r_1 is an arbitrary number in $[0,1]$.

The safety signal is described as demonstrated:

$$safety\_signal = r_2 \tag{8}$$

r_2 denotes a random number between $[0,1]$.

3.2.2. Migration

The migration signifies the exploration stage in the model. Walruses will migrate once the risk factors are significantly high.

$$X_{i,j}^{t+1} = X_{i,j}^{t} + migration_{step} \tag{9}$$

$$migration_{step} = (X_m^t - X_n^t) \cdot \beta \cdot r_3^2 \tag{10}$$

$$\beta = 1 - \frac{1}{1 + \exp\left(-\frac{t - \frac{T}{2}}{T} \times 10\right)} \tag{11}$$

Here, $X_{i,j}^{t+1}$ denotes the new location for the ith walrus on the jth size. $Migration\_step$ is the walrus movement's step size. β denotes the migration step control factor. r_1, r_2, r_3 are arbitrary numbers in $(0,1)$.

3.2.3. Reproduction

Roosting behaviour: The reproduction signifies the exploitation stage. Once the risk factors are lower, the walrus groups tend to breed. The female is affected by the male and the dominant walrus.

$$Female_{i,j}^{t+1} = Female_{i,j}^{t} + \alpha \cdot \left(Male_{i,j}^{t} - Female_{i,j}^{t}\right) + (1-\alpha) \cdot \left(X_{best}^{t} - Female_{i,j}^{t}\right) \quad (12)$$

$Female_{i,j}^{t+1}$ refers to the new location of the ith female on the jth size.

$$Juvenile_{i,j}^{t+1} = \left(O - Juvenile_{i,j}^{t}\right) \cdot P \quad (13)$$

$$O = X_{best}^{t} + Juvenile_{i,j}^{t}.LF \quad (14)$$

Whereas $Juvenile_{i,j}^{t+1}$ denote the new location of the ith juvenile on the jth size. P denotes a distress random coefficient in $[0,1]$. LF is a random number vector made utilizing the Lévy distribution:

$$Lévy\ (a) = 0.05 \times \frac{x}{|y|^{\frac{1}{a}}} \quad (15)$$

x and y denote variables distributed normally, $xN(0,\sigma_x^2)$, $yN(0,\sigma_y^2)$ and σ means standard deviations.

3.2.4. Foraging Behaviour

In this context, they distinguish between two behaviours: gathering and foraging behaviours.

Feeling Behaviours

In underwater hunting, walruses are attacked by predators. In such a case, walruses will flee from their place depending on danger signals from their peers.

$$X_{i,j}^{t+1} = X_{i,j}^{t} \times R - \left|X_{best}^{t} - X_{i,j}^{t}\right| * r_4^2 \quad (16)$$

Whereas, $|X_{best}^{t} - X_{i,j}^{t}|$ refers to the distance between the present walrus and the top walrus; r_4 indicates an arbitrary number in $(0,1)$.

Gathering Behaviour

Walruses can share their place and move with other walruses to discover the ocean area with high-quality food:

$$X_{i,j}^{t+1} = \frac{X_1 + X_2}{2} \quad (17)$$

$$\begin{cases} X_1 = X_{best}^{t} - a_1 \times b_1 \times |X_{best}^{t} - X_{i,j}^{t}| \\ X_1 = X_{\text{second}}^{t} - a_2 \times b_2 \times \left|X_{\text{second}}^{t} - X_{i,j}^{t}\right| \end{cases} \quad (18)$$

$$a = \beta \times r_5 - \beta \quad (19)$$

$$b = \tan(\theta) \quad (20)$$

Whereas X_1 and X_2 weights affect the collecting behaviour. a and β represent gathering coefficients. r_5 denotes a random number between $[0,1]$. θ is a number within the range $[0,\pi]$.

The exploitation and exploration stages are established using the danger signal. If the danger signal is exceeding 1, the warluses travel to a new place, which implies the exploration stage; otherwise, they group replicate, which denotes the exploitation stage. A binary version of Walrus Optimiser is proposed in this work. During binary walrus optimization, the sigmoid function is used to transform constant values to binary values. The binary conversion typically includes thresholding the sigmoid function output:

$$B = \frac{1}{1+\exp(-X)} \quad (21)$$

The Fitness Function (FF) balances classifier accuracy with the number of chosen features, aiming to maximize precision while minimizing the feature set size, as shown in Equation (22).

$$Fitness = \alpha * ErrorRate + (1-\alpha) * \frac{\#SF}{\#All\_F} \quad (22)$$

Here, $ErrorRate$ denotes the classifier error rate and is measured as the percentage of incorrect classifications out of the total classifications made, specified in (0,1), $\#SF$ denotes the number of designated attributes, and $\#All\_F$ indicates total feature counts in the new dataset. α is used for controlling the subset length and classification quality.

3.3. MGU-based Classification Process

Additionally, the MGU is employed for classification [27]. This mitigates the complexity of conventional Gated Recurrent Units (GRU), namely LSTM, and also utilizes fewer gates. The methodology also gives faster training and lower computational costs, thereby reducing the parameters. The MGU is also regarded in constrained environments, and the temporal dependencies in sequential data are effectively captured by this model, which is also crucial for accurate latency prediction in Multihop communication. Its balance between efficiency and overall system provides a clear merit over more intrinsic recurrent units that may be computationally expensive and significantly slower to converge.

MGU is a typical gated RNN that lowers the computational efficiency while conserving the ability for modelling the temporal dependencies. This model is also a lightweight alternative to LSTMs and GRUs, including

smaller parameters and gates, resulting in further computational cost without losing performance.

Compared to GRU, MGU uses only one gate to process either the reset operation or memory updates. The gate of forget is the single gate that controls the amount of candidate activation required to upgrade the present state and how often the preceding Hidden Layer (HL) must be preserved. The gate of forget is measured by utilizing Equation (22):

$$f_l = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \quad (23)$$

Whereas f_t and x_t depict the input and forget gate at time t, respectively. h_{t-1} denotes the preceding HL. W_f and U_f Signify learnable weighted matrices. b_f means bias, and σ indicates the activation function of the sigmoid. Equation (23) utilizes the activation function of tanh to compute $\tilde{h}_t$, which includes new data that is added to the present memory.

$$\tilde{h}_t = \tanh(W_h x_t + U_h(f_t \odot h_{t-1}) + b_h) \quad (23)$$

Now, $\odot$ displays element-to-element multiplication, and W_h, U_h, b_h represent added learnable parameters. Utilizing the forget gate, the last HL h_t is later used by incorporating among the candidate HL and the prior HL, as shown in Equation (24).

$$h_t = (1 - f_t) \odot h_{t-1} + f_t \odot \tilde{h}_t \quad (24)$$

The new and past data are integrated by MGU to capture temporal patterns effectively, thus reducing complexity. It is also appropriate for resource-limited or large-scale time series tasks, mitigating parameters and gates to enable faster training, quicker inference, and lower memory consumption.

3.4. Parameter Optimizer using GTOA Technique

Finally, the GTOA is implemented for learning the parameter tuning technique [29]. This is highly efficient in exploring intrinsic search spaces and is also used to learn parameters by employing mathematical principles from group theory. The method also illustrates convergence properties and robustness against getting trapped in local minima, and results in more precise fine-tuning of learning parameters, related to conventional optimization techniques. Furthermore, the structured search strategy of the GTOA method mitigates computational overhead, making it precise for real-time WSN settings. This also presents significant merit over conventional techniques, thus resulting in enhanced learning efficiency and performance.

GTOA relies on a specific algebraic system and encourages exploration and co-operation with the population by describing the communication and data-sharing mechanisms among individuals. GTOA gives merits such as fewer control parameters, faster convergence, and higher precision of solution. Assume that $G_n = \{0,1, \dots, n-1\}$, n denotes a positive integer and $n \geq 2$. Describe the operation of G_n In Equation (25), subsequently, G_n IT is a group of operations $\oplus$.

$$a \oplus b = (a + b) mod n \; a, b \in G_n \quad (25)$$

If $G_1, G_2, \dots, G_k$ are k groups, their group product is specified as: $G[n_1, n_2, \dots, n_k] = G_1 \times G_2 \times \cdots \times G_k$. It is obtained by $G[n_1, n_2, \dots, n_k]$, which also forms a group. The components of this direct product group are arranged $in\ k$-tuples like $(a_1, a_2, \dots, a_k)$, depicted as a vector $[a_1, a_2, \dots, a_k]$. $InG[m_1 + 1, m_2 + 1, \dots, m_n + 1]$, 3 different arbitrary individuals are chosen, namely $X = [x_1, x_2, \dots, x_n]$, $Y = [y_1, y_2, \dots, y_n]$, and $Z = [z_1, z_2, z_n]$. Over the Random Linear Combination Operator (RLCO), algebraic group operations are executed on X, Y, and Z to produce an innovative individual $R = [r_1, r_2, \dots, r_n] \in G[m_1 + 1, m_2 + 1, \dots, m_n + 1]$, accomplishing group evolution. This method allows new individuals to learn from others. Thus, GTOA possesses robust global exploration proficiency, enhancing the likelihood of discovering high-quality solutions in the solution space.

$$R = X \oplus \left(F\left(Y \oplus (-Z)\right)\right) \quad (26)$$

Now, $r_j = x_j \oplus \left[f_j\left(y_j \oplus \left(m_j + 1 - z_j\right)\right]\right.$, $j = 1,2, \dots, n$. $F = [f_1, f_2, f_n] \in \{-1,0,1\}^n$ indicates a vector of arbitrary integration.

Evolutionary models not only necessitate robust proficiency in global exploration but also require a strong proficiency for effective operation of evolutionary models. To accomplish this, GTOA utilizes a local search operator that relies on the inverse operation in space G, specifically the Inversion and Random Mutation Operator (IRMO).

$$IRMO(R) = \begin{cases} rand\left([0, m_j] - \{r_j\}\right) & h_1 < P_m, h_2 > 0.5 \\ m_j + 1 - r_j & h_1 < P_m, h_2 \leq 0.5, x_j \neq 0 \\ r_j & otherwise \end{cases} \quad j = 1 \dots n \quad (27)$$

P_m indicates the mutation probability, satisfying $0 < P_m \leq 0.5$. h_1 and h_2 are dual arbitrary numbers in $(0,1)$ · $rand\left([0, m_j] - \{r_j\}\right)$ depicts an arbitrary integer with the range $[0, m_j]$ that is not equal to r_j.

Nevertheless, IRMO also has its limitations, as it is not appropriate for 0-1 vectors in the searching area $G[2,2, \dots,2]$. To tackle this, a local searching operator appropriate for space $G[2,2, \dots,2]$ is proposed, drawing inspiration from the mutation operator in GA.

$$SMO(R) = \begin{cases} 1 - r_j & h < P_m \\ r_j & otherwise \end{cases} \quad j = 1 \dots n \quad (28)$$

P_m represents mutation probability, satisfying $0 < P_m \leq 1. h \sim U[0,1]$ signifies an arbitrary number.

Assume that n is the objective function dimension, P_m signifies the mutation probability, N denotes the size of the population, T is the 100 maximum iterations, and $P(t) = \{X_i(t)|1 \leq i \leq N\}$ depicts the population of GTOA in iteration t, where $0 \leq t \leq T$. $X_i(t) = [x_{i1}(t), x_{i2}(t), \ldots, x_{in}(t)]$ is i-th individual. The pseudo-code of GTOA is specified:

Algorithm 1: Algorithm of GTOA

Input: The objective function $\min f(X)$, Parameters N and T, Mutation probability P_m.
Output: An approximate solution (or optimal solution) H and $f(H)$.
Generate initial population $P(0) = \{X_i(0)|i = 1 \cdots N\}$ randomly;
Calculate $f(X_i(0)), i = 1 \cdots N, t \leftarrow 0$;
while $t < T$ do
  for $i \leftarrow 1 to N$ do
    $Y \leftarrow X_{p_1}(t) \oplus \left(F\left(X_{p_2}(t) \oplus \left(-X_{ps}(t)\right)\right)\right)$;
    $Y \leftarrow IRMO(Y, P_m)$ or $Y \leftarrow SMO(Y, P_m)$;
    if $f(Y) > f\left(X_i(t)\right)$ then
      $X_i(t+1) \leftarrow Y$;
    else
      $X_i(t+1) \leftarrow X_i(t)$;
      end if
  end for
  $t \leftarrow t + 1$;
end while
Determine the best individual H in $P(T)$;
return $\left(H, f(H)\right)$

The fitness selection comprises performance in the GTOA. The hyperparameter selection includes the solution encoding to evaluate the efficiency of the candidate solutions. The GTOA deliberates precision as the leading standard to design the FF, as indicated in Equations. (29-30).

$$Fitness = \max(P) \tag{29}$$

$$P = \frac{TP}{TP + FP} \tag{30}$$

Now, TP and FP depict the true and false positive values.

4. Experimental Results and Discussion

In this section, the analysis of the MCLP-DRRNN model's results is examined using the WSN MH dataset [30]. This dataset is intended for research and analysis of MH communication in WSN. It aims to optimize end-to-end latency, considering factors such as routing algorithms, congestion control, EE, and network reliability. The dataset comprises 1,000 samples categorized into three latency groups, as presented in Table 2. The no. of features is 12. They are Node_ID, Hop_Count, Transmission_Delay, Buffer_Occupancy, Channel_Utilization, Energy_Level, Link_Quality, Congestion_Status, Packet_Size, PDR, Traffic_Class, and Routing_Algorithm. But 10 features are only selected such as Transmission_Delay, Buffer_Occupancy, Channel_Utilization, Energy_Level, Link_Quality, Congestion_Status, Packet_Size, PDR, Traffic_Class, and Routing_Algorithm.

Table 2. Details of the dataset

Latency Category	Instances
Medium Latency (20-40ms)	390
High Latency (> 40ms)	359
Low Latency (≤ 20ms)	251
Total	**1000**

Table 3 and Figure 3 illustrate the classifier outcome of the MCLP-DRRNN approach at 80:20. Under 80% TRPHE, the MCLP-DRRNN model attains an average $accu_y$ of 98.75%, $prec_n$ of 98.29%, $reca_l$ of 97.97%, $F1_{Score}$ of 98.12%, and $G_{Measure}$ of 98.13%. Likewise, at 20% TSPHE, the MCLP-DRRNN model attains an average $accu_y$ of 99.33%, $prec_n$ of 98.93%, $reca_l$ of 99.13%, $F1_{Score}$ of 99.02%, and $G_{Measure}$ of 99.03%.

Table 3. Classifier outcome of MCLP-DRRNN model under 80:20

Classes	$Accu_y$	$Prec_n$	$Reca_l$	$F1_{Score}$	$G_{Measure}$
TRPHE (80%)					
Medium Latency	98.75	98.10	98.72	98.41	98.41
High Latency	98.38	97.30	98.29	97.79	97.79
Low Latency	99.12	99.47	96.91	98.17	98.18
Average	**98.75**	**98.29**	**97.97**	**98.12**	**98.13**
TSPHE (20%)					
Medium Latency	99.00	100.00	97.40	98.68	98.69
High Latency	99.50	98.51	100.00	99.25	99.25
Low Latency	99.50	98.28	100.00	99.13	99.13
Average	**99.33**	**98.93**	**99.13**	**99.02**	**99.03**

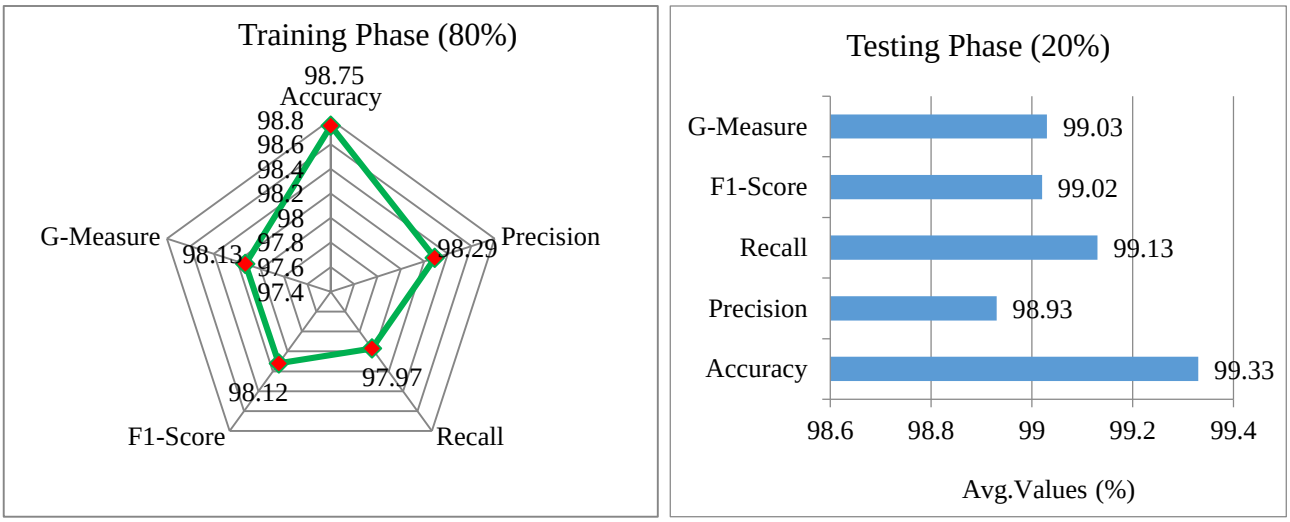


Fig. 3 Average values of the MCLP-DRRNN model (a) Training phase 80% and (b) Testing phase 20%.

Figure 4 exemplifies the Training (TRAIN) and Validation (VALID) $accu_y$ of an MCLP-DRRNN method on an 80:20 split over 30 epochs. Their close alignment throughout training suggests robust regularization and effective feature retention on both seen and unseen data.

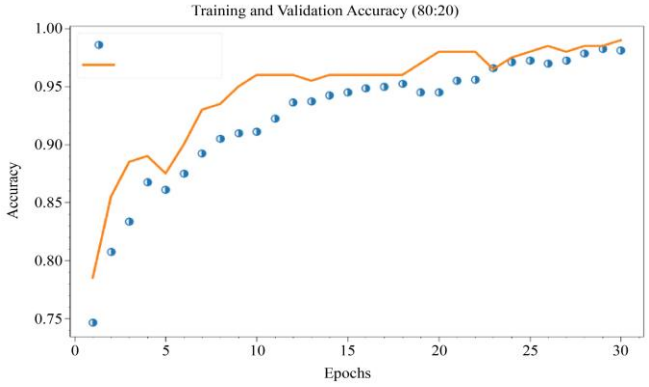


Fig. 4 $Accu_y$ curve of the MCLP-DRRNN model under 80:20

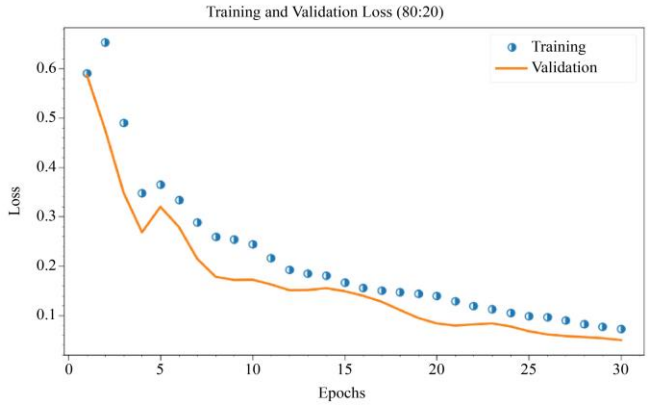


Fig. 5 Loss curve of MCLP-DRRNN model under 80:20

Figure 5 illustrates the TRAIN and VALID losses of the MCLP-DRRNN method at an 80:20 split over 30 epochs. Initially, both TRAIN and VALID losses are high, reflecting a restricted initial understanding. As training progresses, losses steadily decrease, illustrating effective learning. The close alignment suggests robust generalization without overfitting.

Figure 6 exhibits the classifier results of the MCLP-DRRNN model at 80:20. Figure 6(a) illustrates the PR investigation of the MCLP-DRRNN model. The outcomes denoted that the model results in rising PR values. Moreover, the MCLP-DRRNN model can reach maximum PR values on each class. Finally, Figure 6(b) elucidates the ROC inspection of the MCLP-DRRNN methodology. The figure illustrates that the model resulted in increased ROC values. Furthermore, the MCLP-DRRNN methodology can achieve maximum ROC values for every class.

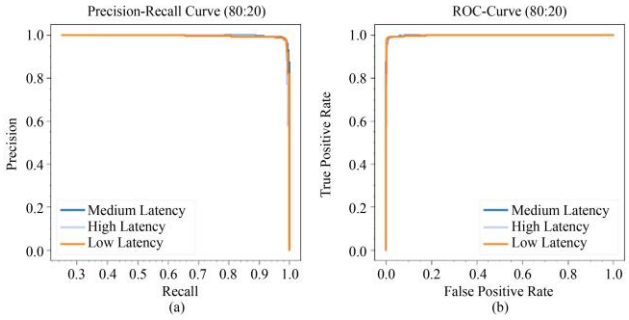


Fig. 6 (a) PR curve, and (b) ROC curve on 80:20.

Table 4 and Figure 7 portray the classifier output of the MCLP-DRRNN methodology at 70:30. Under 70% TRPHE, the MCLP-DRRNN model attains an average $accu_y$ of 98.57%, $prec_n$ of 98.04%, $reca_l$ of 97.82%, $F1_{Score}$ of 97.93%, and $G_{Measure}$ of 97.93%. Also, at 30% TSPHE, the MCLP-DRRNN model attains an $accu_y$ of 98.44%, $prec_n$ of 97.63%, $reca_l$ of 97.83%, $F1_{Score}$ of 97.71%, and $G_{Measure}$ of 97.72%.

Table 4. Classifier output of the MCLP-DRRNN technique under 70:30

Classes	$Accu_y$	$Prec_n$	$Reca_l$	$F1_{Score}$	$G_{Measure}$
TRPHE (70%)					
Medium Latency	98.43	97.46	98.53	98.00	98.00
High Latency	98.00	97.23	97.23	97.23	97.23
Low Latency	99.29	99.42	97.70	98.55	98.55
Average	**98.57**	**98.04**	**97.82**	**97.93**	**97.93**
TSPHE (30%)					
Medium Latency	98.00	99.12	95.73	97.39	97.41
High Latency	98.33	96.33	99.06	97.67	97.68
Low Latency	99.00	97.44	98.70	98.06	98.07
Average	**98.44**	**97.63**	**97.83**	**97.71**	**97.72**

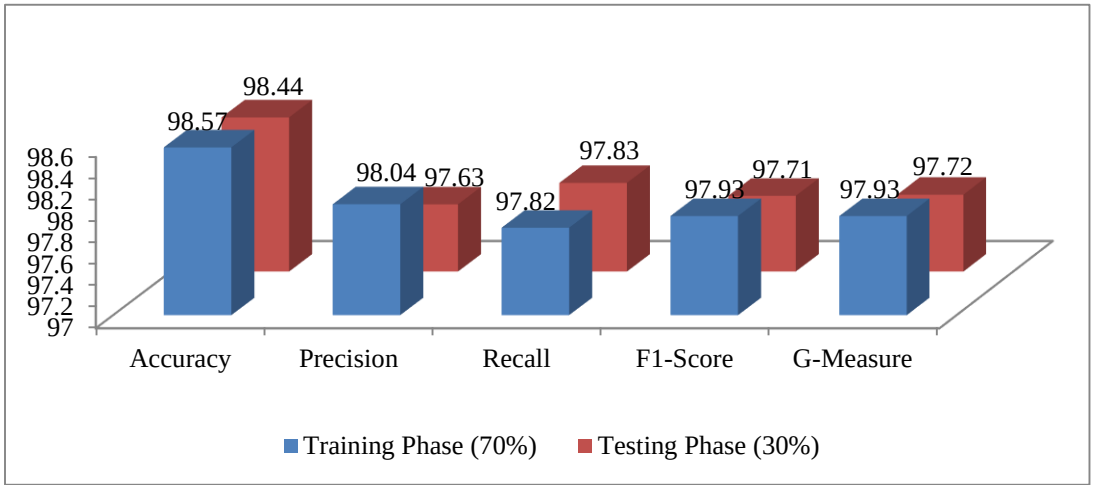


Fig. 7 Average values of the MCLP-DRRNN technique under 70:30

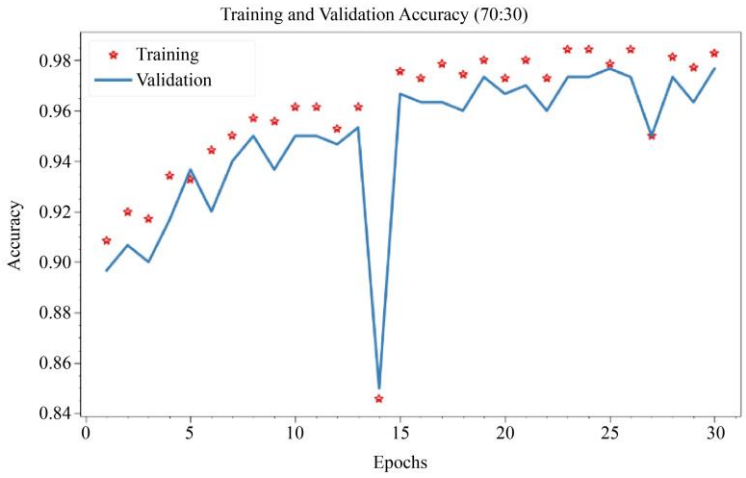


Fig. 8 $Accu_y$ curve of MCLP-DRRNN approach under 70:30

Figure 8 demonstrates the TRAIN and VALID $accu_y$ of an MCLP-DRRNN model on 70:30 over 30 epochs. Both TRAIN and VALID $accu_y$ rise quickly at the start, illustrating efficient learning. VALID slightly surpasses TRAIN, suggesting robust generalization. Their close alignment throughout training reflects good regularization and consistent performance on both seen and unseen data.

Figure 9 exemplifies the TRAIN and VALID losses of the MCLP-DRRNN model over 30 epochs with a 70:30 split. Initially, both TRAIN and VALID losses are high, illustrating limited understanding. As training continues, losses steadily decrease, showing effective learning. The close match between TRAIN and VALID losses suggests that there is no overfitting and robust generalization to new data.

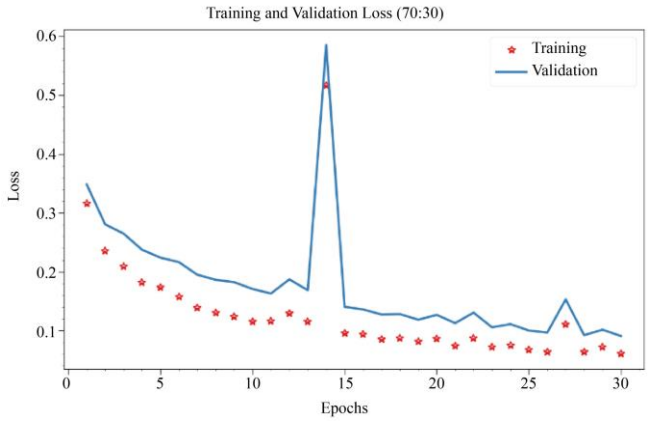


Fig. 9 Loss curve of MCLP-DRRNN approach under 70:30

Figure 10 portrays the classifier results of the MCLP-DRRNN approach at 70:30. Figure 10(a) depicts the PR inspection of the MCLP-DRRNN model. Also, the MCLP-DRRNN model achieves the highest PR values for each class. Finally, Figure 10(b) showcases the ROC examination of the MCLP-DRRNN model. The figure shows that the technique resulted in superior PR and ROC values. Moreover, the MCLP-DRRNN model achieves a higher ROC across all classes.

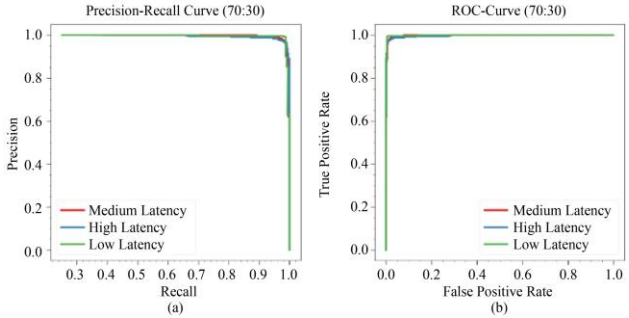
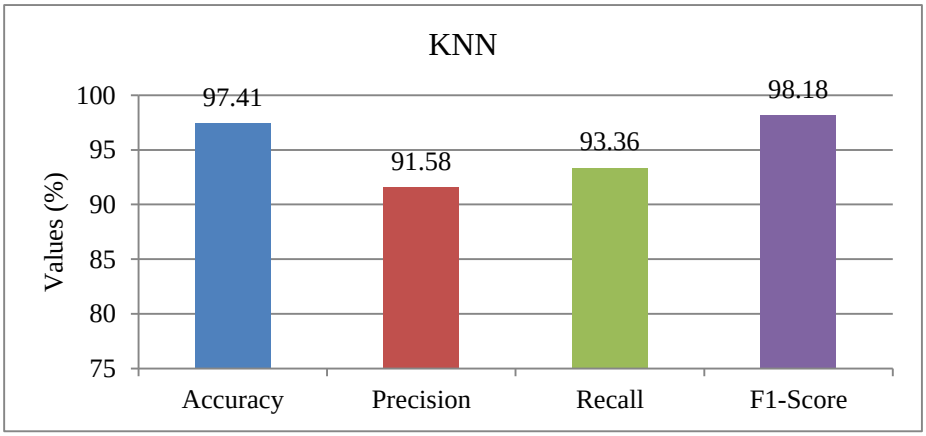


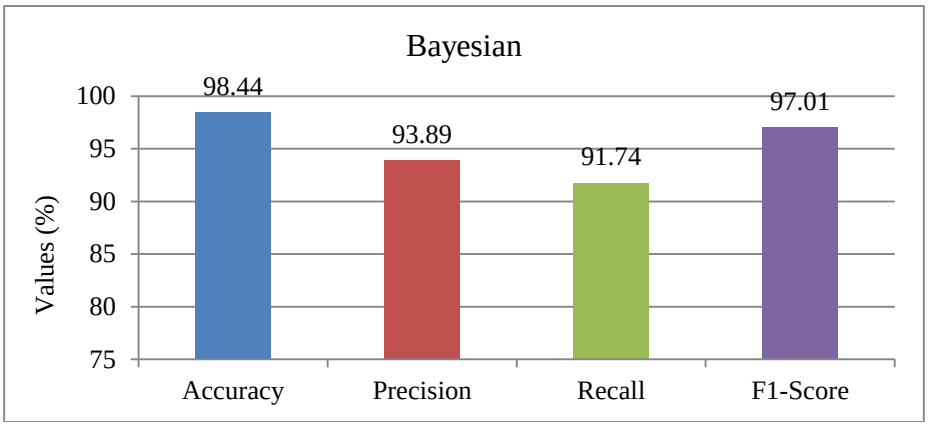
Fig. 10 (a) PR curve, and (b) ROC curve on 70:30.

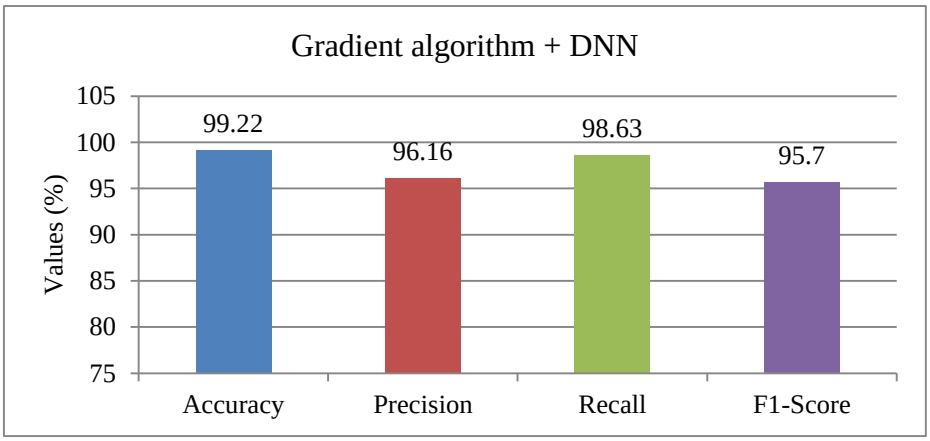
Table 5 and Figure 11 illustrate the advanced performance of the MCLP-DRRNN methodology compared to existing models [31-33]. The MCLP-DRRNN model presents powerful performance with $accu_y$ of 99.33%, $prec_n$ of 98.93%, $reca_l$ of 99.13%, and $F1_{Score}$ of 99.02%. Whereas, the Gradient algorithm + DNN, Random Forest, and Bayesian methodologies achieved slightly better performance with $accu_y$ of 99.22%, 99.09%, and 98.44%, respectively. Whereas, the other existing models, such as KNN, Binary Classifier, PCA, and Genetic Classifier, have obtained worse values with $accu_y$ of 97.41%, 93.66%, 95.75%, and 92.72%, respectively.

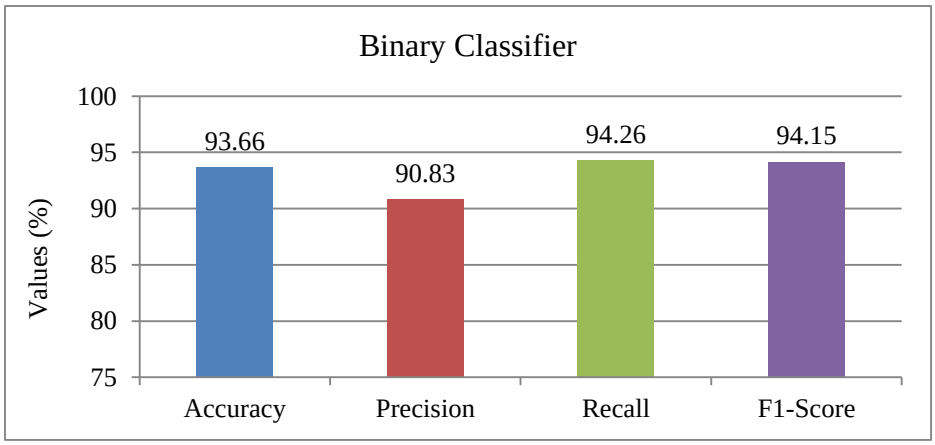
Table 5. Comparative study of the MCLP-DRRNN model with existing approaches

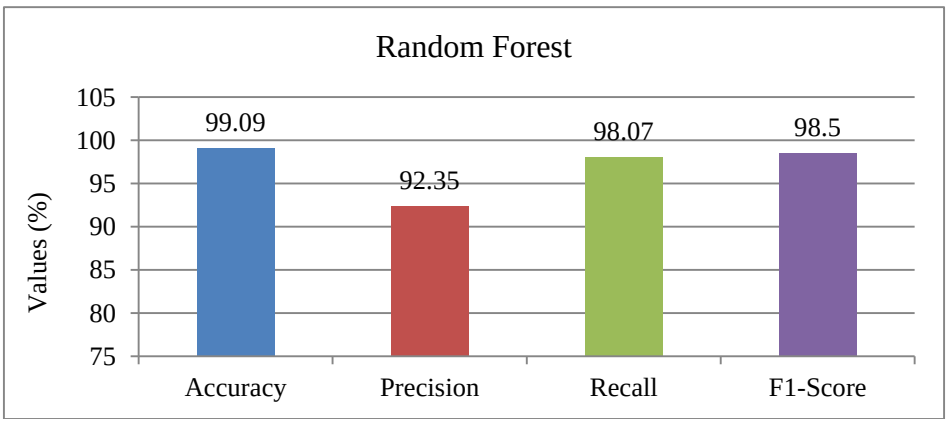
Techniques	$Accu_y$	$Prec_n$	$Reca_l$	$F1_{Score}$
KNN	97.41	91.58	93.36	98.18
Bayesian	98.44	93.89	91.74	97.01
Gradient algorithm + DNN	99.22	96.16	98.63	95.70
Binary Classifier	93.66	90.83	94.26	94.15
Random Forest	99.09	92.35	98.07	98.50
PCA Model	95.75	96.52	95.65	97.04
Genetic Classifier	92.72	90.56	91.97	94.95
MCLP-DRRNN	99.33	98.93	99.13	99.02




Bayesian
Values (%)
100
95
90
85
80
75
98.44
93.89
91.74
97.01
Accuracy
Precision
Recall
F1-Score


Gradient algorithm + DNN
Values (%)
105
100
95
90
85
80
75
99.22
96.16
98.63
95.7
Accuracy
Precision
Recall
F1-Score


Binary Classifier
Values (%)
100
95
90
85
80
75
93.66
90.83
94.26
94.15
Accuracy
Precision
Recall
F1-Score


Random Forest
Values (%)
105
100
95
90
85
80
75
99.09
92.35
98.07
98.5
Accuracy
Precision
Recall
F1-Score

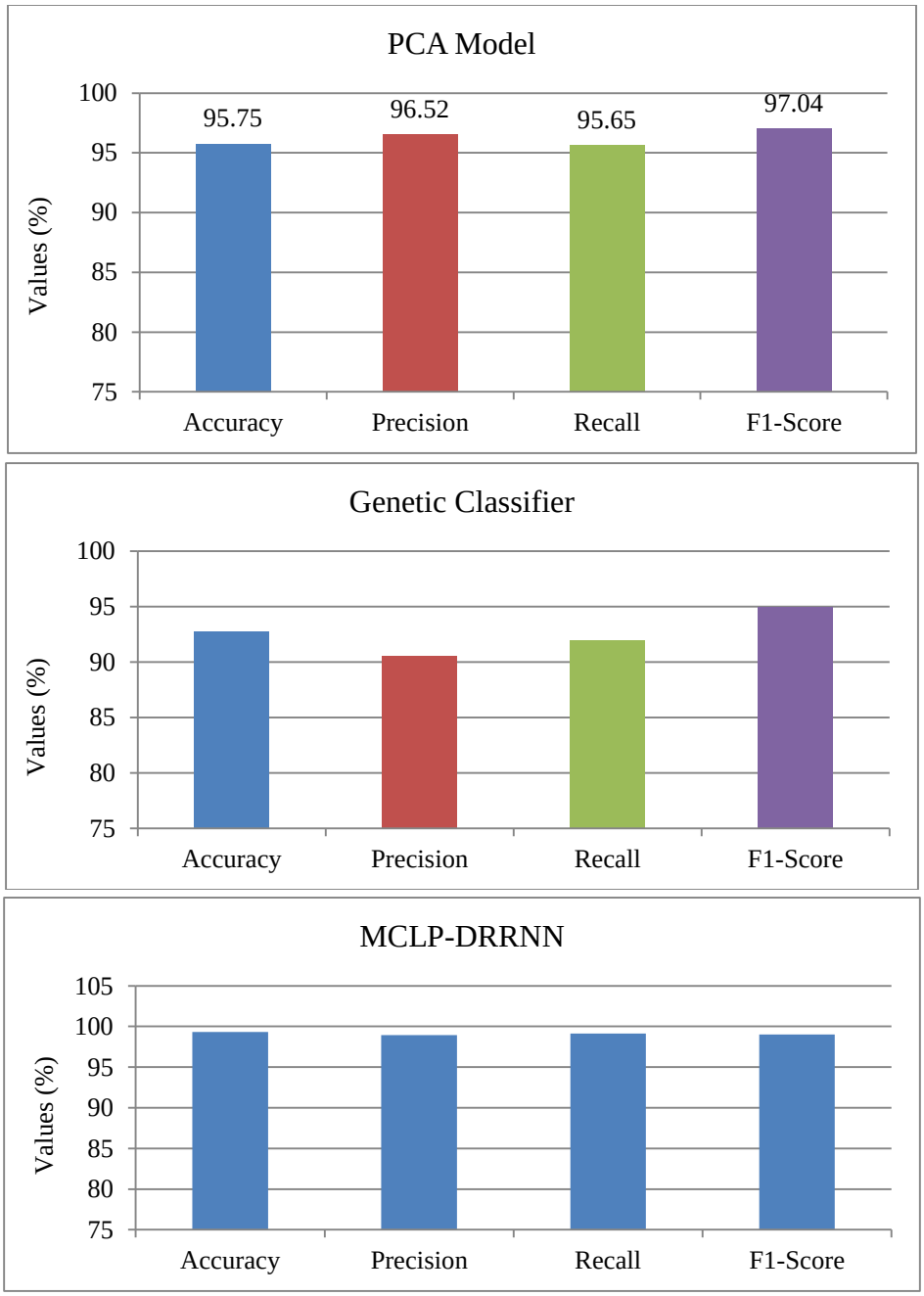


Fig. 11 Comparative study of MCLP-DRRNN model with existing approaches

In Table 6 and Figure 12, the comparative outcome of the MCLP-DRRNN approach is stated in terms of computation time units (CTU). The outcomes imply that the MCLP-DRRNN methodology achieves top performance. According to CTU, the MCLP-DRRNN methodology achieves a lower value of 7.00ms.

At the same time, the KNN, Bayesian, Gradient algorithm + DNN, Binary Classifier, Random Forest, PCA, and Genetic Classifier methodologies yielded the following values: 17.95ms, 25.02ms, 11.90ms, 10.05ms, 19.71ms, 28.41ms, and 17.97ms, respectively.

Table 6. CTU outcome of MCLP-DRRNN method with existing techniques

Techniques	CTU (ms)
KNN	17.95
Bayesian	25.02
Gradient algorithm + DNN	11.90
Binary Classifier	10.05
Random Forest	19.71
PCA Model	28.41
Genetic Classifier	17.97
MCLP-DRRNN	7.00

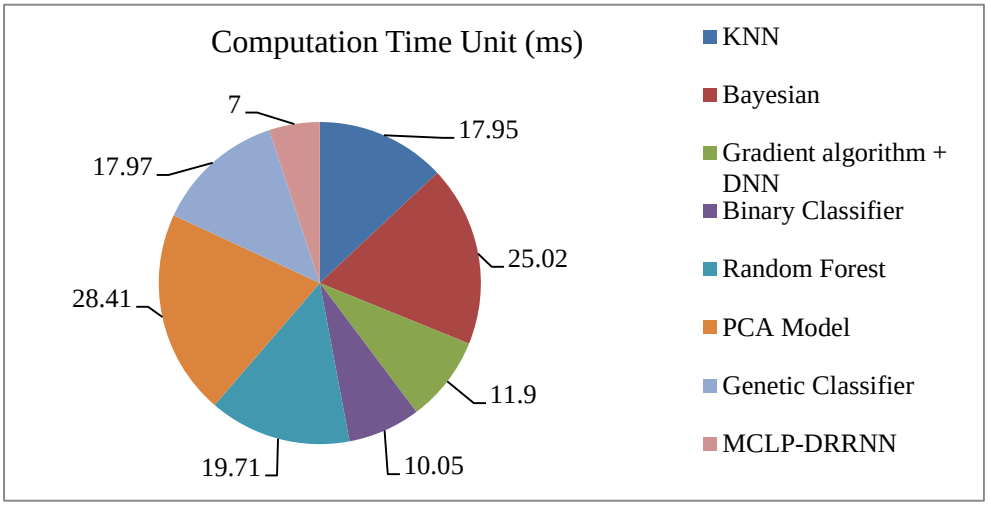


Fig. 12 CTU outcome of MCLP-DRRNN method with existing techniques

From these results and the comparative study, it is evident that the MCLP-DRRNN methodology attained improved latency prediction for MH communication of WSN.

5. Conclusion

In this article, a novel MCLP-DRRNN model is proposed for predicting the latency of MH communication in WSNs. The aim is to develop an effective model for precisely predicting the latency of MH communication in WSN. Initially, the min-max scaling-based data pre-processing is applied. For the FS procedure, the model utilizes the WOA approach to retain and identify the most relevant features. Additionally, the classification process is performed using the MGU technique. Finally, the GTOA is used by the MCLP-DRRNN model for the learning parameter tuning technique. The experimentation of the MCLP-DRRNN model is achieved using the WSN MH dataset, which yields a superior accuracy value of 99.33% compared to existing methods.

References

[1] Saleh M. Altowaijri, "Efficient Next-Hop Selection in Multi-Hop Routing for IoT Enabled Wireless Sensor Networks," *Future Internet*, vol. 14, no. 2, pp. 1-35, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Toan-Van Nguyen et al., "Short-Packet Communications in Multi-Hop WPINs: Performance Analysis and Deep Learning Design," *2021 IEEE Global Communications Conference (GLOBECOM)*, Madrid, Spain, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Toan-Van Nguyen et al., "Short-Packet Communications in Multihop Networks with WET: Performance Analysis and Deep Learning-Aided Optimization," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 439-456, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Aiqi Zhang et al., "Deep Reinforcement Learning-Based Multi-Hop State-Aware Routing Strategy for Wireless Sensor Networks," *Applied Sciences*, vol. 11, no. 10, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Juan Pablo Astudillo León, Luis J. de la Cruz Llopis, and Francisco J. Rico-Novella, "A Machine Learning based Distributed Congestion Control Protocol for Multi-hop Wireless Networks," *Computer Networks*, vol. 231, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Josiah E. Balota, Ah-Lian Kor, and Olatunji A. Shobande, "Multi-Network Latency Prediction for IoT and WSNs," *Computers*, vol. 13, no. 1, pp. 1-32, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[7] Xintao Huan et al., "Improving Multi-Hop Time Synchronization Performance in Wireless Sensor Networks Based on Packet-Relaying Gateways with Per-Hop Delay Compensation," *IEEE Transactions on Communications*, vol. 69, no. 9, pp. 6093-6105, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Salem Omar Sati, and Afif Abugharsa, "Wireless Sensor Network Hop Count Theory Modeling and Deep Learning Prediction," *2024 IEEE 4th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)*, Tripoli, Libya, pp. 124-129, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Ramineni Padmasree, and Aravalli Sainath Chaithanya, "Fault Detection in Single-hop and Multihop Wireless Sensor Networks using a Deep Learning Algorithm," *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 13, no. 3, pp. 453-461, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] S. Regilan, and L.K. Hema, "Machine Learning Based Low Redundancy Prediction Model for IoT-Enabled Wireless Sensor Network," *SN Computer Science*, vol. 4, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[11] Zipeng Li et al., "Latency and Reliability of mmWave Multi-Hop V2V Communications under Relay Selections," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 9807-9821, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[12] Bhaskar Prince, Prabhat Kumar, and Sunil Kumar Singh, "Multi-Level Clustering and Prediction based Energy Efficient Routing Protocol to Eliminate Hotspot Problem in Wireless Sensor Networks," *Scientific Reports*, vol. 15, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[13] G. Pushpa et al., "Optimizing Coverage in Wireless Sensor Networks using Deep Reinforcement Learning with Graph Neural Networks, *Scientific Reports*, vol. 15, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[14] Malika Elmonser et al., "Enhancing Energy Distribution through Dynamic Multi-hop for Heterogeneous WSNs Dedicated to IoT-Enabled Smart Grids," *Scientific Reports*, vol. 14, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[15] Mohammed Alsaeedi, Mohd Murtadha Mohamad, and Anas Al-Roubaiey, "SSDWSN: A Scalable Software-Defined Wireless Sensor Networks," *IEEE Access*, vol. 12, pp. 21787-21806, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[16] Roopali Dogra, Shalli Rani, and Gabriele Gianini, "REERP: A Region-based Energy-Efficient Routing Protocol for IoT Wireless Sensor Networks," *Energies*, vol. 16, no. 17, pp. 1-16, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[17] C. Venkatesan et al., "Energy-Efficient Clustering and Routing Using ASFO and a Cross-Layer-Based Expedient Routing Protocol for Wireless Sensor Networks," *Sensors*, vol. 23, no. 5, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[18] B. Sivasankari et al., "Energy Efficient Multihop Routing Scheme using Taylor based Gravitational Search Algorithm in Wireless Sensor Networks," *International Journal of Electrical and Computer Engineering Systems*, vol. 14, no. 3, pp. 333-343, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[19] Rahul Priyadarshi et al., "AI-based Routing Algorithms Improve Energy Efficiency, Latency, and Data Reliability in Wireless Sensor Networks," *Scientific Reports*, vol. 15, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[20] Shukun He et al., "The Optimization of Nodes Clustering and Multi-hop Routing Protocol using Hierarchical Chimp Optimization for Sustainable Energy Efficient Underwater Wireless Sensor Networks," *Wireless Networks*, vol. 30, pp. 233-252, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[21] Suriya Natarajan, and Palanivelan Manickavelu, "A Novel Deep Learning with Optimization for Energy Prediction and Fractional Wave Elk Herd Optimization for Routing in Internet of Underwater Wireless Sensor Networks," *Cybernetics and Systems*, pp. 1-27, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[22] G. Mahalakshmi, S. Ramalingam, and A. Manikandan, "An Energy Efficient Data Fault Prediction based Clustering and Routing Protocol using Hybrid ASSO with MERNN in Wireless Sensor Network," *Telecommunication Systems*, vol. 86, pp. 61-82, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[23] Rakan Alanazi et al., "Machine Learning-driven Routing Optimization for Energy-efficient 6G-enabled Wireless Sensor Networks," *Alexandria Engineering Journal*, vol. 129, pp. 877-888, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[24] Hoda Jalalinejad et al., "A Hybrid Multi-Hop Clustering and Energy-Aware Routing Protocol for Efficient Resource Management in Renewable Energy Harvesting Wireless Sensor Networks," *IEEE Access*, vol. 12, pp. 137310-137332, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[25] Abdulla Juwaied, and Lidia Jackowska-Strumillo, "DL-HEED: A Deep Learning Approach to Energy-Efficient Clustering in Heterogeneous Wireless Sensor Networks," *Applied Sciences*, vol. 15, no. 16, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[26] Walid K. Ghamry, and Suzan Shukry, "Multi-objective Intelligent Clustering Routing Schema for Internet of things Enabled Wireless Sensor Networks using Deep Reinforcement Learning," *Cluster Computing*, vol. 27, pp. 4941-4961, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[27] Sameer Al-Dahidi et al., "Multistep PV Power Forecasting using Deep Learning Models and the Reptile Search Algorithm," *Results in Engineering*, vol. 27, pp. 1-17, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[28] Seyyid Ahmed Medjahed, and Fatima Boukhatem, "A Binary Walrus Optimizer for Feature Selection Problem," *Computing and Systems*, vol. 29, no. 2, pp. 955-965, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[29] Hongjie Zheng et al., "Research on a Microseismic Signal Picking Algorithm based on GTOA Clustering," *EGUsphere*, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[30] WSN Multi-Hop Dataset, Kaggle, [Online]. Available. https://www.kaggle.com/datasets/ziya07/wsn-multi-hop-dataset
[31] Rami Ahmad, Raniyah Wazirali, and Tarik Abu-Ain, "Machine Learning for Wireless Sensor Networks Security: An Overview of Challenges and Issues," *Sensors*, vol. 22, no. 13, pp. 1-35, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[32] Azita Pourghasem et al., "Machine Learning and Deep Learning-Based Multi-Attribute Physical-Layer Authentication for Spoofing Detection in LoRaWAN," *Future Internet*, vol. 17, no. 2, pp. 1-14, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[33] Abhay R. Gaidhani, and Amol D. Potgantwar, "A Review of Machine Learning-based Routing Protocols for Wireless Sensor Network Lifetime," *Engineering Proceedings*, vol. 59, no. 1, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]