Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.037449
Article

Billiards Optimization with Modified Deep Learning for Fault Detection in Wireless Sensor Network

Yousif Sufyan Jghef[1], Mohammed Jasim Mohammed Jasim[2], Subhi R. M. Zeebaree[3,*] and Rizgar R. Zebari[4]

[1]Department of Computer Engineering, College of Engineering, Knowledge University, Erbil, 44001, Iraq
[2]Engineering College, Al-Kitab University, Kirkuk, Iraq
[3]Energy Eng. Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Iraq
[4]Computer Science Department, College of Science, Nawroz University, Duhok, Iraq
*Corresponding Author: Subhi R. M. Zeebaree. Email: subhi.rafeeq@dpu.edu.krd
Received: 03 November 2022; Accepted: 28 December 2022; Published: 28 July 2023

Abstract: Wireless Sensor Networks (WSNs) gather data in physical environments, which is some type. These ubiquitous sensors face several challenges responsible for corrupting them (mostly sensor failure and intrusions in external agents). WSNs were disposed to error, and effectual fault detection techniques are utilized for detecting faults from WSNs in a timely approach. Machine learning (ML) was extremely utilized for detecting faults in WSNs. Therefore, this study proposes a billiards optimization algorithm with modified deep learning for fault detection (BIOMDL-FD) in WSN. The BIOMDL-FD technique mainly concentrates on identifying sensor faults to enhance network efficiency. To do so, the presented BIOMDL-FD technique uses the attention-based bidirectional long short-term memory (ABLSTM) method for fault detection. In the ABLSTM model, the attention mechanism enables us to learn the relationships between the inputs and modify the probability to give more attention to essential features. At the same time, the BIO algorithm is employed for optimal hyperparameter tuning of the ABLSTM model, which is stimulated by billiard games, showing the novelty of the work. Experimental analyses are made to affirm the enhanced fault detection outcomes of the BIOMDL-FD technique. Detailed simulation results demonstrate the improvement of the BIOMDL-FD technique over other models with a maximum classification accuracy of 99.37%.

Keywords: Wireless sensor network; fault detection; reliability; deep learning; metaheuristics

1 Introduction

The Internet of Things (IoT) is becoming a unique structure for network traffic formed individually by several small devices and elements [1]. Recently, IoT has grabbed substantial interest because of its implications, including traffic management, smart healthcare, mobility, new technologies, and

smart homes. IoT techniques can drastically improve intellectual services' effectiveness, resilience, and stability [2]. The main objective of IoT was to gather data, ensuring the presence of a very critical and susceptible perceptual layer in the mechanism at all times. Thus, the perception layer will be the most vulnerable and crucial, as many sources are required if the node energy is depleted [3].

Moreover, the consistency of the data accumulated through the perception layer will be helpful in the IoT implementation and maturity. Wireless sensor networks (WSNs) were widely utilized in IoT-related mechanisms for gaining the data required by smart environments. Commonly, WSNs are made up of sensor nodes (SN) compiled with wireless communication tools. SNs were widely distributed and self-contained [4]. The WSN structure usually has source SNs, cluster head (CH) nodes, sink nodes, and manager nodes. Such SNs under operation could reach real-time observation of the physical environments, offer comprehensive information for back-end services for the evaluation, and formulate the intellectual structure of the sensing layer [5]. But the common features of such nodes generally need more storage, computing capabilities, and energy.

Some reasons for faults in WSNs are calibration faults, data loss, and aggregation errors. Machine learning (ML) can be broadly employed for fault detection in WSN [6]. Various faults exist that the research community can encounter and are classified under the sensed data. Various classifier methods also include numerous advantages; hence, deciding the optimal classifier for the fusion technique is complex. Thus, the belief function-related decision fusion method will be utilized since it is compatible with any classifier type [7]. ML techniques were commonly utilized for enhancing the fusion accuracy of the belief function fusion method. This work mainly focused on using methods which would minimize the difficulty of the energy consumption and combination operation of the SNs [8]. ML will leverage a large volume of data for training a method and employs the model for prediction purposes.

There are many forms of ML techniques, and each has its features. The naive Bayes (NB) technique could reach simple probability-related classification. Still, it could not be able to study the communication among features and is inclined to low variance and high deviation [9]. Logistic Regression (LR) is implemented efficiently in binary classifications that can be extensively utilized in industry. But it can be disposed to under-fitting and usually contains less accuracy. A decision tree (DT) will be considered a decision support tool that can be easily explained and understood but is susceptible to overfitting. Since an effective technique in ML, artificial neural networks (ANN) will realize artificial intelligence (AI) through the simulation of the neural network (NN) of the human brain [10]. The objectives of WSN are summarized as 4 points: extending network lifespan, enhancing data quality, enhancing network security, and shortening response time.

This study proposes a billiards optimization algorithm with modified deep learning for fault detection (BIOMDL-FD) in WSN. The presented BIOMDL-FD technique employs an attention-based bidirectional long short-term memory (ABLSTM) model for fault detection. In the ABLSTM model, the attention mechanism enables us to learn the relationships between the inputs and modify the probability to give more attention to essential features. At the same time, the BIO algorithm is employed for optimal hyperparameter tuning of the ABLSTM model, which is stimulated by billiard games. Experimental analyses are made to affirm the enhanced fault detection outcomes of the BIOMDL-FD technique.

2 Literature Review

Regin et al. [11] modelled the convex hull technique for computing some extreme points, including the neighbouring nodes. The message's period will remain limited since the SNs raise. Then, the author devised a CNN and NB classifier to enhance the convergence performance and find the

nodes' faults. Next, the author examines CNN, convex hull, and NB techniques with the help of real-time data for finding and organizing the faults. In [12], a supervised ML-related approach will be considered for examining the SN's performance by their data for diagnosing and detecting faults. To promptly diagnose and promptly identify the faults, the author enforced an ensembled learning-related lightweight method named Extra-Trees or Extremely Randomized Trees. This modelled Extra-Trees-based detection technique is robust toward a strong reduction of bias, variance error, and signal noise.

Kalaiselvi et al. [13] devise the link failure detection and malicious nodes detection by malicious nodes in the wireless body area network (WBAN) atmosphere. The malicious SN detection mechanism was devised through the ML technique, and the link failure detection was devised with the help of the DL technique. This presented co-active adaptive neuro-fuzzy inference mechanism classifier related malicious node detection mechanism in WBAN. Jan et al. [14] present distributed sensor-fault detection (FD). The diagnosis mechanism relies upon ML techniques in which the FD block will be enforced in the SN to attain output instantly after data collection. This block has an AE for transforming the input signal into low-dimensional feature vectors that can be presented to SVM for classifying them as faulty or normal.

In [15], an energy-aware intellectual FD technique was modelled for IoT-based WSNs that pointedly enhanced FD accurateness and minimized false alarm rates. A new 3-Tier hard FD system was employed to detect hardware unit faults of the SNs. Additionally, an optimized DL system was utilized for several soft FDs, which averts premature death of SNs. Gavel et al. [16] proposed a new integration of data aggregation-related data fusion with active FD using the properties of KELM and Grey Model (GM). In this study, GM was used as a data fusion technique that records one datum pattern by rejecting redundant data get through various SNs. Trained KELM was used for effective FD, preserving high network confidentiality.

In [17], a modelled multiobjective-deep reinforcement-learning (DRL)-a related technique for fault tolerance in IoT-based WSN. The ultimate aim of this study was to identify the fault nodes having minimal overhead and maximal accuracy. Additionally, this study concentrates on reliable data communication after FD. At last, a mobile sink (MS) was leveraged for energy-efficient data collection that suggestively enhances the network's lifespan. In [18], a new Energy-Efficient Heterogeneous Fault Management technique was devised for managing such heterogeneous errors in IWSN. Effectual heterogeneous FD in this technique is attained through 3 new diagnosis approaches. The novel Tuned SVM classifier eases classifying the heterogeneous faults in which the tuning parameters of the modelled classifier are maximized by utilizing the Hierarchy related Grasshopper Optimization technique.

3 The Proposed Model

In this article, we developed a new BIOMDL-FD technique for fault detection in WSN, thereby extending the network efficiency. The presented BIOMDL-FD technique follows a two-stage process. At first, the fault detection process takes place using the ABLSTM model. Next, the BIO algorithm is applied for hyperparameter tuning in the latter stage. Fig. 1 represents the block diagram of the BIOMDL-FD system.

3.1 Fault Detection Module

In this study, the BIOMDL-FD technique utilizes the ABLSTM model for fault detection in WSN. The BLSTM is utilized to learn the time series feature from input sensor data (ISD). The traffic bytes of all the data ISD were consecutively input as BLSTM that eventually attained an input sensor data

vector (ISDV). The BLSTM is an improved type of LSTM [19]. The BLSTM technique was employed for extracting coarse-grained features concerning forward and backwards LSTMs. LSTM was planned by output gate 0, input gate i, and forget gate f to control overwritten data by relating the inner memory cell C once novel data attains. If the data enters an LSTM network, it is judged whether it can be valued based on significant rules. Only the data which meets techniques authentication remains, and inconsistent data is gone by the forgetting gate. To provide an input order $x = (x0, \ldots, x_t)$ at time t and hidden layers (HLs) of the BLSTM layer, $h = (h_0, \ldots, h_t)$ is the resultant as follows.

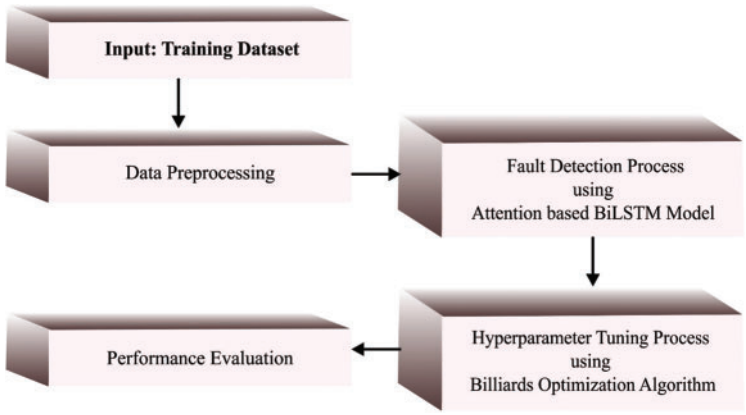

Figure 1: Block diagram of BIOMDL-FD system

The forget gate carries the outcome of HL h_{t-1} at the preceding moment and input x_t at the present moment as input for choosing to forget from the cell state C_t that is represented as [19]:

$$f_t = sigmoid(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \tag{1}$$

An input gate co-operates with the tanh function composed for controlling the addition of novel data. tanh creates a novel candidate vector. An input gate makes a value to all the items in $\widetilde{C}_t$ From zero to one for controlling that several novel data are added that are written as follows [19]:

$$C_t = sigmoid(f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t), \tag{2}$$

$$i_t = sigmoid(W_{xi}x_t + W_{hi}h_{t-1} + b_t), \tag{3}$$

$$\widetilde{C}_t = \tan(W_c x_t + W_c h_{t-1} + b_c), \tag{4}$$

The resultant gate employed for controlling that several of the present unit states are filtered out that is demonstrated as:

$$o_t = sigmoid(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \tag{5}$$

To the BLSTM technique at time t, the HLs of h_t that is an ISDV created in all the ISD is demonstrated as a concatenation of $\overleftarrow{h}_t$ and $\overrightarrow{h}_t$ that is written as:

$$h_t = \overleftarrow{h}_t + \overrightarrow{h}_t, \tag{6}$$

$$\overrightarrow{h}_t = tanh\left(W_{x\overrightarrow{h}}x_t + W_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}\right), \tag{7}$$

$$\overleftarrow{h_t} = \tan\left(W_{x\overleftarrow{h}}x_t + W_{\overleftarrow{h}\overleftarrow{h}}h_{t-1} + b_{\overleftarrow{h}}\right), \tag{8}$$

In which x denotes the input of heterogeneous time series data. '.' stands for the pointwise product. $\vec{H}$ *and* $\leftarrow h_t$ indicates the HLs of forward and backward LSTM layers at time t. Each matrix W is the linked weight betwixt 2 units, and b demonstrates the bias vectors. BLSTM creates an ISDV for all the ISDs. These ISDVs were set in the sequences of connection betwixt the 2 parties from the network stream for developing an order of ISDVs. The attention layer learns the connections in ISDVs. An attention system was utilized for adjusting the probability of ISDVs, so our technique pays further attention to significant features. Primarily, the ISDVs h_t extracting with the BLSTM technique was employed for obtaining their implicit representation u_t with a non-linear transformation that is defined as:

$$u_t = tanb(W_w h_t + b_w), \tag{9}$$

It depends upon the similarity representation u_t with context vector u_w and achieves the normalization significance weighted co-efficient α_t. u_w represents the arbitrary initialized matrix which concentrates on important data on u_t. The weighted coefficient to the above coarse-grained feature is represented as [19]:

$$\alpha_t = \frac{\exp\left(u_t^T u_w\right)}{\sum \exp\left(u_t^T u_w\right)}, \tag{10}$$

Lastly, the fine-grained feature s is estimated utilizing the weighted sum of h_t dependent upon α_t. s is demonstrated as:

$$s = \sum \alpha_t h_t, \tag{11}$$

The fine-grained feature vector s created in the attention method were utilized for malicious traffic detection with softmax classification that is written as:

$$y = softmax(W_h s + b_h), \tag{12}$$

Whereas W_h defines the weighted classification matrix that is map s to a novel vector with length h. h stands for class labels. Fig. 2 demonstrates the infrastructure of the BLSTM technique.

3.2 Hyperparameter Tuning Module

The BIO algorithm is employed for optimal hyperparameter tuning of the ABLSTM model. An important inspiration of BOA is the billiard game and embedding natural physical law from the collision betwixt balls [20]. As a common class of games, billiards are used by a long stick as a cue for striking billiards balls and invoking them for moving from one place to another. A cloth-covered billiards table surrounded by elastic cushions involved confining rails. The billiard sports were so susceptible to natural physical laws. The physics following them mostly contains collisions betwixt balls. The collision betwixt 2 billiard balls was almost elastic. In entirely elastic collisions, the kinetic drives of balls were conserved before and after collisions, also the sum of both moments. If two balls were with the other, the forces betwixt balls in the collision were directed beside an imaginary line which links its centre, simply put, impact line. It is to be noted that the impact velocities decompose into 2 elements, parallel and perpendicular elements. A primary one was parallel to the impact line of balls, and the second element was perpendicular to it. The last ball velocity after collision from the

perpendicular and parallel ways with the influence line of balls was defined as [20]:

$$\vec{v_1'} = v'_{1.\|}\vec{e}_{\|} + \vec{v}_{1.\perp}, \tag{13}$$

$$\vec{v_2'} = v_{2.\|}\vec{e}_{\|} + \vec{v}_{2.\perp}, \tag{14}$$

$$v'_{1.\|} = \frac{m_1 - m_2}{m_1 + m_2} v_{1.\|} + \frac{2m_2}{m_1 + m_2} v_{2.\|}, \tag{15}$$

$$v'_{2.\|} = \frac{2m_1}{m_1 + m_2} v_{1.\|} + \frac{m_2 - m_1}{m_1 + m_2} v_{2.\|}, \tag{16}$$

Whereas v_1' and v_2' stand for their velocities after the collision, v_1 and v_2 define the velocities of the 1st and 2nd balls before the collision. Besides, the symbols $\| \; and \; \perp$ defines the parallel and perpendicular elements correspondingly. The parameters m_1 and m_2 indicate the masses of balls. In addition, the unit vector of the linking vector was represented as $\vec{e}_{\|}$. It can be notable that perpendicular elements of velocities remain unmodified as force is only executed beside the influence line that causes the perpendicular elements of momenta that are preserved to balls. A detailed analysis of the above formulas exposes that once the balls contain equivalent masses, they switch only parallel velocity elements.

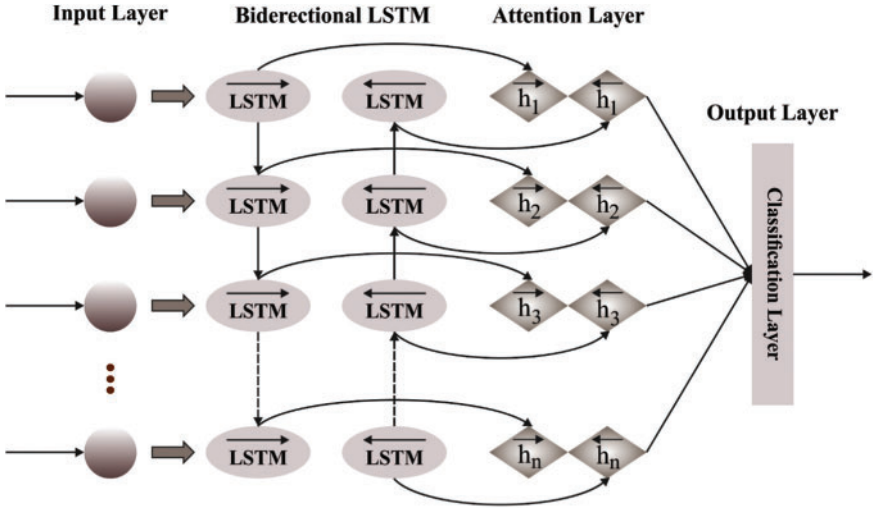

Figure 2: Architecture of BLSTM

$$v'_{1.\|} = v_{2.\|} \; and \; v'_{2.\|} = v_{1.\|}, \tag{17}$$

During this technique, the last places of bodies were identified dependent upon kinematics formulas under this case of constant accelerations as:

$$x(t) = x_0 + v_0 t + \frac{1}{2} a t^2, \tag{18}$$

$$|v|^2 = |v_0|^2 + 2|a|\,|x - x_0|, \tag{19}$$

Algorithm 1: Pseudocode of BIO Algorithm

```
set N = No. of balls;
M = No. of variables;
K = No. of pockets;
ET = escape threshold
itern = 0
Initialization of 2N balls and K pockets;
while (itern < itern_limit)
        Asses the location of balls and pockets based on objective function;
        Upgrade population and pocket memory;
        Generate normal as well as cue ball groups;
        for every pair of ball
                Elect a target pocket using a roulette-wheel selection scheme;
        end
        Upgrade the location of the present normal ball;
        Determine the velocity of normal ball afterwards collisions;
        Determine the velocity of the cue ball afterwards collisions;
        Upgrade the location of the present cue ball
      if (rand < ET)
                Recreate arbitrary ball dimensions;
        end
                Verify the boundary conditions and adjust the balls
      iter = iter + 1;
end while
Report optimal pocket as end solution.
```

4 Results and Discussion

The fault detection performance of the BIOMDL-FD model is inspected under four various types of faults, namely offset fault (OF), gain fault (GF), stuck-at fault (SAF), and out-of-bounds fault (OOBF). The proposed model has experimented on PC i5-8600k, GeForce 1050Ti 4 GB, 16 GB RAM, 250 GB SSD, and 1TB HDD. The parameter settings are learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU.

Table 1 reports the results offered by the BIOMDL-FD model under four fault types. The experimental values indicated that the BIOMDL-FD model had reached enhanced performance under all faults. In addition, it is noted that the BIOMDL-FD model has reached an average DA of 98.60%, 99.33%, 98.43%, 97.82%, and 98.36% under all IFs of 10% to 50%, respectively.

Table 2 and Fig. 3 report a detailed DA examination of the BIOMDL-FD model with other fault detection models with an IF of 10%. The results showed that the BIOMDL-FD model gained maximum fault detection results under all faults. For instance, the BIOMDL-FD model has provided an increased DA of 98.91%, whereas the ERELM, ESVM, EELM, and EKNN models have shown decreased DA of 97.74%, 93.24%, 92.46%, and 92.19% respectively. Moreover, on OOBF, the BIOMDL-FD model has accomplished a higher DA of 99.88%, whereas the ERELM, ESVM, EELM, and EKNN models have demonstrated lower DA of 98.63%, 98.03%, 91.66%, and 90.63% respectively.

Table 1: Detection accuracy analysis of BIOMDL-FD system under distinct faults

	Detection accuracy (%)				
Fault types	Induced fault = 10 (%)	Induced fault = 20 (%)	Induced fault = 30 (%)	Induced fault = 40 (%)	Induced fault = 50 (%)
Offset fault	98.91	99.33	98.55	98.31	98.73
Gain fault	97.67	99.25	96.27	98.29	97.98
Stuck-at fault	97.94	99.47	99.68	96.63	98.73
Out of bounds	99.88	99.28	99.22	98.05	98.01
Average	**98.60**	**99.33**	**98.43**	**97.82**	**98.36**

Table 2: DA analysis of BIOMDL-FD system with other algorithms under IF of 10%

	Detection accuracy (induced faults = 10%)				
Fault types	ERELM	ESVM	EELM	EKNN	BIOMDL-FD
Offset fault	97.74	93.24	92.46	92.19	98.91
Gain fault	96.85	90.13	91.66	79.63	97.67
Stuck-at fault	97.11	91.60	95.45	90.75	97.94
Out of bounds	98.63	98.03	91.66	90.63	99.88
Average	**97.58**	**93.25**	**92.81**	**88.30**	**98.60**

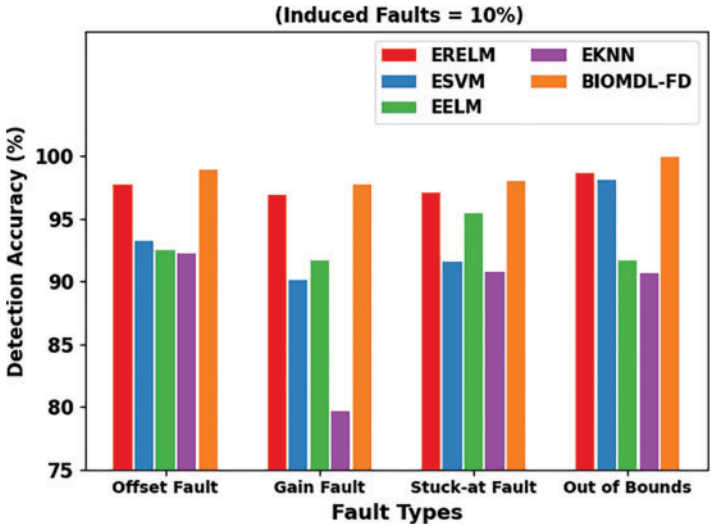

Figure 3: DA analysis of BIOMDL-FD system under IF of 10%

Table 3 and Fig. 4 report a detailed DA inspection of the BIOMDL-FD technique with other fault detection models with an IF of 20%. The results revealed that the BIOMDL-FD approach attained maximum fault detection under all faults. For example, on OF, the BIOMDL-FD technique has offered a high DA of 99.33%, whereas the ERELM, ESVM, EELM, and EKNN approaches have exhibited decreased DA of 99.04%, 93.93%, 81.60%, and 81.78% correspondingly. Furthermore, on OOBF, the BIOMDL-FD method has established a higher DA of 99.28%, whereas the ERELM, ESVM, EELM, and EKNN approaches have established lower DA of 98.20%, 92.93%, 83.32%, and 81.69% correspondingly.

Table 3: DA analysis of BIOMDL-FD system with other algorithms under IF of 20%

Detection accuracy (induced faults = 20%)					
Fault types	ERELM	ESVM	EELM	EKNN	BIOMDL-FD
Offset fault	99.04	93.93	81.60	81.78	99.33
Gain fault	98.15	93.21	84.41	80.99	99.25
Stuck-at fault	98.45	93.37	83.46	80.78	99.47
Out of bounds	98.20	92.93	83.32	81.69	99.28
Average	**98.46**	**93.36**	**83.20**	**81.31**	**99.33**

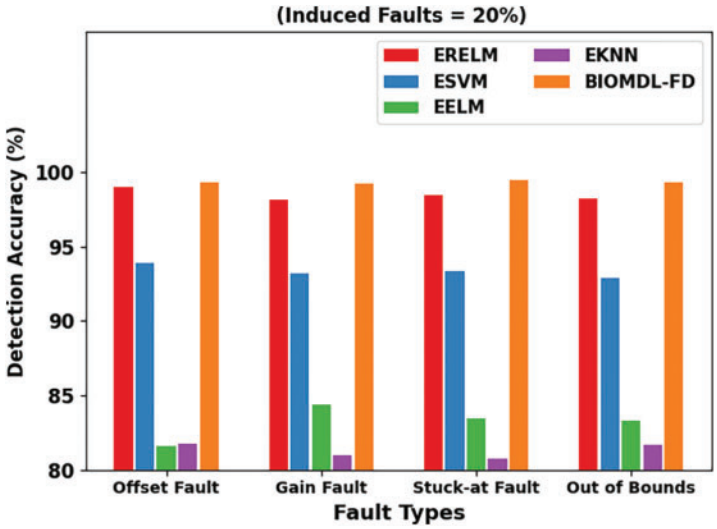

Figure 4: DA analysis of BIOMDL-FD system under IF of 20%

Table 4 and Fig. 5 report a comprehensive DA investigation of the BIOMDL-FD method with other fault detection models with an IF of 30%. The outcomes exhibited the BIOMDL-FD approach attained maximal fault detection results under all faults. For example, on OF, the BIOMDL-FD method has offered a high DA of 98.55%, whereas the ERELM, ESVM, EELM, and EKNN approaches have shown decreased DA of 97.45%, 97.52%, 82.72%, and 80.07% correspondingly. Moreover, on OOBF, the BIOMDL-FD technique has accomplished a higher DA of 99.22%, whereas the ERELM, ESVM, EELM, and EKNN models have illustrated lower DA of 97.31%, 98.24%, 82.94%, and 80.32% correspondingly.

Table 4: DA analysis of BIOMDL-FD system with other algorithms under IF of 30%

Detection accuracy (induced faults = 30%)					
Fault types	ERELM	ESVM	EELM	EKNN	BIOMDL-FD
Offset fault	97.45	97.52	82.72	80.07	98.55
Gain fault	95.10	94.96	83.14	73.17	96.27
Stuck-at fault	97.21	98.82	83.00	80.13	99.68
Out of bounds	97.31	98.24	82.94	80.32	99.22
Average	**96.77**	**97.39**	**82.95**	**78.42**	**98.43**

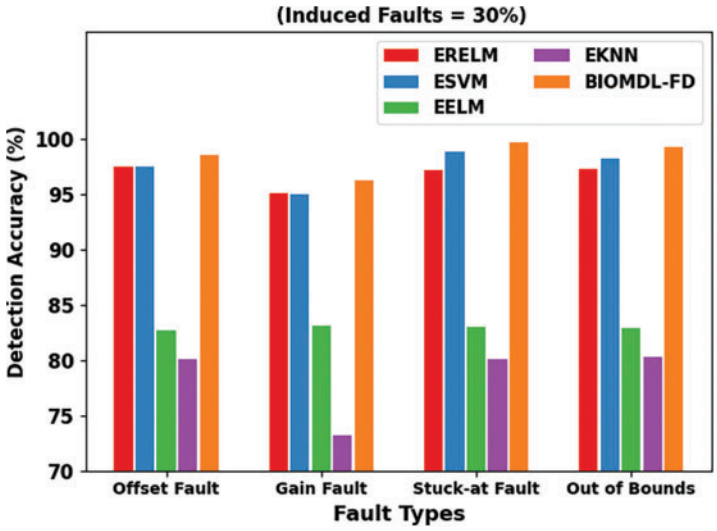

Figure 5: DA analysis of BIOMDL-FD system under IF of 30%

Table 5 and Fig. 6 report a thorough DA analysis of the BIOMDL-FD method with other fault detection models with an IF of 40%. The results exemplified that the BIOMDL-FD algorithm achieved maximum fault detection results under all faults. For example, on OF, the BIOMDL-FD method has offered an increased DA of 98.31%, whereas the ERELM, ESVM, EELM, and EKNN methods have revealed decreased DA of 97.10%, 92.82%, 81.34%, and 62.60% correspondingly. Furthermore, on OOBF, the BIOMDL-FD method has accomplished a higher DA of 98.05%, whereas the ERELM, ESVM, EELM, and EKNN approaches have demonstrated lower DA of 97.25%, 95.11%, 69.67%, and 63.30% correspondingly.

Table 6 and Fig. 7 report a detailed DA inspection of the BIOMDL-FD method with other fault detection models with an IF of 50%. The results exhibited that the BIOMDL-FD approach attained maximum fault detection under all faults. For example, on OF, the BIOMDL-FD method has presented a high DA of 98.73%, whereas the ERELM, ESVM, EELM, and EKNN approaches have exhibited decreased DA of 97.65%, 84.64%, 48.53%, and 59.17% correspondingly. Besides, on OOBF, the BIOMDL-FD method has established a higher DA of 98.01%, whereas the ERELM, ESVM, EELM, and EKNN methods have demonstrated lower DA of 97.12%, 84.03%, 48.57%, and 59.08% correspondingly.

Table 5: DA analysis of BIOMDL-FD system with other algorithms under IF of 40%

Detection accuracy (induced faults = 40%)					
Fault types	ERELM	ESVM	EELM	EKNN	BIOMDL-FD
Offset fault	97.10	92.82	81.34	62.60	98.31
Gain fault	97.26	93.34	80.64	62.18	98.29
Stuck-at fault	95.70	90.58	80.56	60.98	96.63
Out of bounds	97.25	95.11	69.67	63.30	98.05
Average	**96.83**	**92.96**	**78.05**	**62.27**	**97.82**

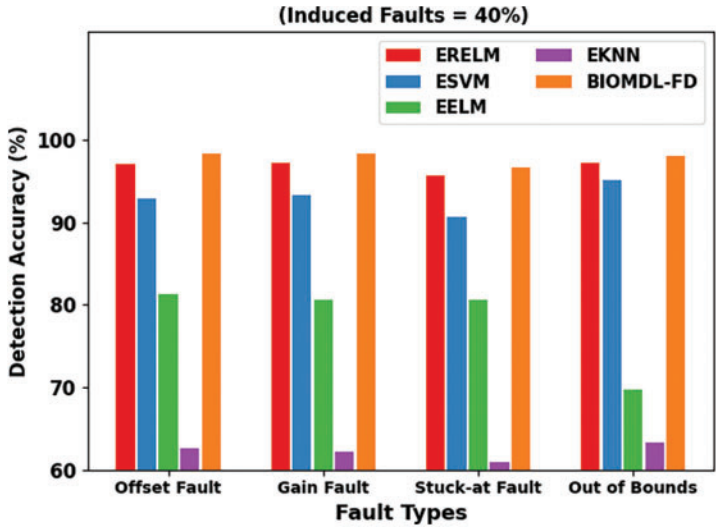

Figure 6: DA analysis of BIOMDL-FD system under IF of 40%

Table 6: DA analysis of BIOMDL-FD system with other algorithms under IF of 50%

Detection accuracy (induced faults = 50%)					
Fault types	ERELM	ESVM	EELM	EKNN	BIOMDL-FD
Offset fault	97.65	84.64	48.53	59.17	98.73
Gain fault	97.17	85.01	69.90	58.46	97.98
Stuck-at fault	97.44	84.43	48.00	58.53	98.73
Out of bounds	97.12	84.03	48.57	59.08	98.01
Average	**97.35**	**84.53**	**53.75**	**58.81**	**98.36**

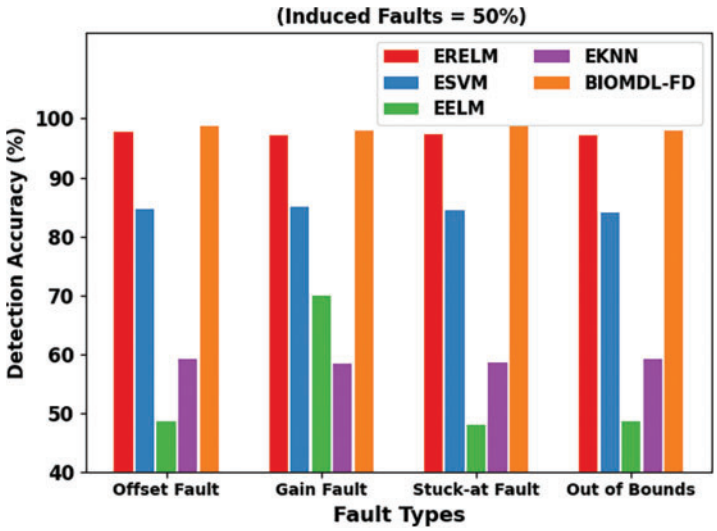

Figure 7: DA analysis of BIOMDL-FD system under IF of 50%

Finally, a detailed classification accuracy (CA) inspection of the BIOMDL-FD model with compared methods take place in Table 7 [14,21]. The experimental values indicated that the BIOMDL-FD model had reached increased CA values under all sensor numbers. For instance, on sensor 2, the BIOMDL-FD model has reached a higher CA of 84.09%, whereas the SS, WMV, NB, and BF models have attained lower CA of 77.91%, 80.49%, 79.99%, and 83.40%, respectively. In addition, on sensor 4, the BIOMDL-FD approach has attained a higher CA of 92.24%, whereas the SS, WMV, NB, and BF methods have reached lower CA of 77.40%, 88.83%, 88.13%, and 89.71% correspondingly.

Table 7: CA analysis of BIOMDL-FD system with existing methodologies

Classification accuracy					
Sensor number	Single sensor (Avg.)	Weighted majority vote	Naïve Bayes	Belief function	BIOMDL-FD
1	78.98	79.29	79.04	79.42	79.23
2	77.91	80.49	79.99	83.40	84.09
3	77.21	87.50	86.05	87.06	88.57
4	77.40	88.83	88.13	89.71	92.24
5	77.46	92.55	91.73	92.93	94.76
6	77.40	94.07	92.99	94.44	97.41
7	77.72	95.58	94.89	95.77	98.48
8	77.59	96.09	96.02	97.29	98.80
9	77.34	96.59	97.35	98.17	98.93
10	77.21	97.03	98.04	98.61	99.18
11	77.21	97.73	98.04	98.67	99.37

Besides, on sensor 11, the BIOMDL-FD model has attained a higher CA of 99.37%, whereas the SS, WMV, NB, and BF models have achieved a lower CA of 72.1%, respectively 97.73%, 98.04%, and 98.67%, correspondingly. Thus, the presented BIOMDL-FD model can be employed for fault detection in the WSN.

5 Conclusion

In this article, we developed a new BIOMDL-FD technique for fault detection in WSN, thereby extending the network efficiency. The presented BIOMDL-FD technique follows a two-stage process: fault detection using the ABLSTM model and hyperparameter tuning. In the ABLSTM model, the attention mechanism enables us to learn the relationships between the inputs and modify the probability to give more attention to essential features. At the same time, the BIO algorithm is employed for optimal hyperparameter tuning of the ABLSTM model, which is stimulated by billiard games. A series of experimental analyses are made to affirm the enhanced fault detection outcomes of the BIOMDL-FD technique. Detailed simulation outcomes demonstrate the improvement of the BIOMDL-FD approach over other models with a maximum classification accuracy of 99.37%. In future, the presented BIOMDL-FD technique can be realized in a large-scale real-time environment.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] F. Fan, S. C. Chu, J. S. Pan, C. Lin and H. Zhao, "An optimized machine learning technology scheme and its application in fault detection in wireless sensor networks," *Journal of Applied Statistics*, pp. 1–18, 2021. https://doi.org/10.1080/02664763.2021.1929089

[2] T. Mahmood, J. Li, Y. Pei, F. Akhtar, S. A. Butt *et al.,* "An intelligent fault detection approach based on reinforcement learning system in wireless sensor network," *The Journal of Supercomputing*, vol. 78, no. 3, pp. 3646–3675, 2022.

[3] U. Saeed, Y. D. Lee, S. U. Jan and I. Koo, "CAFD: Context-aware fault diagnostic scheme towards sensor faults utilizing machine learning," *Sensors*, vol. 21, no. 2, pp. 617, 2021.

[4] A. Singh, J. Amutha, J. Nagar, S. Sharma and C. C. Lee, "Lt-fs-id: Log-transformed feature learning and feature-scaling-based machine learning algorithms to predict the k-barriers for intrusion detection using wireless sensor network," *Sensors*, vol. 22, no. 3, pp. 1070, 2022.

[5] M. Chen, Z. Li, P. Chen, W. Liu and A. Liu, "A novel differential dynamic gradient descent optimization algorithm for resource allocation and offloading in the COMEC system," *International Journal of Intelligent Systems*, vol. 37, no. 11, pp. 8365–8386, 2022.

[6] A. E. Abouelregal and M. Marin, "The size-dependent thermoelastic vibrations of nanobeams subjected to harmonic excitation and rectified sine wave heating," *Mathematics*, vol. 8, no. 7, pp. 1128, 2020.

[7] M. L. Scutaru, S. Vlase, M. Marin and A. Modrea, "New analytical method based on dynamic response of planar mechanical elastic systems," *Bound Value Problems*, vol. 2020, no. 1, pp. 104, 2020.

[8] Y. Hafeez, S. Ali, N. Jhanjhi, M. Humayun, A. Nayyar *et al.,* "Role of fuzzy approach towards fault detection for distributed components," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1979–1996, 2021.

[9] M. Turkyilmazoglu, "An extended epidemic model with vaccination: Weak-immune SIRVI," *Physica A: Statistical Mechanics and its Applications*, vol. 598, pp. 127429, 2022.

[10] M. Tuerkyilmazoglu, "A restricted epidemic SIR model with elementary solutions," *SSRN Journal*, 2021. https://doi.org/10.2139/ssrn.3970806

[11] R. Regin, S. S. Rajest and B. Singh, "Fault detection in wireless sensor network based on deep learning algorithms," *EAI Transactions on Scalable Information Systems*, vol. 8, no. 32, pp. 1–7, 2021.

[12] U. Saeed, S. U. Jan, Y. D. Lee and I. Koo, "Fault diagnosis based on extremely randomized trees in wireless sensor networks," *Reliability Engineering & System Safety*, vol. 205, pp. 107284, 2021.

[13] K. Kalaiselvi, L. Vanitha, K. D. Thilak, T. R. Kumar, S. Saranya *et al.,* "Performance analysis of malicious and link failure detection system using deep learning methodology," *Wireless Personal Communications*, pp. 1–16, 2021. https://doi.org/10.1007/s11277-021-08790-9

[14] S. U. Jan, Y. D. Lee and I. S. Koo, "A distributed sensor-fault detection and diagnosis framework using machine learning," *Information Sciences*, vol. 547, pp. 777–796, 2021.

[15] G. Kaur and P. Chanak, "An energy aware intelligent fault detection scheme for IoT-enabled WSNs," *IEEE Sensors Journal*, vol. 22, no. 5, pp. 4722–4731, 2022.

[16] S. Gavel, R. Charitha, P. Biswas and A. S. Raghuvanshi, "A data fusion based data aggregation and sensing technique for fault detection in wireless sensor networks," *Computing*, vol. 103, no. 11, pp. 2597–2618, 2021.

[17] V. Agarwal, S. Tapaswi and P. Chanak, "Intelligent fault-tolerance data routing scheme for IoT-enabled WSNs," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 16332–16342, 2022.

[18] S. Lavanya, A. Prasanth, S. Jayachitra and A. Shenbagarajan, "A tuned classification approach for efficient heterogeneous fault diagnosis in IoT-enabled WSN applications," *Measurement*, vol. 183, pp. 109771, 2021. https://doi.org/10.1016/j.measurement.2021.109771

[19] T. Su, H. Sun, J. Zhu, S. Wang and Y. Li, "BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset," *IEEE Access*, vol. 8, pp. 29575–29585, 2020.

[20] A. Kaveh, M. Khanzadi and M. R. Moghaddam, "Billiards-inspired optimization algorithm; a new meta-heuristic method," *Structures*, vol. 27, pp. 1722–1739, 2020.

[21] A. Javaid, N. Javaid, Z. Wadud, T. Saba, O. E. Sheta *et al.,* "Machine learning algorithms and fault detection for improved belief function based decision fusion in wireless sensor networks," *Sensors*, vol. 19, no. 6, pp. 1334, 2019.