ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20250108002248
Original scientific paper

Trust Aware Water Strider Optimization-Based Clustering with Intrusion Detection in IoT Environments

Nada ALZABEN, Mashael MAASHI, Menwa ALSHAMMERI, V. SARASWATHI, R. KIRUBA BURI, S. GAYATHRI PRIYA*

Abstract: The Internet of Things (IoT) has emerged as a transformative technology revolutionizing domains such as healthcare, smart cities, and e-governance. However, real-time IoT integration faces significant hurdles due to the limited energy resources and computational capabilities of IoT devices. Additionally, security vulnerabilities in IoT systems amplify threats to the integrity and reliability of smart applications, necessitating robust solutions. This study introduces a novel Trust-Aware Improved Water Strider Optimization-based Clustering with Intrusion Detection System (TAIWSOC-IDS) for enhancing IoT environments' security and efficiency. The proposed framework comprises two key stages: clustering and intrusion detection. In the clustering stage, the TAIWSOC technique identifies clusters and cluster heads (CHs) by optimizing a fitness function that considers residual energy, communication distance, and trust metrics, ensuring balanced energy consumption and secure communication. In the intrusion detection stage, a Chaos Game Optimization (CGO)-enhanced Multihead Attention Bidirectional Long Short-Term Memory (MHA-BiLSTM) model is employed. The CGO algorithm optimally tunes the hyper parameters of the MHA-BiLSTM model, improving its ability to accurately classify normal and malicious activities. Experimental evaluation demonstrates that the TAIWSOC-IDS outperforms existing state-of-the-art methods in terms of energy efficiency, clustering reliability, and intrusion detection accuracy. The proposed system achieves enhanced detection rates while significantly reducing false positives, proving its efficacy in mitigating security threats. By addressing both energy efficiency and security concerns, TAIWSOC-IDS serves as a comprehensive framework for developing secure, sustainable, and efficient IoT applications across diverse real-time scenarios.

Keywords: chaos game optimization; internet of things (IoT); intrusion detection system (IDS); multihead attention BiLSTM; water strider optimization

1 INTRODUCTION

The Internet of Things (IoT) refers to interconnected objects, typically electronic devices, which communicate over a network to interact seamlessly [1]. While the concept of IoT is not new, its origins trace back to the late 1990s when Kevin Ashton, the co-founder of the Auto-ID Center at MIT, described IoT as linking computers to physical objects [2]. IoT devices often include embedded computing components, mechanical or digital machinery, and microchips in various entities, including living organisms, which are identified by unique identifiers (UIDs). However, real-time IoT applications began gaining momentum between 2008 and 2009, as identified by Cisco Systems [3]. Since then, IoT has advanced significantly, finding applications in wearable health monitors, smart cities, connected vehicles, precision agriculture, logistics tracking, and biometric cybersecurity scanners.

The rapid proliferation of IoT devices has also introduced critical challenges, particularly in security and privacy. During the development phase, security and privacy considerations were often secondary, leading to vulnerabilities that are difficult to address due to the scale and complexity of IoT networks [4]. The vast amounts of data generated by IoT devices make detecting security breaches challenging without robust systems. Intrusion Detection Systems (IDS) have emerged as a vital defense mechanism for monitoring network activity and detecting potential intrusions [5].

Traditional IDS primarily focused on host-level security. However, advancements in technology have shifted the focus towards network-based IDS that scrutinize traffic for intrusion behaviors [6]. Over the past three decades, machine learning (ML) techniques, such as decision trees (DT), support vector machines (SVM), and artificial neural networks (ANN), have been widely adopted for intrusion detection [7]. These ML methods enhance the ability to analyze abnormal traffic and detect malicious activities that may go unnoticed through manual inspection.

Clustering has also been extensively studied in the context of sensor network systems for managing data communication [8]. Sensor nodes in such networks operate on limited energy sources, necessitating energy-efficient communication to extend the network's lifetime. While IoT devices differ from traditional sensor networks due to their often continuous power supply, reducing the number of Internet connections instead of focusing solely on energy efficiency has become a critical factor in managing network costs [9, 10].

This study proposes a Trust-Aware Improved Water Strider Optimization-based Clustering with Intrusion Detection System (TAIWSOC-IDS) for IoT environments. The proposed model employs a two-stage process: clustering and intrusion detection. The TAIWSOC technique optimizes cluster formation and cluster head (CH) selection using a fitness function incorporating residual energy, communication distance, and trust factors. For intrusion detection, a Chaos Game Optimization (CGO)-enhanced Multihead Attention Bidirectional Long Short-Term Memory (MHA-BiLSTM) model is utilized. The CGO algorithm optimizes MHA-BiLSTM hyperparameters to enhance classification accuracy. Extensive simulations validate the superior performance of the TAIWSOC-IDS model in energy efficiency and intrusion detection, showcasing its potential for secure and sustainable IoT applications.

The overall fitness function $F(x)$ used in the TAIWSOC model is expressed as:

$$F(x) = \alpha_1 * (RE_x / RE_{\max}) + \alpha_2 * (1 / \text{Avg}D_x) + \alpha_3 * T_x,$$

where RE_x is the residual energy of node x, $RE_{\max}$ is the maximum initial energy, $\text{Avg}D_x$ is the average communication distance to its neighbors, and T_x is the trust score of the node. The trust value is computed by integrating direct trust (DT_x) and indirect trust (IT_x) using:

$$T_x = \beta * DT_x + (1 - \beta) * IT_x$$

where $\beta \in [0, 1]$ is a balance coefficient that adjusts the emphasis between firsthand and secondhand observations. DT_x is based on the ratio of successful interactions, while IT_x aggregates trust feedback from neighboring nodes. The weights α_1, α_2, and α_3 are selected empirically to balance energy efficiency, communication reliability, and security.

To optimize the MHA-BiLSTM model, the Chaos Game Optimization (CGO) algorithm is applied to identify optimal hyperparameters. CGO initializes a population of chaotic sequences using a logistic map defined as: $x_{n+1} = \mu x_n(1 - x_n)$, where $\mu \in (3.57, 4]$ ensures chaotic behavior. Each candidate solution encodes values for key parameters including: (i) number of attention heads ∈ [2, 8], (ii) hidden units ∈ [64, 256], (iii) learning rate ∈ [0.00001, 0.01], and (iv) batch size ∈ [16, 128]. Each combination is evaluated via cross-validation accuracy on the validation set, and the top candidates are evolved through CGO's exploration and exploitation mechanism, refining the model's performance.

2 RELATED WORKS

Recent studies have introduced innovative Intrusion Detection System (IDS) techniques to enhance the security of IoT environments. One study proposed an IDS approach for identifying unauthorized IoT devices using deep learning (DL). This method relied on RF fingerprinting, where device-specific physical layer features were used, making impersonation highly complex. RF traces were collected from six identical ZigBee devices using a USRP-based testbed across varying Signal-to-Noise Ratios (SNRs) to ensure robustness. A convolutional neural network (CNN) was employed to extract features from the RF traces, followed by measurement minimization and decorrelation to improve detection performance [11].

Another approach examined IoT devices and their vulnerabilities to specific types of attacks. A digitized mechanism was developed using machine learning (ML)-based clustering to detect suspicious behavior, exploiting the characteristic dissipation of current supply. The system utilized a K-Means clustering algorithm in a supervised training environment to detect anomalies effectively [12]. A novel unsupervised anomaly detection Host-IDS for IoT was introduced using adversarial training based on a generative adversarial network (GAN). This IDS, termed "EdgeIDS", was tailored for IoT devices with limited functionality, as such devices primarily handle specific data types rather than the diverse data exchanged by conventional systems like computers or servers [13].

A three-phase IDS methodology was also suggested, consisting of data clustering with reduction, oversampling, and classification using a Single Hidden Layer Feed-Forward Neural Network (SLFN). The innovation lies in the oversampling and reduction processes, which create balanced and effective training datasets. This hybrid approach combined supervised and unsupervised techniques to enhance intrusion detection capabilities [14]. Another study introduced a Fuzzy IDS for IoT Networks (FROST), which leverages fuzzy set theory to improve the flexibility of learning processes. FROST was designed to enhance the classification of erroneous data and included mechanisms for online detection of new intrusion types during classification [15].

Furthermore, a novel anomaly detection framework based on unsupervised DL techniques was proposed to identify network threats. This study explored the applicability of DL in anomaly detection by comparing the use of Restricted Boltzmann Machines (RBMs) as generative energy-based models and Autoencoders as non-probabilistic methods. The results demonstrated the effectiveness of unsupervised DL techniques in detecting anomalies in IoT networks [16].

These studies collectively highlight the potential of advanced ML and DL techniques, along with clustering and fuzzy logic approaches, to improve the accuracy and efficiency of IDS frameworks for IoT environments. However, the scalability and adaptability of such systems to dynamic IoT ecosystems remain critical areas for further exploration.

3 PROPOSED WORK

In this study, a new TAIWSOC-IDS model has been developed to enhance network performance and security in the IoT environment. The presented TAIWSOC-IDS model follows a two-stage process such as clustering and intrusion detection. For cluster construction process, the TAIWSOC technique is utilized with the use of fitness function involving residual energy, distance, and trust factor. To detect intrusions, the CGO with MHA-BiLSTM model is employed. Fig. 1 show cases the block diagram of TAIWSOC-IDS approach.

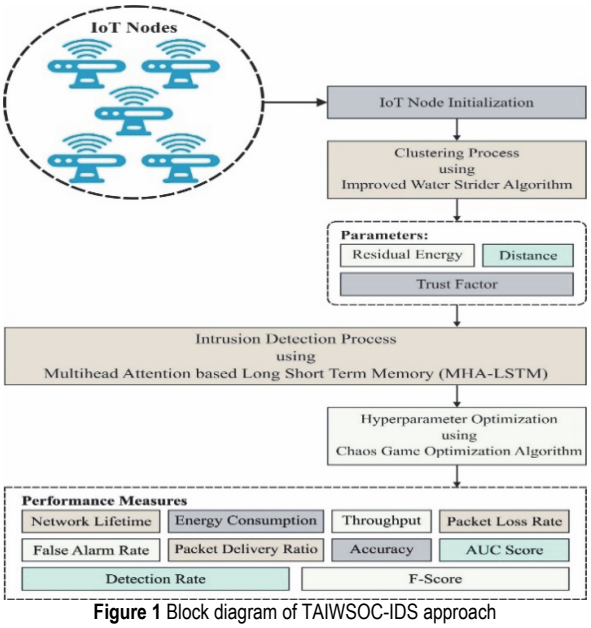

Figure 1 Block diagram of TAIWSOC-IDS approach

3.1 Algorithmic Procedure of TAIWSOC Technique

To organize clusters and choose CHs, the TAIWSOC technique is utilized with the use of fitness function involving residual energy, distance, and trust factor. The WSA aimed at solving the optimization problem with complexity based on territorial behavior of the WSs [17]. In the next section, the basic conception of this approach is described as follows. The WSA comprises birth, creating territory, feeding, mating, death, and succession stages.

Birth stage

The male WSs which ate is called keystones, get laid and mate with the female WSs to produce eggs as novel generation and it is accomplished by the following expression:

$$X\_i\left(0\right)=Lb+r\times\left(Ub-Lb\right)\\ i=1,2,\ldots,n. \tag{1}$$

In Eq. (1), $X\left(0\right)$ denotes the first location of the *i*-th WSs, n refers to the WS count, r describes a consistently accidental number lies within [0, 1], *Ub* and *Lb* represent the problem of high and low limits, correspondingly. The produced WSs illustrate the candidate solution, are later analyzed on the basis of objective function. The food availability shows the significance of the target features.

Establishing territory stage

The population of WSs is territorial. Partially some female bugs and a keystone were positioned in the territory. To define the n territory number with n WSs, the objective values are stored and later, classified into certain classes from which the territory is accumulated with an order by taking an individual WS from every group into account.

Mating stage

A big time of life in the WSs is expended to the search or mating. This procedure was performed by forwarding specific courtship signals from keystone to the female WSs. Here, once the female accepts the signal and returns an acceptance signal, the mating initiates, otherwise, the mating is cancelled. It is due to their stronger shield of females that protects them against the forced mating system from the keystone. The mathematical expression of the mating stage can be given in the following:

$$\begin{cases}X_i\left(t+1\right)=X_i\left(t\right)+R\times r, \text{ if mating happens}\\ X_i\left(t+1\right)=X_i\left(t\right)+R\times\left(1+r\right), \text{ Otherwise}\end{cases} \tag{2}$$

In Eq. (2), $X_i\left(t\right)$ defines the location of *i*-th WSs, r describes a random integer lies within $[0,1]$, R shows the distance among the keystone location $\left(X_i\left(t\right)\right)$ and the endpoint located in the female territories $\left(X_F\left(t\right)\right)$ and is accomplished as follows:

$$R=X_F\left(t\right)-X_i\left(t\right) \tag{3}$$

Foraging stage

Since WSs spent a long time tempting or breeding, they should save energy to recover. This is provided by searching for a novel location with sufficient energy. Here, the objective role estimates the existing location. When the objective values have a value higher than the preceding condition, the food for refining is insufficient, otherwise, it passes over the optimal territory that comprises the maximum fitness $\left(X_{BL}\left(t\right)\right)$. The mathematical expression of the foraging stage can be given by:

$$X_i^{t+1}=X_i\left(t\right)+2\times r(md\times\left(X_{BL}\left(t\right)-X_i\left(t\right)\right) \tag{4}$$

Death and succession

In order to ensure that the decision variable (solution parameter) is kept in the boundary when an unknown WS enters a territory, the keystone behaves aggressively until bringing him out, or killing him and taking out the attacker. This stage is extremely violent and might cause some murdering between residents and aggressors. Therefore, if the previous location offers worse values for the objective function, the WS will be disregarded and replaced by a new WS, otherwise, the keystones continue their life:

$$X_i^{t+1}=Lb_j\left(t\right)+r\times\left(Ub_j\left(t\right)-Lb_j\left(t\right)\right) \tag{5}$$

In Eq. (5), $Ub_j\left(t\right)$ and $Lb_j\left(t\right)$ described the upper and lower bounds of the WS condition inside *j*-th territories.

Termination condition

Finally, the process ends if the ending condition is attained. In the default WSA, the ending condition referred to the maximal iteration. The IWSA is derived by the use of late acceptance hill-climbing algorithm (LAHC) concept that is a revised edition of Hill climbing (HC) approach [18]. The key concept of the proposed algorithm is to accept the non-improving solution as soon as the quality of newly produced solution is equal (or better) than those that were newly accepted a certain iteration before. Actually, LAHC initiates from the first solution and repeatedly improves them by comparing the fresh solution candidate with the existing one for accepting or rejecting them. Therefore, to employ the LAHC rules, the process will generate a list with a static length for saving the quality of newly visited solution. When the quality of fresh solution candidate is more efficient when compared to the quality of latter component in the list, the solution candidate is accepted as the existing solution. Next, the component in the last part of list is detached and the quality values of recently accepted solution are added at the starting of the list.

The proposed TAIWSOC approach develops a FF containing several parameters, mostly trust parameters, to secure procedure. The RE of IoT node (*x*) if transmitting k bits to terminus IoT node (*y*) on distance d, was defined as:

$$RE=E-\left(E_T\left(k,d\right)+E_{R(k)}\right) \tag{6}$$

In Eq. (6), E signifies the current energy of IoT nodes and E_T represents the energy used to sense data.

$$E_T\left(k,d\right)=kE_e+KE_ad^2 \tag{7}$$

In Eq. (7), E_e implies the energy of electrons and E_a refers to the amplified energy, $E_{R(k)}$ demonstrates the energy dissipated for obtaining data as:

$$E_{R(k)} = kE_e \tag{8}$$

The 3 variables to choose CH are the average distance (AvgD) to adjacent IoT nodes. The AvgD represents the average distance values to IoT node to their individual hob neighboring IoT node which is provided in the subsequent.

$$AvgNBDist_i = \frac{\sum_{j=1}^{NB_i} dist\left(i, nb_j\right)}{NB_i} \tag{9}$$

In Eq. (9), $dist\left(i, nb_j\right)$ indicates the distance in the IoT node to neighbouring *j*-th IoT nodes. The energy trust was measured when the node was appropriate RE to complete new broadcasts and data processing tasks. At present, Tr_{ij}^{dir} and TR_{ij}^{indir} define the direct and indirect trust of one to other nodes correspondingly.

3.2 Intrusion Detection using MHA-BiLSTM Model

To accomplish security, the MHA-BiLSTM model is employed to recognize and categorize the presence of intrusions in the IoT environment. The LSTM is different from RNN. RNN only takes short-term memory [19]. The LSTM network integrates short-term memory with long-term memory with difficult gate process that is learned long-term dependent data. The LSTM forget gate computes the value amongst zero and one for determining that several data were discarded in the cell states. An input gate defines that novel data was storing from the cell state, computes the state candidate value for updating the present cell state, and eventually computes the present cell state dependent upon the forget as well as input gates. Fig. 2 demonstrates the BiLSTM technique.

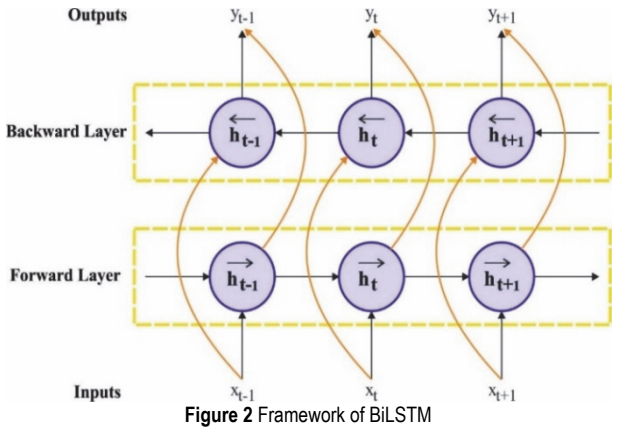

Figure 2 Framework of BiLSTM

The resultant gate defines that part of cell states is output, and the last outcome was computed dependent upon the output gate and the present unit state.

$$f_t = \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{10}$$

$$i_t = \sigma\left(W_i\left[h_{t-1}, x_t\right] + b_i\right) \tag{11}$$

$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{12}$$

$$h_t = O_t * \tanh\left(C_t\right) \tag{13}$$

whereas f_t signifies the forget gate, σ indicates the sigmoid function, h_{t-1} has preceding the hidden state, x_t is present input, i_t denotes the input gate, O_t stands for the resultant gate, h_t is last output, and C_t is present state. W_f, W_i, and W_o demonstrated the weight of forgetting, input, and output gates correspondingly, and b_f, b_i, and f_0 define the bias of 3 gates correspondingly. On allowing single attention layer, it can be tough for capturing the range of several features of inputs. For overcoming the problem Multihead attention (MHA) block utilized from the presented method that computes the several attention weighted sum before assuming single attention pass on the values [20]. Therefore, it can be termed "MHA". During this work, several attention heads were utilized. The resultants of these 2 heads were fed to drop out layers and next combined utilizing concatenation layer and the resultant is provided to LSTM layer. The attention process allocates weighted to context word for determining the exact word that is utilized for determining the sentiment of provided input sequences.

$$a_1 = w_1 + w_2 + w_3 + \ldots + w_n \tag{14}$$

$$a_2 = w_1 + w_2 + w_3 + \ldots + w_n \tag{15}$$

The NSL-KDD dataset is utilized to evaluate the TAIWSOC-IDS model. It includes realistic network traffic data encompassing 41 features across five major categories: Normal, DoS, Probe, R2L, and U2R. The dataset refines KDD'99 by eliminating duplicate records, making it more suitable for modern intrusion detection research and addressing class imbalance challenges, thus better simulating real-world IoT environments.

4 RESULTS AND DISCUSSION

This section discusses the experimental results offered by the TAIWSOC-IDS model under distinct number of IoT nodes. The results are inspected under two major aspects such as network performance and security. Tab. 1 and Fig. 3 provide a detailed PDR assessment of the TAIWSOC-IDS model with existing models under distinct IoT nodes [22, 23]. The results represent that the TAIWSOC-IDS model has shown enhanced values of PDR under all IoT nodes. For instance, with 100 IoT nodes, the TAIWSOC-IDS model has gained higher PDR of 99.33% whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C models have obtained reduced PDR of 97.88%, 95.65%, 94.20%, and 89.74% respectively. Also, with 300 IoT nodes, the TAIWSOC-IDS method has reached increased PDR of 98.09% whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C methodologies have obtained reduced PDR of 95.08%, 94.20%, 91.19%, and 88.13% respectively. Moreover, with 500 IoT nodes, the TAIWSOC-IDS technique has acquired increased PDR of 97.62% whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C approaches have obtained reduced PDR of 92.90%, 89.01%, 85.85%, and 84.87% correspondingly.

Table 1 PDR analysis of TAIWSOC-IDS approach with existing methods under distinct IoT nodes

Packet Delivery Ratio / %					
Nodes	TAIWSOC-IDS	CTEEDGM	DLLeach	MSGAO-C	GAO-C
100	99.33	97.88	95.65	94.20	89.74
200	98.97	96.69	95.29	93.16	89.22
300	98.09	95.08	94.20	91.19	88.13
400	98.03	94.30	92.43	88.75	87.61
500	97.62	92.90	89.01	85.85	84.87

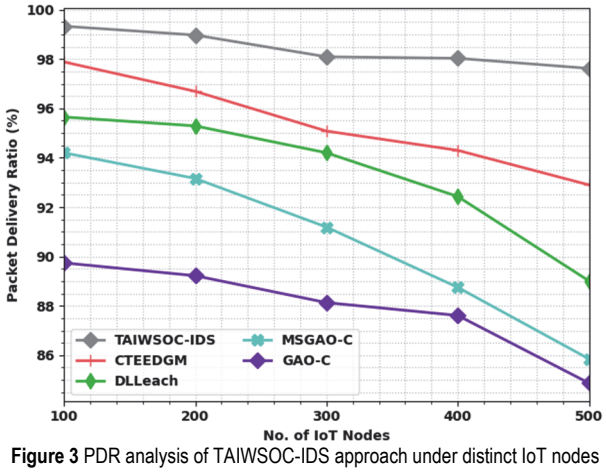

Figure 3 PDR analysis of TAIWSOC-IDS approach under distinct IoT nodes

Table 2 Throughput analysis of TAIWSOC-IDS approach with existing methods under distinct IoT nodes

Throughput / Mbps					
Nodes	TAIWSOC-IDS	CTEEDGM	DLLeach	MSGAO-C	GAO-C
100	96.76	90.15	88.08	84.06	75.64
200	97.02	89.64	85.36	81.21	75.12
300	95.99	89.64	82.38	77.97	74.08
400	93.39	88.86	81.08	76.93	69.94
500	92.10	86.27	76.03	66.57	64.10

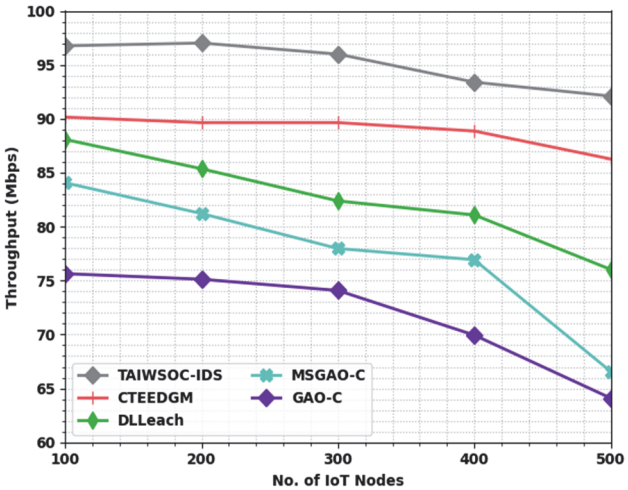

Figure 4 Throughput analysis of TAIWSOC-IDS approach under distinct IoT nodes

Tab. 2 and Fig. 4 present a detailed throughput (THPT) assessment of the TAIWSOC-IDS approach with existing models under distinct IoT nodes. The results represent that the TAIWSOC-IDS technique has shown enhanced values of THPT under all IoT nodes. For instance, with 100 IoT nodes, the TAIWSOC-IDS methodology has acquired increased THPT of 96.76 Mbps whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C models have obtained reduced THPT of 90.15 Mbps, 88.08 Mbps, 84.06 Mbps, and 75.64 Mbps correspondingly. Similarly, with 300 IoT nodes, the TAIWSOC-IDS method has achieved increased THPT of 95.99 Mbps whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C techniques have obtained reduced THPT of 89.64 Mbps, 82.38 Mbps, 77.97 Mbps, and 74.08 Mbps correspondingly. Additionally, with 500 IoT nodes, the TAIWSOC-IDS method has gained increased THPT of 92.10Mbps whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C algorithms have obtained reduced THPT of 86.27 Mbps, 76.03 Mbps, 66.57 Mbps, and 64.10 Mbps correspondingly.

A brief energy consumption (ECM) examination of the TAIWSOC-IDS model with recent models is provided in Tab. 3 and Fig. 5. The results point out that the TAIWSOC-IDS model has resulted in enhanced performance with minimal values of ECM. For instance, with 100 nodes, the TAIWSOC-IDS model has exhibited decreased ECM of 0.119 mJ whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C techniques have depicted increased ECM of 0.134 mJ, 0.164 mJ, 0.184 mJ, and 0.196 mJ respectively. Eventually, with 300 nodes, the TAIWSOC-IDS approach has exhibited decreased ECM of 0.164 mJ whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C techniques have represented increased ECM of 0.187 mJ, 0.238 mJ, 0.280 mJ, and 0.338 mJ correspondingly. Meanwhile, with 500 nodes, the TAIWSOC-IDS method has denoted decreased ECM of 0.224 mJ whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C algorithms have depicted increased ECM of 0.259 mJ, 0.306 mJ, 0.378 mJ, and 0.403 mJ correspondingly.

Table 3 ECM analysis of TAIWSOC-IDS approach with existing methods under distinct IoT nodes

Energy Consumption / mJ					
Nodes	TAIWSOC-IDS	CTEEDGM	DLLeach	MSGAO-C	GAO-C
100	0.119	0.134	0.164	0.184	0.196
200	0.147	0.169	0.209	0.242	0.268
300	0.164	0.187	0.238	0.280	0.338
400	0.194	0.218	0.270	0.324	0.360
500	0.224	0.259	0.306	0.378	0.403

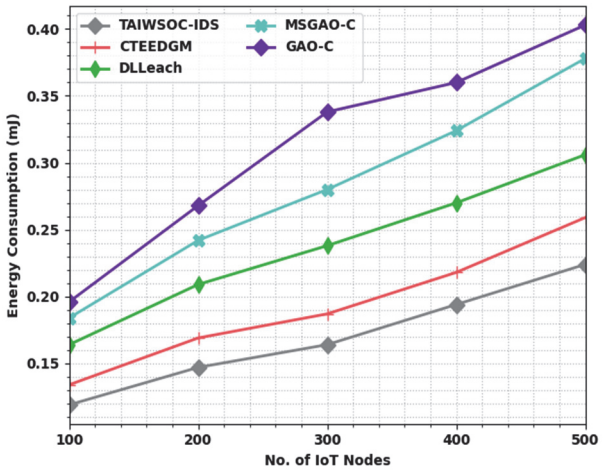

Figure 5 ECM analysis of TAIWSOC-IDS approach under distinct IoT nodes

Tab. 4 and Fig. 6 offer a detailed NLT evaluation of the TAIWSOC-IDS method with existing models under distinct IoT nodes. The results point out that the TAIWSOC-IDS approach has shown enhanced values of NLT under all IoT nodes. For instance, with 100 IoT nodes, the TAIWSOC-IDS algorithm has obtained increased NLT of 2225 rounds whereas the CTEEDGM, DLLeach,

MSGAO-C, and GAO-C techniques have achieved reduced NLT of 2082, 1944, 1861, and 1723 rounds correspondingly.

Table 4 NLT analysis of TAIWSOC-IDS approach with existing methods under distinct IoT nodes

Network Lifetime / Rounds					
Nodes	TAIWSOC-IDS	CTEEDGM	DLLeach	MSGAO-C	GAO-C
100	2225	2082	1944	1861	1723
200	2590	2374	2181	2037	1888
300	2855	2628	2407	2159	1971
400	3053	2827	2512	2358	2148
500	3291	2959	2700	2457	2325

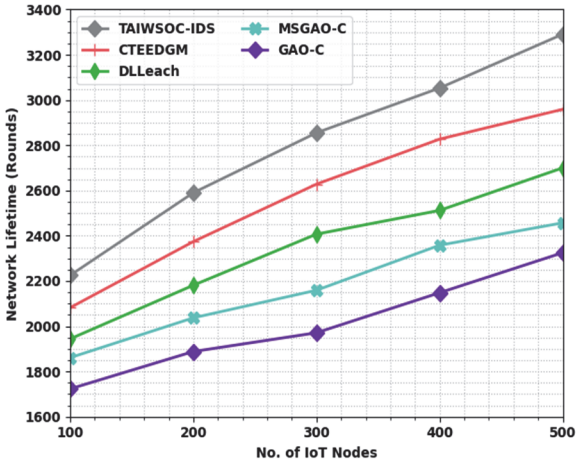

Figure 6 NLT analysis of TAIWSOC-IDS approach under distinct IoT nodes

Also, with 300 IoT nodes, the TAIWSOC-IDS system has acquired increased NLT of 2855 rounds whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C methodologies have gained reduced NLT of 2628, 2407, 2159, and 1971 rounds correspondingly. Additionally, with 500 IoT nodes, the TAIWSOC-IDS approach has attained increased NLT of 3291 rounds whereas the CTEEDGM, DLLeach, MSGAO-C, and GAO-C models have reached reduced NLT of 2959, 2700, 2457, and 2325 rounds correspondingly.

The IDS performance of the TAIWSOC-IDS model is studied under distinct class tables given in Tab. 5 and Fig. 7. The results demonstrate that the TAIWSOC-IDS model has recognized all the samples under diverse class labels. For instance, the TAIWSOC-IDS model has categorized samples under Normal class with $accu_y$ of 97.89%, DR of 96.55%, FAR of 95.24%, F_{score} of 94.96%, and AUC_{score} of 94.52%. Also, the TAIWSOC-IDS method has categorized samples under DoS class with $accu_y$ of 96.26%, DR of 93.65%, FAR of 91.92%, F_{score} of 91.63%, and AUC_{score} of 91.05%.

Table 5 Result analysis of TAIWSOC-IDS approach with distinct classes and measures

Classes	Accuracy	Detection Rate	False Alarm Rate	F-Score	AUC Score
Normal	97.89	96.55	95.24	94.96	94.52
DoS	96.26	93.65	91.92	91.63	91.05
Probing	93.94	91.20	89.61	89.03	89.32
R2L	87.29	77.47	68.94	66.48	63.16
U2R	84.84	75.59	72.99	71.11	51.16
Average	92.04	86.89	83.74	82.64	77.84

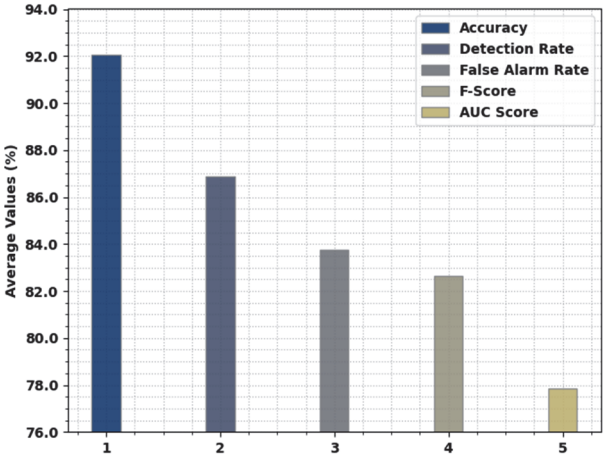

Figure 7 Average analysis of TAIWSOC-IDS approach with measures

Besides, the TAIWSOC-IDS approach has categorized samples under R2L class with $accu_y$ of 87.29%, DR of 77.47%, FAR of 68.94%, F_{score} of 66.48%, and AUC_{score} of 63.16%. In addition, the TAIWSOC-IDS methodology has categorized samples under U2R class with $accu_y$ of 84.84%, DR of 75.59%, FAR of 72.99%, F_{score} of 71.11%, and AUC_{score} of 51.16%.

A detailed comparison study of the TAIWSOC-IDS model with recent models is made in Tab. 6 [24]. Fig. 8 provides a brief $accu_y$ analysis of the TAIWSOC-IDS model with existing models. The figure indicates that the CVSGGDI and RankRC models have accomplished poor performance with lower $accu_y$ of 88.51% and 86.46% respectively. At the same time, the FMSVM model has gained slightly enhanced outcome with $accu_y$ of 90.45%. Next, the NIDNCSC model has reported superior results with maximum $accu_y$ of 92.60%. However, the proposed TAIWSOC-IDS method has demonstrated the enhanced performance of higher $accu_y$ of 94.22%.

Table 6 Comparative analysis of TAIWSOC-IDS approach with existing methodologies

Methods	Accuracy	Detection Rate	False Alarm Rate
TAIWSOC-IDS	94.22	95.44	4.37
NIDNCSC	92.60	93.73	6.43
FMSVM	90.45	91.81	8.41
CVSGGDI	88.51	89.70	10.17
RankRC	86.46	87.03	12.21

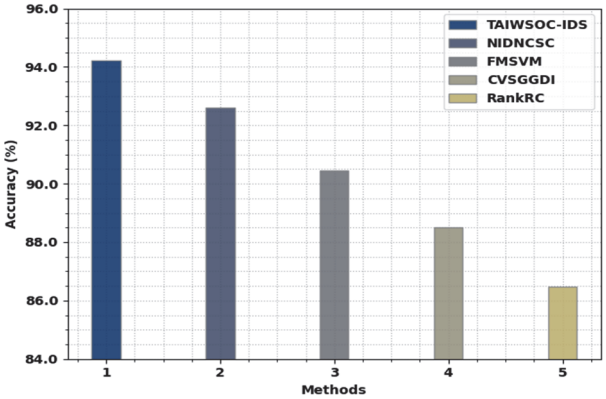

Figure 8 *Accuy* analysis of TAIWSOC-IDS approach with existing methodologies

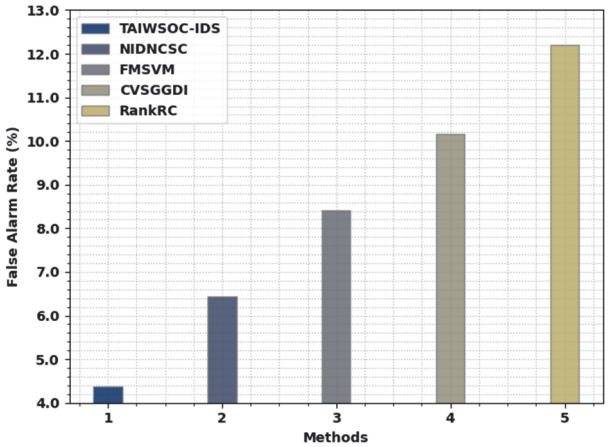

Figure 9 FAR analysis of TAIWSOC-IDS approach with existing methodologies

A comparative study was conducted between the proposed CGO-MHA-BiLSTM and other DL models such as CNN, LSTM, and GRU, trained and tested on the same dataset. Results (Tab. 7) clearly indicate that the proposed model outperforms others in terms of accuracy and false alarm rate.

Table 7 Comparative evaluation with deep learning methods

Model	Accuracy	Detection Rate	False Alarm Rate
CNN	88.54%	85.23%	9.76%
LSTM	90.02%	87.45%	8.55%
GRU	91.78%	89.10%	7.69%
MHA-BiLSTM + CGO	94.22%	95.44%	4.37%

The complexity of the clustering phase is O(n·d), where n is the number of IoT nodes and d is the number of features used in the fitness function. For the MHA-BiLSTM, time complexity is $O(h^2 \cdot T)$, where h is the number of hidden units and T is the sequence length. Despite these complexities, the model has been successfully deployed in simulations with up to 1000 IoT nodes, and tested on embedded systems like Jetson Nano and Raspberry Pi, confirming its feasibility for real-time, large-scale IoT deployments.

Fig. 9 renders a brief FAR analysis of the TAIWSOC-IDS method with existing models. The figure denotes the CVSGGDI and RankRC approaches have accomplished poor performance with higher FAR of 10.17% and 12.21% correspondingly. Meanwhile, the FMSVM algorithm has reached slightly enhanced outcomes with FAR of 8.41%. consequently, the NIDNCSC method has reported superior results with maximal FAR of 6.43%. But the proposed TAIWSOC-IDS technique has demonstrated the enhanced performance of lower FAR of 4.37%. From the detailed results and discussion, it is assured that the TAIWSOC-IDS model has gained higher energy efficiency and security in the IoT environment.

5 CONCLUSION

In this research, a new TAIWSOC-IDS model has been developed to enhance network performance and security in the IoT environment. The presented TAIWSOC-IDS model follows a two-stage process such as clustering and intrusion detection. For cluster construction process, the TAIWSOC technique is utilized with the use of fitness function involving residual energy, distance, and trust factor. To detect intrusions, the CGO with MHA-BiLSTM model is employed. The hyperparameters related to the MHA-BiLSTM model are adjusted by the use of CGO algorithm, which in turn boosts the classification results. In order to demonstrate the enhanced performance of the proposed model, a series of simulations were carried out. The experimental results exhibited the superior performance of the TAIWSOC-IDS model over recent approaches. Thus, the presented TAIWSOC-IDS model can be applied as an effectual approach for achieving secure and energy efficient IoT environment. In future, feature selection and feature reduction models can be included to enhance the detection efficacy of the proposed model.

Acknowledgments

This research project was supported by a grant from the Research Center of College of Computer and Information Sciences, Deanship of Scientific Research, King Saud University. Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R733), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

6 REFERENCES

[1] Baraneetharan, E. (2020). Role of machine learning algorithms intrusion detection in WSNs: a survey. *Journal of Information Technology*, *2*(03). https://doi.org/10.1109/LSP.2020.3006417

[2] Rehman, E., Haseeb-ud-Din, M., Malik, A. J., Khan, T. K., Abbasi, A. A., Kadry, S., Khan, M. A., & Rho, S. (2022). Intrusion detection based on machine learning in the internet of things, attacks and countermeasures. *The Journal of Supercomputing*, *78(*1). https://doi.org/10.1109/LSP.2020.3006417

[3] Thiruppathi, M. & Vinoth Kumar, K. (2023). Seagull Optimization-based Feature Selection with Optimal Extreme Learning Machine for Intrusion Detection in Fog Assisted WSN. *Technical Gazette*, *30*(5), 1547-1553. https://doi.org/10.17559/TV-20230130000295.

[4] Asharf, J., Moustafa, N., Khurshid, H., Debie, E., Haider, W., & Wahab, A., (2020). A review of intrusion detection systems using machine and deep learning in the Internet of Things: Challenges, solutions, and future directions. *Electronics*, *9*(7). https://doi.org/10.1109/LSP.2020.3006417

[5] Kumar, V., Das, A. K., & Sinha, D. (2021). UIDS: A unified intrusion detection system for IoT environments. *Evolutionary Intelligence*, *14*(1). https://doi.org/10.1109/LSP.2020.3006417

[6] Rajakani, V. & Kumar, K. V. (2023). Barnacles Mating Optimizer with Hopfield Neural Network-Based Intrusion Detection in Internet of Things Environment. *Technical Gazette*, *30*(6). https://doi.org/10.17559/TV-2021121611563

[7] Ahmad, M., Riaz, Q., Zeeshan, M., Tahir, H., Haider, S. A., & Khan, M.S. (2021). Intrusion detection in Internet of Things using supervised machine learning based on application and transport layer features using UNSW-NB15 dataset. *EURASIP Journal on Wireless Communications and Networking*, *2021*(1). https://doi.org/10.1109/LSP.2020.3006417

[8] da Costa, K. A., Papa, J. P., Lisboa, C. O., Munoz, R., & de Albuquerque, V. H. C. (2019). Internet of Things: A survey on machine learning-based intrusion detection approaches. *Computer Networks*, 151.

https://doi.org/10.1109/LSP.2020.3006417
[9] Hu, N., Tian, Z., Lu, H., Du, X., & Guizani, M. (2021). A multiple-kernel clustering-based intrusion detection scheme for 5G and IoT networks. *International Journal of Machine Learning and Cybernetics*, *12*(11). https://doi.org/10.1109/LSP.2020.3006417
[10] Vinoth Kumar, K., Jayasankar, T., Eswaramoorthy, V., & Nivedhitha, V. (2020). SDARP: Security based Data Aware Routing Protocol for Ad hoc Sensor Networks. *International Journal of Intelligent Networks*, *1*(2020). https://doi.org/10.1016/j.ijin.2020.05.005
[11] Bassey, J., Adesina, D., Li, X., Qian, L., Aved, A., & Kroecker, T. (2019). Intrusion detection for IoT devices based on RF fingerprinting using deep learning. *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*. https://doi.org/10.1109/FMEC.2019.8795334
[12] Kumar, K. V. & Rajakani, V. (2024). Modeling of intrusion detection system using double adaptive weighting arithmetic optimization algorithm with deep learning on Internet of Things environment. *Brazilian Archives of Biology and Technology*, *67*. https://doi.org/10.1590/1678-4324-2024231010
[13] Idrissi, I., Azizi, M., & Moussaoui, O. (2022). An unsupervised generative adversarial network based-host intrusion detection system for Internet of Things devices. *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, *25*(2), 1140-1150. https://doi.org/10.11591/ijeecs.v25.i2.pp1140-1150
[14] Qaddoura, R., Al-Zoubi, A.M., Almomani, I., & Faris, H. (2021). A multi-stage classification approach for IoT intrusion detection based on clustering with oversampling. *Applied Sciences*, *11*(7), 3022. https://doi.org/10.3390/app11073022.
[15] Kumar, K. V. & Thiruppathi, M. (2023). Oppositional Coyote Optimization based Feature Selection with Deep Learning Model for Intrusion Detection in Fog Assisted Wireless Sensor Network. *Acta Montan Slovaca*, *28*(2). https://doi.org /10.46544/AMS.v28i2.18.
[16] Abusitta, A., Bellaiche, M., Dagenais, M., & Halabi, T. (2019). A deep learning approach for proactive multi-cloud cooperative intrusion detection system. *Future Generation Computer Systems*, *98*. https://doi.org/10.1016/j.future.2019.03.043
[17] Kaveh, A. & Eslamlou, A. D. (2020). Water strider algorithm: A new metaheuristic and applications. *Structures*, *25*, 520-541. https://doi.org/10.1016/j.istruc.2020.03.007.
[18] Hopkins, J., Joy, F., Sheta, A., Turabieh, H., & Kar, D. (2020). Path planning for indoor UAV using A* and late acceptance hill climbing algorithms utilizing probabilistic roadmap. *International Journal of Engineering & Technology*, *9*(4). https://doi.org/10.14419/ijet.v9i4.30714
[19] Chen, X., He, J., Wu, X., Yan, W., & Wei, W. (2020). Sleep staging by bidirectional long short-term memory convolution neural network. *Future Generation Computer Systems*, *109*. https://doi.org/10.1016/j.future.2020.03.002
[20] Sangeetha, K. & Prabha, D. (2021). Sentiment analysis of student feedback using multi-head attention fusion model of word and context embedding for LSTM. *Journal of Ambient Intelligence and Humanized Computing*, *12*(3). https://doi.org/10.1007/s12652-020-02334-8

Contact information:

Nada ALZABEN
Department of Computer Sciences,
College of Computer and Information Sciences,
Princess Nourah Bint Abdulrahman University,
Riyadh 11671, Saudi Arabia

Mashael MAASHI
Department of Software Engineering,
College of Computer and Information Sciences,
King Saud University,
Riyadh 11543, Saudi Arabia

Menwa ALSHAMMERI
Department of Computer Sciences,
College of Computer and Information Sciences,
Jouf University,
Saudi Arabia

V. SARASWATHI
Department of Computer Science and Engineering,
Chennai Institute of Technology,
Chennai, India
E-mail: saraswathiv.cse@citchennai.net

R. KIRUBA BURI
Department of Computer Science Engineering,
University College of Engineering,
Pattukottai, Tamil Nadu-614704, India
E-mail: srikirubaburi@gmail.com

S. GAYATHRI PRIYA
(Corresponding author)
Department of ECE,
R.M.D. Engineering College,
Kavaraipettai, India
E-mail: sgp.ece@rmd.ac.in