Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.041446
Article

Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning Model for Intrusion Detection System

Mahmoud Ragab[1,2,*], Sultanah M. Alshammari[2,3] and Abdullah S. Al-Malaise Al-Ghamdi[2,4]

[1]Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[2]Center of Excellence in Smart Environment Research, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[3]Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[4]Information Systems Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia

*Corresponding Author: Mahmoud Ragab. Email: mragab@kau.edu.sa
Received: 23 April 2023; Accepted: 12 June 2023; Published: 28 July 2023

Abstract: The Internet of Things (IoT) system has confronted dramatic growth in high dimensionality and data traffic. The system named intrusion detection systems (IDS) is broadly utilized for the enhancement of security posture in an IT infrastructure. An IDS is a practical and suitable method for assuring network security and identifying attacks by protecting it from intrusive hackers. Nowadays, machine learning (ML)-related techniques were used for detecting intrusion in IoTs IDSs. But, the IoT IDS mechanism faces significant challenges because of physical and functional diversity. Such IoT features use every attribute and feature for IDS self-protection unrealistic and difficult. This study develops a Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning (MM-WMVEDL) model for IDS. The proposed MM-WMVEDL technique aims to discriminate distinct kinds of attacks in the IoT environment. To attain this, the presented MM-WMVEDL technique implements min-max normalization to scale the input dataset. For feature selection purposes, the MM-WMVEDL technique exploits the Harris hawk optimization-based elite fractional derivative mutation (HHO-EFDM) technique. In the presented MM-WMVEDL technique, a Bi-directional long short-term memory (BiLSTM), extreme learning machine (ELM) and an ensemble of gated recurrent unit (GRU) models take place. A wide range of simulation analyses was performed on CICIDS-2017 dataset to exhibit the promising performance of the MM-WMVEDL technique. The comparison study pointed out the supremacy of the MM-WMVEDL method over other recent methods with accuracy of 99.67%.

Keywords: Internet of Things; intrusion detection system; machine learning; ensemble deep learning; metaheuristics

1 Introduction

In recent times, the Internet of Things (IoT) has advanced and driven the growth of innovative business technologies via a network of devices and computers that can engage and interact with each other [1]. Since the cybersecurity assaults on IoT schemes increases widely and rapidly, businesses and people encounter more difficulties related to business operations, credibility, and funding. It can able to characterize cloud computing (CC) as a framework where various resources and services are obtainable to customers on demand, with any participation either from the customer or the service provider [2]. Many IoT applications in distinct domains rely on CC to process and store data. Security was a main concern with CC, because of the large volume of data that can be stored there. Cyberattack on the CC platform has augmented for numerous reasons [3], which includes the accessibility and availability of hacking tools, which resulted in the attackers not needing exceptional skills or wide knowledge to execute an attack [4].

In the open literature, network intrusion can occur if an intruder launches many potential assaults by using system susceptibilities to make the system down or gain illegal access to the user's data [5]. Undeniably, they are several assaults can be begun in computer networking namely User to Root (U2R), Remote to Local (R2L), Brute Force, Probing (Probe), Port Scanning, etc. [6]. Network intrusion is the result of hackers, assaulting the network through brute force, guessing weak passwords, or utilizing password-guessing software. By using social engineering methods, hackers can even interact with individuals in e-mails, social networks, and messengers for gaining significant data from the mechanism toward network intrusion [7]. Network intrusion involves unusual traffic and has a feature set that differentiates it from regular traffic.

Current paradigms in unusual traffic are identified by several machine learning (ML) and data mining (DM) techniques that enable an Intrusion Detection System (IDS). Network intrusions have a distressing effect on the network and may disable the whole network. Thus, several efforts were intended at devising intrusion and firewall mechanisms to overcome this security challenge [8]. Many ID techniques try to filter or remove unauthorized network traffic through intrusion pattern recognition. Currently, because of the power of computing types of machinery, a big advance, predominantly in the Artificial Intelligence (AI) area, is happening. Advanced technologies of ML [9], mostly DL, were implemented in the security area, and new outcomes and problems were reported. But, with DL, we can significantly raise the robustness and accuracy in the recognition of attacks along with operating detection mechanisms without demanding deep security expert knowledge [10].

This study develops a Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning (MM-WMVEDL) model for IDS. The proposed MM-WMVEDL technique implements min-max normalization to scale the input dataset. For feature selection purposes, the MM-WMVEDL technique exploits the Harris hawk optimization-based elite fractional derivative mutation (HHO-EFDM) technique. In the presented MM-WMVEDL technique, a Bi-directional long short-term memory (BiLSTM), extreme learning machine (ELM) and an ensemble of gated recurrent unit (GRU) models take place. A sequence of simulation analyses was performed to demonstrate the promising performance of the MM-WMVEDL technique.

The rest of the paper is organized as follows. Section 2 provides the related works and Section 3 offers the proposed model. Then, Section 4 gives the result analysis and Section 5 concludes the paper.

2 Literature Review

Saif et al. [11] presented hybrid intelligent IDS (HIIDS) based on metaheuristic and ML approaches for IoT-related applications like health care. In IoT-related smart health care, the biomedical sensor senses the crucial health variables which can be transferred to the cloud server for analysis and storage purposes. Health records or dataset stored as Electronic Health Record (EHR) is security and privacy sensitive. Malibari et al. [12] presented a new metaheuristic with DL-enabled IDS for a secured smart atmosphere called MDLIDS-SSE methodology. Also, the MDLIDS-SSE method entitles an improved arithmetic optimizer algorithm-oriented feature selection (IAOA-FS) system for choosing the best feature subset. On top of that, the quantum-behaved PSO (QPSO) with deep wavelet NN (DWNN) technique can be used for the classification and detection of intrusions in a secured smart atmosphere.

Kareem et al. [13] introduced an innovative FS technique by pushing the performance of Gorilla Troops Optimizer (GTO) relevant to the algorithm for bird swarms (BSA). This BSA was employed for boosting performance exploitations of GTO in designed GTO-BSA since it has the stronger capability for finding possible regions with the best solutions. Zivkovic et al. [14] devised the ID technique by making use of a hybrid method between DNN and the firefly algorithm. The fundamental firefly algorithm, as a frequently used SI approach, has several known deficiencies, and to solve them, an enhanced firefly algorithm has been modelled and leveraged in this manuscript. Kumar [15] offered an HMOFS-OWKELM model for IDS in a big data environment (the abbreviation for HMOFS-OWKELM is Hybrid Metaheuristic Optimization Related Feature Subset Selection-Optimal Wavelet Kernel ELM (OWKELM) oriented Classifier. Apart from that, the HMOFS involves the hybridization of the hill climbing (HC) feature selection process and moth flame optimization (MFO).

Balogun et al. [16] presented a hybrid metaheuristic structures dimensionality reduction technique for IDS. The author leveraged the metaheuristic Bat algorithm for selecting features. The sixteen attributes are selected by the Bat algorithm. Then, RNS has been utilized for acquiring the residues of the 16 features selected. After that, the PCA has been leveraged to get residues by mining it. In [17], an IDS was presented that utilizes DM and ML concepts for detecting network intrusion paradigms. In this presented technique, an ANN was leveraged as a learning approach in ID. To lessen ID errors, the metaheuristic algorithm with a swarm-related technique was utilized. In this technique, for better and more accurate learning of ANNs, a method named the Grasshopper Optimization Algorithm (GOA) was exploited for reducing the ID error rate.

3 The Proposed Model

In this study, we have introduced a new MM-WMVEDL technique for automated detection and classification of intrusions. The presented MM-WMVEDL algorithm follows a three-stage process: data normalization, HHO-EFDM-based feature selection, and ensemble voting-based classification. Fig. 1 represents the workflow of the MM-WMVEDL system.

3.1 Data Normalization

At the primary level, the presented MM-WMVEDL technique carries out min-max normalization to scale the input dataset. The min-max feature scaling technique is used for rescaling the range of feature or observation value of the dataset within [0, 1]. Eq. (1) demonstrates the min-max formula, X denotes the initial real rate whereas X' indicates the normalized rate. *Xmin* value is transformed

into "0", and *Xmax* value was altered into "1", and every other value is transformed into a decimal between zero and one.

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

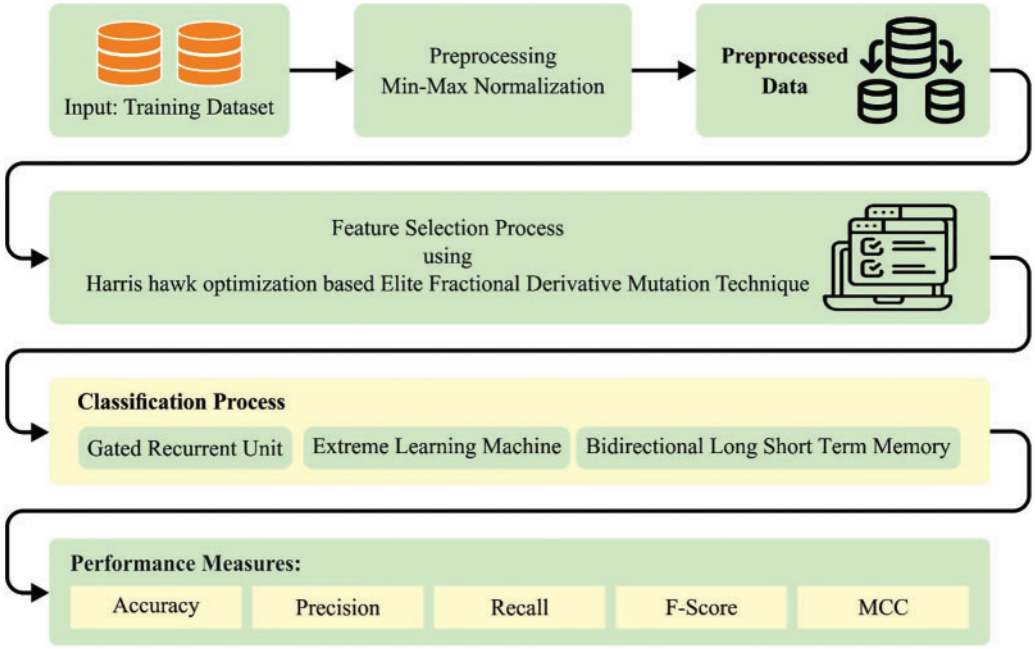

Figure 1: The workflow of the MM-WMVEDL system

3.2 Feature Selection Using HHO-EFDM Technique

For feature selection purposes, the MM-WMVEDL technique employed the HHO-EFDM technique. HHO is a population-based gradient-free optimized technique, motivated by the diverse chasing styles, prey searching capability, and surprise attack of hawk [18]. The proposed HHO technique resolves convergence problems, multiobjective optimization problems, and local optima issues. But to enrich the performance and effectiveness of a model, the EFDM method is developed with HHO thereby improving the exploitation ability. The adoption of these strategies in the HHO-EFDM technique. Similar to other optimization techniques, the HHO technique randomly initializes population member as $X^{(0)} = (P_1^{(0)}, P_2^{(0)}, P_3^{(0)}, \ldots, P_n^{(0)})$ now n portrays the overall hawk population; the population individual is represented as $P_y^{(0)} = \{p_{y1}^{(0)}, p_{y2}^{(0)}, \ldots, ^{by(0)}\}$, where h $H^3$0-LGBM: hybrid HHO based light gradient represent decision variable dimension. Exploitation and exploration are the two important steps in HHO that are mathematically defined in the following.

In HHO, the hawk population was regarded as a solution candidate where they randomly find the prey in the search region. The y^{th} individual in the population is upgraded by using Eq. (2),

$$P_y^{(x+1)} = \begin{cases} P_{\Re}^{(x)} - R_1 |P_{\Re}^{(x)} - 2R_2 P_y^{(x)}|, d_1 \geq 0.5 \\ P_Q^{(x)} - P_s^x - R_3(LB + R_4(UB - LB), d_1 < 0.5) \end{cases} \tag{2}$$

where $P_y^{(x+1)}$ denotes the hawk's location vector for the following iteration, $P_{\Re}^{(x)}$ represents randomly selected Hawks, $P_y^{(x)}$ signifies the location vector of the hawk at the existing iteration, $P_Q^{(x)}$ implies the location of prey and P_s^x represents the mean location of x^{th} generation hawk. As well, R_1, R_2, R_3 and R_4 signifies randomly generated numbers; UB and LB represent the upper and lower boundaries of the decision parameter.

In the HHO algorithm, the hawk gets closer to prey by surprise and pounces, when the prey attempts to escape from a dangerous situation. However, the prey loses its energy while trying to escape from the sight of the hawk. These escape strategies of prey were modelled in two different techniques as hard besiege ($\varepsilon < 0.5$) and soft besiege ($\varepsilon \geq 0.5$) with progressive quick dives. This procedure can be upgraded according to the following equation,

$$P_y^{(x+1)} = \begin{cases} P_Q^{(x)} - P_y^{(x)} - \varepsilon |S_j \cdot P_Q^{(x)} - P_y^{(x)}|, |\varepsilon| \geq 0.5, d_2 \geq 0. \\ P_Q^{(x)} - \varepsilon |S_j \cdot P_Q^{(x)} - P_y^{(x)}|, |\varepsilon| < 0.5, d_2 \geq 0.5 \end{cases} \tag{3}$$

$$P_y^{(x+1)} = \begin{cases} Mf(M) < f(P_y^{(x)}) \\ Nf(N) < f(P_y^{(x)})' d_2 < 0.5 \end{cases} \tag{4}$$

$\varepsilon = 2\varepsilon_0(1 - x/s)$, where ε_0 denotes the uniform distribution random value ranges from $[-1, 1]$, the overall and present iteration is represented by x and S correspondingly, prey's jump strength is represented by $S_j = 2(1 - R_5)$ and randomly generated number R_5 and d_2 ranges within $[0, 1]$.

The optimum solution relies heavily on the exploitation ability of the swarming intelligence. To effectively improve exploitation ability, the features of fractional order derivatives like heritability, memory and storage are exploited and thus prevent premature convergence problems. According to the fitness value, initially, n population individuals are organized as excellent to poor; then initial E^{th} individuals are selected as elite set $P_{ELITE}^{(x)} = \{P_{(1)}^{(x)}, P_{(2)}^{(x)}, \ldots, P_{(E)}^{(x)}\}$, here $E = \left[\frac{n}{2} - \frac{n-2}{2} \cdot \frac{x}{n}\right]$ where they lower from $\frac{n}{2}$ to land signify integer function. With the implementation of fractional derivative mutation for E elite, the exploitation ability of the HHO technique would be increased. To construct proper elite mutation, $1D\ \alpha$ order GL fractional derivative for function $g(y)$ is given as follows,

$$G_\vartheta^\alpha g(y) = e^{-q\vartheta\alpha} \lim_{|k| \to 0} \frac{\sum_{t=0}^{+\infty} (-1)^t \binom{c}{t} g(y - tk)}{|k|^\alpha} \tag{5}$$

$$\binom{c}{t} = \frac{\alpha(\alpha - 1) \ldots (\alpha - t + 1)}{t!} \tag{6}$$

where $\theta \in (-\pi, \pi]$ and $\binom{}{it}$ represents a binomial coefficient. If $\vartheta = \pi$ it is subjected to 2truncation of order δ in terms of left and right derivatives with limit k as k_1 and k_2 is attained by,

$$LG_g^\alpha g(y) = \frac{1}{k_1^\alpha} \sum\nolimits_{t=0}^{\delta} (-1)^t \binom{c}{t} g(y - tk_1) \tag{7}$$

$$RG_g^\alpha g(y) = \frac{1}{k_2^\alpha} \sum\nolimits_{t=0}^{\delta} (-1)^t \binom{c}{t} g(y - tk_2) \tag{8}$$

The step size of mutation ($z_{(p,q)}^{(x)}$) will be decreased with an increasing amount of iterations and obtains an optimum solution. Based on this, the mutation step sizes k_1 and k_2 for p^{th} individual and q^{th} dimension are expressed as,

$$k_1 = \frac{UB_{(q)}^{(x)} - z_{(p,q)}^{(x)}}{\delta + 1} \cdot \left(1 - \frac{x}{S}\right) \tag{9}$$

$$k_2 = \frac{z_{(p,q)}^{(x)} - LB_{(q)}^{(x)}}{\delta + 1} \cdot \left(1 - \frac{x}{S}\right) \tag{10}$$

Furthermore, a greedy selection strategy, an algorithmic model was used to better conserve the optimum individual from the actual elite individual.

In this work, the fitness function is proposed to maintain a balance between several features selected in every solution (lowest) and the classifier accuracy (highest) attained through the features selected, Eq. (11) characterizes the fitness function to assess the solution.

$$Fitness = \alpha \gamma_R (D) + \beta \frac{|R|}{|C|} \tag{11}$$

where α and β parameters are matching to the importance of classifier quality and subset length. $\in [1, 0]$ and $\beta = 1 - \alpha$. $\gamma_R(D)$ signifies the classification error rate. $|R|$ denotes the cardinality of the selected subset and $|C|$ shows the overall amount of features in the given data.

3.3 Ensemble Learning Based Intrusion Detection

In the presented MM-WMVEDL technique, an ensemble of DL models namely GRU, ELM, and BiLSTM models take place. The WMVEDL makes use of the confidence preservation model for increasing the performance of categorizing intrusion [19]. The class prediction of the WMVEDL respective to the classification of intrusion is calculated by Algorithm 1. Hard and Soft voting systems are agglutinated owing to the number of base learners used, for resolving the probability of an even number of predictive output Ξ_j. The average weighted confidence probability $\mu_{\overline{W}}$ of every Ξ_j is represented as follows.

$$\mu_{\overline{\omega_j}} = \frac{1}{n} \sum_{j}^{n} \Xi_j \tag{12}$$

Algorithm 1: Pseudocode of WMVEDL Model

Given input data i
Transmit i to the corresponding handler $(b_1, b_2, \ldots b_n)$ of learners.
Calculate the prediction for every handler through distributed processing.
Accumulate response from every handler of the learner.
Calculate $\overline{\omega_j}$ for each Ξ_j
Aggregate the grades of the handler with $\overline{\omega_j} >= 0.25$.
If accurately one class K_j has the maximum projected output Ξ_j
$P\left(K_j\right) = K_j$
Else
$P\left(K_j\right) = K_j$ with the maximal average weighted confidence $\mu_{\overline{\omega_j}}$
End

3.3.1 GRU Model

The GRU is a subdivision of RNN. The GRU considerably differ from LSTM in that lack cell states and rather applies simple logic circuit comprising a reset gate (R_t) and update gate (Z) [20]. They simplify the training model since they were more literal descriptions of LSTM. At the same time, the GRU was provided by the H_t hidden state that crosses the top of the cell and experiences periodic updates with the gating model. The GRU needs 2 inputs, the pre-hidden state H_{t-1} and the current input X_t. Both states are examined by two gates that evaluate whether the data assist in modifying the hidden state or not. The former is a reset gate that decides what percentage of the preceding hidden state must be retained, thus reducing the amount of data that is stored progressively. Initially, the existing input X_t and the preceding entry H_{t-1} are processed by the non-linear sigmoid function that generates a value within [0, 1], as demonstrated in Eq. (13).

$$R_t = \sigma\left(W_{RH} * H_{t-1} + W_{RX} * X_t\right) \tag{13}$$

Next is the update gate; pre-hidden state H_{t-1} and the inputs X_t are multiplied using the W_{ZX} and W_{ZH}, weights correspondingly. Both products were included, and later the sigmoid activation function was used that clamps resultant within [0, 1], as follows.

$$Z_t = \sigma\left(W_{ZH} * H_{t-1} + Wzx * X_t\right) \tag{14}$$

where Z_t denotes the resultant of the reset gate, W_{RH} and W_{RX} characterize the weight for the reset gate, *and* W_{ZH} and W_{ZX} symbolize the weight of the updating gate of pre-hidden state H_{t-1} and input X_t, correspondingly

Initially, evaluate the product of input X_t and weight $W_{H'X}$ for H'_t. Then find the product of $(H_{t-1} * W_{H'H})$ and reset gate (R), which decides which value needs to be forgotten or remembered and later exploits the non-linear function 'Tanh' by integrating the abovementioned steps, as in Eq. (15).

$$H'_t = \tanh\left\{W_{H'H} * \left(R_t * H_{t-1}\right) + W_{H'X} * X_t\right\} \tag{15}$$

where tanh shows the activation function of output, $W_{H'H}$ and $W_{H'X}$ denote the weight matrix, R_t represents reset gate output, X_t shows input and H_{t-1} indicates the pre-hidden state.

Finally, the update gate decides what must be gathered from H'_t and H_{t-1} present and prior memory content. For the update gate, the initial step was the product of Z_t and H'_t, and the next step was the product of H_{t-1} and $(1 - Z_t)$ are required. By merging those steps, the value of H_t can be defined from the subsequent formula.

$$H'_t = \tanh\left\{W_{H'H} * \left(R_t * H_{t-1}\right) + W_{H'X} * X_t\right\} \tag{16}$$

$$H_t = \left\{\left(1 - Z_t\right) H_{t-1} + Z_T * H'_t\right\} \tag{17}$$

where H_t denotes the outcomes, R_t denotes the reset gate output and H_{t-1} shows the pre-hidden state.

3.3.2 ELM Model

Liu et al. proposed a new type of SLFN model with tremendous performance ELM based on generalized inverse matrix theory [21].

Assume N training instances $\left\{x_i, t_i | x_i \in R^D, T_i \in R^m, i = 1, 2, \ldots, N\right\}$, where $x_i = [\chi_{i1}, \chi_{i2}, \ldots, x_{iD}]^T$ represents the input vector, $t_i = [T_{i1}, T_{i2}, , \tau_{im}]^T$ denotes the corresponding expected output. $g(x)$ shows the activation function viz., a non-linear piecewise continuous function which fulfils the ELM

approximation ability theorem. Gaussian function, Sigmoid function, and so on are the widely applied function:

$$H\beta = T \tag{18}$$

$$\text{where, } H = \begin{bmatrix} h_1(x_1) & \dots & h_L(x_1) \\ \vdots & \ddots & \vdots \\ h_1(x_N) & \dots & h_L(x_N) \end{bmatrix}_{N\times L} = \begin{bmatrix} g(\omega_1 \cdot x_1 + b_1) & \dots & g(\omega_L \cdot x_1 + b_L) \\ \dots & \ddots & \vdots \\ g(\omega_1 \cdot x_N + b_1) & \dots & g(\omega_L \cdot x_N + b_L) \end{bmatrix}_{N\times L}$$

In ELM, H represent the random feature mapping matrix, $\omega_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{iD}]$ denotes the input weight which connects the input layer neuron and the i^{th} hidden layer (HL) neurons, b_i signifies the bias of i^{th} HL neurons, and $\beta = [\beta_1, \dots, \beta_L]^T$ represents the weight matrices between the HLs and the output layers. The HL node parameter (ω_i, b_i) is generated at random and remains the same.

The output weight can be calculated as follows:

$$\beta = H^+ T \tag{19}$$

In Eq. (19), H^+ signifies the Moore–Penrose generalized inverse of the H output matrix.

3.3.3 BiLSTM Model

The Bi-LSTM comprises either forward or backward LSTM layers. The forward layer gets the previous data of orders but the backward layer gets the upcoming data of orders [22]. Fig. 2 showcases the framework of BiLSTM. These 2 layers are connected to the same outcome layer. The network exploits the BiLSTM with a multi-head approach. By linearly offering the context vector to subspace, the multi-head attention layer resolves the secret data. It in turn illustrates the better efficacy, when compared with single-head attention. Then, the outcome was measured by the weighted value and was estimated utilizing the correspondent key and query. The time dimensional computation to attention weighted was illustrated as:

$$s_t = softmax(0_{last} \times (0_{all} \times W_t)^H), 0_{last} \in R^{B,1,Z} \tag{20}$$

$$o_t = s_t \times 0_{all}, 0_{all} \in R^{B,T,Z}, s_t \in R^{B,1,T} \tag{21}$$

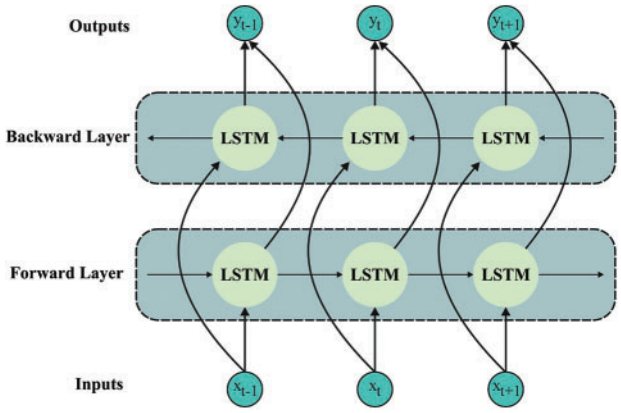

Figure 2: Structure of BiLSTM

At this point, 0_{last} demonstrates the final time outcome, s_t denotes the attention score of time dimensional and 0_{all} signifies the time output. T implies the count of time steps, B suggests the batch

size and Z signify the dimension feature. Variable 1 represents the final time step. H demonstrates the transpose operator and 0_t indicates the results of the time dimensional attention layer. The outcome value of all the times is vital as it could be LSTM resultant database. They stimulate to choose the final time step outcome comprises the redundant data amongst all the time steps.

4 Results and Discussion

The experimental validation of the MM-WMVEDL technique is tested on the CICIDS-2017 dataset. The dataset holds 350000 samples with seven class labels as defined in Table 1. Among the available 77 features, the MM-WMVEDL technique has chosen a set of 33 features.

Table 1: Details of the dataset

Class	No. of instances
BruteForce	50000
DoS	50000
WebAttacks	50000
Infiltration	50000
Bot	50000
DDoS	50000
PortScan	50000
Total no. of instances	350000

The classification performance of the MM-WMVEDL approach is investigated in the form of a confusion matrix in Fig. 3. The outcomes depict that the MM-WMVEDL technique has identified all classes of attacks accurately.

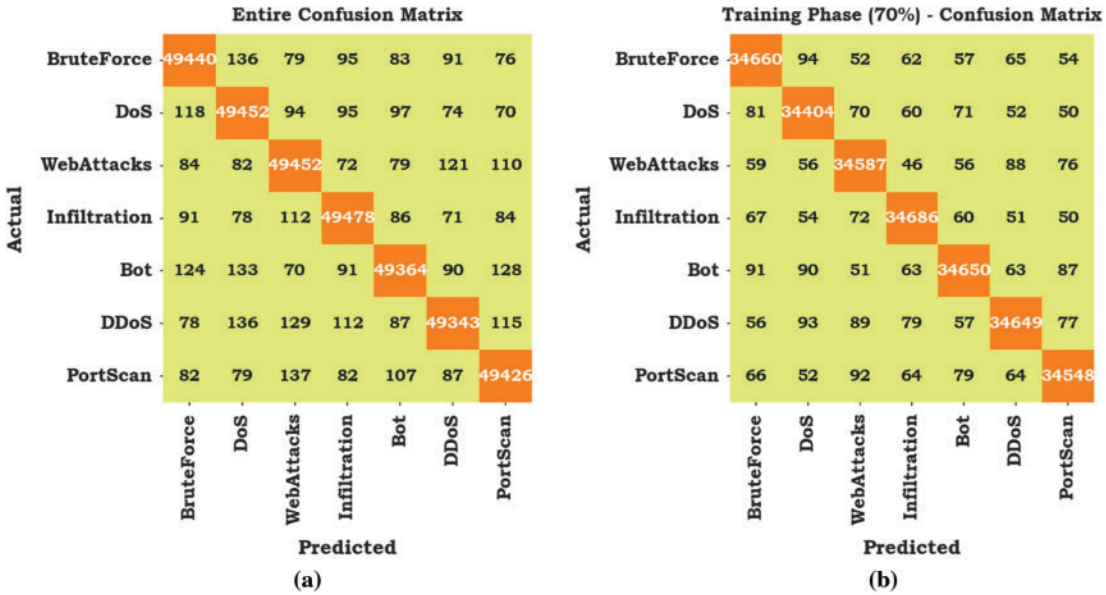

Figure 3: (Continued)

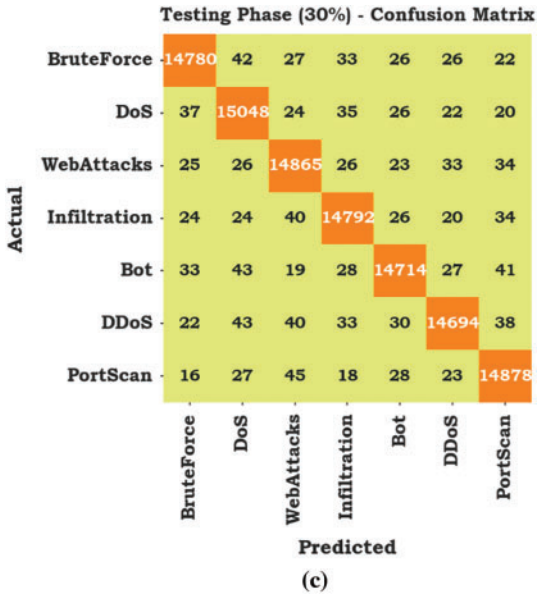

Figure 3: Confusion matrices of MM-WMVEDL approach (a) entire database, (b) 70% of TRS and (c) 30% of TSS

In Table 2 and Fig. 4, the IDS outcomes of the MM-WMVEDL technique on the entire dataset are reported. The experimental outcomes stated that the MM-WMVEDL technique has identified all different kinds of attacks. It is observed that the MM-WMVEDL MM-WMVEDL technique reaches an effectual outcome with an average $accu_y$ of 99.67%, $prec_n$ of 98.84%, $reca_l$ of 98.84%, F_{score} of 98.84%, MCC of 98.65%.

Table 2: IDS outcome of MM-WMVEDL approach on the entire database

Entire dataset					
Class	Accuracy	Precision	Recall	F-score	MCC
BruteForce	99.68	98.85	98.88	98.86	98.67
DoS	99.66	98.71	98.90	98.81	98.61
WebAttacks	99.67	98.76	98.90	98.83	98.64
Infiltration	99.69	98.91	98.96	98.93	98.75
Bot	99.66	98.92	98.73	98.82	98.63
DDoS	99.66	98.93	98.69	98.81	98.61
PortScan	99.67	98.83	98.85	98.84	98.65
Average	99.67	98.84	98.84	98.84	98.65

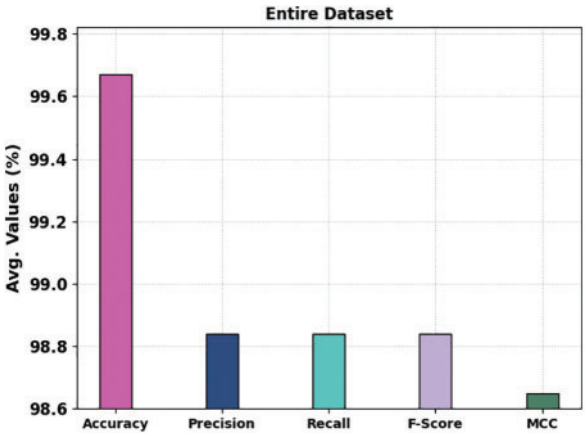

Figure 4: IDS outcome of MM-WMVEDL approach on the entire database

In Table 3 and Fig. 5, the IDS outcomes of the MM-WMVEDL technique on 70% of TRS are reported. The experimental outcomes stated that the MM-WMVEDL technique has identified all different kinds of attacks. It is observed that the MM-WMVEDL technique reaches an effectual outcome with an average $accu_y$ of 99.67%, $prec_n$ of 98.85%, $reca_l$ of 98.85%, F_{score} of 98.85%, and MCC of 98.66%.

In Table 4 and Fig. 6, the IDS outcomes of the MM-WMVEDL method on the entire dataset are reported. The outcomes stated that the MM-WMVEDL system has identified all different kinds of attacks. It is noted that the MM-WMVEDL technique reaches an effectual outcome with an average $accu_y$ of 99.67%, $prec_n$ of 98.83%, $reca_l$ of 98.83%, F_{score} of 98.83%, and MCC of 98.63%.

The TACY and VACY of the MM-WMVEDL method are investigated on IDS performance in Fig. 7. The figure exhibited that the MM-WMVEDL approach has shown improved performance with increased values of TACY and VACY. Visibly, the MM-WMVEDL model has reached maximum TACY outcomes.

Table 3: IDS outcome of MM-WMVEDL approach on 70% of TRS

Training phase (70%)					
Class	Accuracy	Precision	Recall	F-score	MCC
BruteForce	99.67	98.80	98.90	98.85	98.66
DoS	99.66	98.74	98.90	98.82	98.62
WebAttacks	99.67	98.78	98.91	98.85	98.65
Infiltration	99.70	98.93	98.99	98.96	98.79
Bot	99.66	98.92	98.73	98.82	98.63
DDoS	99.66	98.91	98.72	98.81	98.61
PortScan	99.67	98.87	98.81	98.84	98.65
Average	99.67	98.85	98.85	98.85	98.66

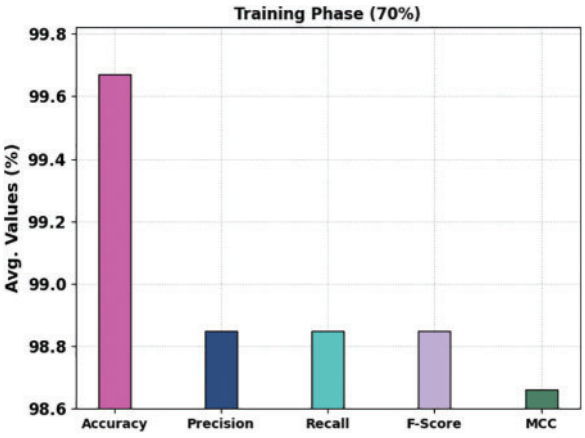

Figure 5: IDS outcome of MM-WMVEDL approach on 70% of TRS

Table 4: IDS outcome of MM-WMVEDL approach on 30% of TSS

Testing phase (30%)					
Class	Accuracy	Precision	Recall	F-score	MCC
BruteForce	99.68	98.95	98.82	98.89	98.70
DoS	99.65	98.66	98.92	98.79	98.58
WebAttacks	99.66	98.71	98.89	98.80	98.60
Infiltration	99.68	98.84	98.88	98.86	98.67
Bot	99.67	98.93	98.72	98.82	98.63
DDoS	99.66	98.98	98.62	98.80	98.60
PortScan	99.67	98.75	98.96	98.85	98.66
Average	99.67	98.83	98.83	98.83	98.63

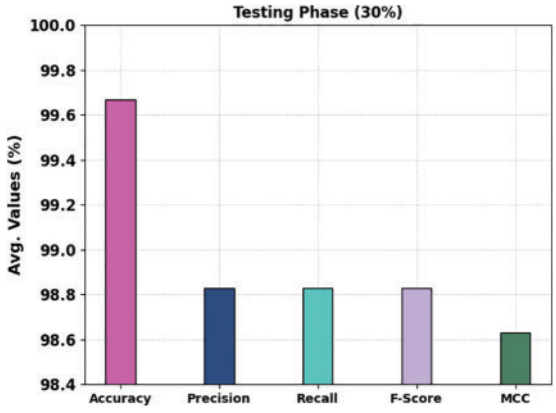

Figure 6: IDS outcome of MM-WMVEDL approach on 30% of TSS

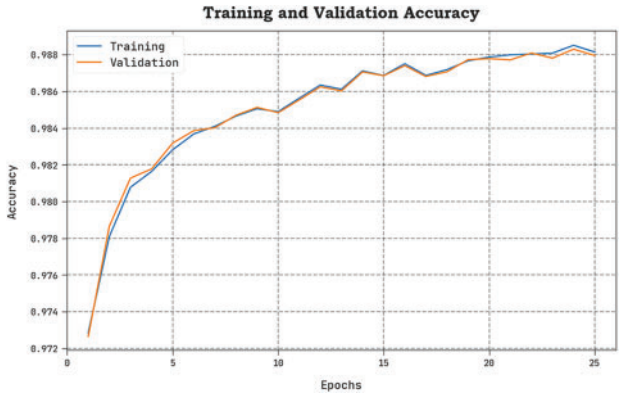

Figure 7: TACY and VACY outcome of MM-WMVEDL approach

The TLOS and VLOS of the MM-WMVEDL methodology are tested on IDS performance in Fig. 8. The figure shows that the MM-WMVEDL algorithm has revealed better performance with the least values of TLOS and VLOS. Particularly, the MM-WMVEDL method has reduced VLOS outcomes.

Figure 8: TLOS and VLOS outcome of MM-WMVEDL approach

Table 5 exhibit the overall classifier outcomes of the MM-WMVEDL technique [23]. The results signify that the CNN model reaches the least outcomes with $accu_y$, $prec_n$, $reca_l$, and F_{score} of 95.41%, 95.47%, 94.12%, and 96.17% respectively. Next, the LSTM, CNN-GRU, CNN-LSTM, and KELM models resulted in moderately closer classification performance with $accu_y$ of 97.89%, 98.82%, 98.71%, and 98.12% respectively. However, the MM-WMVEDL technique reaches maximum performance with an $accu_y$ of 99.67%.

Last, a comprehensive computational time (CT) examination of the MM-WMVEDL technique with recent approaches in terms of Table 6. The experimental values pointed out that the CNN-GRU model and KELM models obtain the least CT of 70.78 and 76.15 s respectively. Next, the LSTM model attains a slightly decreasing CT of 61.74 s. Meanwhile, the CNN-LSTM model results in a reasonable CT of 44.24 s. Although the CNN model reaches a near-optimal CT of 29.85 s, the MM-WMVEDL technique gains a minimal CT of 22.17 s. Therefore, the outcomes ensured the improved performance of the MM-WMVEDL technique over other current techniques.

Table 5: Comparative analysis of MM-WMVEDL system with other systems [23]

Methods	Accuracy	Precision	Recall	F-score
MM-WMVEDL	99.67	98.85	98.85	98.85
CNN–GRU	98.82	98.03	97.46	98.02
CNN-LSTM	98.71	97.09	97.73	97.15
KELM	98.12	97.51	97.31	97.11
CNN	95.41	95.47	94.12	96.17
LSTM	97.89	97.84	96.90	97.51

Table 6: CT analysis of MM-WMVEDL system with other algorithms

Methods	Computational time (sec)
MM-WMVEDL	22.17
CNN–GRU	70.78
CNN-LSTM	44.24
KELM	76.15
CNN	29.85
LSTM	61.74

5 Conclusion

In this study, we have introduced a new MM-WMVEDL technique for automated detection and classification of intrusions. The presented MM-WMVEDL technique follows a three-stage process: data normalization, HHO-EFDM-based feature selection, and ensemble voting-based classification. Primarily, the presented MM-WMVEDL technique implements min-max normalization to scale the input dataset. For feature selection purposes, the MM-WMVEDL technique employed the HHO-EFDM technique. In the presented MM-WMVEDL technique, an ensemble of DL models namely GRU, ELM, and BiLSTM models take place. A comprehensive simulation analysis was made to demonstrate the promising performance of the MM-WMVEDL technique. The comparison study emphasized the supremacy of the MM-WMVEDL approach over other recent methods. In future, the performance of the MM-WMVEDL algorithm can be enhanced by the outlier removal procedure.

Funding Statement: This research work was funded by Institutional Fund Projects under Grant No. (IFPIP: 667-612-1443). Therefore, the authors gratefully acknowledge the technical and financial support provided by the Ministry of Education and Deanship of Scientific Research (DSR), King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

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

References

[1] V. M. Mohana, R. M. Balajee, K. M. Hiren, B. R. Rajakumar and D. Binu, "Hybrid machine learning approach based intrusion detection in cloud: A metaheuristic assisted model," *Multiagent and Grid Systems*, vol. 18, no. 1, pp. 21–43, 2022.

[2] S. E. Quincozes, D. Passos, C. Albuquerque, D. Mossé and L. S. Ochi, "An extended assessment of metaheuristics-based feature selection for intrusion detection in CPS perception layer," *Annals of Telecommunications*, vol. 77, no. 7–8, pp. 457–471, 2022.

[3] M. Tabash, M. A. Allah and B. Tawfik, "Intrusion detection model using naive bayes and deep learning technique," *The International Arab Journal of Information Technology*, vol. 17, no. 2, pp. 215–224, 2020.

[4] R. Salama and M. Ragab, "Blockchain with explainable artificial intelligence driven intrusion detection for clustered IoT driven ubiquitous computing system," *Computer Systems Science and Engineering*, vol. 46, pp. 2917–2932, 2023.

[5] N. A. H. Qaiwmchi, H. Amintoosi and A. Mohajerzadeh, "Intrusion detection system based on gradient corrected online sequential extreme learning machine," *IEEE Access*, vol. 9, pp. 4983–4999, 2021.

[6] J. K. Pandey, S. Kumar, M. Lamin, S. Gupta, R. K. Dubey *et al.,* "A metaheuristic autoencoder deep learning model for intrusion detector system," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–11, 2022.

[7] W. Wiharto, A. K. Wicaksana and D. E. Cahyani, "Modification of a density-based spatial clustering algorithm for applications with noise for data reduction in intrusion detection systems," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 21, no. 2, pp. 189–203, 2021.

[8] S. Bojjagani, B. R. Reddy, M. Sandhya and D. R. Vemula, "CybSecMLC: A comparative analysis on cyber security intrusion detection using machine learning classifiers," in *Machine Learning and Metaheuristics Algorithms, and Applications: Second Symp., SoMMA 2020*, Chennai, India, Springer Singapore, pp. 232–245, 2021.

[9] W. Elmasry, A. Akbulut and A. H. Zaim, "Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic," *Computer Networks*, vol. 168, pp. 107042, 2020.

[10] M. Basheri and M. Ragab, "Quantum cat swarm optimization based clustering with intrusion detection technique for future internet of things environment," *Computer Systems Science and Engineering*, vol. 46, pp. 3783–3798, 2023.

[11] S. Saif, P. Das, S. Biswas, M. Khari and V. Shanmuganathan, "HIIDS: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in IoT based healthcare," *Microprocessors and Microsystems*, 2022. https://doi.org/10.1016/j.micpro.2022.104622

[12] A. A. Malibari, S. S. Alotaibi, R. Alshahrani, S. Dhahbi, R. Alabdan *et al.,* "A novel metaheuristics with deep learning enabled intrusion detection system for secured smart environment," *Sustainable Energy Technologies and Assessments*, vol. 52, pp. 102312, 2022.

[13] S. S. Kareem, R. R. Mostafa, F. A. Hashim and H. M. El-Bakry, "An effective feature selection model using hybrid metaheuristic algorithms for IoT intrusion detection," *Sensors*, vol. 22, no. 4, pp. 1396, 2022.

[14] M. Zivkovic, N. Bacanin, J. Arandjelovic, I. Strumberger and K. Venkatachalam, "Firefly algorithm and deep neural network approach for intrusion detection," in *Applications of Artificial Intelligence and Machine Learning, Proc. of ICAAAIML 2021*, Singapore, Springer Nature, pp. 1–12, 2022.

[15] B. V. Kumar, "Hybrid metaheuristic optimization based feature subset selection with classification model for intrusion detection in big data environment," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 12, pp. 2297–2308, 2021.

[16] B. F. Balogun, K. A. Gbolagade, M. O. Arowolo and Y. K. Saheed, "A hybrid metaheuristic algorithm for features dimensionality reduction in network intrusion detection system," in *Int. Conf. on Computational Science and Its Applications*, Cham, Springer, pp. 101–114, 2021.

[17] S. Moghanian, F. B. Saravi, G. Javidi and E. O. Sheybani, "GOAMLP: Network intrusion detection with multilayer perceptron and grasshopper optimization algorithm," *IEEE Access*, vol. 8, pp. 215202–215213, 2020.

[18] V. Gupta and E. Kumar, "H3O-LGBM: Hybrid Harris hawk optimization based light gradient boosting machine model for real-time trading," *Artificial Intelligence Review*, pp. 1–24, 2023. https://doi.org/10.1007/s10462-022-10323-0

[19] D. A. Okuboyejo and O. O. Olugbara, "Classification of skin lesions using weighted majority voting ensemble deep learning," *Algorithms*, vol. 15, no. 12, pp. 443, 2022.

[20] F. M. Almasoudi, "Grid distribution fault occurrence and remedial measures prediction/forecasting through different deep learning neural networks by using real time data from tabuk city power grid," *Energies*, vol. 16, no. 3, pp. 1026, 2023.

[21] X. Liu, Y. Zhou, W. Meng and Q. Luo, "Functional extreme learning machine for regression and classification," *Mathematical Biosciences and Engineering*, vol. 20, no. 2, pp. 3768–3792, 2023.

[22] H. Tabrizchi, J. Razmara and A. Mosavi, "Thermal prediction for energy management of clouds using a hybrid model based on CNN and stacking multi-layer bi-directional LSTM," *Energy Reports*, vol. 9, pp. 2253–2268, 2023.

[23] A. Henry, S. Gautam, S. Khanna, K. Rabie, T. Shongwe *et al.,* "Composition of hybrid deep learning model and feature optimization for intrusion detection system," *Sensors*, vol. 23, no. 2, pp. 890, 2023.