International Journal of Safety and Security Engineering
Vol. 16, No. 1, January, 2026, pp. 215-224
Journal homepage: http://iieta.org/journals/ijsse

A Hybrid Intrusion Detection System for Secure Vehicular Ad-Hoc Network Communications: Integrating Signature Filtering with a Modified War Optimization Algorithm–Optimized Deep Recurrent Neural Network

Divya Babu*, Terli Sankara Rao

Department of Computer Science and Engineering, GITAM University, Visakhapatnam 530045, India

Corresponding Author Email: dbabu@gitam.in

https://doi.org/10.18280/ijsse.160118

Received: 8 October 2025
Revised: 15 December 2025
Accepted: 21 January 2026
Available online: 31 January 2026

Keywords:
Vehicular Ad-Hoc Networks, Intrusion Detection System, hybrid security model, Deep Recurrent Neural Network, metaheuristic optimization, Modified War Optimization Algorithm, network security

ABSTRACT

Vehicular Ad-Hoc Networks (VANETs) are pivotal to intelligent transportation systems, facilitating real-time data exchange for enhanced safety and traffic efficiency. However, their open architecture and dynamic topology render them vulnerable to diverse cyber threats, including false data injection, Sybil, and Distributed Denial-of-Service (DDoS) attacks. To address the limitations of standalone detection methods, this paper proposes a novel two-stage hybrid Intrusion Detection System (IDS). The first stage employs a lightweight signature-based filter to promptly identify known attack patterns, thereby reducing the computational burden on the subsequent stage. Unmatched traffic is then processed by an anomaly detection module powered by a Deep Recurrent Neural Network (DRNN), which is adept at capturing complex temporal dependencies indicative of novel or evolving threats. Crucially, the hyperparameters of the DRNN are optimized using a Modified War Optimization Algorithm (MWOA) to accelerate convergence and enhance detection robustness. The proposed model is rigorously evaluated on the CIC-IDS2017 dataset and benchmarked against standard Recurrent Neural Network (RNN), One-Dimensional Convolutional Neural Network (1D-CNN), and baseline DRNN models. Performance assessment utilizing accuracy, precision, recall, F1-score, specificity, and Cohen's Kappa demonstrates that our method achieves superior accuracy (97%), significantly lower False Positive Rate (FPR) and False Negative Rate (FNR), and reduced processing latency. These results underscore the efficacy and efficiency of the proposed hybrid IDS, confirming its strong potential for practical, real-time deployment in resource-constrained VANET environments.

1. INTRODUCTION

Vehicular Ad-Hoc Networks (VANETs) have become foundational to intelligent transportation systems by enabling continuous data exchange between vehicles and roadside infrastructure for collision avoidance and emergency dissemination [1]. However, decentralized control, high mobility, and open wireless links make VANETs vulnerable to cyber threats, including false message injection, Sybil identities, and Distributed Denial-of-Service (DDoS) attacks. Traditional Intrusion Detection Systems (IDSs) primarily employ signature-based or anomaly-based strategies [2]. While hybrids offer complementary strengths [3], they often struggle to meet VANETs' latency and accuracy constraints. Recent deep learning models (e.g., Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)) [4-10] capture temporal patterns in traffic flows but depend on careful hyperparameter tuning for reliable performance. To bridge these gaps, we introduced an adaptive Deep Recurrent Neural Network–Modified War Optimization Algorithm (DRNN-MWOA) based hybrid IDS. A lightweight signature filter handles known threats, and an MWOA-tuned DRNN classifies unmatched traffic, improving both detection robustness and computational efficiency [7, 8]. We evaluate the system on CIC-IDS2017 [11-15] and benchmark it against Recurrent Neural Network (RNN), One-Dimensional Convolutional Neural Network (1D-CNN) models, and DRNN, using conventional and extended metrics like Cohen's Kappa.

2. RELATED WORK

In this section, the conventional intrusion detection approach in VANET is reviewed and analyzed. In order to enhance the effectiveness of IDSs, Bangui et al. [16] have introduced a novel machine learning approach, which utilizes posterior identification related to coresets and random forests to improve detection efficiency and accuracy. An IDS was one potential defense against attacks in the VANET. Nevertheless, a major difficulty for IDSs remains handling the massive volume of vehicular information that keeps increasing in the urban environment.

Haydari and Yilmaz [17] have developed new machine learning techniques for targeted attack identification and

extenuation that rely on central communications via roadside units (RSU). Zhang et al. [18] identified forged Basic Safety Messages (BSMs) and Event Report Messages (ERMs) by separating malicious vehicles through clustering techniques.

Thorat et al. [6] have described a unique IDS based on time series classification and deep learning. Traffic characteristics exhibit a strong correlation with time; hence, we use time series feature vectors, obtained from vehicle communications near reported traffic occurrences, to gather traffic parameter time series that are strongly associated with traffic incidents. An LSTM-based traffic event classifier is constructed and trained using time series feature vectors from both normal and collision attack scenarios, which improves the detection accuracy of the pattern of traffic parameters that change over time. Based on the classification result, the emergency message's authenticity can be verified.

Dhar et al. [19] have developed a special IDS for VANETs that focuses more on accurately identifying attack scenarios with few obligatory features by utilizing principal component validation. The second phase of the proposed Cascaded ML architecture involves classifying the assault data, which is initially distinguished from typical scenarios. The idea that an attack shouldn't be conducted in a regular class is emphasized by the framework. The Car Hacking dataset was used to evaluate the suggested framework, which was created using an Artificial Neural Network (ANN). The suggested framework's performance is also contrasted with that of common categorization tasks in the study.

Cui et al. [20] have not only shown federated learning in software-defined VANETs but also developed an accurate and successful model for the Collaborative Intrusion Detection System (CIDS). Without explicitly swapping local network information flows, the model cooperatively trains the CIDS model across local Software-Defined Networks (SDNs), hence expanding the expansibility and globality of IDSs. The Fairness Federated Deep Learning (FFDL) approach uses a two-stage gradient optimization method to enhance fairness and accuracy in the global model, which addresses data heterogeneity while preserving local data privacy. Rashid et al. [21] developed an adaptive, machine learning-based framework for real-time malicious node detection and attack mitigation within VANETs. This approach improves network security by maintaining high detection accuracy while minimizing communication overhead in dynamic environments.

Polat et al. [22] introduced a novel Deep Recurrent Neural Network framework designed to detect DDoS attacks within SDN-based SCADA systems. The approach leverages GRU and LSTM units for improved accuracy in identifying malicious traffic patterns. Due to its shown strengths in pattern identification, deep learning is employed in a diversity of applications, with the detection of anomalous behaviour in computer network activity and software access patterns. The training of optimization-based systems uses machine learning. It enables optimization to improve its accuracy over time automatically, without requiring human input. IDS with Deep learning capabilities provides better detection accuracy. However, it necessitates additional hardware resources and processing power. Due to resource constraints, it is difficult to implement within VANETs.

Though deep learning-based IDSs have proven to be highly accurate in detecting intrusions in VANETs, they have limited execution capability onboard, stringent latency, and time constraints on real-time data processing. Although conventional RNN models are lightweight, they tend to fail to represent long-term temporal dependencies, resulting in a decrease in detection robustness. CNN-based methods, such as 1D-CNNs, are very efficient in feature representation; however, they have higher computational costs because of convolution steps and additional model parameters. Conversely, DRNN models maximize the time modeling, sacrificing complexity and inference time. These trade-offs point to the necessity of optimized deep learning models that trade off detection performance and computation, and the need to balance efficiency and accuracy of these models. The proposed hybrid IDS is driven by these issues and aims to compromise a lightweight signature-based pre-filtering phase with a DRNN optimized by MWOA, thus minimizing unnecessary deep inference and increasing its application in VANET settings.

Contrary to the current hybrid IDS designs, which use parallel execution of signature-based and anomaly-based detectors, thus complicating computational cost and decision conflicts, the designed architecture is strictly sequential and uses two stages. A lightweight signature module first filters known patterns of attacks, and only traffic that matches none of the patterns is sent to the DRNN-based anomaly detector. The design has a major impact on minimizing processing latency and resource consumption, which is a critical constraint in the VANET setting. Moreover, a combination of DRNN parameter adjustment based on MWOA makes the difference between the proposed solution and traditional hybrid IDS models of recovery by means of faster convergence, better generalization, and stronger detection capabilities against unknown attacks.

3. PROPOSED METHOD

The proposed IDS framework (see Figure 1) begins with the collection of raw real-world data (CIC-IDS2017). In the preprocessing stage, redundant flows are removed and essential header features are extracted for analysis. The system then applies a two-stage detection strategy. First, signature matching is performed to quickly identify traffic corresponding to known attack patterns; any matches are immediately flagged as malicious. Unmatched traffic is forwarded to the second stage, where a DRNN optimized with the MWOA [23] learns temporal dependencies and classifies the data. Finally, the system outputs the decision as benign or malicious, with alerts generated for suspicious activity. This layered design ensures both efficiency and adaptability in detecting diverse VANET threats. In this Hybrid method, we can improve the accuracy level of the Deep RNN network with the help of an improved heuristic algorithm, War Optimization Algorithm, a metaheuristic that alternates between attack (exploitation), defense (exploration), and soldier-replacement steps to avoid premature convergence.

The proposed IDS operates in two sequential stages.

The end-to-end workflow is:

1) Preprocess flows (deduplication, header extraction, normalization),

2) Signature matching for known threats,

3) DRNN classification tuned by MWOA,

4) Alert generation and logging.

As opposed to other conventional metaheuristic optimization algorithms like Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), which tend to need a lot

of parameter-tuning and are prone to premature convergence, the MWOA uses adaptive attack, defense, and replacement strategies that actively adjust the balance between the exploration and exploitation of solutions. This is exceptionally beneficial to optimizing DRNN architecture, the loss surface of which is extremely non-convex because of repetitive connectivity and time-delays. The rank/weight-based update scheme in MWOA allows an update step-size to be used in a controlled way, between different iterations, which allows effective exploration of the complex error landscapes and enhances stability of convergence. Subsequently, MWOA offers a stronger optimization framework of RNN-based intrusion detection in VANETs.

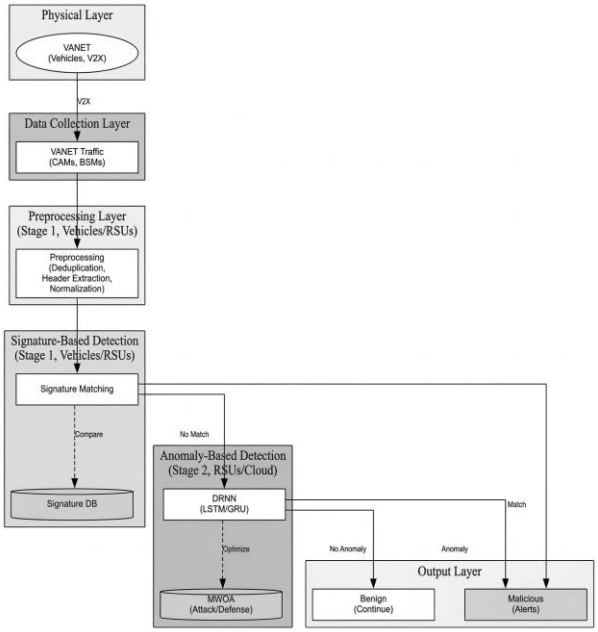

Figure 1. Flowchart of the proposed methodology

3.1 Signature model

A deep learning model was used in the construction of the proposed IDS for VANET. The training and testing phases are the two stages of this deep learning model's progression [16]. The proposed classifier and the signature module are used by the training model to create the learned patterns. The signature module's (training module) learned pattern is used to categorize assaults in the VANET as either unknown or known attacks. The vast amount of VANET data in the VANET architecture reduces the effectiveness of the CIDS. As a result, in the training phase, redundant and duplicate data are removed from network traffic and detected. The signature module receives the pre-processing module's deleted data. The header information is separated from the data using this module to streamline the analysis process, reduce processing overhead, and enhance the accuracy of intrusion detection by focusing on the payload where malicious activities are often encoded. This information is used to test the suggested classifier, which creates the trained patterns in the IDS's training model. The vehicle node is affected by the known attacks in the VANET. The collected database is sent to the classifier for detecting the intruder based on known and unknown attacks.

The signature-based module, which is a component of the proposed two-stage IDS architecture, serves as a first line of network filtering gateway on top of the pre-processed network flows, comparing them to a collection of known attack signatures. Matched signatures of traffic are classified as malicious immediately and logged, whereas the ones that cannot be matched are identified as suspicious and sent to the second-stage anomaly detection module. These filtered flows were transformed into sequential feature vectors and normalized, and given as input to the DRNN classifier. This architecture makes sure that only uncertain instances of traffic reach the DRNN, hence it minimizes the computational burden and allows the DRNN to concentrate on the learning of temporal patterns related to unfamiliar or dynamic attacks.

3.2 Deep Recurrent Neural Network

The RNN is used in this study to detect intrusions in VANET. The processing of data by RNNs is not restricted to one direction. RNNs can cycle through different layers and can also store data temporarily for use at a later time. Figure 2 shows how an RNN is designed. In this design, H_t is the output, X_t is an input.

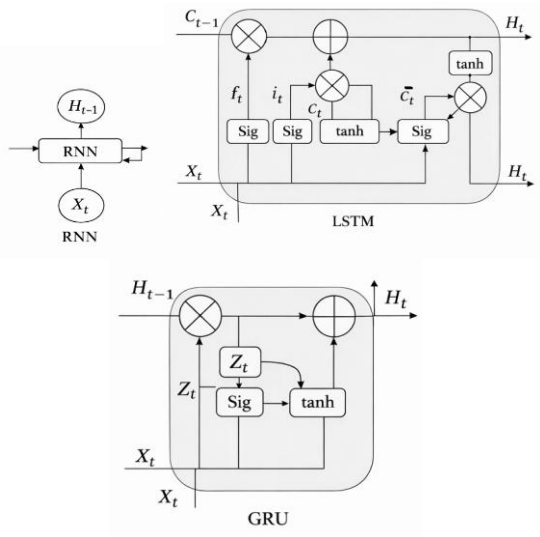

Figure 2. Structural representation of Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) architectures for sequential data processing

The RNN is referred to as a deep neural network and has various layers that are used to analyze input. The Back Propagation Through Time (BPTT) method is used to acquire the RNN model's learning process. Multilayer perceptron (MLP) neural networks frequently employ this strategy. The weight link connecting the present hidden layer and the following hidden layer is denoted as W_{HH}. The connection weight between the input and the first connected hidden layer is defined as W_{Hx}. The weight between the last hidden layer and its output layer is defined as W_{HY}. The biases of the hidden layer and output layer are denoted as B_H and B_Y. H_t is defined as the hidden state at period t. The function G(·) represents a nonlinear activation function, commonly implemented using tanh or ReLU.

$$H_t = G\left(H_{(t-1)}, X_t\right) \tag{1}$$

$$H_t = G(W_{HX}X_t + W_{HH}H_t - 1 + B_H) \tag{2}$$

$$\overline{Y_t} = G(W_{HY}H_t + B_Y) \tag{3}$$

where, $\overline{Y_t}$ is defined as a predicted outcome. The suggested classifier is trained sequentially. The difference between the actual and expected output parameters is calculated at each stage. Error loss (L) function reduction is the main goal of the training process. Low-loss parameters will be obtained by the best model. The mathematical formulation of the training process is as follows:

$$L(\overline{Y_P}, y) = \sum_{t=1}^{n} L(\overline{Y_t} - Y_t) \tag{4}$$

where, n is defined as the maximum learning period. During the learning process, this classifier is known to produce explosive gradients or vanishing gradient problems.

The size of the input dataset is always a factor in this problem.

$$\frac{\partial L}{\partial w} = \sum_{t=1}^{n} \frac{\partial L_t}{\partial w} \tag{5}$$

From the above Eq. (5), the chain rule is applied, and it is reformulated as follows:

$$\frac{\partial L}{\partial w} = \sum_{t=1}^{n} \left(\frac{\partial L_t}{\partial H_t}\right) \sum_{k=1}^{t} \left(\prod_{i=k+1}^{t} \frac{\partial H_i}{\partial H_{i-1}} \right) \partial Hk/\partial w \tag{6}$$

The above Eq. (6) shows that the gradient at a given time depends on the product of partial derivatives across successive hidden states, which explains the vanishing and exploding gradient problems in standard RNNs. The use of GRU and LSTM in the RNN has helped to mitigate the bursting and vanishing gradients problems. The following is a presentation of the LSTM and GRU mathematical formulations. It describes the LSTM architecture, in which gating mechanisms regulate information flow and preserve the long-term dependencies.

$$\begin{cases} f_t = \sigma\left(W_f . [H_{t-1}, X_t] + B_f\right) \\ i_t = \sigma(W_i . [H_{t-1}, X_t] + B_i) \\ C_t^{`} = tanh(W_c . [H_{t-1}, X_t] + B_c) \\ C_t = f_t * C_{t-1} + it * C_t^{`} \\ O_t = \sigma(W_o . [H_{t-1}, X_t] + B_o) \\ H_t = O_t * tanh(C_t) \end{cases} \tag{7}$$

where, B as biases, w as weights, v as output vector, $\sigma_a = \frac{a}{1+e^{-a}}$ as sigmoid, $\tanh(a) = \frac{1-e^{-2a}}{1+e^{-2a}}$ is a hyperbolic tangent activation function and s is defined as the cell state.

$$\begin{cases} Z_t = \sigma(W_z . [H_{t-1}, X_t] + Bz) \\ r_t = \sigma\left(W_r . \left[H_{t-1}, X_t\right] + Br\right) \\ H_t^{`} = tanh(Wh . [r_t * H_{t-1}, X_t] + Bh) \\ H_t = (1 - Z_t) * H_{t-1} + Z_t * H_t^{`} \end{cases} \tag{8}$$

The above Eq. (8) represents the GRU architecture, where update and reset gates control information flow and alleviate vanishing gradient problems.

where, Z_t is defined as the update gate, r_t is defined as the reset gate, X_t is defined as the input vector, and the related weight is denoted by W. The GRU regulates the flow of information using the sigmoid activation function for the update and reset gates. The hyperbolic tangent function is used for computing the candidate hidden state. To improve the performance of the VANET, the optimal weighting parameter is selected by using enhanced WOA.

3.3 Modified War Optimization Algorithm

This optimization is a metaheuristic approach, and it maintains a military to defend itself from attackers (see Figure 3). This army consists of different forces like elephants, chariots, and infantry. In every iteration, whole soldiers have a similar probability of becoming either commanders or kings based on their combat strength [23]. In this strategy, the commander and king are considered the leaders. The changes in the commander and king in the war field will manage the remaining soldiers. There is a requirement for the commander or king to face competition from the soldiers.

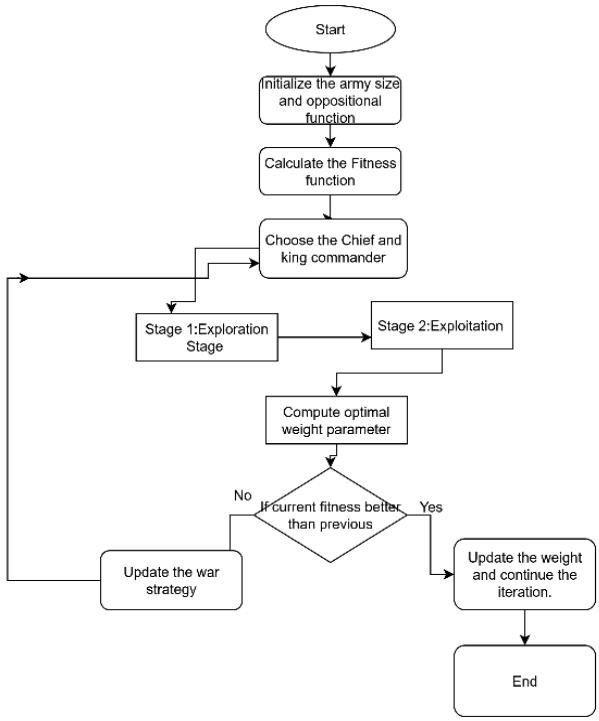

Figure 3. Modified War Optimization Algorithm (MWOA)

The optimization algorithm and the mathematical formulation of the strategy are presented below.

1. Initialization

• Define population size N (number of solutions/warriors).

• Randomly initialize each warrior's position (candidate DRNN parameters).

• Set maximum iterations, T.

2. Fitness Evaluation

For each warrior:

• Train the DRNN with the candidate parameters.

• Compute fitness (e.g., classification error).

3. Attack Phase (Exploitation)

• Select the current best warrior (lowest error).

• Update positions of warriors close to the best solution to exploit promising areas.

4. Defense Phase (Exploration)

• Allow some warriors to move randomly in the search space.

• Ensures diversity and avoids local optima.

5. Replacement Strategy

• Replace the weakest warriors (worst solutions) with new random solutions.

• Maintains population diversity.

6. Update DRNN Parameters

• Assign the best warrior's position as the new set of optimized DRNN parameters.

7. Termination Check

• If maximum iterations T is reached or convergence achieved → stop.

• Else, go back to Step 2.

8. Final Classification

• Use optimized DRNN to classify VANET traffic as Benign or Malicious.

3.3.1 Attack strategy

Two methods are being used in this process. Each soldier in the initial scenario positioned themselves relative to their commander and king. The monarch recommends the best location from which to launch a powerful assault on the opposition. The soldier having the highest attack force or fitness, based on the results, is related to the king. In the early stages of a battle, the weight and rank of all soldiers will be comparable. His rank will rise if the soldier uses the tactic to its full potential. In this algorithm, the oppositional function is also considered for empowering the WOA. Additionally, the update procedure is designed as follows [24]:

$$x_i(t+1) = x_i(t) + 2 \times \rho \times (c - k) + rand \times \left(w_i \times k - x_i(t)\right) \quad (9)$$

where, w_i is a weight, k is a king position, x_i is the last position, $x_i(t+1)$ is a novel position.

3.3.2 Fitness evaluation

The best weighting parameter is chosen in this evaluation process by taking fitness functions into account. This is how it is put together:

$$FE = Min\ (training\ error) \quad (10)$$

Based on this fitness function evaluation, the optimal weight parameter is selected, which increases the prediction accuracy of the classifier.

3.3.3 Weight and rank update

In this updating process, the search agent is related to the position of the king, the commander, and the rank of every soldier. Each soldier's rank is related to their success history in the war, which will similarly affect the weighting factor. The rank of every soldier reflects how close the soldier is to the final target [24]. If the attack force in the novel position is lower than that of the last position, the soldier considers it as the last position.

$$x_i(t+1) = x_i'(t+1)) \times \left(f_n \geq f_p\right) + \left(x_i(t) \times \left(f_n < f_p\right)\right) \quad (11)$$

where, f_n is fitness of novel position, f_p is fitness of previous position. r_i is the rank of ith soldier. Based on the above formulation, the soldier's position is updated optimally. After that, the rank r_i of the soldier will be upgraded.

$$r_i(t+1) = \left(r_{i(t)} + 1\right)) \times \left(f_n \geq f_p\right) + \left(r_{i(t)} \times \left(f_n < f_p\right)\right) \quad (12)$$

w_i is the weight of i[th] soldier. Let maximum iteration is T. α is a control parameter. The new weight of soldiers is computed by:

$$w_i = w_i \times \left(1 - \frac{r_i}{T}\right)^{\alpha} \quad (13)$$

3.3.4 Defense strategy

The position of the king, the army commander, and a random soldier are all factors in the following process for updating positions [24]. ρ is the control parameter. k & c are king and commander position respectively. The mechanism for updating the rankings and weights is similar to the first method.

$$x_i'(t+1) = x_i(t) + 2 \times \rho \times \left(k - x_{rand}(t)\right) + rand \times w_i \times \left(c - x_i(t)\right) \quad (14)$$

3.3.5 Replacement or transfer of weak soldiers

Find the weak warriors in each iteration and judge them to be the least fit. Here, we verified several replacement methods. The straightforward method of substituting a random soldier for a weak soldier is written as follows:

$$x_w(t+1) = LB + rand \times (UB - LB) \quad (15)$$

The next tactic is to replace the weak soldier in the middle of an entire army in a combat zone, as shown in the Eq. (16) below. The algorithm's convergence properties are improved by this method.

$$x_w(t+1) = -(1 - rand\ n) \times \left(x_w(t) - median\ (X)\right) + k \quad (16)$$

Soldiers with higher fitness levels are assigned lower weights, and a larger weight if their fitness level is poor. Each soldier takes high steps during the early stages of the conflict, which causes weight shifts. When the fight is over, the soldiers consider taking small steps to accomplish the goal and adjusting the weight gradually. The soldiers move in random directions and do not strictly follow the king, as this strategy is randomly selected to enhance the algorithm's exploratory capability. At the end of the war, the army personnel are able to identify the final optimal location. This approach effectively balances exploration and exploitation, leading to improved optimization performance.

4. OUTCOME EVALUATION

In this section, the proposed methodology is validated and compared with traditional in the implementation process. This proposed methodology is implemented in Python, and the performance measures are validated based on a confusion matrix. To validate the proposed methodology, the standard dataset is considered. The CIC-IDS2017 dataset [25] contains

benign and the most up-to-date common attacks, which resemble true real-world data (PCAPs). It also includes the results of the network traffic analysis using CICFlowMeter with labelled flows based on the time stamp, source and destination IPs, source and destination ports, protocols, and attack (CSV files). Also available is the extracted feature definition. Infiltration attacks, online attacks, DDoS attacks, botnet attacks, Heartbleed attacks, and brute force attacks are all included in this attack database. The suggested approach is also contrasted with well-established methods such as DRNN, RNN, and 1D-CNN.

Though CIC-IDS2017 is a general-purpose network-based intrusion data set, it has found extensive use in VANET security studies because of its real-to-life traffic creation, varied attack instances, and abundant flow-based temporal characteristics. Most VANET communication patterns, including short-lived connections, bursty traffic patterns, and swift movement characteristic alterations, can be estimated by the timestamped bi-directional flow found in CIC-IDS2017. Moreover, DDoS, botnets, and infiltration attacks are examples of attacks in the dataset that are similar to the threats to vehicular communication infrastructures. However, we admit that CIC-IDS2017 is not structured to explicitly capture an example of VANET-specific considerations, including high-speed mobility, continual topology, and DSRC/C-V2X protocol semantics. Thus, the dataset gives a relevant reference point of detection capacity and temporal learning behaviour; however, future research will have to expand validation to VANET-specific data and simulation platforms (e.g., SUMO Veins-based traffic traces) to further gauge the feasibility of real-world implementation.

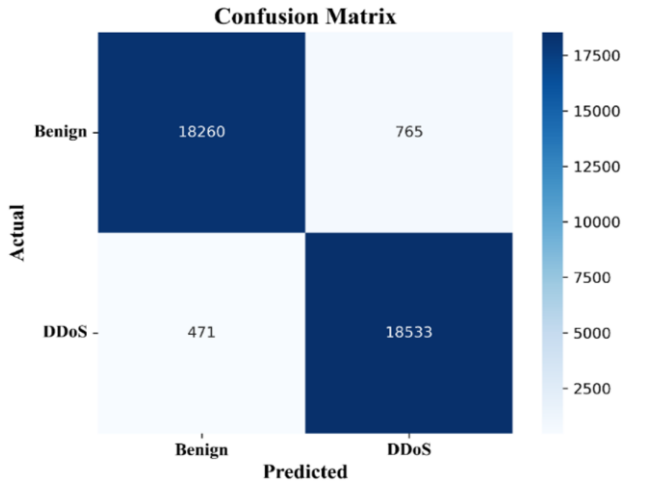

Figure 4. Confusion matrix

The proposed method is analyzed with performance measures:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + TN}$$

where, FN and FP are falsely detected attack-free records and falsely detected attack records, respectively. TP and TN represent correctly detected attack and attack-free records, respectively. Based on the confusion matrix, the normal and abnormal attacks are identified. The confusion matrix is given in Figure 4. It consists of two classes: benign and DDoS attacks.

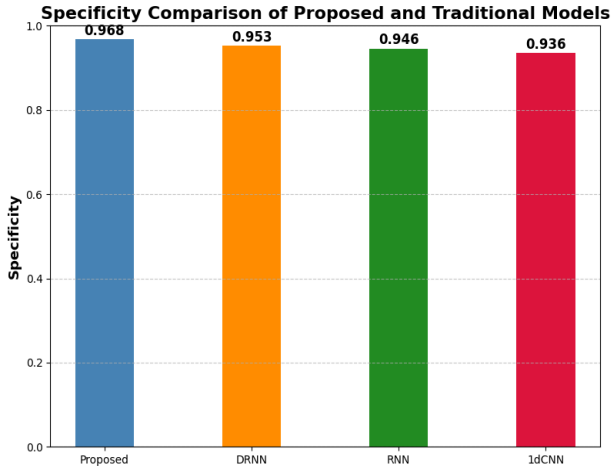

Figure 5. Specificity comparison of proposed and traditional models

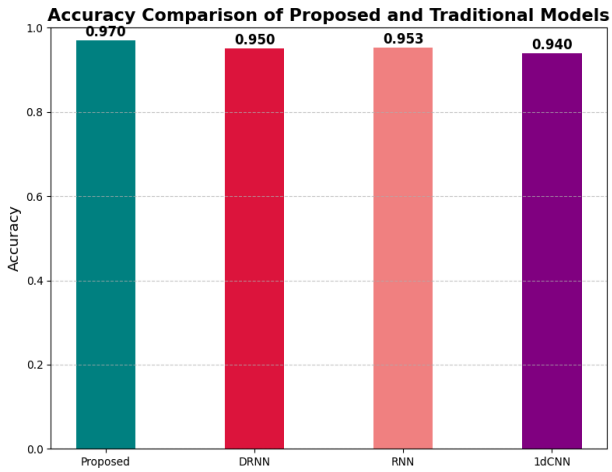

Figure 6. Accuracy of proposed and traditional models

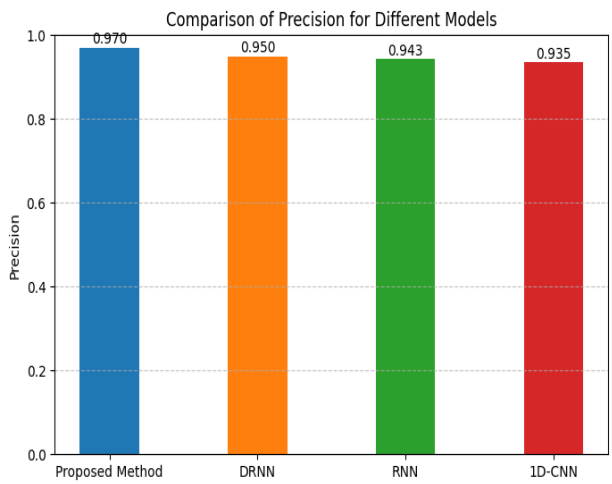

Figure 7. Comparison of precision for different models

The specificity measure is produced and shown in Figure 5 based on the confusion matrix. The proposed approach is contrasted with the traditional DRNN, RNN, and 1D-CNN approaches. The suggested method yielded 0.97 accuracy. Similarly, the DRNN, RNN, and 1D-CNN are achieved at 0.95, 0.953, and 0.94 accuracy measures, respectively. The proposed method achieved a high sensitivity measure during intrusion detection in this validation of comparison analysis. Precision, accuracy, sensitivity, F1-score, and Recall are generated and plotted (see Figures 6-9). The proposed approach is in contrast with the traditional DRNN, RNN, and 1D-CNN approaches. The proposed method achieved a high accuracy and precision measure during intrusion detection in this validation of comparison analysis.

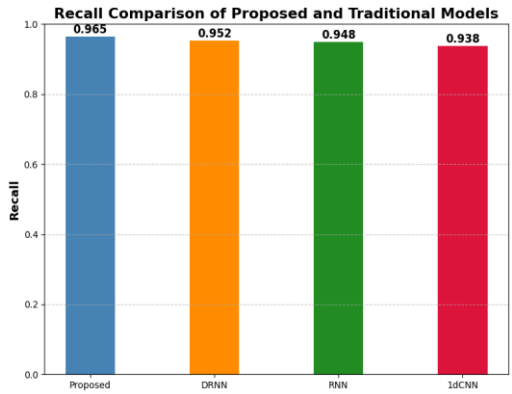

Figure 8. Recall comparison of proposed and traditional models

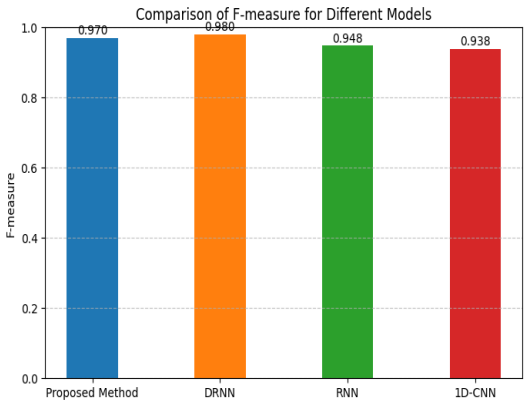

Figure 9. Comparison of F-measure for different models

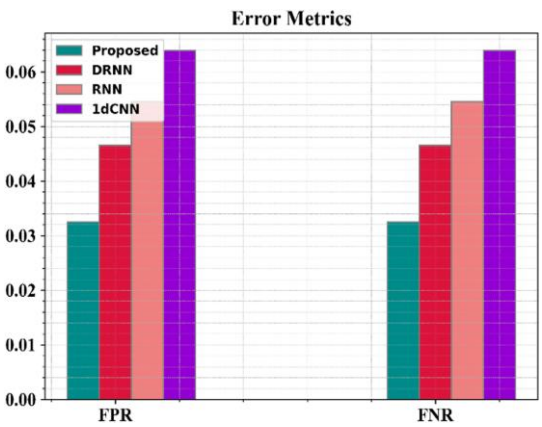

Figure 10. False Positive Rate (FPR) and False Negative Rate (FNR)

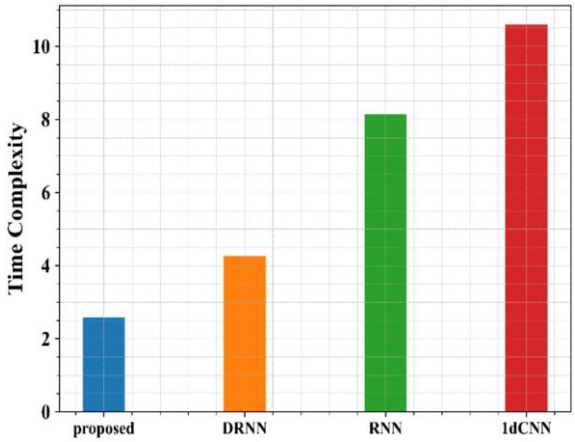

Figure 11. Time complexity

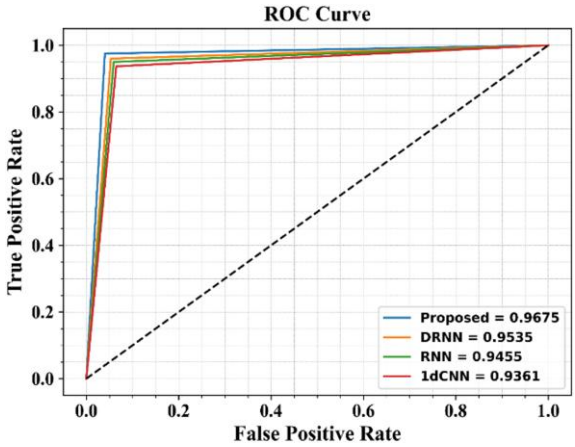

Figure 12. Receiver Operating Characteristic (ROC) curve

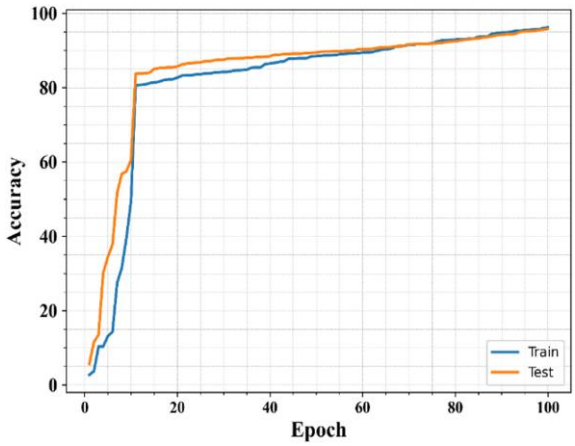

Figure 13. Training accuracy

The error value of the proposed method is analyzed with two measures, False Positive Rate (FPR) and False Negative Rate (FNR). The error value should be low for an optimal intrusion detection model. The FPR is produced and shown in Figure 10 based on the confusion matrix. The proposed approach is contrasted with the traditional DRNN, RNN, and 1D-CNN approaches. The suggested method yielded 0.033 FPR. Similarly, the DRNN, RNN, and 1D-CNN are achieved at 0.047, 0.051, and 0.065 FPR, respectively. The proposed method achieved a low FPR during intrusion detection in this validation of comparison analysis. The FNR is produced and shown in Figure 10 based on the confusion matrix. The

proposed approach is contrasted with the traditional DRNN, RNN, and 1D-CNN approaches. The suggested method yielded 0.032 FNR. Similarly, the DRNN, RNN, and 1D-CNN are achieved at 0.047, 0.055, and 0.064 FNR, respectively. The proposed method achieved a low FNR during intrusion detection in this validation of comparison analysis.

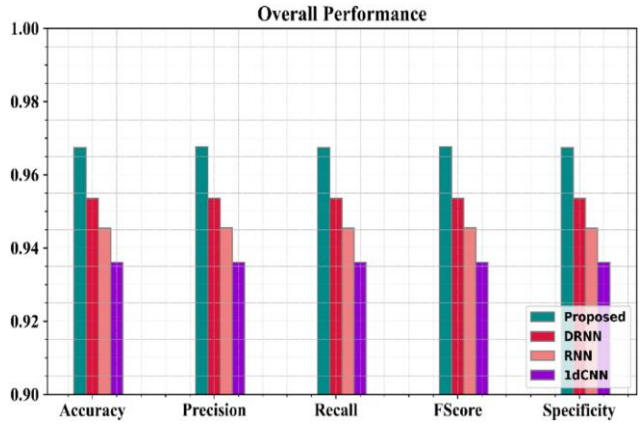

Figure 14. Complete performance

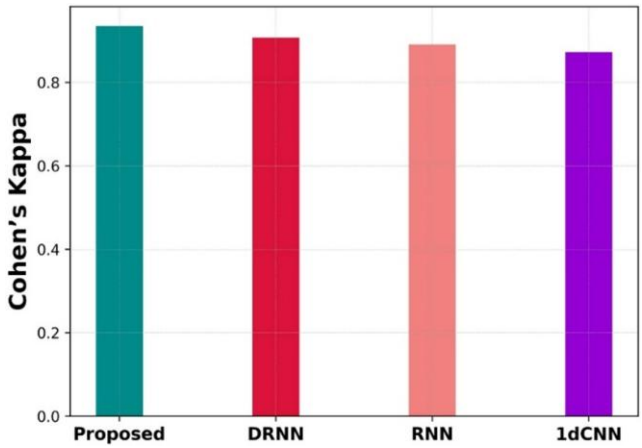

Figure 15. Validation of Cohens' kappa

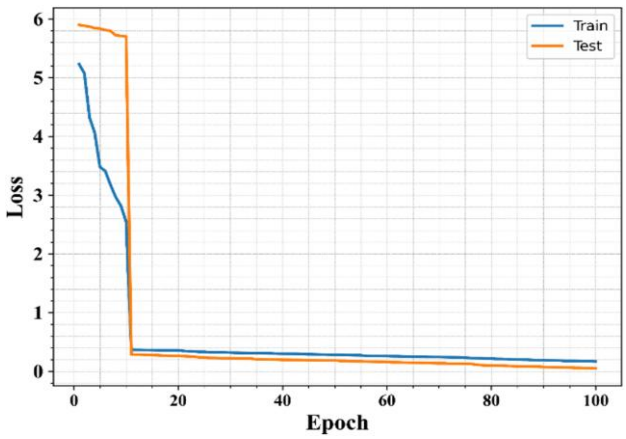

Figure 16. Convergence analysis

Additionally, the time complexity of the proposed approach and conventional methods is presented in Figure 11. The proposed method obtained 3s of processing time; the DRNN, RNN, and 1D-CNN obtained 4s, 8s, and 10.5s, respectively. The proposed method has a low processing time. The Receiver Operating Characteristic (ROC) curve of the proposed technique is presented in Figure 12. In this analysis, the proposed approach obtained 0.9675, and DRNN, RNN, and 1D-CNN obtained 0.9535, 0.9455, and 0.9361. This ROC should be high, so the proposed method has obtained a high value. The convergence analysis of the proposed approach is illustrated in Figure 13. The complete performance of this intrusion detection is presented in Figure 14. In this validation, the proposed approach obtained optimal intrusion detection with a low-loss parameter.

Figure 15 compares the performance of different models for detecting intrusions in VANETs using Cohen's Kappa score, where higher values indicate better agreement between predicted and actual results. The proposed method, which combines an MWOA with a DRNN, achieves the highest score of approximately 0.92, demonstrating superior detection accuracy and reliability. The DRNN model follows with a score of about 0.90, while the standard RNN records 0.88, and the 1D-CNN model achieves 0.86. These results show that the proposed method significantly improves intrusion detection by optimizing model parameters, thereby enhancing the security and trustworthiness of data transmission among vehicles in VANETs compared to other traditional methods.

To further remove any doubts about the effectiveness of the MWOA in enhancing the training of the DRNN, a convergence analysis is provided in Figure 16. The findings suggest that MWOA converges to a lower value of the loss, faster and more stable than the usual optimization methods, proving that it is able to prevent premature convergence and local minima. The adaptive weight and rank updating factor allows the dynamic control of the step-size in training and results in better parameter choices for the DRNN. Subsequently, the optimized DRNN has a larger generalization capacity, which is measured by greater detection accuracy, reduced false rates, and better measures of agreement, including Cohen's Kappa. This discussion has proved that performance improvements can be directly caused by the MWOA-based hyperparameter tuning and not by the complexity of the model itself.

5. CONCLUSIONS

In this study, we developed a finely tuned deep learning model for intrusion detection in VANETs. A DRNN and an MWOA were combined in the suggested strategy. The datasets are gathered from Canadian Institute for Cybersecurity data sources that contain the intrusion data. Data pre-processing is used in the training module to find and eliminate redundant and duplicate data from network traffic. Data from the pre-processing module was transferred to the signature module, which separates the contents from the header data. The proposed IDS's training model for trained patterns is provided by the finely tuned classifier using header data from known malicious attacks and unknown assaults. The MWOA is used to optimize the weight parameter in the DRNN. The suggested solution is put into practice using Python, and it is assessed using performance metrics. It is contrasted with traditional methods as well. The accuracy of the suggested approach was 0.97. In this validation of comparison analysis, the proposed method acquired a high accuracy and a lower time complexity measure during intrusion detection.

In addition to the analogy with generic deep learning networks, such as RNN and 1D-CNN, the provided DRNN-MWOA-based IDS is contrasted with the VANET-specific

intrusion detection techniques. One such example is the LSTM-based temporal IDS models [9], which are superior in the detection accuracy but require a higher inference latency, as they entail complex recurrent structures. The cascaded machine learning model proposed in the study by Dhar et al. [19] is far sparser in terms of features, but it needs centralized processing and little attack generalization. At the expense of increased communication overhead and heightened synchronization complexity, federated and collaborative VANET IDS models [20] improve system scalability and data privacy. In contrast, the proposed method, owing to its sequential hybrid structure and the optimal training of the DRNN, offers a more favorable trade-off among detection accuracy, false rate reduction, and computational efficiency compared to the aforementioned approaches.

This renders the proposed IDS a competitive and real-time VANET security system. In practical deployment terms, the proposed two-stage hybrid IDS is ideal to accommodate the high-speed VANET environment with real-time processing needs. The signature-based lightweight module is capable of being effectively installed on on-board units (OBUs) or RSUs to quickly sieve off known attacks with the minimum of computation. Another, more computationally intensive anomaly detection module obtained through the DRNN-MWOA can be deployed at the RSUs or at the edge/cloud server to assist in scalable traffic analysis on suspicious traffic. This distributed deployment architecture allows real-time intrusion detection as well as resource limitations and scalability of the network. There are, however, other challenges like communication latency between vehicles and infrastructure, model update synchronization, and large-scale deployment that are still open. Overcoming the challenges with the help of edge intelligence and federated learning represents a positive future outlook for work.

In addition to the VANET context, the hybrid intrusion detection model that is proposed based on the DRNN-MWOA can be applied to other dynamic and latency-sensitive networks, such as Internet of Things (IoT) systems and Mobile Ad-Hoc Networks (MANETs). The typical features of these environments are decentralized architecture, temporally varying workload, and resource limitations. An efficient temporal modeling and adaptive optimization are needed. Sequential hybrid detection and MWOA-optimized DRNN training may be generalized to deal with various security threats in these networks, and this is why the proposed solution is a universal tool to handle a broad range of real-time intrusion detection applications.

Limitations: Although the proposed DRNN-MWOA-based hybrid IDS has shown promising performance, it must be noted that there are a number of limitations to this type of IDS. First, the CIC-IDS2017 dataset is used in the experimental assessment; however, this commonly used dataset does not explicitly model the specifics of communication protocols in VANETs, high vehicular mobility, and dynamically varying network topology. Second, the testing environment is a comparatively fixed traffic condition, and it does not reflect the changing topology conditions of real vehicular networks. Third, there is no direct analysis of the energy consumption and computational footprint of the DRNN model when deployed to resource-constrained vehicular devices, including OBUs. The future research will focus on addressing these limitations by validating on VANET specific datasets, mobility-sensitive simulations, and energy-efficient profiling on vehicular platforms.

REFERENCES

[1] Lee, M., Atkison, T. (2021). VANET applications: Past, present, and future. Vehicular Communications, 28: 100310. https://doi.org/10.1016/j.vehcom.2020.100310

[2] Kaur, G., Kakkar, D. (2022). Hybrid optimization enabled trust-based secure routing with deep learning-based attack detection in VANET. Ad Hoc Networks, 136: 102961. https://doi.org/10.1016/j.adhoc.2022.102961

[3] Al-Garadi, M.A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., Guizani, M. (2020). A survey of machine and deep learning methods for Internet of Things (IoT) security. IEEE Communications Surveys & Tutorials, 22(3): 1646-1685. https://doi.org/10.1109/COMST.2020.2988293

[4] Otoum, Y., Liu, D., Nayak, A. (2022). DL-IDS: A deep learning-based intrusion detection framework for securing IoT. Transactions on Emerging Telecommunications Technologies, 33(3): e3803. https://doi.org/10.1002/ett.3803

[5] Gad, A.R., Nashat, A.A., Barkat, T.M. (2021). Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset. IEEE Access, 9: 142206-142217. https://doi.org/10.1109/ACCESS.2021.3120626

[6] Thorat, S., Rojatkar, D., Deshmukh, P. (2025). Comparative analysis of various intrusion detection systems using deep learning for VANET and their issues and challenges with IoT devices integration. Indian Journal of Technical Education, 48(1): 177-185.

[7] Shu, J., Zhou, L., Zhang, W., Du, X., Guizani, M. (2021). Collaborative intrusion detection for VANETs: A deep learning-based distributed SDN approach. IEEE Transactions on Intelligent Transportation Systems, 22(7): 4519-4530. https://doi.org/10.1109/TITS.2020.3027390

[8] Ben Rabah, N., Idoudi, H. (2022). A machine learning framework for intrusion detection in VANET communications. In Emerging Trends in Cybersecurity Applications, pp. 209-227. https://doi.org/10.1007/978-3-031-09640-2_10

[9] Yu, Y., Zeng, X., Xue, X., Ma, J. (2022). LSTM-based intrusion detection system for VANETs: A time series classification approach to false message detection. IEEE Transactions on Intelligent Transportation Systems, 23(12): 23906-23918. https://doi.org/10.1109/TITS.2022.3190432

[10] Alsarhan, A., Al-Ghuwairi, A.R., Almalkawi, I.T., Alauthman, M., Al-Dubai, A. (2021). Machine learning-driven optimization for intrusion detection in smart vehicular networks. Wireless Personal Communications, 117(4): 3129-3152. https://doi.org/10.1007/s11277-020-07797-y

[11] Singh, G., Khare, N. (2022). A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques. International Journal of Computers and Applications, 44(7): 659-669. https://doi.org/10.1080/1206212X.2021.1885150

[12] Maseer, Z.K., Yusof, R., Bahaman, N., Mostafa, S.A., Foozy, C.F.M. (2021). Benchmarking of machine learning for anomaly-based intrusion detection systems in the CICIDS2017 dataset. IEEE Access, 9: 22351-22370. https://doi.org/10.1109/ACCESS.2021.3056614

[13] Schmidt, D.A., Khan, M.S., Bennett, B.T. (2020). Spline-based intrusion detection for VANET utilizing knot flow classification. Internet Technology Letters, 3(3): e155. https://doi.org/10.1002/itl2.155

[14] Raza, A., Bukhari, S.H.R., Aadil, F., Iqbal, Z. (2021). An UAV-assisted VANET architecture for intelligent transportation system in smart cities. International Journal of Distributed Sensor Networks, 17(7): 15501477211031750. https://doi.org/10.1177/15501477211031750

[15] Bangui, H., Buhnova, B. (2021). Recent advances in machine-learning driven intrusion detection in transportation: Survey. Procedia Computer Science, 184: 877-886. https://doi.org/10.1016/j.procs.2021.04.014

[16] Bangui, H., Ge, M., Buhnova, B. (2022). A hybrid machine learning model for intrusion detection in VANET. Computing, 104(3): 503-531. https://doi.org/10.1007/s00607-021-01001-0

[17] Haydari, A., Yilmaz, Y. (2022). RSU-based online intrusion detection and mitigation for VANET. Sensors, 22(19): 7612. https://doi.org/10.3390/s22197612

[18] Zhang, Y.A., Cheong, C., Li, S., Cao, Y., Zhang, X., Liu, D. (2024). False message detection in internet of vehicle through machine learning and vehicle consensus. Information Processing & Management, 61(6): 103827.https://doi.org/10.1016/j.ipm.2024.103827

[19] Dhar, A.C., Roy, A., Akhand, M.A.H., Kamal, M.A.S. (2023). Cascadmlids: A cascaded machine learning framework for intrusion detection system in VANET. Electronics, 12(18): 3779. https://doi.org/10.3390/electronics12183779

[20] Cui, J., Sun, H., Zhong, H., Zhang, J., Wei, L., Bolodurina, I., He, D. (2023). Collaborative intrusion detection system for SDVN: A fairness federated deep learning approach. IEEE Transactions on Parallel and Distributed Systems, 34(9): 2512-2528. https://doi.org/10.1109/TPDS.2023.3290650

[21] Rashid, K., Saeed, Y., Ali, A., Jamil, F., Alkanhel, R., Muthanna, A. (2023). An adaptive real-time malicious node detection framework using machine learning in vehicular ad-hoc networks (VANETs). Sensors, 23(5): 2594. https://doi.org/10.3390/s23052594

[22] Polat, H., Türkoğlu, M., Polat, O., Şengür, A. (2022). A novel approach for accurate detection of the DDoS attacks in SDN-based SCADA systems based on deep recurrent neural networks. Expert Systems with Applications, 197: 116748. https://doi.org/10.1016/j.eswa.2022.116748

[23] Ayyarao, T.S., Ramakrishna, N.S.S., Elavarasan, R.M., Polumahanthi, N., et al. (2022). War strategy optimization algorithm: A new effective metaheuristic algorithm for global optimization. IEEE Access, 10: 25073-25105. https://doi.org/10.1109/ACCESS.2022.3153493

[24] Ayyarao, T.S., Kumar, P.P. (2022). Parameter estimation of solar PV models with a new proposed war strategy optimization algorithm. International Journal of Energy Research, 46(6): 7215-7238. https://doi.org/10.1002/er.7629

[25] Canadian Institute for Cybersecurity. Intrusion detection evaluation dataset (CIC-IDS2017). https://www.unb.ca/cic/datasets/ids-2017.html.