PENINSULA PUBLISHING LLC

SHIFRA

Vol. **(2025)**, 2025, **pp**. 110–117
ISSN: 3078-3186

Research Article

An Integrated Federated Learning Framework with Optimization for Industrial IoT Intrusion Detection

Aitizaz Ali[1, *], Aseel AlShuaibi [2], Muhammad Waqas Arshad[3]

[1] *School of Technology, Asia Pacific University of Technology and Innovations, Kuala Lumpur, Malaysia*

[2] *Department of Computer Networks and Communications, King Faisal University, Al-Ahsa 31982, Saudi Arabia*

[3] *Department of Computer Science and Engineering, University of Bologna, Italy*

ARTICLE INFO

Article History

Received 20 Nov 2024

Revised: 11 Jan 2025
Accepted 12 Feb 2025
Published 1 Mar 2025

Keywords

Industrial Internet of Things (IIoT),

Federated Learning (FL),

Intrusion Detection System (IDS),

Convolutional Neural Networks (CNN),

Differential Privacy.

ABSTRACT

Because of the rapidly growing number of connected devices, the Industrial Internet of Things (IIoT) has presented significant security challenges. IoT networks must be secure and efficient in order to remain sustainable. We present an integration of federated learning and optimization techniques to detect intrusions in IIoT. By combining CNNs with the proposed framework, model accuracy can be enhanced while communication costs are minimized. Data security and user privacy are ensured through the implementation of differential privacy mechanisms. In addition to high rates of accuracy, precision, and recall, the framework has an efficient training management process. With its robustness and scalability, the proposed method offers a better solution for IIoT security concerns than traditional centralized machine learning methods.

1. INTRODUCTION

With the industrial Internet of Things (IIoT) advancing rapidly in recent years, wireless transmission and processing technologies have been playing an increasingly important role. With the development of IoT networks, cutting-edge portable devices such as smartphones, smartwatches, and applications are becoming more common. Smart cities, smart manufacturing, navigation systems, and smart healthcare are among the business sectors that have extensively incorporated these technologies. As IoT networks proliferate rapidly, they still present a number of significant challenges. Managing IoT systems efficiently and flexibly, expanding applications, and accommodating future expansion are some of the challenges. Moreover, IoT networks require constant computing resources, which makes them cognitively demanding and time-efficient, as well as highly secure against unauthorized access. A new generation of users is becoming aware of the importance of protecting their private information as a result of rapid advancements in digital technology [1], [2].

An integrated learning platform that includes local training should be possible with devices that are capable of working together. Machine learning (ML) can be implemented on a decentralized platform called Federated Learning (FL) [3], [4]. Because FL is an end-to-end learning framework, it automatically promotes confidentiality and privacy since the end device does not transmit the data. Participants' data is used to train distributed learning models. Only the settings that have been changed are shared between a client device and a cloud server.

The security of IIoT devices is becoming increasingly important to both consumers and manufacturers[5]. As IIoT devices lack the necessary security mechanisms, they are susceptible to attacks. Among the main reasons for this are their confined environments and low computational power. As a result of their low power consumption, IoT devices are usually limited in terms of the functions they can perform. This ultimately leads to security measures failing. The security risks associated with individuals using default settings, no passwords, or weak passwords are enormous. Most IIoT devices come with

*Corresponding author email: aitizaz.ali@apu.edu.my
DOI: https://doi.org/10.70470/SHIFRA/2025/007

default passwords that are easy to remember or don't require a password at all. This vulnerability can be exploited relatively easily by hackers once they gain access to these devices. This vulnerability may enable attackers to commit serious attacks on IIoT devices, compromising the privacy of users [6]. Efforts are underway to design quick and effective systems that will prevent catastrophic harm. Intruders are constantly monitored for signs of intrusion by intrusion detection systems (IDS) [7].

Using pre-established rules, it decides whether network packets will be allowed or denied based on their metadata. An intrusion can be detected in one of two ways: through detection or deployment. Additionally, each of these categories has two subcategories. Intruder detection systems can either be host-based (HIDS) or network-based (NIDS). A signature-based system detects intrusions, while an anomaly-based system detects them. The first case involves predefined patterns of behaviour. Because of this, signature-based IDS are not capable of detecting unknown threats. Anomaly-based IDS identify potential threats based on specific network traffic characteristics when detecting anomalous behaviours in the system. IIoT networks currently lack a practical IDS that uses ML-based methods to detect irregularities in their systems. Nonetheless, typical centralized systems suffer from single points of failure (SPOF) [8], among other shortcomings. This problem is addressed by the FL protocol, which allows devices to train a machine-learning model without exchanging or receiving data.

Due to the expansion of the IoT, the number of security vulnerabilities and threats associated with IoT devices and systems is increasing significantly. Detecting IoT attacks and cyber threats early is possible through intrusion detection systems (IDSs) [9]. A number of IDS mechanisms based on artificial intelligence (AI) have been incorporated into the market in recent years. These mechanisms study the network traffic of devices to detect anomalous behaviour that could indicate a particular type of attack [10]. In heterogeneous and untrusted IoT devices, AI-based IDS have been trained to detect attacks through the analysis of network traffic and behaviour[11]. As a result, this approach raises privacy concerns since private information can be shared between different domains [12].

2. RELATED WORK

Two recent surveys have been conducted about IoT intrusion detection [13]. This paper presented a taxonomy that organizes IDS based on their specificity to the IoT, provided by the author [14]. They also compared IDS with IoT and looked at installation strategies, detection mechanisms, and validation strategies. According to the author [13], IoT intrusion detection processes should be improved. IoT architecture was the focus of their study. An in-depth and critical analysis was presented. A paper by [3] discussed how machine learning could be used to protect massive IoT devices from various attacks. IoT devices and machine learning were the focus of the author [15]. Data generated by modern communication systems generates massive amounts of information, and machine learning improves performance and resource allocation. Research was conducted to improve the connectivity of wireless devices through distributed learning. Wireless sensor networks (WSNs) are commonly attacked by denial-of-service (DoS) attacks [16], [17]. To address the problem, they also studied ML-based IDS. In addition, most conventional solutions lack ML-based approaches; therefore, ML IDS was recommended for TCP/IP security. The author proposes a system that detects IoT anomalies using dimension reduction and a classifier [18].

Author [19] says that IoT environments can be made safer by alerting users to abnormal events based on footage from connected surveillance systems. In the proposed solution, deep neural networks, multi-classifiers, and kernel density functions are utilized. It uses the Random Forest Differential Evolution with Kernel Density (RFKD) method to detect abnormal behaviour, and when it detects abnormality, it sends an MQTT signal to IoT devices. A multi-anomaly detection system, however, is not addressed in this work. An artificial neural network, a convolutional neural network, and a recurrent neural network were used by the author [20] to detect distributed DoS (DDoS). Two new traffic datasets, CIC-DDoS2019 and TON-IoT, included various DDoS attacks, which were used to evaluate each model. Using two-tier classification and dimension reduction, [21] presents a method for detecting intrusions.

Furthermore, this method is able to identify cyberattacks that originate at the local level and travel up the root level (U2R) and from the root level to the local level (R2L). Minimizing dimensions can be achieved using several methods, including linear discriminant analysis and component analysis. It identifies anomalous behaviour by using two versions of KNN, Naive Bayes and certainty factor. The proposed intrusion detection system consists of two steps [22].

According to [23], detecting network intrusions can be achieved using FL-based methods. As far as privacy protection and network intrusion detection are concerned, it's supposed to resolve those issues. A local version of the GRU deep learning model is run during iterative training. Using federated learning techniques, the parameters of network traffic can be aggregated and averaged locally. Their experiment relied on the widely used lab simulation dataset CICIDS2017. IoT networks are becoming larger and more complex, making distributed learning important [24]. For IoT networks to achieve ML accuracy, global algorithms must be updated regularly, resulting in high data costs. A wireless FL system was investigated as a means of reducing communication costs and improving learning performance at the same time. By decoupling power, bandwidth, and transmission indicators, the Lagrange multiplier method maximizes information flow

through networks. Power and bandwidth are allocated using an algorithm based on linear search. When using local differential privacy (LDP) on native IoT terminals, power IoT faces a major challenge to balance utility and privacy [25].

3. METHODOLOGY

In industrial environments, this deep learning framework is used to detect intrusions by combining multiple data sources and advanced machine learning techniques. It is necessary to preprocess the data after it has been collected in order to ensure accuracy, quality, and accuracy. There are three steps that machine learning models need in order to function: handling missing values, normalizing numerical features, and encoding categorical variables. Preprocessing is essential to ensure data integrity and improve the reliability of subsequently analyzed data. After preprocessing, Convolutional Neural Networks (CNNs) are used to extract features. A CNN's ability to identify complex patterns and features in large datasets makes it an ideal tool for analyzing large datasets [26]. Mechanisms for detecting anomalies are integral to the framework. As part of this step, the extracted features are analyzed to find any patterns that are unusual or unexpected. During the framework's final phase, real-time monitoring and performance assessment are conducted.

3.1 Data Preprocessing

Transforming data values to make the acquisition and processing of information more efficient is part of preprocessing. Since most datasets have large contrasts between maximum and minimum values, normalizing them minimizes the complexity of the algorithm used to process them. Normalizing the data allows neural network algorithms to be classified appropriately. Normalization usually involves converting the current range of data to the norm using a min-max algorithm. Using Equation (1), we can find the normalization formula.

$$p = \frac{((x - x_{min})(Max - Min)}{(x_{max} - x_{\min}) + min} \tag{1}$$

Consider $(min,\ max)$ as representing the specified input variables $(x_{min},\ x_{max})$ as representing the initial input variables, and *P* as representing the converted input variables. Z-scores are used to normalize datasets by standardizing their features. The feature values of a dataset can be normalized using its properties. An equation representing these values in Equation (2), in which a feature's mean (average over all of its values in a dataset) and standard deviation of the average are given.

$$x'^{(j)} = \frac{x^j - \mu^j}{\delta^j} \tag{2}$$

3.2 CNN-based AI Framework

This section describes the CNN-based AI framework proposed by the author. The CNN-based AI framework's specific architecture is shown in Figure 1. As seen in Figure 3, 3D convolutional layers with decreasing kernel sizes are used to extract features. A twofold reduction in feature maps is achieved when MaxPooling is introduced before the Rectified Linear Unit layer (ReLU). In this way, the model becomes less computationally complex. The developed model adds ReLU layers to reduce the effect of gradient vanishing. Dropout layers are added to prevent overfitting. A dropout layer produces output by connecting it to a fully connected layer.

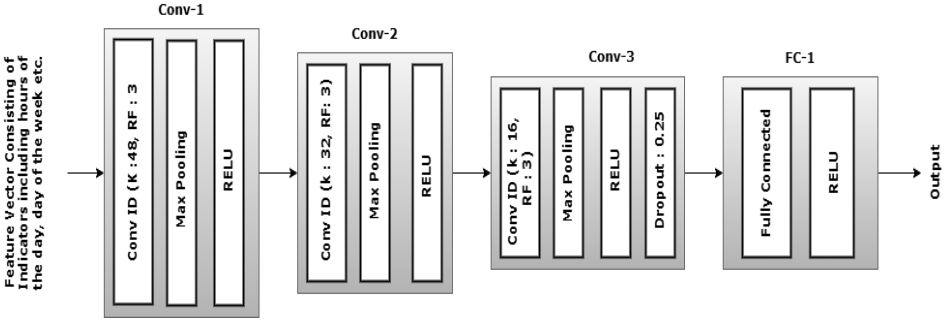

Fig. 1. Deep learning model based on CNN-based AI framework

CNN-based deep learning models are commonly developed using the coarse-to-fine framework. As a result of the high proportion of trainable parameters involved in the coarse-to-fine framework, the computational cost is significantly high. Our implementation uses the well-known pyramid architecture. Initially, there are a large number of kernels in this pyramid architecture, but as we move deeper into the network, they become fewer.

A description of the various layers within the developed deep learning model is given below, including MaxPooling, 1D convolution, ReLU, dropout, and FC.

1. 1D Convolution Layer
   A feature extraction block would not be complete without the operational model for this layer. A feature extraction block would not be complete without the operational model for this layer. A CNN layer's learnable filters have similar

receptive fields. As a result of convolving inputs i_1, i_2, i_3, I_4, I_5 with kernel weights w_1, w_2, w_3, the feature maps FM_1, FM_2, FM_3 can be obtained. Feature maps are obtained using Equation 3 in Figure 2.

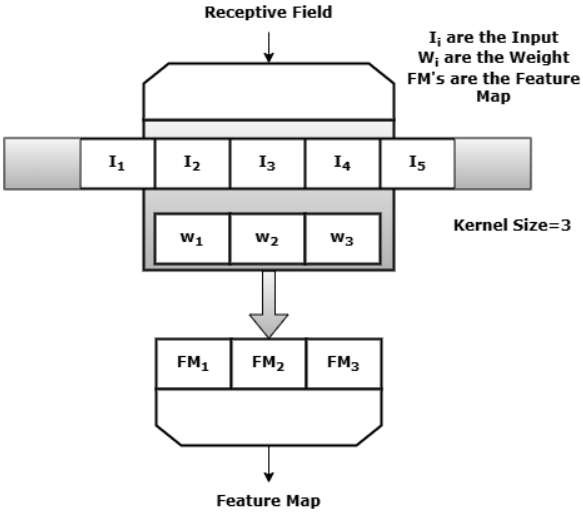

Fig. 2. An overview of the 1D convolution process

$$FM_2 = w_1 * I_2 + w_2 * I_3 + w_3 * I_4 \quad (3)$$

As a result of the CNN layers of our developed model, we can produce feature maps based on the inputs with learnable filters. In deep learning models, convolutional layers can be stacked appropriately to learn features from input data early on. Input signal features are tracked by the feature maps produced. If you slightly shift the location of various features, you'll get a completely different map of features.

2. Max pooling Layer
   An additional pooling layer is normally applied after the convolutional layer to mitigate feature map invariance. The feature map created from this method is affected differently when it is used to create new pooled feature maps. Pooling filter sizes are typically much smaller than feature map sizes. With the pooling layer incorporated into a deep model, feature maps (pooled) are produced that show a summary of the input signals. We have chosen MaxPooling, which is a down-sampling method. An input signal is slid across MaxPooling, and its maximum value is chosen within the overlapped area. Convolution was followed by a MaxPooling layer in our proposed CNN-based AI framework.
3. RELU
   In deep learning models, Rectified Linear Units (ReLUs) are added as activation functions to enhance the ability to learn complex structures. During the development of our deep learning model, we add ReLUs after MaxPooling and after the fully connected layer. Using Equation 4, ReLU operates by thresholding.

$$f(z) = \max(0, z) \quad (4)$$

4. Dropout
   Deep learning models can be overfitted by adding a dropout layer. As part of the deep learning training process, it randomly selects some neurons and deactivates them. Neurons that have been deactivated have no effect.
5. Fully Connected Layer
   In any deep learning model, a fully connected (FC) layer is added for nonlinear mappings between inputs and outputs. Deep learning models are usually finished with an FC layer.

3.3 AI-powered Anomalies Detection

It is important to detect and investigate time series anomalies, especially when working with sensor data from physical infrastructure. We define normal behaviour as deviations from normal patterns on a daily basis. Each of the methods below assumes that the data will be used daily, whereas week-to-week usage can vary. Either technique can be used on any periodicity, and there is no restriction on its use to daily patterns. In the first method, recent measurements are taken into consideration more than older measurements, resulting in a weighted prediction. It involves transforming the observed daily pattern into a transformed space and looking for differences.

The prediction method used must be accurate and expressive in order to compute anomaly scores that are reliable and accurate. Our prediction process incorporates developments over time as described above, assuming daily patterns. So, we used a weighted averaging scheme, where recent values are given a higher weight, while older values are given a lower weight. The following documents provide further details: [27]. This prediction method is not affected by holidays or external events, so it is very effective when predicting weekly patterns. Changing seasons will affect the prediction model, but alternating behaviours cannot be predicted. The quality of the prediction will be negatively affected by a randomly

distributed power usage pattern throughout the day. When a time series is highly unpredictable, any difference between predicted and actual values will have less significance than in a pattern with very small deviations. Anomaly scores are calculated as follows:

$$anomaly[time] = \frac{|predVal[time] - obsVal[time]|}{avg_{t \in Time}(|predVal[t] - obsVal[t])} \quad (5)$$

An anomaly score is calculated at a specific time point in a time series. Observations are normalized using the model deviation average.

4. RESULT AND DISCUSSION

On a centralized model, various machine learning approaches were applied and evaluated by metrics such as Precision, Accuracy, and Recall. The model uses these metrics to identify benign and attack classes in the data. A combination of RNN and the proposed methods achieves 93% and 93% Accuracy and F1-score, respectively. The proposed method was able to correctly identify benign as well as malicious attacks with a Precision and Recall score of 94% and 93%, respectively. Although centralized models deliver commendable performance, they are still prone to failure. (shown in Figure 3).

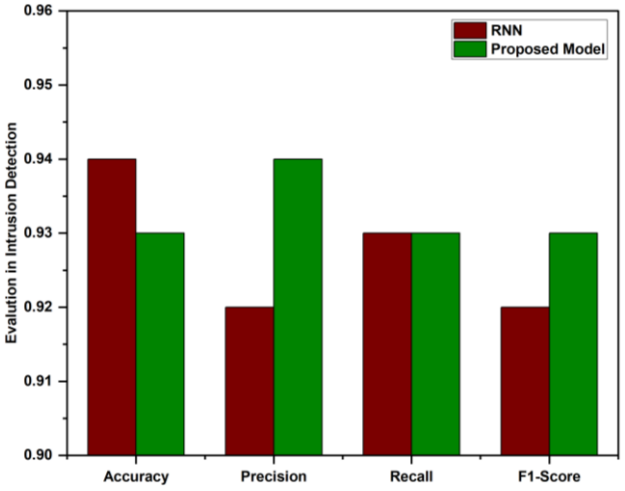

Fig. 3. Detection of intrusions using a centralized model.

After 50 FL rounds, Figure 4 illustrates the FL training time. As shown in the figure, three, nine, and fifteen FL clients consume more time when using different types of data distribution, IID and Non-IID. With an increasing number of clients, training time increases (Figure 4). The figure also clearly shows that the proposed method obtains faster training results than RNNs.

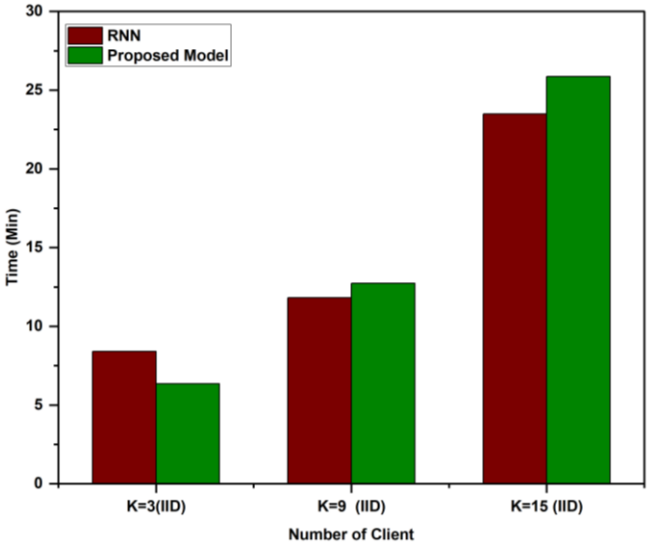

Fig. 4. Number of clients versus Training time (for IID).

According to Figure 5, different data distribution types have little effect on training time, although non-IID scenarios show a slight increase. Because of this, the proposed method requires a considerable amount of training, both in terms of machine learning as well as the number of clients involved.

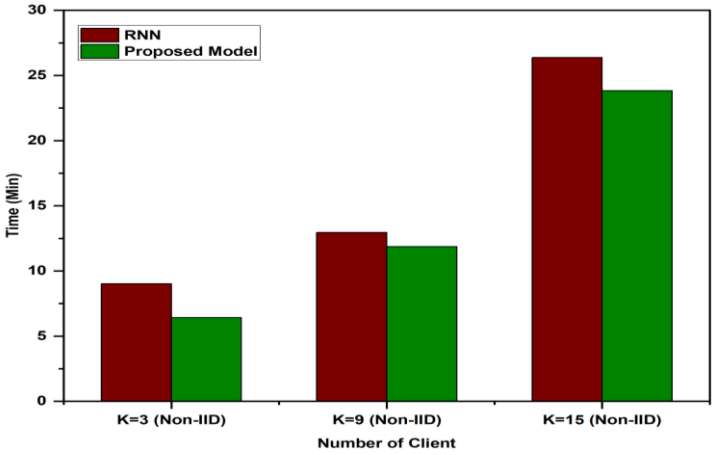

Fig. 5. Number of clients versus Training time (for Non IID).

As shown in Figures 6 and 7, both centralized and FL techniques show the learning process versus accuracy graphs. Initial results indicate that the centralized methods (including both the proposed model and the RNN) are significantly less efficient than the FL technique. As the results narrow in subsequent rounds, the results become extremely competitive by the 50th round. In addition to performing effectively compared to centralized ML approaches, the proposed FL method provides additional advantages, which are detailed in the following sections.

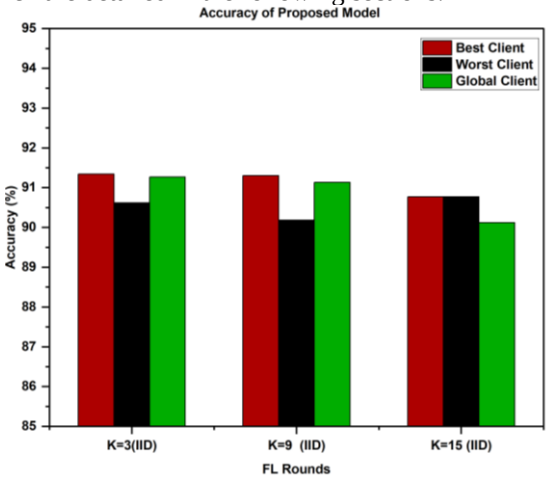

Fig. 6. Accuracy for proposed model (for IID).

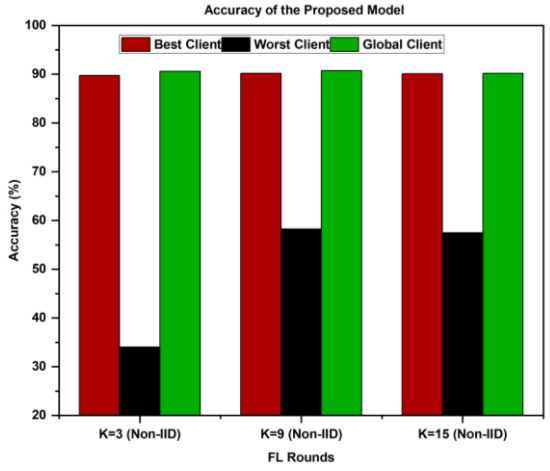

Fig. 7. Accuracy for proposed model (for Non-IID).

5. CONCLUSION

Using a CNN-based deep learning model, we introduce an innovative framework for federated learning to address the security challenges in the IIoT. Using the framework, industrial Internet of Things data can be efficiently processed, robust features can be extracted, and anomalies can be detected. In addition to improving accuracy, precision, and recall, the proposed method is easier to implement and is less costly to communicate. By implementing differential privacy, data is protected without compromising system efficiency. In IIoT networks, the proposed FL framework proves to be an effective security solution. Developing additional privacy-preserving security mechanisms for complex IIoT environments, as well as expanding the capacity of the model to handle dynamic data distributions, will be the focus of future research.

Funding:

The authors declare that no financial aid or sponsorship was received from any external agencies or institutions for this study. All research activities were independently carried out.

Conflicts of Interest:

The authors declare no conflicts of interest.

Acknowledgment:

The authors are sincerely grateful to their institutions for their invaluable guidance and technical support.

References

[1] Y. Guo, Z. Zhao, K. He, S. Lai, J. Xia, and L. Fan, "Efficient and flexible management for industrial Internet of Things: A federated learning approach," *Comput. Netw.*, vol. 192, p. 108122, Jun. 2021, doi: 10.1016/j.comnet.2021.108122.

[2] P. Rani and R. Sharma, "Intelligent transportation system for internet of vehicles based vehicular networks for smart cities," *Comput. Electr. Eng.*, vol. 105, p. 108543, 2023.

[3] M. Panigrahi, S. Bharti, and A. Sharma, "Federated learning for beginners: Types, simulation environments, and open challenges," in Proc. 2023 Int. Conf. Computer, Electronics & Electrical Engineering & Their Applications (IC2E3), Jun. 2023, https://doi.org/10.1109/IC2E357697.2023.10262769

[4] P. Rani *et al.*, "Federated Learning-Based Misbehavior Detection for the 5G-Enabled Internet of Vehicles," *IEEE Trans. Consum. Electron.*, vol. 70, no. 2, pp. 4656–4664, May 2024, doi: 10.1109/TCE.2023.3328020.

[5] A. Singh *et al.*, "Smart Traffic Monitoring Through Real-Time Moving Vehicle Detection Using Deep Learning via Aerial Images for Consumer Application," *IEEE Trans. Consum. Electron.*, vol. 70, no. 4, pp. 7302–7309, Nov. 2024, doi: 10.1109/TCE.2024.3445728.

[6] R. Ahmad and I. Alsmadi, "Machine learning approaches to IoT security: A systematic literature review," *Internet Things*, vol. 14, p. 100365, Jun. 2021, doi: 10.1016/j.iot.2021.100365.

[7] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 16–24, 2013.

[8] M. M. Rashid *et al.*, "A Federated Learning-Based Approach for Improving Intrusion Detection in Industrial Internet of Things Networks," *Network*, vol. 3, no. 1, pp. 158–179, Jan. 2023, doi: 10.3390/network3010008.

[9] N. Hussain and P. Rani, "Comparative studied based on attack resilient and efficient protocol with intrusion detection system based on deep neural network for vehicular system security," in *Distributed Artificial Intelligence*, Boca Raton, FL, USA: CRC Press, 2020, pp. 217–236.

[10] N. Garcia *et al.*, "Distributed real-time SlowDoS attacks detection over encrypted traffic using Artificial Intelligence," *J. Netw. Comput. Appl.*, vol. 173, p. 102871, Jan. 2021, doi: 10.1016/j.jnca.2020.102871.

[11] N. K. Agrawal *et al.*, "TFL-IHOA: Three-Layer Federated Learning-Based Intelligent Hybrid Optimization Algorithm for Internet of Vehicle," *IEEE Trans. Consum. Electron.*, vol. 70, no. 3, pp. 5818–5828, Aug. 2024, doi: 10.1109/TCE.2023.3344129.

[12] Z. Ji, Z. C. Lipton, and C. Elkan, "Differential Privacy and Machine Learning: a Survey and Review," *arXiv preprint*, arXiv:1412.7584, 2014, doi: 10.48550/ARXIV.1412.7584.

[13] E. Benkhelifa, T. Welsh, and W. Hamouda, "A Critical Review of Practices and Challenges in Intrusion Detection Systems for IoT: Toward Universal and Resilient Systems," *IEEE Commun. Surv. Tutor.*, vol. 20, no. 4, pp. 3496–3509, 2018, doi: 10.1109/COMST.2018.2844742.

[14] A. D. Jasim, "A survey of intrusion detection using deep learning in internet of things," *Iraqi J. Comput. Sci. Math.*, vol. 3, no. 1, pp. 83–93, 2022.

[15] W. Samek, S. Stanczak, and T. Wiegand, "The Convergence of Machine Learning and Communications," *arXiv preprint*, arXiv:1708.08299, 2017, doi: 10.48550/ARXIV.1708.08299.

[16] P. Rani and M. H. Falaah, "Real-Time Congestion Control and Load Optimization in Cloud-MANETs Using Predictive Algorithms," *NJF Intell. Eng. J.*, vol. 1, no. 1, pp. 66–76, 2024.

[17] P. Rani, U. C. Garjola, and H. Abbas, "A Predictive IoT and Cloud Framework for Smart Healthcare Monitoring Using Integrated Deep Learning Model," *NJF Intell. Eng. J.*, vol. 1, no. 1, pp. 53–65, 2024.

[18] S. Zhao *et al.*, "A Dimension Reduction Model and Classifier for Anomaly-Based Intrusion Detection in Internet of Things," in *Proc. IEEE DASC/PiCom/DataCom/CyberSciTech*, Orlando, FL, USA, Nov. 2017, pp. 836–843, doi: 10.1109/DASC-PICom-DataCom-CyberSciTec.2017.141.

[19] G. Vallathan *et al.*, "Suspicious activity detection using deep learning in secure assisted living IoT environments," *J. Supercomput.*, vol. 77, no. 4, pp. 3242–3260, Apr. 2021, doi: 10.1007/s11227-020-03387-8.

[20] M. A. Ferrag, L. Shu, H. Djallel, and K.-K. R. Choo, "Deep Learning-Based Intrusion Detection for Distributed Denial of Service Attack in Agriculture 4.0," *Electronics*, vol. 10, no. 11, p. 1257, May 2021, doi: 10.3390/electronics10111257.

[21] H. H. Pajouh *et al.*, "A Two-Layer Dimension Reduction and Two-Tier Classification Model for Anomaly-Based Intrusion Detection in IoT Backbone Networks," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 2, pp. 314–323, Apr. 2019, doi: 10.1109/TETC.2016.2633228.

[22] M. E. Pamukov, V. K. Poulkov, and V. A. Shterev, "Negative Selection and Neural Network Based Algorithm for Intrusion Detection in IoT," in *Proc. IEEE TSP*, Athens, Greece, Jul. 2018, pp. 1–5, doi: 10.1109/TSP.2018.8441338.

[23] Z. Tang, H. Hu, and C. Xu, "A federated learning method for network intrusion detection," *Concurrency Computat.: Pract. Exper.*, vol. 34, no. 10, p. e6812, May 2022, doi: 10.1002/cpe.6812.

[24] H. Chen *et al.*, "Federated Learning Over Wireless IoT Networks With Optimized Communication and Resources," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16592–16605, Sep. 2022, doi: 10.1109/JIOT.2022.3151193.

[25] H. Cao, S. Liu, R. Zhao, and X. Xiong, "IFed: A novel federated learning framework for local differential privacy in Power Internet of Things," *Int. J. Distrib. Sens. Netw.*, vol. 16, no. 5, p. 155014772091969, May 2020, doi: 10.1177/1550147720919698.

[26] J. Castro-Godinez *et al.*, "Approximate Acceleration for CNN-based Applications on IoT Edge Devices," in *Proc. IEEE LASCAS*, San Jose, Costa Rica, Feb. 2020, pp. 1–4, doi: 10.1109/LASCAS45839.2020.9069040.

[27] M. C. Hao *et al.*, "A Visual Analytics Approach for Peak-Preserving Prediction of Large Seasonal Time Series," *Comput. Graph. Forum*, vol. 30, no. 3, pp. 691–700, Jun. 2011, doi: 10.1111/j.1467-8659.2011.01918.x.