Vol. 05, No. S1 (2023) 63-68, doi: 10.24874/PES.SI.01.008

Proceedings on Engineering Sciences

www.pesjournal.net

RESEARCH ON IIOT SECURITY: NOVEL MACHINE LEARNING-BASED INTRUSION DETECTION USING TCP/IP PACKETS

Neha Agarwal[1]
Rajendra Pandey P.
Smitha Rajagopal

Received 26.04.2023.
Accepted 24.06.2023.

Keywords:

Internet of things (IoT), industrial systems, intrusion detection system (IDS), TCP/IP packet, security, hybrid deep convolutional autoencoder, and splinted decision tree (HDCA-SDT).

A B S T R A C T

The Industrial Internet of Things (IIoT) explosive expansion has raised questions regarding the safety of industrial systems. Networks like these are crucially protected from a variety of cyber threats by intrusion detection systems (IDSs). In order to detect intrusions in the IIoT environment utilizing TCP/IP packets, this work introduces a novel Hybrid Deep Convolutional Autoencoder and Splinted Decision Tree (HDCA-SDT) technique. High-level features are extracted from the unprocessed TCP/IP packet data using the DCA. The retrieved features are then classified using the SDT algorithm into various intrusion categories. In order to enable quicker decision-making yet preserve accurate results, the SDT technique effectively divides the feature space. The NSL-KDD dataset is used to train and assess the model. The efficiency of the suggested hybrid strategy is shown by experimental findings. Comparing the proposed hybrid approach to conventional intrusion detection methods, it acquired higher detection accuracy. The model also demonstrates robustness to fluctuations in traffic on the network and possesses the ability to identify known and unidentified intrusions with high recall rates.

1. INTRODUCTION

IoT has pushed for modernization and sped up progress in the wireless and communication industries. It presents a new technology that enables simple communication between people and things by connecting a number of embedded devices, commonly referred to as things, to the Internet. With such widespread adoption, these structures are announced and widely used in a variety of industries, including production, intelligent factories, intelligent houses, and industrial programmers (Nwakanma et al., (2019)). The Industrial IoT (IIoT) is a growing class of IoT-enabled contemporary production constructions that, when used successfully, provides significant viable and economic benefits to system setup, simplicity, reliability, flexibility, and compatibility (Long et al., (2018)). However, IoT-enabled mechanization devices have exposed industrial and habitation areas to a plethora of new risks. Either electronic communication disruption or security breaches are the result of malicious operations. Attackers may quickly and easily steal or delete information from computers as well as network

[1] Corresponding author: Neha Agarwal
Email: neha.agarwal@vgu.ac.in

infrastructure, encouraging the use of cyber warfare (Hafeez et al., (2020)). Numerous assaults, including port scans, network scans, address sweeps, "Man-in-the-Middle (MitM)" attacks, port sweeps, vulnerability scans, theft of information, and botnet assaults, are listed. Formosa Petrochemical Corporation (FPCC) and CPC company, two controlled oil companies in Taiwan, both underwent malware cyber attacks on May 1, 2020. In July 2020, unauthorized access to its IT infrastructure caused security difficulties for Swvl, a Cairo-based platform for booking transit. These illustrations show the weaknesses in the effects of network security. A lot of data is impacted and taken by attackers as a result of the broad multiplicity of IoT gadgets and unprotected "Machine-to-Machine (M2M)" and "Machine-to-People (M2P)" links, which breach individual security and secure equipment use. When an intrusion detection system (IDS) has both excellent detection performance and low error rates, it is seen as strong or precise in identifying disruptions and may detect hostile activities and achieve internet traffic dependability (Muna et al., (2018)). The field of research has undergone a fresh revolution because of anomaly detection systems (ADS). Both labeled and unlabeled data may be easily taken into consideration by the Deep Learning approach in the case of anomalous network packet behavior. There are several well-known machine-learning methods for identifying risky discoveries (Kasongo and Sun, (2019)).In order to increase the accuracy of our detection, we use a Hybrid Deep Convolutional Autoencoder and Splinted Decision Tree (HDCA-SDT) model in this study to track down harmful activity and system performance.

The remainder of the paper is divided into subsequent parts. Part 3 contains the method explained. Part 4 contains the result. Part 5 discusses the conclusions.

2. RELATED WORKS

Anton et al. (2018) converted data from a simulated industrial network into a time interval and analyzed it using three different techniques. Because the data includes labeled assaults, the effectiveness may be evaluated. Da Costa et al. (2019) did a detailed examination of recent research focusing on the IOT and ML, as well as a range of smart methodologies and their applications to intrusion detection designs in computer networks. Ferrag et al. (2021) explored three models for a deep learning-based IDS for DDoS attacks: CNN, DNN, and RNN. Vaiyapuriet al. (2021) offered a variety of DL-based IDS identification methods, databases, and comparative analyses. The study's ultimate goal is to pinpoint the shortcomings, difficulties, and possible future paths of earlier investigations. In order to identify intrusions in the data-centric IoMT-based system, this research suggests a model based on swarm-neural networks. At the network's edge, medical information may be accurately and quickly analyzed because of the suggested model's ability to identify intrusions while information transmission (Awotunde et al., (2022)).
Anton et al. (2018) created a data set of Modbus/TCP communication from a fake industrial situation that is used to build anomaly detection techniques that utilize machine learning to locate fraudulent traffic. Mothukuri et al. (2021) also tried to highlight the benefits of the suggested methodology above conventional ML methods in terms of protecting user data and reaching the highest level of accuracy when identifying threats. Rashid et al. (2023) provided a Federated Learning (FL) technique for securing the security of IoT networks by identifying undesirable intrusions. This technique uses federated training of local IoT device data to assure privacy and security. Small Internet of Things devices only exchange new parameters with a central, international server, which combines them and disseminates a better-detecting method. Friha et al. (2022) suggested the use of FELIDS, a federated learning-based intrusion detection system, to safeguard farm IoT infrastructure. The FELIDS technology particularly safeguards personal information via local learning, whereby gadgets gain the expertise of other devices by exchanging only changes from their algorithm with an aggregate server that generates a better-detecting model. Khan et al. (2021) proposed a DNN for intrusion detection in the MQTT-based protocol and compared the performance of the DNN with other conventional ML methods.

3. METHODOLOGY

In Fig. 1, we show an IIoT scenario where HDCA-SDT models are used to analyze network traffic that has been altered by hostile activity. According to TCP/IP protocols, these IoT devices broadcast their data through gateways like base stations and modems. When unwanted activity is mixed in, malware or ransomware can disrupt network traffic and provide attackers the chance to steal and erase important data. We use the HDCA-SDT model to look for interruptions.

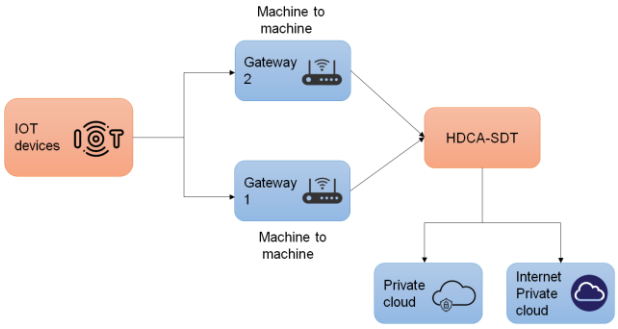

Figure 1. HDCA-SDT models in network traffic

3.1 Dataset

Records from the NSL-KDD dataset have 41 attributes that are either labeled as normal or as a particular kind of assault (Javaid et al., 2016). These characteristics

include the fundamentals of a TCP/IP relationship, traffic characteristics over a period of time or a connection interval, and content features from application layer data. A total of 34 continuous characteristics, four binary characteristics, and three nominal characteristics are included. Twenty-two assault classes, one normal class, and 23 traffic classes make up the training data. In the test data, there are 38 traffic classes, including one normal class, 16 brand-new attacks, and 21 attack classes from training. DoS, probing, U2R, and R2L are the different types of attacks. The statistics of records for training and test data are shown in Table 1 and include both normal and different attack classes.

Table 1. Traffic data allocation for both regular and attack traffic in training and test data.

Traffic	Normal	Attack			
		DoS	U2R	R2L	Probe
Training	67343	45927	52	995	11656
Test	9711	7458	67	2887	2421

3.2 Deep Convolutional Autoencoder (DCA)

A typical sort of neural network design for unsupervised learning tasks, such as anomaly detection, is convolutional autoencoders. They are extremely good at spotting threats in many different fields, including cyber security. Convolutional autoencoders are trained on a dataset containing normal or genuine data in the context of attack detection. The autoencoder gains the ability to compress the input data and then decode it to restore it to its original form. While providing a certain amount of noise tolerance during training, the model learns to properly recreate the normal data. The autoencoder is unable to properly reconstruct the input when it receives information that considerably deviates from the learned normal patterns, indicating the existence of an attack or anomaly. The autoencoder can efficiently identify and indicate possible attacks or departures from normal behavior by comparing the reconstruction error or divergence between the original and reconstructed data.

A typical autoencoder is made up of two layers, which stand for the encoder eu and the decoderhw accordingly. By minimizing the mean squared errors (MSE) among its inputs and its results over all samples, it seeks to identify a code for every input data.

$$\frac{1}{m}\sum_{j=1}^{m} \quad \|hw\left(eu(v_j)\right) - v_j\|2\ 2 \tag{1}$$

$$eu(v) = \sigma(Uv) \equiv g\ hw(g) = \sigma(Ug) \tag{2}$$

Where x and h are variables and are a sigmoid activating function similar to ReLU, for ease of explanation, the bias is ignored. Following training, the included code h acts as the new representation of the input data.

3.3 Splinted decision tree (SDT)

The C4.5 decision tree method is the foundation of the Splinted Decision Tree (SDT) algorithm. The splint value of a node is the key problem while building a decision tree. The suggested technique offers a fresh method for choosing the splint value. The algorithm's stages are as follows:

Step 1: A leaf node for the decision tree is formed by selecting the class from which all of the provided training samples come.

Step 2: Determine the gain ratio for each feature 'a by dividing the characteristic's data gain by the attribute's splinting value. For the gain ratio, use the equation.

$$GainRatio(b) = \frac{JH(b)}{Splint(b)} \tag{3}$$

Where is the collection of all the training set's instances? Step 3: An attribute's information gain is calculated as

$$JH(b = Ent(T) - \sum_{b-val\ \epsilon\ values(b)} \quad \frac{|T-b|}{|b|} * Ent(T-b) \tag{4}$$

Step 4: Calculating entropy is as follows:

$$Ent(T) - \sum_{i=1}^{num\_class} \quad \frac{freq(Ki,T)}{|T|} * log2\,\frac{freq(Ki,T)}{|T|} \tag{5}$$

If L is the set of classes and $K = K_1, K_2, \ldots, K_m$ is the number of distinct classes, and num_class is the number of classes in L. For our purposes, num_class has just two possible values: "normal" and "anomaly."

Step 5: An attribute's splint value is determined by averaging all of the inputs for that attribute throughout the domain. It may be stated as follows:

$$Spplit(b) = \frac{\sum_{j=1}^{n} \quad (b\_val)j}{n} \tag{6}$$

Where n is the number of attribute 'b' values.

Step 6: Determine which attribute has the greatest gain ratio. Assume that the attribute $'b\_best'$ has the greatest gain ratio.

Step 7: Construct a decision node that partitions the dataset based on the $'b\_best'$ attribute.

Step 8: Follow steps 1 through 4 for every subset created by splinting the set on the property "b_best" and insert the resultant nodes as children of the parent node.

The C4.5 algorithm uses the following function to determine an attribute's splint value:

$$Spilnt(b) - \sum_{b\_val\ \epsilon\ values(b)} \quad \frac{|T-b|}{|b|} * log2\left(\frac{|T-b|}{|b|}\right) \tag{7}$$

3.4 Hybrid Deep Convolutional Autoencoder and Splinted Decision Tree (HDCA-SDT)

An approach for classifying data and spotting anomalies is called Hybrid Deep Convolutional Autoencoder and Splinted Decision Tree (HDCA-SDT). In order to get results that are reliable and accurate, it combines the benefits of deep convolutional autoencoders with splinted decision trees.

There are two main parts that make up the HDCA-SDT framework. Initially, a feature extractor is used with the deep convolutional autoencoder. By encoding the input data into a compressed representation using convolutional layers, it learns to capture hierarchical and spatial data. Following that, the autoencoder's decoder makes an effort to recreate the original input while stressing any errors. The second component serves as a classifier and is a splinted decision tree. It applies decision tree-based division to the encoded representations from the autoencoder in order to classify instances as normal or abnormal. The decision tree algorithm performs notably well with structured and multidimensional data. The HDCA-SDT framework can effectively detect and classify anomalies in complex datasets by combining the feature extraction capabilities of the deep convolutional autoencoder with the classification capability of the splinted decision tree. It provides a holistic approach to anomaly detection by capitalizing on the strengths of both techniques, resulting in enhanced precision and robustness.

4. RESULT AND DISCUSSION

All experiments are conducted utilizing the Python3 programming language on the Ubuntu 21.04 OS running on an HP laptop with an AMD Ryzen 9-5950X, 16Giga Byte memory, and an Nvidia GeForce(TM) 990M GPU. The neural network "Application Programming Interfaces (APIs)" TensorFlow and Python's Keras are closely related. TensorFlow is a program used to build machine learning algorithms. Accuracy (%), Precision (%), Recall (%), and F-Measure (%) are analyzed in this section. The existing methods are "Deep learning enabled intrusion detection and prevention system (DL-IDPS)" [17], "Probability risk identification based intrusion detection system (PRI-IDS)" [18], and "Convolutional neural network-Long short term memory (CNN-LSTM)" [19] compared with the proposed method (HDCA-SDT).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

The level of accuracy of both the recommended and recognized procedures is seen in Figure 2. In contrast to the suggested method HDCA-SDT, which achieves an accuracy of 98.9%, DL-IDPS, PRI-IDS, and CNN-LSTM can only achieve an accuracy of 87.6%, 89.2%, and 91.7%, respectively. The proposed approach by HDCA-SDT is more accurate than the methods that have traditionally been used.

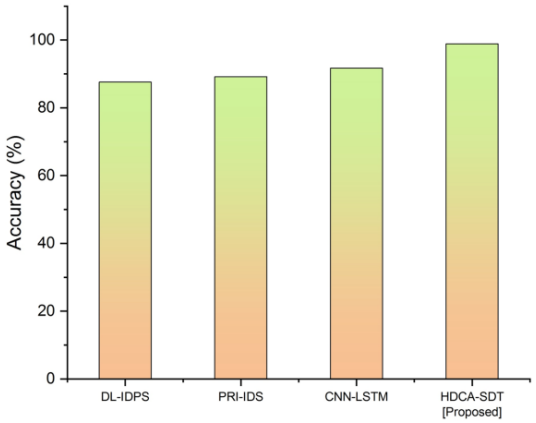

Figure 2. Accuracy

The precision of positive predictions provided by a model or system is measured by precision, a frequently used statistic in binary classification tasks. Out of the total number of positively anticipated cases, the percentage of accurately predicted positive instances is calculated. Figure 3 displays the degree of precision for both the suggested and accepted approaches. DL-IDPS, PRI-IDS, and CNN-LSTM are only able to reach an accuracy of 87.8%, 89.1%, and 90.2%, respectively, in comparison to the recommended approach HDCA-SDT, which achieves 98.7% accuracy. In comparison to conventional techniques, the HDCA-SDT methodology is more precise.

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

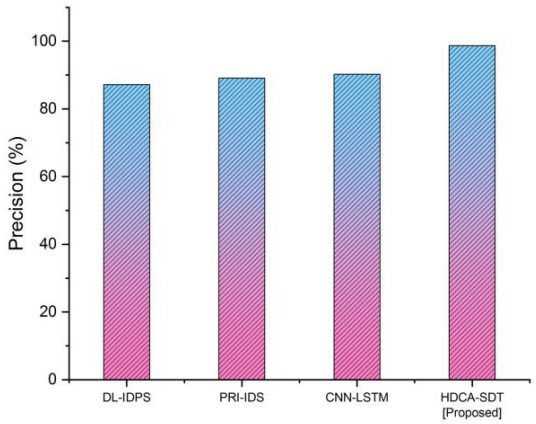

Figure 3. Precision

A recall is a statistic used in binary classification tasks to assess the percentage of real positive cases that are properly detected by a model or system. A recall is also known as sensitivity or true positive rate (TPR). Figure 4 illustrates the degree of recall possessed by both the suggested and the recognized processes. However, in comparison to the proposed approach HDCA-SDT, which achieves a recall of 98.2%, DL-IDPS, PRI-IDS, and CNN-LSTM are only able to obtain a recall of 88.3%, 89.5%, and 91.2%, respectively. The methodology that is being offered by HDCA-SDT is more recall than the approaches that have been typically utilized in the past.

$$Recall = \frac{TP}{TP+FN} \quad (10)$$

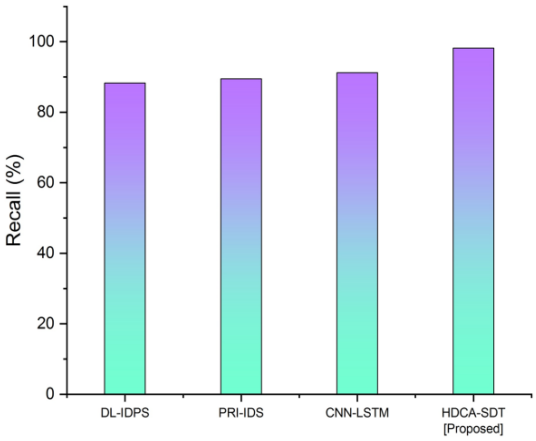

Figure 4. Recall

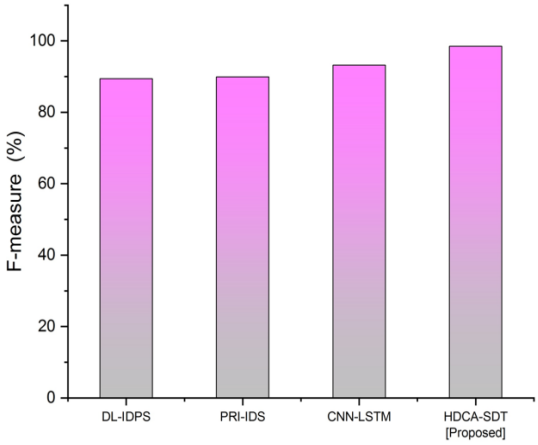

Figure 5. F-Measure

The F-measure, usually referred to as the F1 score, is a statistic frequently employed in binary classification tasks to balance the trade-off between recall and precision. It offers a single metric that combines recall and accuracy into a single number. The level of F-Measure that both the proposed and the recognized procedures have is shown in Figure 5. However, DL-IDPS, PRI-IDS, and CNN-LSTM are only able to produce F-measure of 89.4%, 89.9%, and 93.2%, respectively, in compared to the suggested technique HDCA-SDT, which obtains a recall of 98.5%. In comparison to earlier methods that were frequently used, the HDCA-SDT methodology has a higher F-measure.

$$F - Measure = 2 \times \frac{Precision * recall}{Precision + recall} \quad (11)$$

5. CONCLUSION

The Hybrid Deep Convolutional Autoencoder and Splinted Decision Tree (HDCA-SDT) approach is a revolutionary one that is introduced in this paper to detect intrusion in IIOT. Using the DCA, high-level characteristics are retrieved from the raw TCP/IP packet data. The NSL-KDD dataset was used in this work to evaluate the suggested HDCA-SDT method. Accuracy (98.9%), Precision (98.7%), Recall (98.2%), and F-measure (98.5%) values were obtained for our proposed method's performance measures. It has been demonstrated experimentally that the proposed strategy was superior to the current method for detecting intrusion. The IIoT method's ability to acquire data from the actual world in order to assess how well it performs in these circumstances will be taken into account in future analyses. The suggested model will also be expanded in further work to include more protocols.

References:

Anton, S.D., Ahrens, L., Fraunholz, D. and Schotten, H.D., (2018), November. Time is of the essence: Machine learning-based intrusion detection in industrial time series data. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 1-6). IEEE. https://doi.org/10.1109/ICDMW.2018.00008

Anton, S.D., Kanoor, S., Fraunholz, D. and Schotten, H.D., (2018), August. Evaluation of machine learning-based anomaly detection algorithms on an industrial Modbus/TCP data set. In Proceedings of the 13th international conference on Availability, reliability, and Security (pp. 1-9). https://doi.org/10.1145/3230833.3232818

Awotunde, J.B., Abiodun, K.M., Adeniyi, E.A., Folorunso, S.O. and Jimoh, R.G., (2022, January). A deep learning-based intrusion detection technique for a secured IoMT system. In Informatics and Intelligent Applications: First International Conference, ICIIA 2021, Ota, Nigeria, November 25–27, 2021, Revised Selected Papers (pp. 50–62). Cham: Springer International Publishing. DOI: 10.1007/978-3-030-95630-1_4

Da Costa, K.A., Papa, J.P., Lisboa, C.O., Munoz, R. and de Albuquerque, V.H.C., (2019). Internet of Things: A survey on machine learning-based intrusion detection approaches. Computer Networks, 151, pp.147-157. https://doi.org/10.1016/j.comnet.2019.01.023

Ferrag, M.A., Shu, L.,Djallel, H. and Choo, K.K.R., 2021. Deep learning-based intrusion detection for distributed denial of service attack in agriculture 4.0. Electronics, 10(11), p.1257. https://doi.org/10.3390/electronics10111257

Friha, O., Ferrag, M.A., Shu, L., Maglaras, L., Choo, K.K.R. and Nafaa, M., (2022). FELIDS: Federated learning-based intrusion detection system for agricultural Internet of Things. Journal of Parallel and Distributed Computing, 165, pp.17-31. https://doi.org/10.1016/j.jpdc.2022.03.003

Hafeez, I., Antikainen, M., Ding, A.Y. and Tarkoma, S., (2020). IoT-KEEPER: Detecting malicious IoT network activity using online traffic analysis at the edge. IEEE Transactions on Network and Service Management, 17(1), pp.45-59. https://doi.org/10.1109/TNSM.2020.2966951

Javaid, A., Niyaz, Q., Sun, W. and Alam, M., (2016), May. A deep learning approach for network intrusion detection system. In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS) (pp. 21-26). DOI 10.4108/eai.3-12-2015.2262516

Kasongo, S.M. and Sun, Y., (2019). A deep learning method with filter-based feature engineering for wireless intrusion detection system. IEEE Access, 7, pp.38597-38607. https://doi.org/10.1109/ACCESS.2019.2905633

Khan, M.A., Khan, M.A., Jan, S.U., Ahmad, J., Jamal, S.S., Shah, A.A., Pitropakis, N. and Buchanan, W.J., (2021). A deep learning-based intrusion detection system for MQTT enabled it. Sensors, 21(21), p.7016. https://doi.org/10.3390/s21217016

Kim, T.Y. &Cho, S.B. (2019). CNN-LSTM neural networks for anomalous database intrusion detection in RBAC-administered model. In Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part IV 26 (pp. 131–139). Springer International Publishing. DOI: 10.1007/978-3-030-36808-1_15

Lee, T.H., Chang, L.H. and Syu, C.W., (2020), June. Deep learning enabled intrusion detection and prevention system over SDN networks. In 2020 IEEE International Conference on Communications Workshops (ICC Workshops) (pp. 1-6). IEEE. https://doi.org/10.1109/ICCWorkshops49005.2020.9145085

Long, N.B., Tran-Dang, H. and Kim, D.S., (2018). Energy-aware real-time routing for the large-scale industrial Internet of Things. IEEE Internet of Things Journal, 5(3), pp.2190-2199. https://doi.org/10.1109/JIOT.2018.2827050

Marsden, T., Moustafa, N., Sitnikova, E. & Creech, G. (2018). Probability risk identification based intrusion detection system for SCADA systems. In Mobile Networks and Management: 9th International Conference, MONAMI 2017, Melbourne, Australia, December 13-15, 2017, Proceedings 9 (pp. 353–363). Springer International Publishing. DOI: 10.1007/978-3-319-90775-8_28

Mothukuri, V., Khare, P., Parizi, R.M., Pouriyeh, S., Dehghantanha, A. and Srivastava, G., (2021). Federated-learning-based anomaly detection for IoT security attacks. IEEE Internet of Things Journal, 9(4), pp.2545-2554. https://doi.org/10.1109/JIOT.2021.3077803

Muna, A.H., Moustafa, N. &Sitnikova, E. (2018). Identification of malicious activities in the industrial Internet of Things based on deep learning models. Journal of information security and applications, 41, pp.1-11. https://doi.org/10.1016/j.jisa.2018.05.002

Nwakanma, C.I.,Nwadiugwu, W., Lee, J.M. and Kim, D.S., (2019), June. Real-Time validation scheme using blockchain technology for Industrial IoT, in Proceedings of 2019 Korean Institute of Communications and Information Sciences Summer Conference (pp. 379-382).

Rashid, M.M., Khan, S.U., Eusufzai, F., Redwan, M.A., Sabuj, S.R. and Elsharief, M., (2023). A Federated Learning-Based Approach for Improving Intrusion Detection in Industrial Internet of Things Networks. Network, 3(1), pp.158-179. https://doi.org/10.3390/network3010008

Vaiyapuri, T., Sbai, Z., Alaskar, H. and Alaseem, N.A., (2021). Deep learning approaches for intrusion detection in IIoT networks–opportunities and future directions. International Journal of Advanced Computer Science and Applications, 12(4). https://dx.doi.org/10.14569/IJACSA.2021.0120411

Neha Agarwal
Vivekananda Global University, Jaipur, India
neha.agarwal@vgu.ac.in
ORCID 0000-0003-2647-3458

Rajendra Pandey P
Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India
panday_004@yahoo.co.uk
ORCID 0000-0003-4769-2465

Smitha Rajagopal
Jain (deemed to be)University, Bangalore, India
smitha.rajagopal@jainuniversity.ac.in
ORCID 0000-0002-0643-1931