University of Ha’il–Journal of Science (UOHJS)

Volume IV - Second Issue

ISSN: 1658-8096
e-ISSN: 1658-8800

December 2023

An Improved Hybrid Machine Learning Security Model for IoT Systems

Ashwag Albakri*

Department of Computer Science, College of Computer Science & Information Technology, Jazan university, Jazan 45142, SaudiArabia

[*] *Corresponding author; E-mail: aoalbakri@jazanu.edu.sa*

Abstract:

Internet of Things (IoT) security has been receiving more attention from both the academic and corporate sectors. There are various security threats that target IoT devices, including network intrusions as well as data breaches. Thus, to detect security threats in an IoT network, an efficient security model is required to protect the network. An enhanced security architecture grounded in machine learning (ML) approaches that robotically adjust to the changing in security requirements of the IoT domain is presented in this research paper. The proposed system uses support vector machines (SVM) for cybersecurity threat identification and convolutional neural networks (CNN) for feature extraction. An AI-based detection model associates with a monitoring agent, and an AI-based reaction agent are utilized to look at the network patterns, using ML models, and detect intrusions into the IoT devices. The proposed approach is effective when it is compared with other existing approaches; the experiment results shows that the proposed model achieve nearly 99% accuracy.

Key words*: Machine Learning, IoT, Convolutional Neural Network, Security, Support Vector Machine.*

1. INTRODUCTION

With an enormous count of cellular IoT devices anticipated to be positioned in the upcoming years, the IoT is significantly altering the ICT environment, Bagaa et al. (2020). IoT is one of the most popular technologies in use today, and it is referred to as a networked system of heterogeneous components that allows intelligent systems along with some services to sense, gather, distribute, and analyze data, Sarker et al. (2022). The IoT technology has offered the first preview of the future IoT-based ecosystem. This IoT has some issues, mostly with security, including intrusions into the IoT network, Ghillani and Diptiban (2022).

Several approaches have been proposed to solve the security issues in IoT. One of the approaches utilize a CNN-grounded method for anomaly-based intrusion detection systems (IDS), Saba *et al.* (2022) . The proposed method makes use of the IoT's promise along with the necessary tools to accurately analyze all the IoT traffic. That proposed model has the ability to detect possible intrusions and odd traffic patterns, Saba *et al.* (2022). Another research provids an effective AI-based IDS mechanism for

IoT systems in which Deep learning (DL) and metaheuristic (MH) algorithms' improvements are utilized, and they prove their effectivness in solving difficult engineering problems , Fatani *et al.* (2021).

In another study by Lin and Tao (2020), the overall points of adversarial machine learning (ML) are initially presented. Then, it demonstrates how conventional approaches, like Petri Net, cannot effectively address the security problem. The proposed method provides privacy along with trustworthiness. This study then employs a triage (filter) study example from the IoT cyber security operations center, Lin and Tao (2020). According to Sharma *et al.* (2021), the authors discuss the security of the Industrial IoT (IIoT) devices, and how the security of the IIoT that are integrated with other technologies such as 5G and blockchain can be improved. An extensive examination of the deployment of cutting-edge security goes on to evaluate the IoT device product life cycle.

Arachchige *et al.* (2020) proposes a system known as PriModChain that combines differential privacy, federated ML, the Ethereum blockchain, as well as smart contracts. It ensures the privacy along with reliability of industrial IoT (IIoT) data. Also, the research examines the self-normalizing neural network (SNN), a variation of the forward neural network (FNN), and compares its effectiveness with the FNN for classifying intrusion threats in an IoT network. Ibitoye *et al.* (2019) research makes use of the BoT-IoT dataset. Though several approaches have been proposed in the existing studies, security is still a concern. Thus, in this research work, an improved ML-based model is proposed.

The rest of this paper is organized as follows: Section 2 discusses the literature review on IoT security. Section 3 demonstrates the proposed enhanced ML-based model architecture and the associated technologies. The performance analysis of the proposed enhanced ML-based model, and the results of its assessments are shown in Section 4. The conclusion is drawn in Section 5 along with the limitations of the proposed work.

2. LITERATURE REVIEW

The most recent research work related to the various ML approaches to detect security issues in IoT are reviewed in this section.According to Makkar et al. (2020), the quality of the data generated by the IoT devices varies due to the device's speed, which is affected by the time and location dependencies. The volume of the data generated by the IoT devices has also increased significantly. In that case, ML algorithms played a key role in ensuring the security of IoT systems through biotechnology-based security and authorization, and in improving the usability through anomaly detection techniques. However, ML algorithms are also used by hackers to target the security holes in IoT-based smart equipment. Makkar et al. (2020) suggested that by using ML to detect spam, IoT device security may be accomplished. To achieve that, a ML framework for spam detection in the IoT was proposed.

Uprety et al. (2020) discussed that ML techniques, i.e., reinforcement learning, have been a burgeoning option for IoT security. In contrast to the other ML approaches, reinforcement learning could learn from the environment with only a small amount of input on the parameters to be learned. By interacting with the environment and changing the parameters instantly, it resolved the optimization problem. The authors' study provided a thorough analysis of cyberattacks that have been launched against various IoT systems, and it provided security methods based on the reinforcement learning and deep reinforcement learning to counteract those various cyberattacks against the IoT systems.

Li et al. (2021) highlighted a key problem of improving the industrial development that focused on how to connect high-risk network actions with entities. The authors discussed unresolvable conflicts of the security challenges that are emerged due to the growth of the IoT technologies. The proposed solution was to create a security architecture that would concentrate on finding ways to replace human monitoring with intelligent system defense. To do so, the authors examined the possibility of utilizing

DL to improve the IoT security architecture, and they discussed how the IoT can detect and react to cyberattacks, how to encrypt edge data flow, and how to incorporate current security solutions.

Sarker et al. (2021) provided "CyberLearning," a ML-grounded cybersecurity modeling with correlated feature selection. That research paper also provided a detailed empirical assessment of several ML security models. The cyberlearning model offered a binary and a multi-class classification model for recognizing different cyberattacks. The model was made with efficient ML classification methods such as naive Bayes (NB), logistic regression (LR), decision trees (DT), K-nearest neighbors (KNN), stochastic gradient descent (SGD), SVM, random forests, adapted boosts, extreme boosts, and linear discriminant analysis.

Object detection was emphasized by Zhou et al. (2021) as being essential for activity tracking and surveillance system environment monitoring. Due to the expanding edge-cloud computing architecture, the authors showed how to handle the constantly rising amount of surveillance data locally across the IoT devices. That research focused on multitarget discovery for monitoring IoT systems in real time. It was advised to employ A-YONet, a deep neural network model with restricted computational resources. The advantages of You Only Look Once (YOLO) and Multi-Task CNN (MTCNN) were combined to build the model.

According to Ullah et al. (2021), classic ML methods became useless when they were deployed in an environment that encopassed inconsistent network technologies and various infiltration techniques. In a number of academic domains, DL approaches have shown they could accurately spot irregularities. The CNNs were a fantastic option for anomaly detection and classification since they are effective at performing speedier computations and have the ability to automatically categorize important attributes in input data. The authors proposed an intrusion detection model for IoT networks that utilized CNN model to generate a multiclass classification method.

In order to screen the cutting process and preserve Controller Numerical Control (CNC) machines' cutting stability, Tran et al. (2022) developed a solution that guarantees effective cutting procedures that could help in improving the quality of the finished product. In order to monitor vibration conditions, a force sensor was installed inside the milling CNC machine center. An IoT architecture was built to establish a connection between the sensor node and the cloud server, enabling the use of the message queue telemetry transport (MQTT) protocol for real-time machine status gathering. A refined deep neural network (DNN) model has been devised to effectively categorize various cutting circumstances, namely stable cutting and unstable cuts. The objective was to ensure the CNC machine remains in an optimal condition..

Mukhtar et al. (2023) argued that the use of ML techniques amplifies the implementation deficiencies of algorithms on edge devices, rendering them more vulnerable to side-channel attacks. The present work introduced a DL-based system that was included in the edge device for the purpose of identifying side-channel leakages. The process of designing a deep learning-based system entails several problems, among which lies the task of accurately identifying the appropriate attack model for the underlying target algorithm. Three machine learning-based side-channel attack models were collected and examined in order to evaluate the security of the edge devices, building upon the previous findings. In this particular instance, the elliptic-curve encryption approach, which is well recognized, was used as a test case.

While detecting intrusions in an IoT context, intrusion detection in IoT still has significant challenges. This paper offers a comprehensive methodology for quickly and effectively identifying and stopping cybersecurity assaults using ML and DL approaches. The current research contributions are as follows:

- By checking, identifying, and blocking cybersecurity threats in IoT, a unified AI-grounded security detection model is aligned.
- To handle intrusion detection, the implementation along with the validation of an AI security detection model for the IoT use a mix of ML and DL approaches.
- For detecting cybersecurity threats, a hybrid CNN-SVM model known as the HCS model is proposed in this paper. The goal is to use CNN to extract characteristics from the data, then utilizing those characteristics by the SVM classifier to identify the cybersecurity attacks.
- By using that technique, it is not necessary to gather previous information or feature design specifics. The main benefit of the CNN model is that it makes use of the input's topological data and is invariant to simple transformations such as rotation and translation.

3. PROPOSED METHODOLOGY

For detecting cybersecurity assaults in IoT, a hybrid HCS model is proposed. The proposed model combines SVM and CNN classifiers' best features. A CNN has a supervised learning process and numerous dense or fully connected layers. The CNN absorbs invariant local features extremely well and functions a human. It obtains the data's most discerning information. Each layer's output serves as the subsequent layer's input. Effective features are determined from the input data using CNN. The goal of SVM is to represent multi-dimensional datasets in a space where hyperplanes serve as the boundaries between data components from various classes. SVM is terrible for noisy data but has been verified to be excellent for binary classification. Learning deep features might be non-effective because of the shallow design of SVM, according to Ahlawat, Savita, and Amit Choudhary (2020). The proposed HCS model is designed to include the benefits of CNN as well as SVM, as shown in Figure. 1. The layers that make up CNN's general design are the convolution layer, dropout layer, flatten layer, pooling layer, along with dense layer, Khairandish *et al.* (2022). The proposed HCS model is demonstrated in the follwing subsections.

3.1. Preprocessing

Principal Component Analysis (PCA) is used in this research to transform high-dimensional data into a lower-dimensional representation by identifying the most significant features that contain the most amount of information within the dataset. PCA is a key method for compressing data and extracting features from it. It has also been used in intrusion or attack detection. A data space is changed into a feature space, which has fewer dimensions, through a method called feature selection. The selection of features is based on the variation they induce in the output. The primary determinant of variance is the first principal component. The feature that exhibits the second largest variation is referred to as the second principal component, and so forth. It is essential to note that the principal components exhibit no association among themselves. The PCA is often used as a preprocessing technique to effectively shorten the training time of algorithms by reducing the dimensionality of the feature space, Geladi, Paul, and Johan Linderholm (2020). In this work, after pre-processing 41 features, only 5 features are considered and the others are neglected.

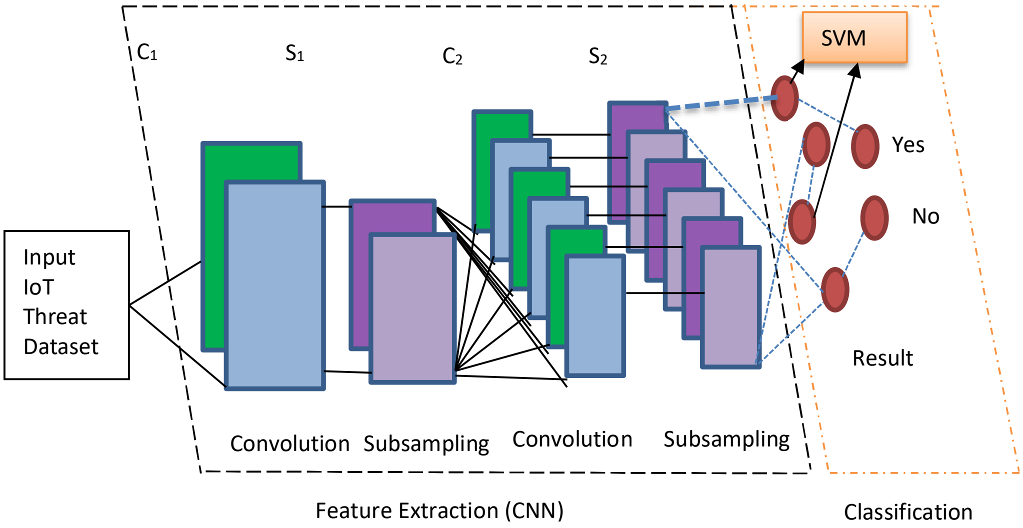

Figure 1. Proposed HCS Model Architecture

3.2. Convolutional Neural Netowrk (CNN)

A Convolutional Neural Network (CNN) typically has three layers: a hidden layer, an input layer, along with an output layer, Yang A *et al.* (2019). Here, the output from one layer is often utilized as the subsequent layer's input. The different layers are the convolution layer, dropout layer, pooling layer, flatten layer, as well as dense layer. The convolution layer is the initial layer of the proposed CNN model that extracts input features, where the pixel properties and the connections between neighboring pixels are retrieved by a mathematical process.

The dropout layer enhances the proposed model's stability and efficacy, and is used as a solution to overfitting during the training phase. In contrast to the dropout layer, where neurons are randomly selected, weights are changed and altered throughout the training. The pooling layer selectively retains crucial data by discarding redundant input after feature extraction. The subsampling procedure is used to decrease the data map size while retaining crucial information. The use of a pooling layer in conjunction with the mapping of key features has the potential to lead to overfitting. Before adding a dense layer, two-dimensional (2D) arrays from the previous layer are flattened into one-dimensional (1D) arrays. An output from a dense layer is the last output.

The success of deep networks serves as the driving force behind this study, which proposes an improvement in the CNN to overcome the limitations of current techniques. The proposed model has a dense layer, two convolution layers, two dropout layers, two max-pooling layers, along with a flatten layer. The Adam optimizer is used for training the model with some parameters like learning rate, decay, beta_1, epsilon, beta_2, and amsgrad, Kumar *et al.* (2020). Also, that gives the CNN stacks nearly eight layers more than the normally used which is three to five layers.

3.3. Support Vector Machine

Following the completion of pre-processing and feature extraction, the identification of cybersecurity threats is conducted via the use of a SVM classifier. The SVM classifier was trained by using feature vectors that were stored in matrix format. The evaluation of the numerical value has been conducted using the outcome of the instructional process, Alkhaleefah M and Wu Chao-Cheng (2018).

3.4. Proposed Hybrid CNN-SVM (HCS) Model

In the proposed hybrid CNN-SVM (HCS) model, the features from the CNN are used in the SVM for the purposes of training as well as evaluating the cybersecurity threat dataset. Initially, the CNN features are gathered, and then the CNN features output is given to the SVM as input. The testing data undergoes a similar pre-processing procedure before being employed to assess the classifier. The proposed HCS performance is evaluated by assessing its accuracy on the datasets. The acquired findings are recorded for further analysis. The proposed HCS model utilizes an RBF function as its kernel.

This is the main justification for combining CNN's and SVM's benefits. Although SVM is different from other algorithms in terms of how it extracts features, it also offers excellent efficiency and speed. For cybersecurity threats, this combination of ML and DL algorithms could provide the greatest benefits.

4. RESULTS AND DISCUSSION

The proposed HCS model is implemented with Python, Jupyter environment programming, and the Anaconda package. The comparison is done with the aid of some confusion matrix parameters. Though ML algorithms are well suited for small datasets, this study combines DL algorithms. The DL models have the capability to autonomously extract pertinent features from the data, hence diminishing the need for human feature engineering.

4.1. Dataset Description

The NSL-KDD dataset is used in this research because the proposed research is related to network intrusion detection, see https://www.kaggle.com/datasets/hassan06/nslkdd/data. Researchers often use these datasets as a means to evaluate and compare the effectiveness of their intrusion detection algorithms for IoT networks. The dataset used in this study has been enhanced by eliminating duplicate connection records from the original KDDCup-99 dataset. Furthermore, the dataset is formatted in CSV, including a total of 41 features and encompassing five distinct assault types. The 42nd characteristic stores data on the five different types of network link vectors, which are divided into one network normal class and four network attack classes. DoS, R2L, Probe, and U2R are additional attack types that are subdivided. The MIT Lincoln Laboratory's dataset created for the 1998 DARPA intrusion detection competition served as the source of the KDDCup-99 dataset. This is tabulated in Table 1.

Table 1. Attacks Description

Attacks	Type of Attacks
DoS	Back, Land, Neptune, Pod, Smurf, Worm, Teardrop, Mailbomb, Processtable, Udpstorm, Apache2
R2L	Guess_password, Ftp_ write,Imap, Phf, Multihop, Warezmaster, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Named, Httptunnel,Sendmail
Probe	Satan, IPsweep, Nmap, Saint, Portsweep, Mscan
U2R	Buffer_overflow, Loadmodule, Perl, Rootkit, Sqlattack, Xterm, Ps

4.2. Discussions

The parameters used in the proposed HCS model are described in Table 2. Convolutional layers are the first layer. It is followed by a maxpooling layer as the second layer with a pool size of 4. The third layer is the dropout layer with 0.2 as the dropout rate. Another set of convolutional layers, the maxpooling layer and the dropout layer serve as the fourth, fifth and sixth layers. The seventh layer is the flattening layer. The final eight layers, which are dense, have 50 units along with the sigmoid activation function. Categorical cross entropy, the user-defined Adam optimizer, and 50 epochs make up the additional training parameters. The parameters of the user-defined Adam optimizer are tabulated in Table 3.

Table 2. Proposed HCS model parameters

Name	Layer	Parameter
Conv1D	Convolutional layer	kernel size =3, Filters = 32, padding = same, activation = relu, input shape
Maxpooling1D	Pooling layer	Pool size =4
Dropout	Dropout layer	Drop rate = 0.2
Conv1D	Convolutional layer	Filters = 32, kernel size =3, padding = same, activation = relu
Maxpooling1D	Pooling layer	Pool size =4
Dropout	Dropout layer	Drop rate = 0.2
Flatten	Flatten layer	-
Dense	Dense layer	Unit =50, activation = softmax

Table 3. Adam parameters

Parameters	Values
Learning rate	0.001
decay	0.0
amsgrad	False
beta_1	0.9
Epsilon	1e-07
beta_2	0.999

The assessment of the performance of the HCS model is an essential stage in determining its efficacy in handling hypothetical data. In this work, a total of eight independent metrics were used to provide a comprehensive comparison between the proposed HCS model and the current CNN and SVM models. These metrics enable the evaluation of the relevance of the results for each classification label and facilitate the comparison of several models. The evaluation metrics used in this study include the f1 score, precision, accuracy, recall, macro average, confusion matrix, support, and weighted average.

Accuracy may be quantified by calculating the ratio of correctly recognized observations to the total number of observations. This metric offers insights into the percentage of inputs that are accurately recognized, serving as an initial gauge of the model's success. Equation (1) denotes the mathematical representation of accuracy.

$$\text{accuracy} = \frac{\text{correct observations}}{\text{All Observations}} \qquad (1)$$

The confusion matrix has a comprehensive record of all the predictions made by the model. The matrix's columns are associated with the actual class classifications, while rows denote each respective class prediction. In order to conduct a comprehensive analysis, the words true negative (TN), false negative (FN), true positive (TP), and false positive (FP) are used. Precision is a quantitative measure used to evaluate the level of accuracy or excellence in the predictions made by a model pertaining to a certain class. The concept of precision is represented by equation (2).

$$\text{prec} = \frac{\text{TP}}{\text{FP+TP}} \qquad (2)$$

Recall refers to the ratio of correctly categorized samples by the model to the total number of samples in a certain class. It is sometimes termed to as "sensitivity," since it characterizes the model's responsiveness to the presence of a certain class. The concept of recall is represented by equation (3).

$$\text{recall} = \frac{\text{TP}}{\text{FN+TP}} \qquad (3)$$

The F1 Score, a harmonic ratio, is used to attain equilibrium between recall along with accuracy. The F1 score is calculated using the formula (4).

$$\text{F1 score} = 2 * \left(\frac{\text{prec}*\text{recall}}{\text{prec+reca}}\right) \qquad (4)$$

The support function provides data on the aggregate occurrence of models belonging to a certain class. This information provides insight into the number of samples that were used in the research, enabling the examination of the distribution of classes within the dataset. There are two methods for summarizing these measures, namely macro and weighted average. When computing a macro-average, the individual values of a measure, such as recall, accuracy, f1 score, etc., for each class are summed together. After that, divide the total by the overall number of classes. The mathematical representation of this is denoted as (5).

$$\text{Macro avg} = \frac{\text{class 1 score +cl} \qquad \text{2 score}}{2} \qquad (5)$$

A statistical technique known as weighted averaging is used to account for the imbalanced distribution of class samples within a dataset. The mathematical representation of this phenomenon is (6), which divides the total sample count by the sum of the metric scores obtained for each class and the support value.

$$\text{Weighted avg} = \frac{\sum(\, i^{th} \text{ class score } * i^{th} \text{ class support})}{\text{total dataset samples count}} \qquad (6)$$

The proposed HCS model for cybersecurity threat detection comparison is done with the existing CNN and SVM. This detail is tabulated in Table 4.

Table 4. Comparison Table

Methods	Classes	Precision	Recall	F1-score	Support
CNN	0	0.95	0.96	0.96	13334
	1	0.96	0.941	0.95	11861
	accuracy	-	-	0.96	25195
	Macro avg	0.96	0.95	0.95	25195
	Weighted avg	0.96	0.96	0.96	25195
SVM	0	0.95	0.95	0.95	13334
	1	0.95	0.94	0.94	11861
	accuracy	-	-	0.95	25195
	Macro avg	0.95	0.94	0.95	25195
	Weighted avg	0.95	0.95	0.95	25195
Proposed HCS	0	0.99	1.00	0.99	13334
	1	1.00	0.99	0.99	11861
	accuracy	-	-	0.99	25195
	Macro avg	0.99	0.99	0.99	25195
	Weighted avg	0.99	0.99	0.99	25195

Table 5. Comparison Table

Methods	Precision	Recall	F1-score
CNN	0.96	0.96	0.95
SVM	0.95	0.95	0.94
LR Ullah et al. (2021)	0.98	0.98	0.98
Proposed HCS	0.99	0.99	0.99

Table 4 provides a comprehensive comparison of performance metrics across various approaches. Upon analyzing all the parameters, it is obvious that the proposed model exhibits good performance in comparison to the existing models. In this particular instance, the HCS model under consideration has an approximate accuracy rate of 99%. The performance of this model exhibits a 3.13% improvement compared to the CNN model and a 4.21% improvement compared to the SVM model. Also, when compared with the existing LR model, the accuracy of the proposed model is 1.02% better, as shown in Table 5.

5. CONCLUSION AND FUTURE STUDY

In this paper, we employ the proposed HCS model to identify and mitigate cybersecurity threats in the IoT domain. Methodologies such as SVM and CNN are used to compare and find real positive results about how well the proposed model works. Our research work determined that the proposed HCS model had much better performance in comparison to statistical techniques. This paper provides evidence for the conclusions drawn from several investigations, which shows that the proposed model is the most efficient approach to identifying cybersecurity vulnerabilities. In the analysis of the NSL-KDD dataset, many performance measures including precision, F1 score, accuracy, recall, macro average, support, and weighted average are evaluated and compared. The findings indicate improvements to the proposed model. The proposed model shows an improvement of 3.13% with the existing CNN and an improvement of 4.21% with the existing SVM method. The objective of continuing the research is to further enhance the algorithm. Despite the improved accuracy of the proposed research work, it has not been implemented in any real-time applications which we considered as future work.

References

Ahlawat, Savita, and Amit Choudhary, "Hybrid CNN-SVM classifier for handwritten digit recognition," Procedia Computer Science, vol. 167, pp.2554-2560, 2020.

Alkhaleefah M, Wu Chao-Cheng, "A hybrid CNN and RBF-based SVM approach for breast cancer classification in mammograms," In: 2018 IEEE international conference on systems, man, and cybernetics (SMC). IEEE, 2018.

Arachchige, Pathum Chamikara Mahawaga, Peter Bertok, Ibrahim Khalil, Dongxi Liu, Seyit Camtepe, and Mohammed Atiquzzaman, "A trustworthy privacy preserving framework for machine learning in industrial IoT systems," IEEE Transactions on Industrial Informatics, vol. 16, no. 9, pp. 6092-6102, 2020.

Bagaa, Miloud, Tarik Taleb, Jorge Bernal Bernabe, and Antonio Skarmeta, "A machine learning security framework for iot systems," IEEE Access, vol.8, pp.114066-114077, 2020.

Fatani, Abdulaziz, Mohamed Abd Elaziz, Abdelghani Dahou, Mohammed AA Al-Qaness, and Songfeng Lu, "IoT intrusion detection system using deep learning and enhanced transient search optimization," IEEE Access, vol. 9, pp. 123448-123464, 2021.

Geladi, Paul, and Johan Linderholm, "Principal component analysis," pp. 17-37, 2020.

Ghillani, Diptiban, "Deep learning and artificial intelligence framework to improve the cyber security," Authorea Preprints, 2022.

Ibitoye, Olakunle, Omair Shafiq, and Ashraf Matrawy, "Analyzing adversarial attacks against deep learning for intrusion detection in IoT networks," In 2019 IEEE global communications conference (GLOBECOM), pp. 1-6. IEEE, 2019.

Khairandish, Mohammad Omid, Meenakshi Sharma, Vishal Jain, Jyotir Moy Chatterjee, and N. Z. Jhanjhi, "A hybrid CNN-SVM threshold segmentation approach for tumor detection and classification of MRI brain images," Irbm, vol. 43, no. 4, pp. 290-299, 2022.

Kumar, Avinash, Sobhangi Sarkar, and Chittaranjan Pradhan, "Malaria disease detection using cnn technique with sgd, rmsprop and adam optimizers, " Deep learning techniques for biomedical and health informatics, pp. 211-230, 2020.

Li, Yuxi, Yue Zuo, Houbing Song, and Zhihan Lv, "Deep learning in security of internet of things," IEEE Internet of Things Journal, vol. 9, no. 22, pp. 22133-22146, 2021.

Lin, Tao, "Deep learning for IoT," In 2020 IEEE 39th International Performance Computing and Communications Conference (IPCCC), pp. 1-4. IEEE, 2020.

Makkar, Aaisha, Sahil Garg, Neeraj Kumar, M. Shamim Hossain, Ahmed Ghoneim, and Mubarak Alrashoud, "An efficient spam detection technique for IoT devices using machine learning," IEEE Transactions on Industrial Informatics, vol. 17, no. 2, pp. 903-912, 2020.

Mukhtar, Naila, Ali Mehrabi, Yinan Kong, and Ashiq Anjum, "Edge enhanced deep learning system for IoT edge device security analytics," Concurrency and Computation: Practice and Experience, vol. 35, no. 13, pp. e6764, 2023.

Saba, Tanzila, Amjad Rehman, Tariq Sadad, Hoshang Kolivand, and Saeed Ali Bahaj, "Anomaly-based intrusion detection system for IoT networks through deep learning model," Computers and Electrical Engineering, vol. 99, pp.107810, 2022.

Sarker, Iqbal H., Asif Irshad Khan, Yoosef B. Abushark, and Fawaz Alsolami, "Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions," Mobile Networks and Applications, pp. 1-17, 2022.

Sarker, Iqbal H, "CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks," Internet of Things, vol. 14, pp. 100393, 2021.

Sharma, Parjanay, Siddhant Jain, Shashank Gupta, and Vinay Chamola, "Role of machine learning and deep learning in securing 5G-driven industrial IoT applications," Ad Hoc Networks, vol. 123, pp. 102685, 2021.

Tran, Minh-Quang, Mahmoud Elsisi, Meng-Kun Liu, Viet Q. Vu, Karar Mahmoud, Mohamed MF Darwish, Almoataz Y. Abdelaziz, and Matti Lehtonen, "Reliable deep learning and IoT-based monitoring system for secure computer numerical control machines against cyber-attacks with experimental verification," IEEE Access, vol.10, pp. 23186-23197, 2022.

Ullah, Imtiaz, and Qusay H. Mahmoud, "Design and development of a deep learning-based model for anomaly detection in IoT networks," IEEE Access, vol. 9, pp. 103906-103926, 2021.

Uprety, Aashma, and Danda B. Rawat, "Reinforcement learning for iot security: A comprehensive survey," IEEE Internet of Things Journal, vol. 8, no. 11, pp. 8693-8706, 2020.

Yang A, Yang X, Wu W, Liu H, Zhuansun Y, "Research on feature extraction of tumor image based on convolutional neural network," IEEE Access, vol. 7, pp. 24204–13, 2019.

Zhou, Xiaokang, Xuesong Xu, Wei Liang, Zhi Zeng, and Zheng Yan, "Deep-learning-enhanced multitarget detection for end–edge–cloud surveillance in smart IoT," IEEE Internet of Things Journal, vol. 8, no. 16, pp. 12588-12596, 2021.