IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 37, No. 5, October, 2023, pp. 1121-1131
Journal homepage: http://iieta.org/journals/ria

A Novel CNN Model with Dimensionality Reduction for WSN Intrusion Detection

Ruqaya Abed Alhasan*, Ekhlas Kadhum Hamza

Department of Computer Engineering, University of Technology, Baghdad 10066, Iraq

Corresponding Author Email: Ekhlas.K.Hamza@uotechnology.edu.iq

https://doi.org/10.18280/ria.370504

Received: 24 April 2023
Revised: 20 May 2023
Accepted: 25 May 2023
Available online: 31 October 2023

Keywords:
wireless sensor network, deep learning, convolutional neural networks, intrusion detection, dimensionality reduction

ABSTRACT

Wireless sensor networks (WSNs) play a critical role in cyber-physical systems, enabling communication between autonomous sensors. When integrated with the Internet of Things (IoT), WSNs unfortunately become vulnerable to various attacks, such as blackhole, grayhole, flooding, and scheduling, thereby posing significant security threats. Existing methods for intrusion detection in WSNs often suffer from low detection rates, significant computational overhead, and false alarms, primarily due to resource constraints and data correlations. This study introduces IDS-CNN, a novel intrusion detection method leveraging Convolutional Neural Networks (CNNs). The proposed IDS-CNN model, designed to optimize efficiency and reduce processing time, comprises nine convolutional neural network layers and six Max-Pooling1D layers. To alleviate computational demands, dimensionality reduction techniques, specifically Principal Component Analysis and Singular Value Decomposition, are applied to raw traffic data. The IDS-CNN model is then employed to classify and categorize network threats. Experimental evaluations suggest that the IDS-CNN approach yields a high accuracy rate of 99% compared to existing methods, based on tests performed on two datasets, WSN-DS and UNSW-NB15. Notably, with the UNSW-NB15 dataset, accuracy rates were further improved to 99.99% and 100%. By leveraging deep learning techniques to enhance intrusion detection in WSNs, this study presents a significant contribution to the field. The IDS-CNN model advances our understanding of WSN security by exceeding the accuracy rates of prior models. As it addresses the limitations of existing methods, the implications of this research are substantial, offering a more reliable and efficient solution for WSN intrusion detection. The findings underscore the potential of IDS-CNN in safeguarding WSNs and IoT systems from sophisticated and evolving cyber threats.

1. INTRODUCTION

The burgeoning development of the Internet of Things (IoT) has led to a marked increase in the prevalence of wireless sensor networks (WSNs). These networks serve a crucial function, connecting a variety of devices and facilitating a wide range of applications that enrich daily life [1, 2]. However, with the expansion of WSN deployment, security assurance has emerged as a pressing concern. Existing security measures, such as encryption and authentication, have been integrated into WSNs to enhance their protection. Despite these precautions, sophisticated attack methods have been devised that can circumvent traditional security measures. The task of preserving sensitive information within the framework of large-scale WSN deployments demands the implementation of more robust and advanced security strategies [3].

Passive defense approaches alone are insufficient to provide foolproof protection for WSNs. Active defense technologies, such as intrusion detection systems (IDS), are essential for proactive threat detection [4, 5]. IDSs driven by data analysis have demonstrated the ability to detect attacks even when traditional safeguards are lacking [6]. However, the increasing volume of data transmitted over WSNs poses significant challenges for real-time analysis by IDSs [7]. Additionally, distinguishing between normal and abnormal network traffic in WSNs is complicated by the presence of attacks such as wormholes, sinkholes, flooding, and jamming, which disrupt the typical network behavior [8]. The sheer magnitude of network traffic data slows down classifiers and introduces difficulties in detecting suspicious behavior due to noise and irrelevant features, further impeding effective investigation and decreasing the likelihood of successful detection [9, 10].

Machine learning approaches, while promising, are not without limitations. The performance of machine learning models is heavily reliant on the quality of the data and features used in the algorithms. Thus, feature selection plays a crucial role in optimizing the effectiveness of machine learning algorithms. As computers continue to permeate various aspects of human life, datasets with high-dimensional feature spaces have become commonplace. However, to accurately represent the underlying essence, it is necessary to focus on a selected set of relevant features. Deep learning algorithms, in particular, suffer from performance degradation when confronted with a multitude of irrelevant and duplicate features. To address this issue, researchers have successfully integrated feature selection techniques with machine learning, finding widespread application in domains such as network traffic monitoring and security [11-13].

The unique characteristics of WSNs necessitate specialized intrusion detection approaches. Traditional methods used in computer networks are ill-suited to safeguard WSNs due to disparities in terminal types, data transmission, network

topology, and other factors [14]. An effective WSN IDS must possess high detection efficacy for both known and unknown threats while imposing minimal impact on the WSN infrastructure [15]. This research introduces a novel approach to intrusion detection in WSNs, aiming to bridge the identified gaps. The key contributions of this study include:

1. Analysis of feature selection techniques, such as principal component analysis (PCA) and singular value decomposition (SVD), to address the computational complexity and detection performance limitations arising from the vast volume and diversity of data processed in WSNs.

2. Utilization of a convolutional neural network (CNN)-based algorithm for intrusion detection, leveraging a 9-layer design with 6-Max-Pooling1D. This design optimizes resource efficiency and reduces time complexity.

3. Development of the IDS-CNN model, specifically tailored to detect traffic attacks in WSNs with a focus on minimizing false positives. This model overcomes the limitations of conventional WSN intrusion detection techniques, offering improved detection performance, real-time capabilities, reduced complexity, and resource requirements.

The remaining sections of this paper are organized as follows: Section 2 provides an overview of relevant studies in the field. Section 3 discusses the application of deep learning in WSNs. Section 4 presents the proposed approach for protecting WSNs from intrusion. Experimental environments are detailed in Section 5, followed by the presentation and interpretation of results in Section 6. Finally, Section 7 outlines future objectives for further research.

2. RELATED WORK

With the proliferation of wireless LANs [16], notably Ad Hoc networks and wireless sensor networks, traditional wired network intrusion detection system (IDS) solutions are incompatible. This highlights the critical necessity for an intrusion detection system for wireless sensor networks. Intruder detection systems that use anomaly detection will look at any suspicious behaviour [17]. Researchers have used these findings to construct a variety of powerful anomaly detection systems, most of which are variants on artificial immunity algorithms, clustering algorithms, machine learning algorithms, and statistical learning models. To identify anomalies in the NSL-KDD dataset, Liu et al. [18] used an EM approach to Expectation Maximization. In this paper, we looked at several distinct kinds of attacks, including Synflood, land, ping of death, sweeping, and UDP flood. To achieve smart, sustainable energy management, Hemanand et al. [19] suggested applying the existing Glow worm Swarm Optimization technique across IoT sensors to detect the devices in need of energy and distribute appropriate energy on a need basis. According to Jayalakshmi et al. [20], the routing protocol should be one of the factors examined when gauging a network's efficacy. It was proposed by Gopalakrishnan et al. that the security of the system may be enhanced by deploying highly secured cryptographic algorithms on each node in the network.

MQTTset, presented by Vaccari et al. [21], is a dataset dedicated to the MQTT protocol, which is commonly used in IoT networks. By combining the official dataset with cyberattacks against the MQTT network, we show the creation of the dataset and validate it through the definition of a hypothetical detection system. The obtained results show how machine learning models may be trained using MQTTset to create detection systems that can secure IoT environments. A innovative misuse-based intrusion detection system is proposed by Kumar et al. [22], which can identify five types of attacks in a network: exploit, DOS, probe, generic, and normal. In addition, the KDD99 or NSL-KDD 99 data set is used in the majority of the works that are similar to IDS. When it comes to detecting modern threats, these data sets are now regarded useless and antiquated. In this paper, we use the UNSW-NB15 dataset as an offline resource for developing our own integrated classification-based algorithm for sniffing out cybercrime. As shown by the research carried out by Chandre et al. [23], it is possible to predict how successful an attack will be against an IoT system that is based on MQTT by employing one of a variety of machine learning models. We used the precision, accuracy, and F1 score as evaluation criteria to make a direct comparison between the models' levels of effectiveness. The findings showed that the performance of random forest was extremely accurate, with a degree of certainty that was equivalent to 96 percent.

The study provided by Hemanand et al. [24] creates a smart IDS using the Cuckoo Search Greedy Optimization (CSGO) and the Likelihood Support Vector Machine (LSVM) models to improve WSN security. Some of the most popular network datasets, such as NSL-KDD and UNSW-NB15, are used to validate this model. The first step in normalizing the attributes is to do dataset pre-processing, which involves removing any unnecessary data, making educated guesses about the values that are absent, and applying any necessary filters. The CSGO algorithm needs to be given the optimal number of features, which was calculated during the pre-processing stage, for it to be able to select the best possible features. The very last step is to predict whether the label should be considered normal or abnormal by utilizing a machine-learning classification algorithm that is based on the linear support vector machine (LSVM). During the process of evaluating the findings, a multitude of performance measurements is utilized to verify and assess the efficacy of the suggested security model.

The effectiveness of an attack on a MQTT-based IoT system may be predicted using a number of different machine learning models, as demonstrated by the work of Makhija et al. [25]. To compare the efficacy of the models, we employed the precision, accuracy, and F1 score as evaluation criteria. Results demonstrated that random forest's performance was very accurate, with a 96 percent degree of certainty. Using the Cuckoo Search Greedy Optimization (CSGO) and Likelihood Support Vector Machine (LSVM) models, Hemanand et al. [24] proposed work creates an intelligent IDS system for improving WSN security. This model takes into account the most popular network datasets for validation, including NSL-KDD and UNSW-NB15. At first, the attributes are normalized via dataset pre-processing via the elimination of extraneous data, the prediction of missing values, and the application of filters. In order to pick the optimum features, the CSGO algorithm must be fed the optimal number of features that were determined during pre-processing. The final step is to forecast the categorized label as normal or abnormal using a machine learning classification technique based on the linear support vector machine (LSVM). During the results evaluation process, many performance measurements are used to verify and compare the effectiveness of the suggested security model.

Feature selection in IDS using a hybrid optimization approach was proposed by Alkanhel et al. [26]. The suggested

approach, known as GWDTO, is inspired by the grey wolf (GW) and dipper throated optimization (DTO) algorithms. The suggested technique may be more effective because it strikes a better balance between the optimization process' exploration and exploitation phases. The suggested GWDTO algorithm's performance was measured against a set of evaluation metrics and compared to other optimization strategies in the literature on the used IoT-IDS dataset to prove its efficacy. Furthermore, a statistical analysis is carried out to evaluate the reliability and efficiency of the method.

Edwin Singh and Celestin Vigila [27] used WOA (Whale Optimization Algorithm) and DNN (Deep Neural Network) to optimize the pre-processed data to build a system for detecting and classifying incursions in MANET (Whale Optimized Deep Neural Network Model), with the goal of predicting unanticipated cyber-attacks. As a result, intruder-proof data transfer to other nodes is made possible. Using a combination of ML-IDS and WOA-DNN, we can identify the intruders. Principal Component Analysis (PCA) reduces the dimensionality of the data, leading to more precise results. Forward propagation uses a classifier to determine the likely safety or danger of a result. Classification accuracy, attack detection rate, precision rate, and F-Measure, Recall are used to evaluate both the conventional and new models. The suggested WOA-DNN model achieves an accuracy rate of 99.1 percent and improved assessment metrics. WOA-DNN has a higher attack detection rate than competing methods, which translates to fewer false alarms. The suggested WOA-DNN model achieves a 99.1 percent accuracy rate in its classifications.

To mitigate the wide-ranging effects of denial-of-service (DoS) attacks while keeping energy consumption to a minimum, Feature selection models for NIDSs are proposed by Almomani [28]. Particle swarm optimization (PSO), grey wolf optimization (GWO), firefly algorithm (FFA), and the genetic algorithm (GA) all form the basis of this concept (GA). The proposed model is made with the intention of enhancing NIDS functionality. The suggested model uses Anaconda Python Open Source's wrapper-based methods with the GA, PSO, GWO, and FFA algorithms to pick features, as well as filtering-based methods for the mutual information (MI) of the GA, PSO, GWO, and FFA algorithms, which yielded 13 sets of rules. Support vector machine (SVM) and J48 ML classifiers are used on the UNSW-NB15 dataset to assess the proposed model's output characteristics.

Table 1. Review on existing strategies

Research	Year	Dataset	Algorithm	Attacks Executed	Accuracy	Precision	Recall	F1 Score
Liu et al. [18]	2020	NSL-KDD dataset	RF DT Bagging SVM NB BN AdaBoost XGBoost	Syn Flood Land UDP Flood Ping of Death (PoD) Smurf IP Sweeping Port Scan	0.966 0.996 0.967 0.957 0.452 0.882 0.740 0.970	0.969 0.969 0.969 0.948 0.904 0.944 0.663 0.970	0.967 0.967 0.967 0.957 0.452 0.882 0.740 0.968	0.968 0.968 0.968 0.951 0.545 0.902 0.646 0.968
Vaccari et al. [21]	2020	MQTTset Message (Queue Telemetry) Transport	Neural network, random forest, Naïve Bayes, Decision tree, Gradient boost, Multilayer perceptron	flooding denial of service, MQTT Publish Flood, SlowITe malformed Data, brute force authentication	0.993268 0.994299 0.987903 0.977972 0.991131 0.94688	N/A	N/A	0.9932 0.9943 0.9897 0.9850 0.9916 0.9636
Kumar et al. [22]	2020	UNSW-NB15 and real time data set at NIT Patna CSE lab (RTNITP18)	Different decision tree models (C5, CHAID, CART, QUEST) are trained with selected 13 features of the dataset	Exploit, DOS, Probe, Generic and Normal	high	69.9 50.37 99.21 99.7 81.17	54.6 5 71.7 96.7 98	high
Almomani [28]	2020	UNSW-NB15 dataset.	the support vector machine (SVM) and J48 ML classifiers	----------	90.119 90.484	N/A	N/A	N/A
Chandre et al. [23]	2021	WSN-DS	CNN	Denial of Service (DoS), Black hole, Gray hole, Flooding and TDMA	97	99	99	99
Makhija et al. [25]	2022	MQTTset (Message Queue Telemetry Transport)	RF, KNN, and SVM classifier	unauthorized access, denial of service, packet sniffing, and malware injection	96	N/A	N/A	N/A
Hemanand et al. [24]	2022	NSL-KDD and UNSW-NB15	Cuckoo Search Greedy Optimization (CSGO) and Likelihood Support Vector Machine (LSVM)	Probe DoS R2L U2R	99.56	93.56 99.99 58.96 96.47	98.45 98.69 29.45 64.36	95.56 99.5 43.21 59.65
Alkanhel et al. [26]	2023	RPL-NIDDS17	grey wolf (GW), and dipper throated optimization (DTO)	Sybil, blackhole, sinkhole, and clone	1.18 average error	N/A	N/A	N/A
Edwin Singh and Celestin Vigila [27]	2023	NSL-KDD	WOA-DNN	Probe DoS R2L U2R	99.01	99	99.05	98.1

The assessment criteria and experimental designs used by the reference techniques vary considerably from one another. To better understand the significance of the data shown in Table 1, it would be helpful to have a more in-depth

explanation of these variances. The observed performance differences across the baseline approaches are heavily influenced by factors like as the features used, the datasets utilised, and the types of assaults evaluated. Because of these variations, it is essential that they be taken into account throughout the planning stages of the proposed method's development.

In their analysis of the NSL-KDD dataset, Liu et al. [18] found that different methods had diverse degrees of accuracy, precision, recall, and F1 score for various attack types. This indicates that the efficacy of intrusion detection systems is highly dependent on the algorithm selected and how well it handles various types of attacks. Vaccari et al. [21] used a variety of techniques to identify DDoS, MQTT Publish Flood, SlowITe, and malformed data assaults. These algorithms' results in terms of accuracy and precision vary widely, demonstrating the need to use the right algorithms for the right kinds of attacks.

Using attributes hand-picked from the UNSW-NB15 dataset, Kumar et al. [22] used several decision tree algorithms. The models' efficiency ranged depending on the type of assault, with some showing better specificity and recall than others. This demonstrates how feature selection and model selection are crucial for effective intrusion detection. When working with the NSL-KDD and UNSW-NB15 datasets, Hemanand et al. [24] utilized the Cuckoo Search Greedy Optimization (CSGO) and Likelihood Support Vector Machine (LSVM) techniques. The results demonstrated discrepancies in detection efficiency across attacks of various sorts, further evidencing the impact of dataset features on the efficacy of intrusion detection methods.

Noting that the reported findings show the performance of existing solutions, they provide a solid basis for the suggested approach. The results are discussed to provide light on the reasoning behind the suggested method's design decisions. Researchers may make educated selections about what to focus on while creating new intrusion detection techniques if they have a firm grasp of the strengths and weaknesses of the existing systems.

In conclusion, the suggested technique benefits greatly from a thorough description of the experimental settings, assessment criteria, and variances in performance among the baseline methods. The suggested technique aims to overcome the limits of existing tactics by drawing attention to the elements causing performance variations and taking into account the unique needs of the datasets and attack scenarios, hence improving the intrusion detection capabilities.

3. CONVOLUTIONAL NEURAL NETWORKS

This problem could be addressed with the use of Convolutional Neural Networks (CNN), which offer an automatic and accurate means of detecting network irregularities. Common applications of convolutional neural networks (CNNs) in computer vision include object recognition and classification in images. CNNs are ideally suited for IDS because they eliminate the requirement for human-engineered feature extraction and can learn complex properties automatically from raw network traffic data. CNNs' capacity to capture geographical and temporal relationships in network traffic data is invaluable for detecting sophisticated attacks that evolve over time [25].

As can be seen in Figure 1, a typical CNN architecture for IDS often consists of several layers of convolutional, pooling, and fully linked networks. The CNN's first layer employs convolution methods to help extract elementary features like edges and corners from the input data. Successive layers perform increasingly complex convolution techniques to extract higher-level features like shapes and textures. By using pooling layers to reduce the dimensionality of the feature maps, computation time can be reduced. The collected attributes are then classified as "normal" or "abnormal" network activity based on the presence of completely linked layers. Most intrusion detection system convolutional neural networks (IDS CNNs) are taught through supervised learning, where the network is taught using a labelled dataset of typical and abnormal network traffic. Following training, the network is tested on an independent dataset to gauge its ability to detect outliers. Improve the CNN's ability to spot network abnormalities by expanding the size and diversity of the training dataset, tweaking hyperparameters like learning rate and regularization, and using data augmentation techniques to add variety to the training data [26, 27].

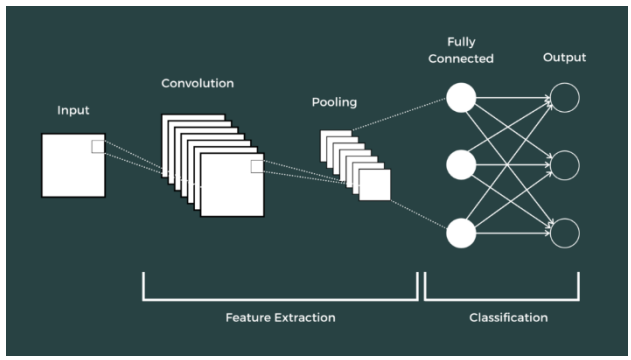

Figure 1. Convolutional neural network [28]

Using CNNs in IDS has many advantages, one of which is the detection of previously unknown threats. Because, unlike more conventional methods, CNNs may be taught to identify specific attack patterns in network traffic data. With the aid of specialized training datasets, CNNs can be educated to identify many forms of cyberattacks, including Distributed Denial of Service (DDoS) attacks and SQL injection attacks. Finally, Convolutional Neural Networks offer a promising solution to the problem of detecting advanced, until unknown attacks in IT systems. Due to their ability to automatically learn complex features from raw network traffic data, CNNs are an essential aspect of network security in today's complex and interconnected world.

4. THE PROPOSED RESEARCH METHODOLOGY

Many researchers, seeking superior IDS performance in WSN, have turned to ever-more-intricate data mining strategies. However, real-time implementation of such algorithms in wireless sensor networks is impractical due to their high processing overhead. Large feature dimensions in the input data, an abundance of redundant data, and insufficient data preparation all contribute to the high computational cost of IDS.

Feature selection is a technique for reducing the number of potential features from a big pool to a more manageable one. Given the significance of the data, pre-processing techniques like Principal component analysis (PCA) and Singular Value Decomposition (SVD) are used to improve the detection accuracy of the classification algorithm, reduce the

computational load of IDS, and preserve as much information as possible. The data analysis module depicted in Figure 1 can receive event information and assess it to determine if the observed behaviour constitutes an intrusion using this method. In Figure 2 we can see the overall structure of the algorithms. Figure 2 shows the recommended architecture for the system based on this study. The first part of this text is devoted to data processing in its various forms.

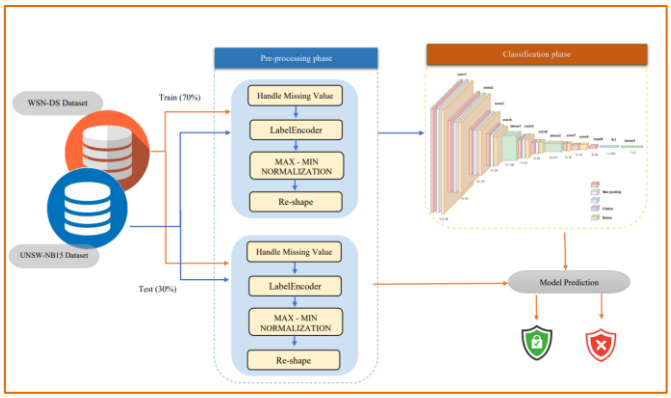

Figure 2. Specification of the algorithm framework for WSN-DS and UNSW-NB15 dataset

The term "data engineering process" is often used to describe this approach. This is a crucial step in the learning process. There are three phases of data processing: cleaning, normalization, and feature selection. The most important features are selected using a filter-based method inspired by principal component analysis and singular value decomposition. After the required feature vector has been selected, the training set is used to train the model. A trained model can then be validated using data from the validation set. Finally, the validated model is used to analyze data from the test dataset.

4.1 The Preparation of data

(1) Collecting and mapping information

The label characteristic of the sample data is a string of letters; to get rid of it from the algorithm, we need to convert those letters into integers. The Attack classification includes the five different forms of data (Normal, Blackhole, Grayhole, Flooding, and Time Division Multiple Access). Due to the incalculability of this data, it is organized non a sequential sequence using the ordinal digits 0, 1, 2, 3, and 4. Adjustment in light of Table 2.

Table 2. Attack-type-characteristic-value conversion table

Original Eigenvalue	Transformed Eigenvalue
Normal	0
Grayhole	1
Blackhole	2
TDMA	3
Flooding	4

(2) LabelEncoder

Categories' nominal and ordinal feature labels are represented as Strings. Some labels might have ordered information (ordinal qualities) while others might not (nominal features). Labels must be encoded as numbers during data pre-processing to increase the likelihood that the learning algorithm will correctly interpret the features. LabelEncoder's encoding method assigns numbers to labels.

(3) Maximum and minimum normalization

Since the continuous data ranges from less than 1 characteristic to hundreds of thousands, normalization is necessary for numerous classification techniques. Here, we employ the outliers of Eq. (1) to provide a baseline for comparison. Where x_j is the original feature data, Minj is the minimum value for the feature, M_{axj} is the maximum value for the feature, and x_j is the normalized feature data.

$$x_j^* = \frac{x_j - Min_j}{Max_j - Min_j} \quad (1)$$

4.2 Features extraction

(1) Principal component analysis

Principal component analysis (PCA) is widely applied to the problem of discovering patterns in high-dimensional data. The goal of PCA is to represent both recognized and unknown faces using a smaller number of typical feature photographs (called Eigenobject). PCA has been shown to be useful for detecting and validating facial features, as shown by statistical data. The PCA technique requires transforming a two-dimensional matrix of face images into a one-dimensional vector. A one-dimensional vector can be orientated in either the row or the column without affecting its value [22, 23].

(2) Singular value decomposition

Singular value decomposition (SVD) is another technique for data partitioning. It's used for things like feature extraction, matrix approximation, and pattern recognition in signal processing and statistics. When applied to a single signal, PCA fails to extract features, and when applied to a signal with varying frequencies, PCA fails to give information about the features existing in the signal. For feature extraction, SVD can be more useful than principal component analysis due to the fact that frequency differences may mask genuine differences across physiological states [29-31].

4.3 Classification model

Using information gathered from wireless sensor networks and filtered with a sequence backward feature selection strategy, an intrusion can be detected using the IDS-ML classification method. IDS-ML is a rapid, distributed, high-performance gradient boosting system built on gradient-based approaches [32]. IDS-ML is built around a variant of the histogram method that significantly reduces the feature and sample sizes required during training. An Intrusion Detection System based on Convolutional Neural Networks allows for real-time detection of network intrusions (IDS). By examining labeled data, the IDS may be trained to tell the difference between typical and malicious network activity.

In order to efficiently analyze and extract characteristics from input data, the suggested Convolutional Neural Network (CNN) architecture comprises of many layers. Intricate data patterns may be captured and learned by these layers, allowing the model to make precise predictions. The function of each stage is described in the next paragraph:

A Convolutional layer with 16 filters, a kernel size of 3, and a stride of 1 is the foundation of the Convolutional Neural Network (CNN) model. The input data is a 15-by-1 matrix, and this layer attempts to extract neighborhood-level patterns and characteristics. Next, a MaxPooling layer applies

downsampling to the feature maps in order to make them more manageable. The non-linear LeakyReLU activation function is used to improve the model's capability of learning intricate representations. The next levels are more Convolutional layers, each with 32, 64, and 64 filters. To further extract and improve the learnt features, a MaxPooling layer and the LeakyReLU activation function are added to each layer after the learner. The goal of these layers is to provide the model with the building blocks it needs to extract more complicated representations of the input.

After the Convolutional layers is a Dense layer with 128 units and a linear activation function. The retrieved features are aggregated and made ready for further processing with the aid of this layer. Two more Convolutional layers, each with 32 filters and the same padding strategy to preserve spatial dimensions, are added to the model. MaxPooling and LeakyReLU activation, which follow these layers, also aid in feature extraction.

To further improve the model's capability of capturing complicated patterns and correlations in the data, a Dense layer with 512 units and a linear activation function is added. The LeakyReLU activation function is used to two extra Convolutional layers of 16 filters each. These additional layers aid in the extraction of more nuanced information. After the multidimensional feature maps are flattened using the Flatten layer, a Convolutional layer is applied with 35 filters, a kernel size of 3, and a stride of 1. Class probabilities may be predicted with the help of the model's last layer, a Dense layer with 2 units and a softmax activation function.

In conclusion, the proposed CNN architecture uses Convolutional, MaxPooling, Dense, and activation layers to efficiently extract features and train representations from the input data, allowing for accurate classification or prediction in the context of the task at hand. According to Table 3, the proposed CNN design employs 47 342 parameters.

Table 3. Proposed CNN layers parameters

Layer (Type)	Output Shape	Param #
conv1d-1 (Conv1D)	(None, 13, 16)	64
maxpooling1d-1 (MaxPooling1)	(None, 13, 16)	0
leakyrelu-1 (LeakyReLU)	(None, 13, 16)	0
conv1d-2 (Conv1D)	(None, 11, 32)	1568
maxpooling1d-2 (MaxPooling1)	(None, 11, 32)	0
leakyrelu-2 (LeakyReLU)	(None, 11, 32)	0
conv1d-3 (Conv1D)	(None, 9, 64)	6208
maxpooling1d-3 (MaxPooling1)	(None, 9, 64)	0
leakyrelu-3 (LeakyReLU)	(None, 9, 64)	0
conv1d-4 (Conv1D)	(None, 7, 64)	12352
maxpooling1d-4 (MaxPooling1)	(None, 7, 64)	0
leakyrelu-4 (LeakyReLU)	(None, 7, 64)	0
dense-1 (Dense)	(None, 7, 128)	8320
conv1d-5 (Conv1D)	(None, 7, 32)	12320
maxpooling1d-5 (MaxPooling1)	(None, 7, 32)	0
leakyrelu-5 (LeakyReLU)	(None, 7, 32)	0
conv1d-6 (Conv1D)	(None, 7, 32)	3104
maxpooling1d-6 (MaxPooling1)	(None, 7, 32)	0
leakyrelu-6 (LeakyReLU)	(None, 7, 32)	0
dense-2 (Dense)	(None, 7, 512)	16896
conv1d-7 (Conv1D)	(None, 7, 16)	24592
leakyrelu-7 (LeakyReLU)	(None, 7, 16)	0
conv1d-8 (Conv1D)	(None, 7, 16)	784
leakyrelu-8 (LeakyReLU)	(None, 7, 16)	0
conv1d-9 (Conv1D)	(None, 7, 35)	1715
flatten-1 (Flatten)	(None, 245)	0
dense-3 (Dense)	(None, 2)	492

The suggested CNN architecture can be trained using labelled network traffic data to classify traffic flows as normal or pathological. The testing phase involves feeding the network traffic into a CNN and comparing the results to a threshold in order to determine whether or not the traffic is normal.

5. EXPERIMENTAL SETTINGS

Here, the publicly available WSN-DS dataset was employed [33] for the experiment. created specifically for use with WSNs, this dataset contains information used to detect intrusions (WSN). Blackhole, Grayhole, flooding, and scheduling are the four forms of DoS assaults seen in WSN-DS. Table 2 contains the comprehensive statistical data. Of the total number of samples in both the training and testing sets, 224796 (or 70%) were drawn at random from the former, and 149865 (or 30%) from the latter. The experiments were also conducted with data from the UNSW-NB15 assaults dataset [34]. As may be shown in Table 3, the 42 elements present in the UNSW-uncluttered NB15's design. There are a total of 42 features, of which only three are not numerical in nature (categorical features).

But the confusion matrix (CM) is utilized to evaluate the Accuracy, Recall, Precision, and F-measure of our approach on the dataset. In equations (5) through (8), the true positive (TP) and false negative (TN) counts are balanced by the false positive (FP) and false negative (FN) counts, respectively [35-38]. Precision: the percentage of instances that are accurately classified; Take into account again the percentage of "good" components that were properly assigned to the "good" group; Accuracy: the percentage of false alarms that occur when using a detection model that initially misclassified some components as false positives; The F-Score is the Mean.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F_1 = 2 * \frac{precision * recall}{precision + recall} = \frac{2TP}{2TP + FP + FN} \tag{5}$$

6. RESULT ANALYSIS AND DISCUSSION

In this section, the efficiency of the model that was suggested is evaluated. The researchers carried out a variety of separate studies, including: 1) Comparing IDS-CNN with and without a feature extraction step against machine learning classification methods using PCA or SVD with ten or fifteen characteristics. 2) Analysing the performance of IDS-ML in contrast to the performance of other machine learning classification methods; 3) Analysing the performance of IDS-CNN in light of four distinct measures; the accuracy and recall rates of the detection system must be high enough for usage on the network and eventual integration with the intrusion detection system used on the network. Improved classification technique, including WSN-DS and UNSW-NB15 dataset without feature selection phase, are compared in Table 4.

Table 4. Comparison of multiple classification measurements on WSN-DS and UNSW-NB15 dataset without feature selection phase

Dataset	Measures			
	Accuracy	Precision	Recall	F1-Measure
WSN-DS	0.9807	0.9832	0.9807	0.9807
UNSW-NB15	0.9289	1	0.9348	0.9663

For better estimator accuracy or better performance on very high-dimensional datasets, the feature selection module can be used to perform feature selection/dimensionality reduction on sample sets. This paper uses two algorithm PCA and SVD with 10 or 15 features. Figures 3 to 6 the results obtained from applying feature selection with ML algorithms on WSN-DS dataset.

Figures 7 to 10 shows the findings that were achieved by applying feature selection using DL algorithms to the UNSW-NB15 dataset via PCA and SVD with either 10 or 15 features.

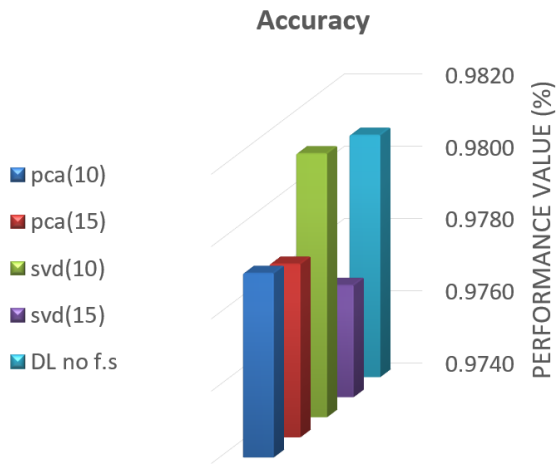

Figure 3. Accuracy comparison of multiple feature selection approaches on WSN-DS

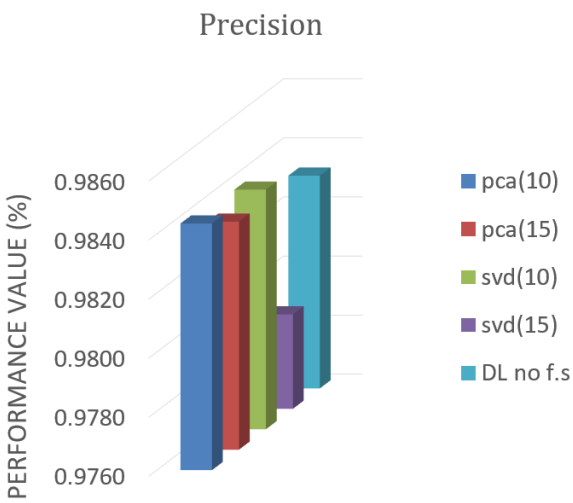

Figure 4. Precision comparison of multiple feature selection approaches on WSN-DS

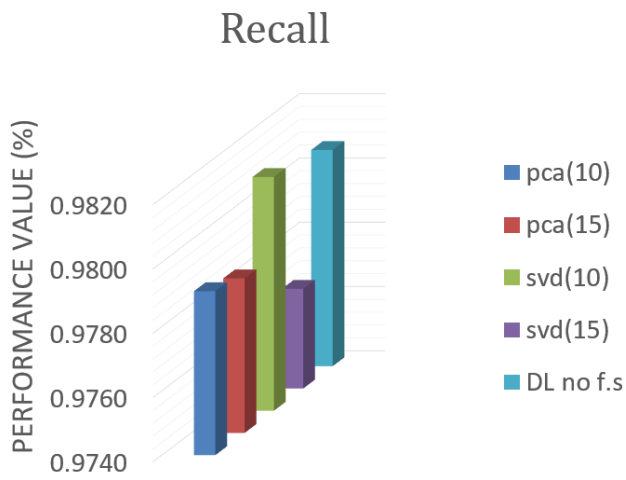

Figure 5. Recall comparison of multiple feature selection approaches on WSN-DS

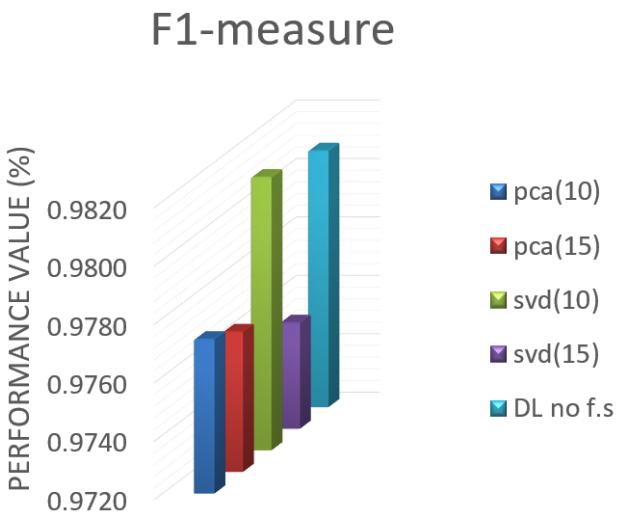

Figure 6. F1-measure comparison of multiple feature selection approaches on WSN-DS

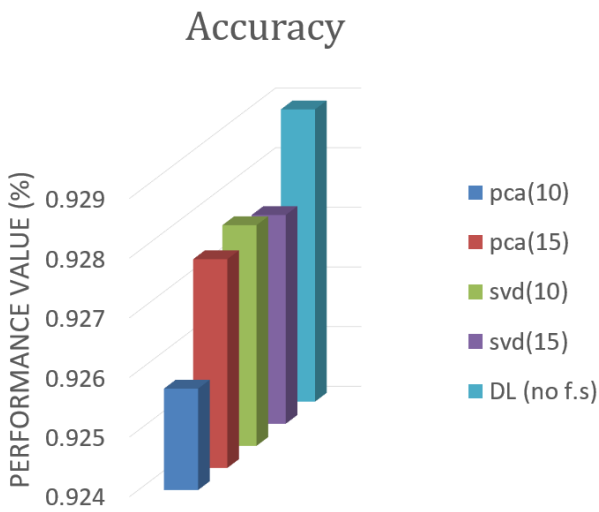

Figure 7. Accuracy comparison of multiple feature selection approaches on UNSW-NB15 dataset

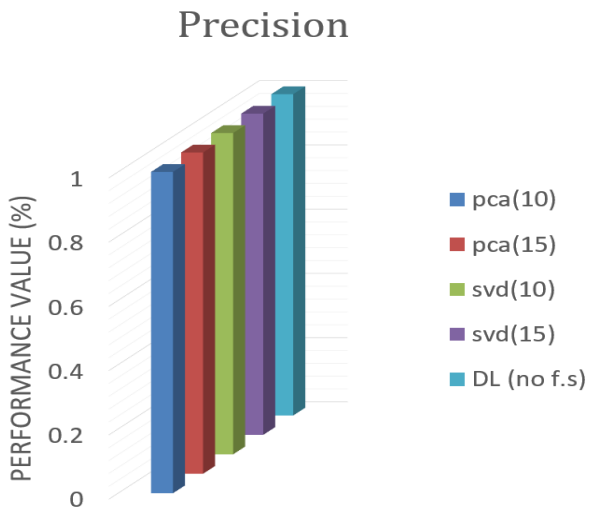

Figure 8. Precision comparison of multiple feature selection approaches on UNSW-NB15 dataset

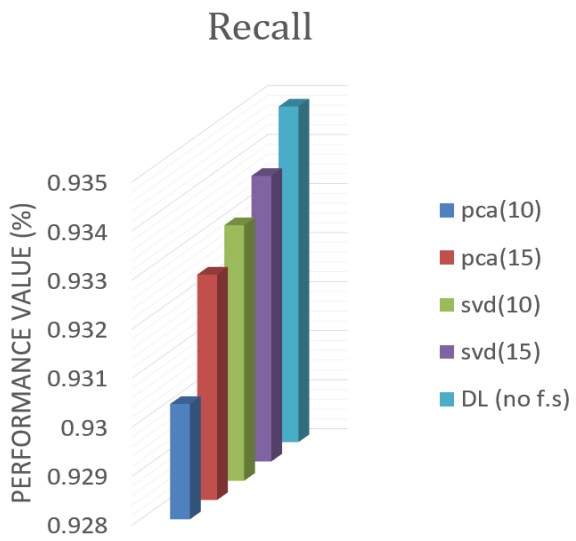

Figure 9. Recall comparison of multiple feature selection approaches on UNSW-NB15 dataset

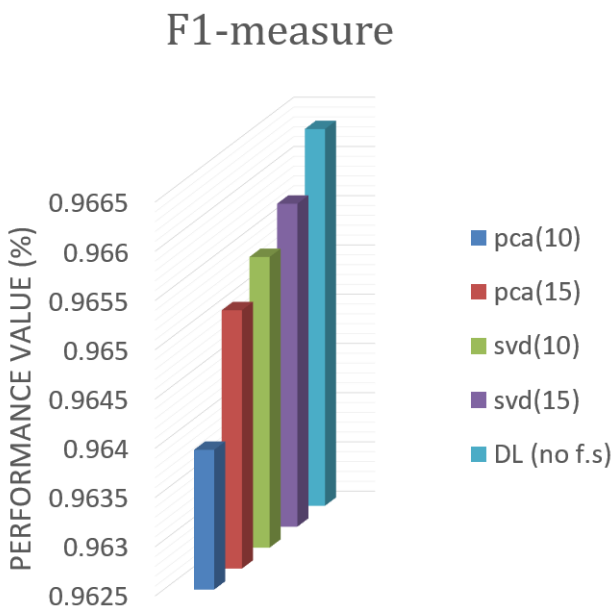

Figure 10. F1-measure comparison of multiple feature selection approaches on UNSW-NB15 dataset

To compare the suggested algorithms to the state-of-the-art, ran them through the IDS-CNN with feature selection methods; the feature ranking for these algorithms is shown in Table 5. The proposed model has greater performance. The outcomes of the experiments are detailed in Table 5, as well as in Figure 11. In order to evaluate the classification performance of our system, and make use of the confusion matrix, abbreviated as CM.

Focusing on the feature extraction method and accuracy, Table 5 compares the IDS-CNN algorithm to various approaches on the UNSW-NB15 dataset. In comparison to the Deep CNN method's 97% accuracy, the IDS-CNN algorithm's results are clearly superior (99.66%) (see table). Data from the UNSW-NB15 dataset shows that the IDS-CNN algorithm is more effective at identifying intrusions than the Deep CNN approach.

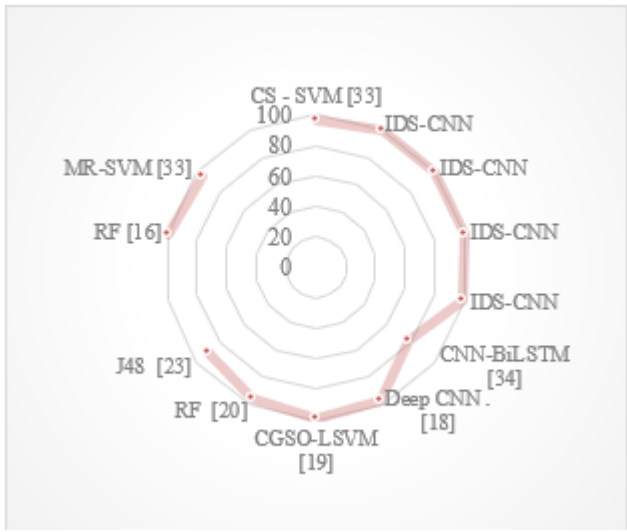

Figure 11. Evaluation of several classification models

Table 5. Comparison of IDS-CNN algorithm and other methods on UNSW-NB15 dataset

Algorithms	Feature Extraction Technique	Accuracy
J48 [28]	N/A	90.484
CGSO-LSVM [24]	CSGO	99.65
CNN-BiLSTM [38]	O-SS-SMOS	77.16
IDS-CNN	PCA10	99.66
IDS-CNN	SVD10	99.86

The IDS-CNN algorithm is compared against other approaches in Table 6 using the WSN-DS dataset. The feature extraction method employed by the Deep CNN approach is unfortunately not specified in the table. The IDS-CNN algorithm, on the other hand, clearly hits an accuracy of 99.26%. The high accuracy attained by the IDS-CNN algorithm supports its efficacy in identifying intrusions in the WSN-DS dataset, however there is currently no way to compare its accuracy to that of the Deep CNN approach.

The overall strengths of the IDS-CNN algorithm are shown by the findings in both tables. The IDS-CNN system successfully detects intrusions across many datasets, with an accuracy of 99.66% on the UNSW-NB15 dataset and 99.26% on the WSN-DS dataset respectively. These results confirm

IDS-CNN's efficiency as an intrusion detection system, demonstrating its promise as a dependable and robust solution for bolstering the safety of wireless sensor networks.

Table 6. Comparison of IDS-CNN algorithm and other methods on WSN-DS dataset

Algorithms	Feature Extraction Technique	Accuracy
Deep CNN [23]	NA	97
IDS-CNN	99.66	99.26
Algorithms	Feature Extraction Technique	Accuracy
Deep CNN [23]	NA	97
IDS-CNN	99.66	99.26

Table 5 contrasts the strategies suggested in this paper with those found in the literature review. According to the findings, when comparing other methods for the feature selection scheme, IDS-CNN performed better than any of the others, and when comparing DL methods for the multiclass configuration, it was the best option. After conducting these studies, the researchers have determined the following:

Tables 4 and 5 show that when compared to the other algorithms tested on the wireless sensor network datasets WSN-DS and UNSW-NB15, the IDS-CNN model obtains higher levels of accuracy and recall. Figures 3 to 10 illustrate how the reduced feature dimension that follows feature selection of the data affects the algorithm's accuracy, F-measure, and other indications. The feature selection algorithm is clearly the winner among the three methods. It is crucial to be able to address feature dependencies and the interplay between feature subset search and model selection when working with WSN-DS and UNSW-NB15 data. It is easy to get rid of certain unnecessary internally dependant qualities because the other three approaches don't consider the classifier's interaction with the data. However, when the data comprising those characteristics is processed as a whole, the discrimination performance of those features is low, despite the fact that the features themselves provide significant potential for discrimination. The wrapper's learning algorithm, which relies on the precision of its predictions, is responsible for weighing the pros and cons of the chosen subset. The ability to employ classifiers in conjunction with feature selection to zero in on a smaller set of traits that will be most useful during the learning process.

7. CONCLUSIONS

When it comes to identifying malicious software, the most often used method is a combination of feature selection algorithms and machine learning approaches. By lowering the number of features and the dimensionality through feature selection, this method not only enhances generalization but also helps reduce the problem of overfitting. It also helps clarify the connection between characteristics and the values they represent. In particular, the IDS-CNN model shows substantial improvements in accuracy over its rivals. IDS-CNN is ideally suited for intrusion detection in WSNs because of its fast-training efficiency, low memory usage, high precision of 100%, and the ability to handle large-scale data processing, all of which are benefits of its use of decision-based learning within a gradient boosting framework.

It is critical, however, to recognize the constraints of the suggested IDS-CNN approach. There will always be situations or varieties of attacks when your chosen approach won't fare as well as it could. To overcome these obstacles and improve the method, more study is required. To further reduce the computing cost of IDS, future research should investigate innovative feature selection strategies to enhance dimensionality reduction and data pretreatment. When combined with other machine learning techniques for WSN intrusion detection, the performance of IDS-CNN can be even better. However, it is crucial to think about whether or not these results are generalizable to other WSN intrusion detection situations, despite the fact that the results acquired from experiments and research done on similar processes illustrate the efficacy of IDS-CNN. To further understand IDS-CNN's resilience and dependability, it would be helpful to evaluate and test it on a wider range of datasets and under varied network circumstances.

Potentially improving real-world WSN security is the real-world relevance of the suggested IDS-CNN approach. IDS-CNN can help improve the safety and reliability of WSN installations by detecting intrusions with a high degree of precision, a low incidence of false alarms, and just a modest amount of processing power. This has significant repercussions in many fields, including as manufacturing process monitoring, environmental sensing, healthcare delivery, and smart infrastructure. The results of this research show promise for improving the security and dependability of the IoT ecosystem by, among other things, protecting sensitive data and maintaining WSN uptime.

REFERENCES

[1] Ali, I., Ahmedy, I., Gani, A., Munir, M.U., Anisi, M.H. (2022). Data collection in studies on Internet of things (IoT), wireless sensor networks (WSNs), and sensor cloud (SC): Similarities and differences. IEEE Access, 10: 33909-33931. https://doi.org/10.1109/ACCESS.2022.3161929

[2] Zhou, M., Wang, Y., Tian, Z., Lian, Y., Wang, Y., Wang, B. (2018). Calibrated data simplification for energy-efficient location sensing in internet of things. IEEE Internet of Things Journal, 6(4): 6125-6133. https://doi.org/10.1109/JIOT.2018.2869671

[3] Salman, K.D., Hamza, E.K. (2021). Visible light fidelity technology: Survey. International Journal of Computers, Communications & Control (IJCCC), 21(2): 1-15. https://doi.org/10.33103/uot.ijccce.21.2.1

[4] Batra, I., Verma, S., Kavita, Alazab, M. (2020). A lightweight IoT-based security framework for inventory automation using wireless sensor network. International Journal of Communication Systems, 33(4): e4228. https://doi.org/10.1002/dac.4228

[5] Osanaiye, O.A., Alfa, A.S., Hancke, G.P. (2018). Denial of service defence for resource availability in wireless sensor networks. IEEE Access, 6: 6975-7004. https://doi.org/10.1109/ACCESS.2018.2793841

[6] Hussein, M.A., Hamza, E.K. (2022). Secure mechanism applied to big data for IIoT by using security event and information management system (SIEM). International Journal of Intelligent Engineering & Systems, 15(6): 667-681. https://doi.org/10.22266/ijies2022.1231.59

[7] Hamza, E.K., Jaafar, S.N. (2022). Nanotechnology application for wireless communication system. In Nanotechnology for Electronic Applications, pp. 115-130. https://doi.org/10.1007/978-981-16-6022-1_6

[8] Park, T., Cho, D., Kim, H. (2018). An effective classification for DoS attacks in wireless sensor networks. In 2018 Tenth international conference on ubiquitous and future networks (ICUFN), Prague, Czech Republic, pp. 689-692. https://doi.org/10.1109/ICUFN.2018.8436999

[9] Selvamani, D., Selvi, V. (2019). A comparative study on the feature selection techniques for intrusion detection system. Asian Journal of Computer Science and Technology, 8(1): 42-47. https://doi.org/10.51983/ajcst-2019.8.1.2120

[10] Li, P., Zhao, W., Liu, Q., Liu, X., Yu, L. (2018). Poisoning machine learning based wireless IDSs via stealing learning model. In Wireless Algorithms, Systems, and Applications: 13th International Conference, WASA 2018, Tianjin, China, pp. 261-273. https://doi.org/10.1007/978-3-319-94268-1_22

[11] Ibraheem, E.K., Hamza, E.K. (2022). Load balancing performance optimization for LI-Fi/Wi-Fi HLR access points using particle swarm optimization and DL algorithms. International Journal of Intelligent Engineering & Systems, 15(6): 364-381. https://doi.org/10.22266/ijies2022.1231.34

[12] Nafea, S., Hamza, E.K. (2020). Path loss optimization in WIMAX network using genetic algorithm. Iraqi Journal of Computers, Communications, Control and Systems Engineering, 20(1): 24-30. https://doi.org/10.33103/uot.ijccce.20.1.3

[13] Hamza, E., Aziez, S., Hummadi, F., Sabri, A.H. (2022). Classifying wireless signal modulation sorting using convolutional neural network. Eastern-European Journal of Enterprise Technologies, 6(9): 120. https://doi.org/10.15587/1729-4061.2022.266801

[14] Wali, S.S., Abdullah, M.N. (2022). Efficient energy for one node and multi-nodes of wireless body area network. International Journal of Electrical and Computer Engineering, 12(1): 914-923. https://doi.org/10.11591/ijece.v12i1.pp914-923

[15] Pandey, S.K. (2019). An anomaly detection technique-based intrusion detection system for wireless sensor network. International Journal of Wireless and Mobile Computing, 17(4): 323-333. https://doi.org/10.1504/IJWMC.2019.103110

[16] Zhou, M., Liu, Y., Wang, Y., Tian, Z. (2019). Anonymous crowdsourcing-based WLAN indoor localization. Digital Communications and Networks, 5(4): 226-236. https://doi.org/10.1016/j.dcan.2019.09.001

[17] Hamza, E., Al-asady, H. (2018). Indoor localization system using wireless sensor network. Iraqi Journal of Computer, Communication, Control and System Engineering, 18(1): 29-38. https://doi.org/10.33103/uot.ijccce.18.1.3

[18] Liu, J., Kantarci, B., Adams, C. (2020). Machine learning-driven intrusion detection for Contiki-NG-based IoT networks exposed to NSL-KDD dataset. In Proceedings of the 2nd ACM workshop on wireless security and machine learning, pp. 25-30. https://doi.org/10.1145/3395352.3402621

[19] Hemanand, D., Jayalakshmi, D.S., Ghosh, U., Balasundaram, A., Vijayakumar, P., Sharma, P.K. (2021). Enabling sustainable energy for smart environment using 5G wireless communication and internet of things. IEEE Wireless Communications, 28(6): 56-61. https://doi.org/10.1109/MWC.013.210015

[20] Jayalakshmi, D.S., Hemanand, D., Kumar, G.M., Rani, M.M. (2021). An efficient route failure detection mechanism with energy efficient routing (EER) protocol in MANET. International Journal of Computer Network & Information Security, 13(2). https://doi.org/10.5815/IJCNIS.2021.02.02

[21] Vaccari, I., Chiola, G., Aiello, M., Mongelli, M., Cambiaso, E. (2020). MQTTset, a new dataset for machine learning techniques on MQTT. Sensors, 20(22): 6578. https://doi.org/10.3390/s20226578

[22] Kumar, V., Sinha, D., Das, A.K., Pandey, S.C., Goswami, R.T. (2020). An integrated rule based intrusion detection system: Analysis on UNSW-NB15 data set and the real time online dataset. Cluster Computing, 23: 1397-1418. https://doi.org/10.1007/s10586-019-03008-x

[23] Chandre, P.R., Mahalle, P.N., Shinde, G.R. (2021). Intrusion prevention framework for WSN using deep CNN. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(6): 3567-3572.

[24] Hemanand, D., Reddy, G.V., Babu, S.S., Balmuri, K.R., Chitra, T., Gopalakrishnan, S. (2022). An intelligent intrusion detection and classification system using CSGO-LSVM model for wireless sensor networks (WSNs). International Journal of Intelligent Systems and Applications in Engineering, 10(3): 285-293.

[25] Makhija, J., Shetty, A.A., Bangera, A. (2022). Classification of attacks on MQTT-based IoT system using machine learning techniques. In International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021, Singapore, pp. 217-224. https://doi.org/10.1007/978-981-16-3071-2_19

[26] Alkanhel, R., El-kenawy, E.S.M., Abdelhamid, AA., Ibrahim, A., Alohali, M.A., Abotaleb, M., Khafaga, D.S. (2023). Network intrusion detection based on feature selection and hybrid metaheuristic optimization. Computers, Materials & Continua, 74(2): 2677–2693. https://doi.org/10.32604/cmc.2023.033273

[27] Edwin Singh, C., Celestin Vigila, S.M. (2023). WOA-DNN for intelligent intrusion detection and classification in MANET services. Intelligent Automation & Soft Computing, 35(2): 1737–1751. https://doi.org/10.32604/iasc.2023.028022

[28] Almomani, O. (2020). A feature selection model for network intrusion detection system based on PSO, GWO, FFA and GA algorithms. Symmetry, 12(6): 1046. https://doi.org/10.3390/sym12061046

[29] Jameel, N., Abdullah, H.S. (2021). Intelligent feature selection methods: A survey. Engineering and Technology Journal, 39(1): 175-183. https://doi.org/10.30684/etj.v39i1b.1623

[30] Hamza, E.K., Aziez, S.A., Reja, A.H., Sabry, A.H. (2022). Identifying some regularities of radio frequency propagation of a radar system by analyzing different environmental effects. (2022) Eastern-European Journal of Enterprise Technologies, 5(9-119): 61-67. http://dx.doi.org/10.15587/1729-4061.2022.264093

[31] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30.

[32] Almomani, I., Al-Kasasbeh, B., Al-Akhras, M. (2016). WSN-DS: A dataset for intrusion detection systems in

wireless sensor networks. Journal of Sensors, 2016: 4731953. https://doi.org/10.1155/2016/4731953
[33] Nasser, T.H., Hamza, E.K., Hasan, A.M. (2023). MOCAB/HEFT algorithm of multi radio wireless communication improved achievement assessment. Bulletin of Electrical Engineering and Informaticsthis, 12(1): 224-231. http://dx.doi.org/10.11591/eei.v12i1.4078
[34] Liu, Q., Wang, D., Jia, Y., Luo, S., Wang, C. (2022). A multi-task based deep learning approach for intrusion detection. Knowledge-Based Systems, 238: 107852. https://doi.org/10.1016/j.knosys.2021.107852
[35] Sabbah, T.S. (2022). Hybridized dimensionality reduction method for machine learning based web pages classification. Iraqi Journal of Computers, Communications, Control and Systems Engineering, 22(3): 97-110. https://doi.org/10.33103/uot.ijccce.22.3.9
[36] Fadhil, H.M., Abdullah, M.N., Younis, M.I. (2022). A framework for predicting airfare prices using machine learning. Iraqi J. Comput. Commun. Control Syst. Eng, 22: 81-96. https://doi.org/10.33103/uot.ijccce.22.3.8
[37] Hu, J., Ma, D., Liu, C., Shi, Z., Yan, H., Hu, C. (2019). Network security situation prediction based on MR-SVM. IEEE Access, 7: 130937-130945. https://doi.org/10.1109/ACCESS.2019.2939490
[38] Jiang, K., Wang, W., Wang, A., Wu, H. (2020). Network intrusion detection combined hybrid sampling with deep hierarchical network. IEEE Access, 8: 32464-32476. https://doi.org/10.1109/ACCESS.2020.2973730