[1] **Hind Ali abdul Hassan**

[2] **Mina zolfy**

# Exploring Lightweight Deep Learning Techniques for Intrusion Detection Systems in IoT Networks: A Survey

Journal of Electrical Systems

***Abstract: -*** The proliferation of Internet users has coincided with a commensurate increase in the amount of very important, sensitive, and private information being transferred across the Internet. Malicious actors are increasingly targeting networks to breach them and obtain illegal access to critical information since this trend has revealed holes in security systems. In addition to endangering the privacy of the data concerned, these breaches disrupt the smooth functioning of systems. Therefore, in light of these dangers, intrusion detection systems (IDSs) are now an essential part of any cybersecurity program. The goal of these systems is to detect and report any suspicious activity by constantly monitoring and analyzing network traffic. Numerous review articles have investigated various methods for network intrusion detection. To improve detection accuracy while keeping computing efficiency high, this survey study investigates lightweight deep learning techniques for intrusion detection systems. These techniques include pruning, quantization, clustering, and collaborative optimization. This study analyzes five different types of new real-world traffic datasets (i.e., CSE-CIC-IDS2018, NSL-KDD, Bot-IoT, ToN IoT Network, and UNSW-NB15) and evaluates the performance of several machine learning and deep learning techniques. This survey provides metrics for measuring the accuracy of intrusion detection across various systems, which may be used to assess performance.

***Keywords:*** Intrusion detection system, Deep Learning, Lightweight model, pruning, quantization, clustering, Dataset-IoT.

## I. INTRODUCTION

Internet of Things (IoT) is a part of daily life and has a place in many applications as it allows a variety of devices to communicate with each other over the Internet [1]. One of the biggest problems with the Internet of Things is security, posing a significant security threat because of how hackers can get access to it [2]. To secure the network against different forms of attacks several defense algorithms are available such as the Intrusion Detection System (IDS) represents one of the defense techniques and crucial components in ensuring the security of computer networks by identifying and mitigating malicious of activities. Traditional IDS techniques relying on handcrafted features and rule-based systems, often struggle to effectively detect emerging and sophisticated attacks [3].

IDS have developed throughout the years to detect malicious and authentic network traffic. Complex network threats are the driving force behind the development of the next-generation firewall. Its system module could identify intrusions using signatures, behavioral analysis, and harmful actions. The term intrusion detection system can be defined as a system that detects and reports any suspicious activity in a network. The main objective of it is to safeguard a system from potential threats through coordinated processes in a coordinated fashion. In most cases, a security analyst can take the required steps to lessen the impact of the breaches [4].

Artificial intelligence (AI) techniques play a crucial role in the development of IDS and offer several benefits over other techniques [5]. Intrusion detection has been a hotspot for machine learning, deep learning approaches recently because of their capacity to automatically identify useful features from unstructured network data. A successful technique that developed from the shallow neural network, deep neural networks (DNNs) have recently attracted a lot of attention in the field of intrusion detection. DNNs can mimic extremely complex models and are better at modeling or abstracting representations [6], [7].

Deep learning models offer superior anomaly detection accuracy, their computational requirements and model complexities can be challenging, especially in resource-constrained environments. Lightweight deep learning models developed to solve this problem, have characteristics of fewer parameters, simpler architectures, and lower computational and memory requirements, making them ideal for deployment on devices with limited resources. Lightweight techniques including network pruning, weight clustering, quantization, and collaborative optimization have shown promise in developing lightweight deep learning models for intrusion detection [8].

[1] *Corresponding author: Tabriz university
Waist unversity
[2] Tabriz university

This survey paper explores the concept of lightweight deep learning for intrusion detection systems, which aims to leverage the power of deep learning models while maintaining computational efficiency. In this work, we review the literature on intrusion detection systems (IDS) that make use of lightweight machine learning and deep learning techniques, examine the most popular IDS datasets for (IoT) network, and compare the results of lightweight deep learning models on various IoT datasets.

The outline for the rest of the paper.: A concise overview of lightweight approaches is provided in Section 2. Types of intrusion detection system datasets are briefly described in Section 3. Section 4 delves into the methods employed by intrusion detection systems, specifically machine learning and deep learning. Describe the intrusion detection system for the Internet of Things in Section 5. An intrusion detection system that uses software to create a network is briefly described in Section 6.

## II. LIGHTWEIGHT TECHNIQUE

lightweight technique generally refers to methods and approaches that aim to reduce the complexity, computational resources, or memory requirements of artificial models while still maintaining reasonable performance. These techniques are particularly important for scenarios where deploying large and resource-intensive models is not feasible due to constraints such as limited hardware capabilities or real-time processing requirements [9]. In following lightweight techniques used in DNN.

### *A. Model* Pruning:

It is a technique that involves removing unnecessary connections or parameters from a deep neural network. There are two types of pruning weight and filter pruning. Weight pruning is a technique used in machine learning, particularly in deep learning, to optimize and reduce the size of neural network models by selectively removing less important connection weights or setting their values to zero. This process leads to a more efficient and compact model with fewer parameters, which can have several benefits including improved model deployment and reduced computational resource requirements [10], [11]. Fig. 1 shows an illustration of pruning techniques.

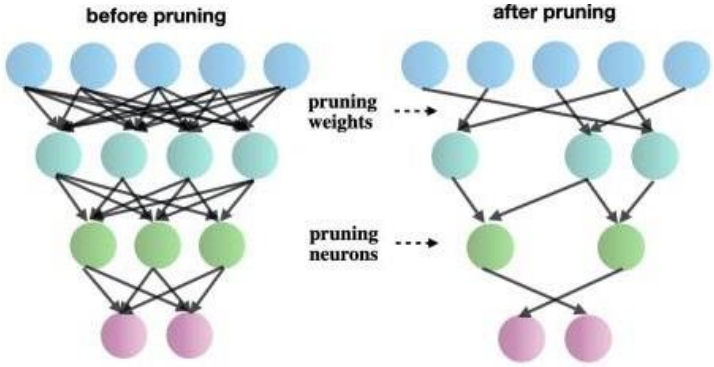


Figure. 1: Pruning model [12]

By deleting unnecessary filters from the pre-trained model according to a predetermined criterion, filter pruning can reduce the size of deep convolutional neural networks (CNNs). Filter pruning is a three-step method that begins with training a large model on the target dataset and ends with retraining (fine-tuning) the model to remove any extraneous filters. It is necessary to retrain the trimmed model for it to perform as before. Several techniques exist for filter pruning, including learning filter pruning criteria, stripe-wise pruning, differentiating layer pruning based on RFC, and receptive field criterion (RFC) (LFPC) [13], [14].

### *B. Quantization:*

The accuracy of the model's activations and weights is diminished during quantization. To implement quantization, lower-width integers are used in place of 32-bit floating-point numbers. The use of 8-bit integers for weights and activations, for instance, can drastically cut down on computation time and memory use. [11] . There are two forms of quantization:

I. Quantization-Aware Training: This method involves training the model with quantization as an objective. During training, simulated quantization operations are applied to activations and weights,

allowing the model to learn to be robust to quantization error. This often results in better post-quantization accuracy [15]

II. Post-Training Quantization (PTQ): It is a technique where an already trained model is quantized after training. This is a common approach to quantizing pre-trained models for deployment [15]

*C. Weight clustering:*

When it comes to deep learning and machine learning, weight clustering is a tool for making smaller, more efficient neural networks. Clustering is a technique that groups neural network weights that are similar and assigns each cluster a single value—the centroid or cluster center—to represent them. A smaller number of network parameters can be achieved with this procedure. [16].

*D. Collaborative optimization:*

Collaborative optimization is a comprehensive method that uses multiple techniques to create a model that, when deployed, has the optimal balance of target properties including accuracy, model size, and inference speed. To accomplish the cumulative optimization effect, collaborative optimizations aim to build on separate techniques by applying them sequentially. [17].

The area of IoT has been the subject of numerous reviews and surveys. Idrissi *et al* [18] applied a (CNN) optimization strategy that includes pruning, post-training quantization, and clustering to create a deep learning-based host intrusion detection system (DL-HIDS) for usage with a subset of commercial IoT devices. It achieves an accuracy of up to 99.74 when implemented on the MQTTIOT-IDS2020 dataset. Ching-Hao Wang *et.al* [19] recently developed approaches for compressing models, such as pruning, quantization, knowledge distillation, and search for network design, which are presented. Two types of pruning exist: filter pruning and weight pruning. These methods can be used by any model to reduce power consumption and speed up inference time; they are all considered compression methods. These strategies can be used by models such as CNN or DNN. R.F. Bikmukhamedov& A.F. Nadeev [20] presented methods for lightweight machine learning classifiers using IoT traffic flows as a dataset: logistic regression, support vector machines with linear kernels, decision trees, and linear models. The writers gathered dataset packets from smart home appliances and used the Principal Component Analysis (PCA) algorithm to exclude or decrease similar data in the comparison line and reduce dimensionality without losing any data. linear models (SVM and Logistic Regression) attained a performance accuracy of approximately 99.6%, with the most recent proposal achieving an accuracy of up to 99.8%.

**Zhao *et. al*** [21] offered a network intrusion detection system for the Internet of Things (IoT) based on a lightweight neural network (LNN). Using the principle component analysis (PCA) method in the data preprocessing stage, this paper achieved high classification performance with low computational cost by reducing the dimensionality of traffic features. It then implemented this model in two datasets, the UNSW-NB15 Data Set and the Bot-IoT Data Set. The cross-entropy function was utilized as the loss function for binary classification, whereas network intrusion detection was utilized as the loss function for multiclassification. In the first data set, the accuracy for binary classification reaches 98.94 percent, whereas in the second data set, it reaches 99.99 percent. 86.11% and 96.15% in multiclassification. Consequently, the UNSW-NB15 data set and the Bot-IoT data set both show an 84.11 percent and an 80.63 percent reduction in LNN model complexity, respectively. Mingjian **Lei *et.al*** [22] the P-DNN, a novel approach to intrusion detection that utilizes pruning deep neural networks. Step one involves training a deep neural network with a complicated structure using the expanded features of the original data set. Step two involves gradually pruning the network to get a smaller and simpler model. Connections with higher absolute weights in a DNN model undergo pruning because they contain more crucial information than connections with lower absolute weights. This model was applied to the KDD Cup 99 dataset, which has 494021 instances in training and 311029 instances in testing, and it achieved an accuracy of 93%. The pruning operation reduced the model's complexity. Only connections with more significant information were kept in the weight.

Godswill Lucky et.al [23] a decision-tree method, was suggested as a model to be used with a lightweight network monitoring approach that makes use of feature selection to detect distributed denial of service attacks. The analyses presented here make use of three datasets: CAIDA 2007, CIC 2017, and 2019 The Low Variance Filter was used to select the features from these datasets. Using just three features, they can attain an accuracy of up to 99.69 percent across all datasets, according to the final technique examination of the design. Tailin Liang et al [24] suggested the Prunin and quantization methods for optimizing networks. As part of the pruning process, any neurons or duplicated parameters that do not add much to the reliability of the results are removed. It can be classified as static if executed before or during runtime, or as dynamic if executed during runtime. In signal processing, quantization refers to the transformation of a continuous signal into a representation in a discrete

symbol or integer form. Its bias and weighted activation function are quantified. Put the weights into numbers instead of the activations. The sensitivity of activation to numerical precision is higher. Bais is not subject to quantization since it does not necessitate storage. B Sharmila and R. Nagapadma [25] both quantized autoencoder uint8 and quantized autoencoder float16 (QAE-float16) are proposed as separate AI models by the authors, who employ post-training quantization (QAE-uint8). Anomaly data is assumed to generate high reconstruction error (RE) via autoencoder models, from which QAE models are created. The quantization process that follows training makes use of pruning, grouping, and other similar methods. We put the suggested models through their paces using the 24-feature RT-IoT23 dataset. Attacks such as SSH brute-force, UFONet, and distributed denial of service were the primary areas of attention for the writers (Distributed Denial of Service ). By a significant margin, QAE-uint8 is the most efficient model. It reduced peak CPU consumption by 27.94%, compressed memory size by 92.23%, and decreased average memory utilization by 70.01% .

To our knowledge, there is very little research paper that considers the optimization techniques in IDS. This led us to investigate this field with more attention and focus the light on these approaches. Table I shows a comparison analysis of IDS.

TABLE I. COMPARATIVE ANALYSIS OF LIGHTWEIGHT TECHNIQUES FOR IDS

| Ref | Learning model | Optimization | Dataset | Accuracy before optimization | Accuracy after optimization | Model depths | Model size before optimization | Model size after optimization |
|---|---|---|---|---|---|---|---|---|
| [18] | CNN | Pruning,Quantizatio clustering | MQTTIOT-IDS2020 | 99.74% | 97.74% | 7 layers, 16 Features | 343 Kb | 106 Kb |
| [22] | DNN | Pruning | KDDCPU 99 | 91.2% | 93.71% | 11 layers 41 features | NA | NA |
| **[25]** | Autoencoder | QAE-uint8 | RT-IoT23 | 98.40% | 96.35% | 8 layers 23 features | 79 Kb | 6 Kb |

## III. BENCHMARK DATASETS USED IN IDS MODELS:

To assess the intrusion detection model, several datasets were utilized as benchmark datasets. A higher detection rate and more accurate classifications are the goals of the work performed on the different datasets [26]. Intruder detection databases have proliferated in recent years [27]. Table II shows the details of the IDS dataset used in IoT.

1. **BoT-IoT Dataset**: Its creation was facilitated by the Cyber Range Lab of the UNSW Canberra Cyber Center's realistic network environment architecture. Various types of the dataset's source files are made available, such as the original pcap files, generated argus files, and CSV files. To facilitate the labeling procedure, the files were split according to the assault category and subcategory. Data exfiltration, keylogging, OS and service scans, distributed denial of service, and distributed denial of service attacks are all part of the dataset. When it comes to DDoS and DoS attacks, the data is further grouped by protocol. To make the dataset more manageable [28].
2. **ToN_IoT Network Datasets**: The goal is to gather and analyze data from many sources related to the IoT and industrial IoT. It contains a variety of data gathered from many sources, such as system network traffic, telemetry data from linked devices, and system logs from Linux and Windows. Connecting numerous virtual computers, cloud layers, blur, edges, and physical systems, the ToN-IoT dataset is built from a realistic network to assess the efficacy and precision of different AI-based cybersecurity technologies [29].
3. **NSL-KDD Dataset:** It is a refined version of the KDD Cup 1999 dataset. It addresses some of the limitations and shortcomings of the original dataset. NSL-KDD has become a popular choice for evaluating models due to its balanced distribution of attacks and non-attacks, reduced redundancy, and updated feature selection [30].
4. **UNSW-NB15 Dataset:** It was created by the University of New South Wales (UNSW) in Australia and is a more recent dataset that includes more recent attacks. It is a comprehensive dataset specifically designed for network intrusion detection systems. UNSW-NB15 captures a wide range of modern network intrusion scenarios, including various types of attacks and normal traffic [31], [32].
5. **CICIDS2017 Dataset:** It was created by the Canadian Institute for Cybersecurity and provides a comprehensive collection of benign and malicious network traffic for NIDS evaluation. It includes a wide range of attacks, such as DoS, DDoS, botnets, and ransomware, and features realistic network scenarios [33].

TABLE II. : SUMMARY OF THE MOST COMMON BENCHMARK USED IN THE IDS MODEL

| Ref | Dataset | Attack type | No. of futures | No. of class | Generation year | Dataset size |
|---|---|---|---|---|---|---|
| [28], [34] | Bot-IoT | DDoS, DoS, OS, Service Scan, Keylogging and Data exfiltration attacks | 48 | 6 | 2019 | 16.7 G |
| [29] | TON_IoT Network | DoS, DDoS, Ransomware, Backdoor, injection, password, scanning, Man in the middle (MITM) and cross-site scripting (XS S) | 46 | 9 | 2020 | 148.4 M |
| [30] | NSL-KDD | DOS, R2L, U2Rand a probing | 41 | 4 | 2009 | 15MB |
| [35] | UNSW-NB15 | Normal, Fizzers analysis, Back-Doors, DoS, Exploits, Generic, Reconnaissance, Shell Code, and Worms. | 49 | 9 | 2010 | 156MB |
| [33] | CICIDS2017 | Web-based, Brute force, DoS, DDoS, Infiltration, Heart bleed, Bot, and Scan. | 81 | 8 | 2017 | 241 MB |

## IV. BENCHMARK DATASETS USED IN IDS MODELS:

There are several techniques designed to be used with IDS based on AI. describing the algorithms used in Machine Learning (ML) and Deep Learning (DL) techniques and also optimization techniques. The classification depends on types of algorithms, which include supervised and unsupervised algorithms [36], [37]. Supervised learning algorithms to detect known attacks and unsupervised learning to detect unknown and zero-day attacks [38]. Fig. 2 Illustration IDS techniques

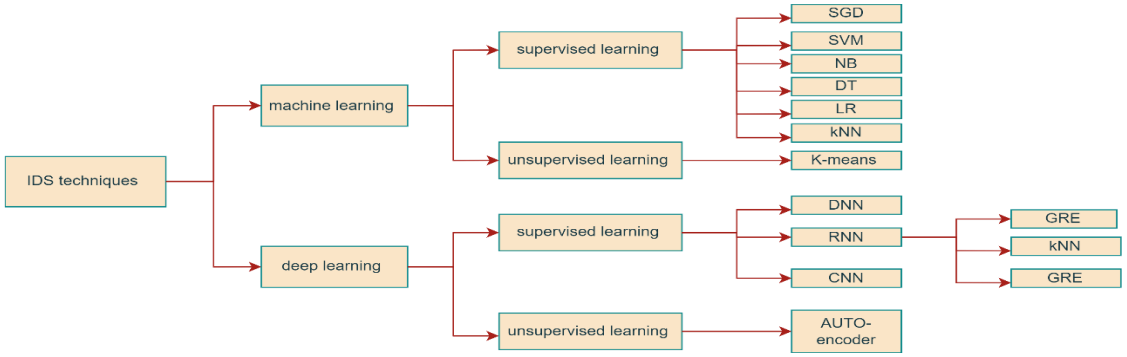


Figure. 2: IDS techniques Based Atrifical Intelligent

### A. *Machine* Learning Algorithms in Intrusion Detection Systems

IDS are undergoing a revolutionary change because of machine learning, which allows them to adapt and learn from new threats as they emerge. Through vast data analysis, anomaly detection, and pattern recognition, they enhance threat detection, reduce false alarms, and automate security processes making networks more resilient and secure. In [39], [40] A plethora of machine learning (ML) classifiers, including k-nearest neighbor (k-NN), support vector machine (SVM), naive Bayes (NB), random forest (RF), decision tree (DT), and stochastic gradient descent (SGD), have been developed to construct advanced and efficient intrusion detection systems (IDS). In [41] classifiers like KNN, SGD, RF, LR, and NB are used for training based on a taxonomy of classifiers that involves both lazy and eager learners to refine the feature selection. The chi-square a filter-based technique is applied to the UNSW-NB15 dataset. In [42] SVM and BN known for their effectiveness in solving classification problems form part of the evaluation process on the NSL-KDD dataset, consisting of 19,000 samples, using accuracy and misclassification rates as evaluation metrics, In [43], [44] as a result of optimization, the IDS achieved maximum classification accuracy. Feature selection variations implemented on the IDS played a pivotal role in this achievement by constructing robust machine-learning models. Table III shows different machine-learning.

TABLE III. ML TECHNIQUES USED IN IDS MODELS

| References | Tools used with IDS | Dataset | Accuracy |
|---|---|---|---|
| [39] [41], [42], [43] | NB , k-NN,SVM,NB, RF,DT SGD & LR | NSL-KDD,UNSW_NB15& CIC-IDS2017, MQTT-IOT-IDS2020, | 99.98% to 76.96% |

### *A. Deep Learning Algorithms in Intrusion Detection Systems*

Compared to ML-based methods, deep learning (DL) approaches perform better when dealing with massive datasets. Because of its ability to automatically extract complicated representations from data, DL is regularly utilized in cybersecurity, and its approaches have matured into the most practical and extensively used intrusion detection system in networks [46], [47],[48]. Recent studies are reviewed, the DL approaches are used to propose solutions of IDS [49].DL is a crucial tool for improving IDS performance since it explains what IDS is, how it works and gives definitions and examples of various forms of IDS [50]. When it comes to huge data, IDS has its work cut out for it, but DL is up to the task. When opposed to ML, DL can automatically extract features without feature engineering [51]. Hence, recent work on DL with network anomaly detection has demonstrated and addressed the possibility of DL in network traffic analysis [52]. There were benefits and drawbacks of using deep learning in intrusion detection that were demonstrated by certain intrusion detection systems [53], [54], [55], [56]. To effectively identify possible threats, the NSL-KDD dataset was fed into a Deep Neural Network (DNN). To develop a model using the DNN algorithm, the dataset has to be preprocessed and normalized first. The complete NSL-KDD dataset was employed for testing purposes. Subsequently, accuracy and precision matrices were utilized to assess the effectiveness of the DNN model. This proposed strategy, based on DNN, significantly improves the identification of network anomalies and introduces new avenues for analysis within intrusion detection systems [57].

### *B. Features Selection Optimized*

Feature selection involves picking a subset of important features from a dataset for model creation. Its goal is to enhance predictor performance, speed up, economize predictions, and gain a clearer insight into the data's underlying generation process [58]. The particle swarm optimization PSO is one of the optimization methods of feature selection [59]. In [60] To eliminate extraneous and noisy attributes using the random forest (RF) technique, the PSO algorithm was applied to the selective features of the NSL-KDD dataset. This dataset consisted of just 10 features out of 41 originally. Using the NSL-KDD dataset's training and testing sets, an RF classifier is trained to identify the most important features, rank them from most to least, and then remove the irrelevant ones. In [61] a new anomaly-based detection system utilizes a novel feature selection technique termed mutation cuckoo fuzzy MCF to identify optimal feature subsets. It employs multiverse optimization with an artificial neural network MVO-ANN for classification. This model is specifically applied to intrusion detection and validated using the widely recognized NSL-KDD dataset. Identifies 22 out of 41 features as the most influential, significantly improving the performance of the anomaly-based intrusion detection system.

In [62], [63] One new optimization technique that has just been developed is PIO or Pigeon-Inspired Optimizer. It is a swarm intelligence algorithm. The PIO algorithm paired with the DT classifier underwent evaluation across three well-known datasets: KDDCUP99, NLS-KDD, and UNSW-NB15. Using PIO's feature selection approach, the number of features in the datasets was drastically reduced. Specifically, from 41 to just 7 features for KDDCUPP99, 41 to 5 for NSL-KDD, and 49 to 5 features for UNSW-NB15. This reduction maintained a high true positive rate of TPR and accuracy while significantly decreasing the time required for model development. Additionally, the PIO cosine similarity method for binarization displayed faster convergence than the sigmoid method. In [64], [65] the Genetic Algorithm GA is suggested as a prevalent optimization technique, particularly employed in solving combinatorial optimization issues. In a Fog environment, the Genetic Algorithm wrapper-based feature selection technique is combined with NB for the anomaly detection model GANBADM. This process eliminates surplus attributes, aiming to reduce time complexity while creating an improved model capable of more accurate result prediction. The NSL-KDD dataset is utilized for training, where 19 attributes are derived from the original 41 attributes forming the new attribute set.

In [66] The authors proposed a mixed-model approach that combines RNNs and limited Boltzmann machines. Without using feature engineering, RBM can detect malicious communications. In [67] a novel approach blending the fruit fly algorithm FFA and ant lion optimizer ALO is proposed for crucial feature selection in constructing an IDS. After feature selection using FFA and ALO the SVM, KNN, NB, and DT were employed to assess the chosen features across KDD Cup99, NSL-KDD, and UNSW-NB15 datasets. In [68] an approach employing Genetic

Algorithm GA proposed for feature selection aims to boost IDS accuracy. This was tested across three benchmark datasets and compared against standard feature selection methods. In [69] imagined structures for ANNs Integrating convolutional neural networks (CNNs) for spatial feature extraction with long short-term memory networks (LSTMs) for temporal feature extraction form a CNN-LSTM hybrid intrusion detection system (LSTMs). In [70] proposed optimization using the golden eagle method Applying this method to the datasets NSL-KDD and UNSW-NB15, the GEO-SMPIF self-constructing multi-layer perceptron interfaced fuzzy system increases privacy and security within the professional network architecture. In [71] A collection of features was chosen using the embedded feature selection technique known as GIWRF, which stands for gini impurity-based weighted random forest. To apply these features to the UNSW-NB15 and Network TON IoT datasets, we used DT, Gradient Boosting Tree GBT, AdaBoost, multilayer perceptron MLP, long short-term memory LSTM, and gated recurrent unit GRU models. In [72] an additional feature selection algorithm is suggested combining conditional random field CRF and spider monkey optimization SMO to identify the most pertinent features within a dataset. Initially, CRF is utilized for the initial selection of contributed features followed by the application of SMO to refine and finalize the useful features from the reduced dataset. Additionally, a convolutional neural network (CNN) is employed for classifying the NSL.KDD dataset into normal and attack categories. In [73] the process employs an ensemble feature selection-based DNN to efficiently identify anomalous behaviors in network traffic data. It combines a light gradient boosting machine LightGBM for feature selection and a DNN integrated with batch normalization and embedding techniques as the classifier. Table IV shows a summary of different feature selection algorithms in IDS.

TABLE IV. OPTIMIZATION TECHNIQUES USED IN IDS MODELS

| **Ref.** | **Classifier** | **Optimization techniques** | **Dataset** | **No, of features before the optimizer** | **No, features after the optimizer** | **Average accuracy** |
|---|---|---|---|---|---|---|
| [60] | FR | PSO | NLS-KDD | 41 | 10 | 99.3% |
| [61] | MCF&MVO-ANN | MCF | NLS-KDD | 41 | 22 | 98.16 |
| [63] | Sigmoid PIO DT, Cosine PIO DT | PIO | KDD CUP 99,NLS-KDDand UNSW-NB15 | 41,41,49 | 7,5,5 | 96 %to 86.30% |
| [64] | NB | GA | NLS-KDD | 41 | 19 | 99.73 |
| [67] | SVM, KNN, NB, DT | FFA–ALO | KDDCup99,NSL-KDD and UNSW-NB15 | 41,41 49 | 12,16,15 | 99.73% to 99.12% |
| [68] | SVM, KNN, XgBoost | GA | KDD Cup'99, UNSWNB15, and Bot-IoT | 34,49,29 | 10, variable | 99.8% |
| [70] | GEO-SMPIF | CHO | NLS-KDD, UNSW-NB15 | 41,49 | 11,13 | 99.99%, 99.97% |
| **[71]** | DT | GIWRF | UNSW-NB15, Network TON_IoT | 42,41 | 20,10 | 93.01%,99.80% |
| **[72]** | CNN | SMO | NSL-KDD | 41 | 16 | 99.90% |
| [73] | DNN | LightGBM | KDD 99, NSL-KDD, and UNSW-NB15 | 41,41,49 | 14,15,15 | 99.92%to 88.34% |

## V. INTRUSION DETECTION SYSTEM IN INTERNET OF THINGS ENVIRONMENTS

The phrase "Internet of Things" (IoT) describes the interconnected system of physical objects and the software that allows them to exchange data with one another and with the cloud. This term is used to characterize electronic devices that may collect data from sensors, run programs, and communicate with other devices and systems through networks such as the Internet now more than ever, security is a major issue for the Internet of Things. Cyberattacks on IoT devices have a long history of being successful[74]. Problems with IoT devices' fundamental design are the source of this issue. Their power source is sometimes restricted, and they need to be able to endure years of use on a single charge while out in the field. Many Internet of Things devices cannot encrypt, authenticate, or use security protocols since doing so would substantially raise the power consumption of simple transmissions. As more

sophisticated methods of exploiting firmware vulnerabilities become available, more and more flaws will be found in the firmware of the devices. These vulnerabilities can build up during the device's lifespan if updates aren't applied. [75], [76], [77]. An IoT attack is a compromise of (IoT) system. Products, services, information, and people all fall into this category. A hacker could take control of an automated or IoT system, steal data from it, or even disable it by launching an Internet of Things (IoT) assault. IoT attacks can compromise devices connected to the IoT system including phones and computers[78]. Internet of Things (IoT) ecosystems are particularly vulnerable to DoS and DDoS attacks, the two most common forms of distributed denial of service (DDoS). These assaults cause an oversaturation of the IoT network or devices, which causes them to become inoperable [79]. Malware is another type of attack that specifically targets the interconnected devices within the IoT aiming to disrupt their functionality steal data or take control of the devices for malicious purposes [80]. IDS has three types namely, anomaly-based detection, signature-based detection, and specification-based detection.

1. **Anomaly-based IDS (AIDS)** is an approach to identify cyber threats by detecting unusual behavior in a system making them effective against new or unknown attacks. They establish a baseline during a training phase monitor for deviations and generate alerts for potentially malicious activities. They complement signature-based IDS for a more comprehensive security approach. This study focused on IDS based on an anomaly. Thus the existing sub-classification of this system is as follows:

- Statistical-based approach is one of the techniques used in AIDS during the training use of statistical measures to identify deviations from expected behavior. These measures can be simple such as mean and standard deviation or more complex involving multivariate analysis or time series analysis.
- **Knowledge-based (AIDS)** is the second technique of the AIDS security system that relies on a predetermined understanding of normal system behavior to identify deviations or anomalies. In knowledge-based refers to the system's reliance on a predefined set of rules, thresholds, or models to characterize what is considered normal within a network or system. Unlike statistical anomaly detection which may adapt to changes over time, knowledge-based AIDS relies on a fixed set of rules and may not easily adapt to evolving threats or variations in system behavior. It can be effective in detecting known types of anomalies or attacks for which explicit rules are defined. However, it may be less suitable for identifying novel or sophisticated threats that do not conform to the established rules [81], [82].
- **Machine Learning-based (AIDS)** offers the advantage of adaptability and the ability to detect unseen threats. They can continuously improve their detection capabilities over time as they encounter new data. However, they may also face challenges such as false positives or the need for large amounts of diverse training data to effectively capture the range of normal behaviors [83].

2. **Signature-Based Technique** This form, which compares the attack's signature to the present traffic, is called knowledge-based detection or abuse. If a match is detected, an attack report will be generated; in the absence of a match, no attack will be considered. This method stands out from the competition due to its minimal false alarm rate and constant signature updating requirement [84].
3. **Specification-Based Technique** For this type, detecting a program's activity and alerting the user to a breach of those specifications rely on matching the memorized and predetermined specifications with those specifications. [85]. With its ability to identify novel Internet of Things (IoT) risks, anomaly-based NIDS is the subject of this research. The NIDS examines data sent over a network to identify previously unseen threats. A continuing research challenge, the feature set design is vital for identifying network traffic. [86].

In [87] A highly extendable DNN model was created for IoT networks; it successfully detected IoT DDoS botnet attacks with a headstrong detection rate of 0.94. In [88] Security attacks on IoT networks can now be more accurately detected and intelligently stopped with the use of convolutional neural networks (CNNs), bidirectional long short-term memory (Bi-LSTM), and other machine learning and deep learning techniques. Training and testing the model on two separate datasets, UNSW-NB151 and NSL-Botnet2, ensure its flexibility to handle various data types. In [89] a model built on autoencoder neural networks is suggested as an anomaly-based detection system that may detect botnet activity in the Internet of Things (IoT) using unsupervised deep learning methods. We apply the model to the BOT-IoT dataset after collecting, preprocessing, and normalizing the data. In [90]The DeBot model, a deep learning tool for BoT detection in industrial network traffic, makes use of a unique Cascade Forward Back Propagation Neural Network (CFBPNN) with a subset of features selected using the correlation-based feature selection (CFS) technique. It has been tested extensively on five BoT-IoT datasets: NF-UNSW-NB15, NF-ToN-IoT, NF-BoT-IoT, NF-CSE-CIC-IDS2018, and ToN-IoT-Windows.

In [91] convolutional neural network (CNN)-based anomaly-based intrusion detection systems (IDS) were established. These systems take advantage of the IoT's capabilities by thoroughly analyzing all network data. The

convolutional neural network (CNN) model can detect suspicious traffic patterns and possible intrusions. Using the NID and BoT-IoT datasets, the model is trained and evaluated. In [92] Long short-term memory and feed-forward neural networks have been introduced. The performance and detection of various types of assaults are assessed using two distinct datasets, NSL-KDD and BoT-IoT. In [93] three three-level IoT-BoT databases have used the following machine learning classifiers: DT, ensemble bag, K-NN, linear discriminant, and SVM. The bot-IoT original dataset is the first of three databases; the second became smaller through random sampling; and the third became balanced through the use of the synthetic minority oversampling technique SMOTE function, which improved the effectiveness of the IDS model level including labels and reduced classification errors. The labels are structured hierarchically, with "attack" or "normal" at the top level and "category" and "subcategory" of "attack" at lower levels. In [94] IDS employs fog computing to identify distributed denial of service (DDoS) assaults on mining pools in Internet of Things (IoT) networks that include blockchain technology. The BoT-IoT dataset is used to test the suggested model, which is assessed using RF and an improved gradient tree boosting system on distributed fog nodes. According to the results, XGBoost is better at detecting binary attacks, whereas the RF is better at detecting multi-attacks. Furthermore, compared to XGBoost, the RF's training and testing times on dispersed fog nodes are significantly shorter. In [95] built an IoT-centric intrusion detection model with a K-NN classifier and proposed feature selection methods. To achieve better results, this model proposes building an NIDS using the K-NN algorithm. Using principal component analysis The features are chosen using GA and a PCA univariate statistical test. In [96] Testing a GA-based feature selection strategy with a DT classifier on the Bot-IoT botnet detection dataset revealed that, out of 40 features, 6 were effectively chosen. Table V shows different IDS techniques used in IoT.

TABLE V. IDS MODELS WITH IOT

| Refs. | Algorithm used | IoT dataset | Type of classification | Accuracy average |
|---|---|---|---|---|
| [97] | DNN | DDose- Iot attack | Binary | 94% |
| [88] | CNN, BiLSTM | UNSW-NB151& NSL-Botnet2 | Both binary and multiclassification | 99.4% to 83% |
| [89] | Autoencoder | BOT-IoT | Binary | 99% |
| [90] | Neural network | NF-UNSW NB1 NF-ToN-IoT, NF-BoT IoT, NF-CSE-CIC-IDS2018&ToN-IoT-Windows | Binary | 100% |
| [91] | CNN | NID data &Bot-IoT | Binary | 99.51%&98.85 % |
| [92] | Neural network, LSTM | BoT-IoT &NLS-KDD | Binary | 99.97% to 96.44% |
| [93] | DT,EnsembleBag,KNN,LD&SVM | BoT-Iot DB2 BoT-Iot DB3 (two databases with three levels) | Binary and multiclassification | 100%to 99.9% |
| [94] | 1-RF &Xgbooest | BoT-IoT dataset | Multiclassification | 99% |
| [95] | KNN | BoT-IoT dataset | Multiclassification | 99.99% |
| [96] | DT based GA | BoT-IoT dataset | Multiclassification | 99.87% |

## VI. INTRUSION DETECTION SYSTEM WITH SOFTWARE DEFINE NETWORK ENVIRONMENTS

The idea behind Software Defined Networks (SDNs) is to partition the networking control OS from the hardware functionality and put the control OS in a central location so it can govern the hardware functions underneath it. [98]. While software-defined networking (SDN) can build a secure network, it has double-edged swords: it increases the likelihood of assaults and clarifies the possibility of applying deep learning (DL) for an anomaly detection system based on the flow of data [99]. Simplifying the implementation, speeding up incident responses, and allowing for a prompt reaction to cyberattacks through appropriate countermeasures are all possible outcomes of incorporating IDS into software-defined networking SDN [100]. SDN-based network intrusion detection systems have lately implemented machine learning (ML) for data network security and problem resolution [101]. Figure 3 describes the SDN layer architecture

In [102] To identify attacks in software-defined networking (SDN), researchers employ a hybrid feature selection algorithm. This algorithm has two parts: the first uses the correlation-based feature selection (CFS)

algorithm to get a subset of features, and the second uses the Random Forest Recursive Feature Elimination (RF-RFE) algorithm to get the best set of features. Finally, the algorithm employs the LightGBM algorithm to identify and categorize various SDN attacks. In [103] RF, DT, and KNN methods have been proposed for classifying malicious traffic in the InSDN dataset with an SDN environment. After reducing the number of features in the InSDN dataset using the cross-correlation feature, we found that the reduced dataset had the quickest learning time compared to the original dataset. The RF achieved the highest accuracy among all methods used. In [104] Features selected via cross-correlation, We present the CCFS approach and compare it to two existing algorithms, CFA and MIFS, which use four different classifiers: SVM, NB, DT, and KNN. According to the findings of the experiments conducted on the KDD Cup 99, NSL-KDD, AWID, and CIC-IDS2017 datasets, DT is the best classifier and CCFS is the best feature selection method. In[105] ML classifiers included SVM, KNN, NB, and RF and were employed to achieve early and accurate detection of DDoS attacks within SDN environments. This was accomplished by leveraging optimal feature subsets identified through three feature selection techniques: filter, wrapper, and embedded methods with recursive feature elimination RFE being specifically implemented on the KDD-NLS dataset. Table VI shows the different techniques used by IDS in SDN environments.

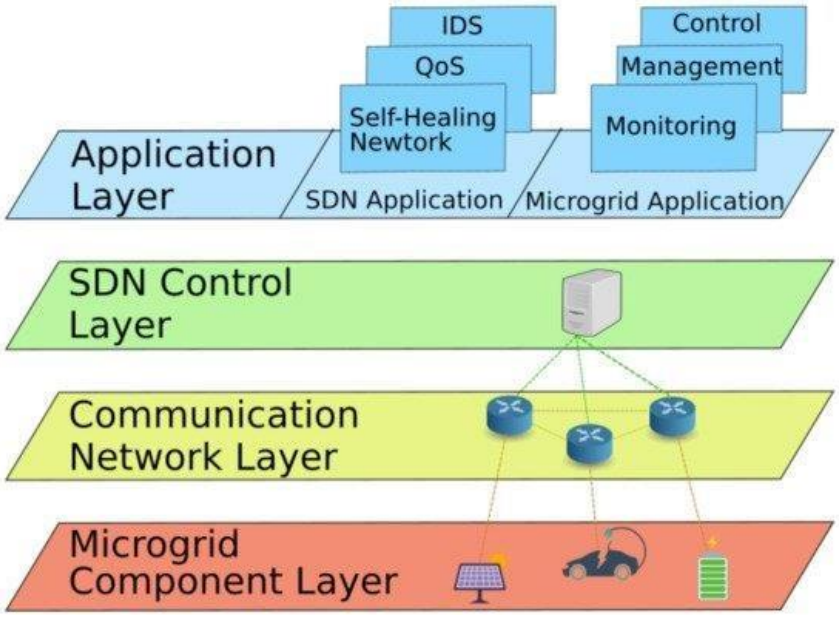


Figure 3: SDN layered architecture [106]

TABLE VI. IDS IN SDN ENVIRONMENTS

| Ref. | Classifiers | Optimizer | Dataset | Features selection | Accuracy |
|---|---|---|---|---|---|
| [102] | HFS-LGBM | RF-RFE | NLS-KDD | 8 | 98.72% |
| [103] | RF, DT, KNN | Feature correlation | InSDN | 56 | 99.9962% to 99.9699% |
| [104] | DT | CCFS | KDD Cup, 99NSL-KDD , AWID,and CIC-IDS2017 | 20,20,84.78 | 85.19 %t:98.42% |
| [105] | RF | REF | NLS-KDD | 28 | 99.97% |

## VII. CONCLUSION

The use of lightweight approaches, deep learning (DL), and machine learning (ML) in intrusion detection systems is thoroughly examined in this survey. This study sheds light on the effectiveness and generalizability of these approaches by analyzing their application to five separate IDS datasets. As a fundamental method, machine learning demonstrates its adaptability by efficiently identifying dataset trends and outliers. This survey shows that traditional ML techniques are still relevant in many security scenarios and highlights how they strengthen intrusion detection.

Intrusion detection systems have become much more efficient with the use of deep learning (IDS). The use of complex neural network topologies allows intrusion detection systems based on deep learning to identify and

respond to security threats with exceptional accuracy. The sophistication and frequency of security breaches on computer networks have only grown as these systems have evolved to counter them. A potential approach is to combine deep learning with lightweight approaches like pruning, quantization, clustering, and collaborative optimization for Intrusion Detection System (IDS) datasets. Model complexity, computing demands, clustering, and pattern recognition are all optimally addressed by these methods, highlighting the importance of collaborative optimization.

## REFERENCES


[1] M. Kuzlu, C. Fair, and O. Guler, "Role of Artificial Intelligence in the Internet of Things (IoT) cybersecurity," *Discover Internet of Things*, vol. 1, no. 1, Dec. 2021, doi: 10.1007/s43926-020-00001-4.

[2] C. A. de Souza, C. B. Westphall, R. B. Machado, L. Loffi, C. M. Westphall, and G. A. Geronimo, "Intrusion detection and prevention in fog based iot environments: A systematic literature review," *Computer Networks*, vol. 214, p. 109154, 2022.

[3] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[4] S. P. Thirimanne, L. Jayawardana, L. Yasakethu, P. Liyanaarachchi, and C. Hewage, "Deep Neural Network Based Real-Time Intrusion Detection System," *SN Comput Sci*, vol. 3, no. 2, Mar. 2022, doi: 10.1007/s42979-022-01031-1.

[5] A. Shenfield, D. Day, and A. Ayesh, "Intelligent intrusion detection systems using artificial neural networks," *ICT Express*, vol. 4, no. 2, pp. 95–99, Jun. 2018, doi: 10.1016/j.icte.2018.04.003.

[6] W. A. Ali, M. Bendechache, M. FadhelAljunaid, and P. Sandhya, "A Review of Current Machine Learning Approaches for Anomaly Detection in Network Traffic."

[7] S. P. Thirimanne, L. Jayawardana, L. Yasakethu, P. Liyanaarachchi, and C. Hewage, "Deep Neural Network Based Real-Time Intrusion Detection System," *SN Comput Sci*, vol. 3, no. 2, Mar. 2022, doi: 10.1007/s42979-022-01031-1.

[8] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning Efficient Convolutional Networks through Network Slimming," Aug. 2017, [Online]. Available: http://arxiv.org/abs/1708.06519

[9] X. Long, Z. Ben, and Y. Liu, "A Survey of Related Research on Compression and Acceleration of Deep Neural Networks," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2019. doi: 10.1088/1742-6596/1213/5/052003.

[10] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning Filters for Efficient ConvNets," Aug. 2016, [Online]. Available: http://arxiv.org/abs/1608.08710

[11] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[12] L. Deng, G. Li, S. Han, L. Shi, and Y. Xie, "Model compression and hardware acceleration for neural networks: A comprehensive survey," *Proceedings of the IEEE*, vol. 108, no. 4, pp. 485–532, 2020.

[13] Y. He, P. Liu, L. Zhu, and Y. Yang, "Filter Pruning by Switching to Neighboring CNNs With Good Attributes," *IEEE Trans Neural Netw Learn Syst*, vol. 34, no. 10, pp. 8044–8056, 2023, doi: 10.1109/TNNLS.2022.3149332.

[14] Z. He *et al.*, "Filter pruning via feature discrimination in deep neural networks," in *European Conference on Computer Vision*, Springer, 2022, pp. 245–261.

[15] X. Zhao, R. Xu, and X. Guo, "Post-training Quantization or Quantization-aware Training? That is the Question," in *2023 China Semiconductor Technology International Conference (CSTIC)*, IEEE, 2023, pp. 1–3.

[16] S. Ye *et al.*, "A Unified Framework of DNN Weight Pruning and Weight Clustering/Quantization Using ADMM," Nov. 2018, [Online]. Available: http://arxiv.org/abs/1811.01907

[17] S. and Communication Networks, "Retracted: Research on Collaborative Optimization Model of Tourism Resources and Highway Network Based on IoT Network and Deep Learning," *Security and Communication Networks*, vol. 2022, pp. 1–1, Dec. 2022, doi: 10.1155/2022/9803909.

[18] I. Idrissi, M. Azizi, and O. Moussaoui, "A Lightweight Optimized Deep Learning-based Host-Intrusion Detection System Deployed on the Edge for IoT," *International Journal of Computing and Digital Systems*, vol. 11, no. 1, pp. 209–216, 2022, doi: 10.12785/ijcds/110117.

[19] C.-H. Wang *et al.*, "Occam's razor in deep learning Lightweight Deep Learning: An Overview", doi: 10.1109/MCE.2022.Doi.

[20] ÎAroslavskiĭ gosudarstvennyĭ universitet, Institute of Electrical and Electronics Engineers. Russia Section. CAS Chapter, Moscow Technical University of Communications and Informatics, and Institute of Electrical and Electronics Engineers, *2019 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO) : 01-03 July 2019, P.G. Demidov Yaroslavl State University, Russia, Yaroslavl*.

[21] R. Zhao *et al.*, "A Novel Intrusion Detection Method Based on Lightweight Neural Network for Internet of Things," *IEEE Internet Things J*, vol. 9, no. 12, pp. 9960–9972, Jun. 2022, doi: 10.1109/JIOT.2021.3119055.

[22] Mingjian Lei, Xiaoyong Li, Binsi Cai, Yunfeng Li, Limengwei Liu, and Wenping Kong, *P-DNN: An Effective Intrusion Detection Method based on runing Deep Neural Network*. 2020.

[23] G. Lucky, F. Jjunju, and A. Marshall, "A Lightweight Decision-Tree Algorithm for detecting DDoS flooding attacks," in *Proceedings - Companion of the 2020 IEEE 20th International Conference on Software Quality, Reliability, and Security, QRS-C 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 382–389. doi: 10.1109/QRS-C51114.2020.00072.

[24] T. Liang, J. Glossner, L. Wang, S. Shi, and X. Zhang, "Pruning and quantization for deep neural network acceleration: A survey," *Neurocomputing*, vol. 461, pp. 370–403, Oct. 2021, doi: 10.1016/j.neucom.2021.07.045.

[25] B. S. Sharmila and R. Nagapadma, "QAE-IDS: DDoS anomaly detection in IoT devices using Post-Quantization Training," *Smart Science*, vol. 11, no. 4, pp. 774–789, 2023, doi: 10.1080/23080477.2023.2260023.

[26] V. R. Balasaraswathi, M. Sugumaran, and Y. Hamid, "Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms," *Journal of Communications and Information Networks*, vol. 2, no. 4, pp. 107–119, Dec. 2017, doi: 10.1007/s41650-017-0033-7.

[27] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network-based intrusion detection data sets," *Computers and Security*, vol. 86. Elsevier Ltd, pp. 147–167, Sep. 01, 2019. doi: 10.1016/j.cose.2019.06.005.

[28] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, Nov. 2019, doi: 10.1016/j.future.2019.05.041.

[29] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain Cities Soc*, vol. 72, p. 102994, 2021.

[30] Canadian Institute for Cybersecurity, "NSL-KDD dataset," UNIVERSITY OF NEW BRUNSWICK.

[31] T. Janarthanan and S. Zargari, "Feature Selection in UNSW-NB15 and KDDCUPâ€™99 datasets," 2017. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[32] V. Kanimozhi and P. Jacob, "UNSW-NB15 dataset feature selection and network intrusion detection using deep learning," *International Journal of Recent Technology and Engineering*, vol. 7, no. 5, 2019.

[33] Canadian Institute for Cybersecurity, "CSE-CIC-IDS2018 on AWS," UNIVERSITY OF NEW BRUNSWICK.

[34] N. Koroniotis and N. Moustafa, "Enhancing Network Forensics with Particle Swarm and Deep Learning: The Particle Deep Framework," Academy and Industry Research Collaboration Center (AIRCC), Mar. 2020, pp. 41–60. doi: 10.5121/csit.2020.100304.

[35] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems (UNSW-NB15 Network Data Set)." [Online]. Available: https://cve.mitre.org/

[36] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, "A systematic review on supervised and unsupervised machine learning algorithms for data science," *Supervised and unsupervised learning for data science*, pp. 3–21, 2020.

[37] N. Prazeres, R. L. de C. Costa, L. Santos, and C. Rabadão, "Engineering the application of machine learning in an IDS based on IoT traffic flow," *Intelligent Systems with Applications*, vol. 17, Feb. 2023, doi: 10.1016/j.iswa.2023.200189.

[38] L. Shahbandayeva, U. Mammadzada, I. Manafova, S. Jafarli, and A. Z. Adamov, "Network Intrusion Detection using Supervised and Unsupervised Machine Learning," in *2022 IEEE 16th International Conference on Application of Information and Communication Technologies (AICT)*, IEEE, Oct. 2022, pp. 1–7. doi: 10.1109/AICT55583.2022.10013594.

[39] M. Vishwakarma and N. Kesswani, "A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection," *Decision Analytics Journal*, vol. 7, Jun. 2023, doi: 10.1016/j.dajour.2023.100233.

[40] N. Saran and N. Kesswani, "A comparative study of supervised Machine Learning classifiers for Intrusion Detection in Internet of Things," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 2049–2057. doi: 10.1016/j.procs.2023.01.181.

[41] A. Halimaa and K. Sundarakantham, "Machine learning based intrusion detection system," in *2019 3rd International conference on trends in electronics and informatics (ICOEI)*, IEEE, 2019, pp. 916–920.

[42] A. Halimaa and K. Sundarakantham, "Machine learning based intrusion detection system," in *2019 3rd International conference on trends in electronics and informatics (ICOEI)*, IEEE, 2019, pp. 916–920.

[43] C. Gambella, B. Ghaddar, and J. Naoum-Sawaya, "Optimization problems for machine learning: A survey," *European Journal of Operational Research*, vol. 290, no. 3. Elsevier B.V., pp. 807–828, May 01, 2021. doi: 10.1016/j.ejor.2020.08.045.

[44] V. Priyalakshmi and R. Devi, "An Innovative Machine Learning Optimization Algorithm for Feature Selection," in *2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, IEEE, Jul. 2022, pp. 1–5. doi: 10.1109/ICSES55317.2022.9914132.

[45] G. Kocher and G. Kumar, "Analysis of Machine Learning Algorithms with Feature Selection for Intrusion Detection using UNSW-NB15 Dataset," *International Journal of Network Security & Its Applications*, vol. 13, no. 1, pp. 21–31, Jan. 2021, doi: 10.5121/ijnsa.2021.13102.

[46] H. Li, K. Ota, and M. Dong, "Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing," *IEEE Netw*, vol. 32, no. 1, pp. 96–101, Jan. 2018, doi: 10.1109/MNET.2018.1700202.

[47] Z. M. Fadlullah *et al.*, "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2432–2455, Oct. 2017, doi: 10.1109/COMST.2017.2707140.

[48] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, pp. 1646–1685, Jul. 2020, doi: 10.1109/COMST.2020.2988293.

[49] H. Li, K. Ota, and M. Dong, "Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing," *IEEE Netw*, vol. 32, no. 1, pp. 96–101, Jan. 2018, doi: 10.1109/MNET.2018.1700202.

[50] F. Gottwalt, E. Chang, and T. Dillon, "CorrCorr: A feature selection method for multivariate correlation network anomaly detection techniques," *Comput Secur*, vol. 83, pp. 234–245, 2019.

[51] B. I. Farhan and A. D. Jasim, "Survey of Intrusion Detection Using Deep Learning in the Internet of Things," *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 1, pp. 83–93, 2022, doi: 10.52866/ijcsm.2022.01.01.009.

[52] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," *Cluster Comput*, vol. 22, pp. 949–961, 2019.

[53] R. C. Staudemeyer, "Applying long short-term memory recurrent neural networks to intrusion detection," *South African Computer Journal*, vol. 56, no. 1, pp. 136–154, 2015.

[54] R.-H. Hwang, M.-C. Peng, V.-L. Nguyen, and Y.-L. Chang, "An LSTM-based deep learning approach for classifying malicious traffic at the packet level," *Applied Sciences*, vol. 9, no. 16, p. 3414, 2019.

[55] G. Kim, H. Yi, J. Lee, Y. Paek, and S. Yoon, "LSTM-based system-call language modeling and robust ensemble method for designing host-based intrusion detection systems," *arXiv preprint arXiv:1611.01726*, 2016.

[56] K. Kim and M. E. Aminanto, "Deep learning in intrusion detection perspective: Overview and further challenges," in *2017 International Workshop on Big Data and Information Security (IWBIS)*, IEEE, 2017, pp. 5–10.

[57] Z. Liu *et al.*, "Deep Learning Approach for IDS," in *Fourth International Congress on Information and Communication Technology*, X.-S. Yang, S. Sherratt, N. Dey, and A. Joshi, Eds., Singapore: Springer Singapore, 2020, pp. 471–479.

[58] S. Venkatesan, "Design an intrusion detection system based on feature selection using ML algorithms," *Mathematical Statistician and Engineering Applications*, vol. 72, no. 1, pp. 702–710, 2023.

[59] M. Couceiro, P. Ghamisi, M. Couceiro, and P. Ghamisi, *Particle swarm optimization*. Springer, 2016.

[60] N. Kunhare, R. Tiwari, and J. Dhar, "Particle swarm optimization and feature selection for intrusion detection system," *Sādhanā*, vol. 45, no. 1, p. 109, Dec. 2020, doi: 10.1007/s12046-020-1308-5.

[61] S. Sarvari, N. F. Mohd Sani, Z. Mohd Hanapi, and M. T. Abdullah, "An Efficient Anomaly Intrusion Detection Method with Feature Selection and Evolutionary Neural Network," *IEEE Access*, vol. 8, pp. 70651–70663, 2020, doi: 10.1109/ACCESS.2020.2986217.

[62] R. Ramalingam, D. Karunanidy, S. S. Alshamrani, M. Rashid, S. Mathumohan, and A. Dumka, "Oppositional Pigeon-Inspired Optimizer for Solving the Non-Convex Economic Load Dispatch Problem in Power Systems," *Mathematics*, vol. 10, no. 18, Sep. 2022, doi: 10.3390/math10183315.

[63] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer," *Expert Syst Appl*, vol. 148, Jun. 2020, doi: 10.1016/j.eswa.2020.113249.

[64] J. O. Onah, S. M. Abdulhamid, M. Abdullahi, I. H. Hassan, and A. Al-Ghusham, "Genetic Algorithm based feature selection and Naïve Bayes for anomaly detection in fog computing environment," *Machine Learning with Applications*, vol. 6, p. 100156, Dec. 2021, doi: 10.1016/j.mlwa.2021.100156.

[65] K. P. Sit, I. Medi, and C. R. Ramachandiran, "Enhanced Genetic Algorithm in Image Reconstruction," in *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, IEEE, 2022, pp. 1456–1463.

[66] C. Li, J. Wang, and X. Ye, "Using a recurrent neural network and restricted Boltzmann machines for malicious traffic detection," *NeuroQuantology*, vol. 16, no. 5, 2018.

[67] M. Samadi Bonab, A. Ghaffari, F. Soleimanian Gharehchopogh, and P. Alemi, "A wrapper-based feature selection for improving performance of intrusion detection systems," *International Journal of Communication Systems*, vol. 33, no. 12, Aug. 2020, doi: 10.1002/dac.4434.

[68] Z. Halim *et al.*, "An effective genetic algorithm-based feature selection method for intrusion detection systems," *Comput Secur*, vol. 110, Nov. 2021, doi: 10.1016/j.cose.2021.102448.

[69] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," *IEEE Access*, vol. 10, pp. 99837–99849, 2022, doi: 10.1109/ACCESS.2022.3206425.

[70] S. Siva Shankar, B. T. Hung, P. Chakrabarti, T. Chakrabarti, and G. Parasa, "A novel optimization based deep learning with artificial intelligence approach to detect intrusion attack in network system," *Educ Inf Technol (Dordr)*, pp. 1–25, 2023.

[71] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique," *Cybersecurity*, vol. 5, no. 1, Dec. 2022, doi: 10.1186/s42400-021-00103-8.

[72] G. Parimala and R. Kayalvizhi, "An Effective Intrusion Detection System for Securing IoT Using Feature Selection and Deep Learning," in *2021 International Conference on Computer Communication and Informatics (ICCCI)*, IEEE, Jan. 2021, pp. 1–4. doi: 10.1109/ICCCI50826.2021.9402562.

[73] Z. Wang, J. Liu, and L. Sun, "EFS-DNN: An Ensemble Feature Selection-Based Deep Learning Approach to Network Intrusion Detection System," *Security and Communication Networks*, vol. 2022, pp. 1–14, Apr. 2022, doi: 10.1155/2022/2693948.

[74] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: a review," *J Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0268-2.

[75] A. Khanna and S. Kaur, "Internet of Things (IoT), Applications and Challenges: A Comprehensive Review," *Wirel Pers Commun*, vol. 114, no. 2, pp. 1687–1762, Sep. 2020, doi: 10.1007/s11277-020-07446-4.

[76] S. Bagchi *et al.*, "New Frontiers in IoT: Networking, Systems, Reliability, and Security Challenges," *IEEE Internet Things J*, vol. 7, no. 12, pp. 11330–11346, Dec. 2020, doi: 10.1109/JIOT.2020.3007690.

[77] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine Learning in IoT Security: Current Solutions and Future Challenges," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, pp. 1686–1721, Jul. 2020, doi: 10.1109/COMST.2020.2986444.

[78] K. Doshi, Y. Yilmaz, and S. Uludag, "Timely Detection and Mitigation of Stealthy DDoS Attacks via IoT Networks," *IEEE Trans Dependable Secure Comput*, 2021, doi: 10.1109/TDSC.2021.3049942.

[79] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: A survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, Jun. 2017, doi: 10.1016/j.jnca.2017.04.002.

[80] J. Ferdous, R. Islam, A. Mahboubi, and M. Z. Islam, "A Review of State-of-the-Art Malware Attack Trends and Defense Mechanisms," *IEEE Access*, vol. 11, pp. 121118–121141, 2023.

[81] P. Spadaccino and F. Cuomo, "Intrusion Detection Systems for IoT: opportunities and challenges offered by Edge Computing and Machine Learning," *arXiv preprint arXiv:2012.01174*, 2020.

[82] V. Jyothsna and K. M. Prasad, "Anomaly-Based Intrusion Detection System," in *Computer and Network Security*, J. Sen, Ed., Rijeka: IntechOpen, 2019, p. Ch. 3. doi: 10.5772/intechopen.82287.

[83] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset," *IEEE access*, vol. 9, pp. 22351–22370, 2021.

[84] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença Jr, "Network anomaly detection system using genetic algorithm and fuzzy logic," *Expert Syst Appl*, vol. 92, pp. 390–402, 2018.

[85] G. Dupont, J. den Hartog, S. Etalle, and A. Lekidis, "Network intrusion detection systems for in-vehicle network-Technical report," *arXiv preprint arXiv:1905.11587*, 2019.

[86] S. M. Mehibs and S. H. Hashim, "Proposed network intrusion detection system in cloud environment based on back propagation neural network," *Journal of University of Babylon for Pure and Applied Sciences*, vol. 26, no. 1, pp. 29–40, 2018.

[87] P. Jithu, J. Shareena, A. Ramdas, and A. P. Haripriya, "Intrusion Detection System for IOT Botnet Attacks Using Deep Learning," *SN Comput Sci*, vol. 2, no. 3, May 2021, doi: 10.1007/s42979-021-00516-9.

[88] S. Jain, P. M. Pawar, and R. Muthalagu, "Hybrid intelligent intrusion detection system for internet of things," *Telematics and Informatics Reports*, vol. 8, Dec. 2022, doi: 10.1016/j.teler.2022.100030.

[89] I. Apostol, M. Preda, C. Nila, and I. Bica, "Iot botnet anomaly detection using unsupervised deep learning," *Electronics (Switzerland)*, vol. 10, no. 16, Aug. 2021, doi: 10.3390/electronics10161876.

[90] P. L. S. Jayalaxmi, G. Kumar, R. Saha, M. Conti, T. Kim, and R. Thomas, "DeBot: A deep learning-based model for bot detection in industrial internet-of-things," *Computers and Electrical Engineering*, vol. 102, p. 108214, 2022.

[91] T. Saba, A. Rehman, T. Sadad, H. Kolivand, and S. A. Bahaj, "Anomaly-based intrusion detection system for IoT networks through deep learning model," *Computers and Electrical Engineering*, vol. 99, p. 107810, 2022.

[92] O. Jullian, B. Otero, E. Rodriguez, N. Gutierrez, H. Antona, and R. Canal, "Deep-Learning Based Detection for Cyber-Attacks in IoT Networks: A Distributed Attack Detection Framework," *Journal of Network and Systems Management*, vol. 31, no. 2, p. 33, Apr. 2023, doi: 10.1007/s10922-023-09722-7.

[93] S. Alosaimi and S. M. Almutairi, "An Intrusion Detection System Using BoT-IoT," *Applied Sciences (Switzerland)*, vol. 13, no. 9, May 2023, doi: 10.3390/app13095427.

[94] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, S. Garg, and M. M. Hassan, "A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network," *J Parallel Distrib Comput*, vol. 164, pp. 55–68, 2022.

[95] M. Mohy-eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection," *Multimed Tools Appl*, pp. 1–19, 2023.

[96] X. Liu and Y. Du, "Towards Effective Feature Selection for IoT Botnet Attack Detection Using a Genetic Algorithm," *Electronics (Switzerland)*, vol. 12, no. 5, Mar. 2023, doi: 10.3390/electronics12051260.

[97] P. Jithu, J. Shareena, A. Ramdas, and A. P. Haripriya, "Intrusion Detection System for IOT Botnet Attacks Using Deep Learning," *SN Comput Sci*, vol. 2, no. 3, May 2021, doi: 10.1007/s42979-021-00516-9.

[98] R. Deb and S. Roy, "A comprehensive survey of vulnerability and information security in SDN," *Computer Networks*, vol. 206, p. 108802, 2022.

[99] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *2016 international conference on wireless networks and mobile communications (WINCOM)*, IEEE, 2016, pp. 258–263.

[100] A. Fausto, G. Gaggero, F. Patrone, and M. Marchese, "Reduction of the Delays Within an Intrusion Detection System (IDS) Based on Software Defined Networking (SDN)," *IEEE Access*, vol. 10, pp. 109850–109862, 2022, doi: 10.1109/ACCESS.2022.3214974.

[101] S. Kranthi, M. Kanchana, and M. Suneetha, "A study of IDS-based software-defined networking by using machine learning concept," in *Advances in Data and Information Sciences: Proceedings of ICDIS 2021*, Springer, 2022, pp. 65–79.

[102] G. Logeswari, S. Bose, and T. Anitha, "An Intrusion Detection System for SDN Using Machine Learning," *Intelligent Automation and Soft Computing*, vol. 35, no. 1, pp. 867–880, 2023, doi: 10.32604/iasc.2023.026769.

[103] A. M. Ibrahimy, F. Dewanta, and M. E. Aminanto, "Lightweight Machine Learning Prediction Algorithm for Network Attack on Software Defined Network," in *APWiMob 2022 - Proceedings: 2022 IEEE Asia Pacific Conference on Wireless and Mobile*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/APWiMob56856.2022.10014244.

[104] G. Farahani, "Feature Selection Based on Cross-Correlation for the Intrusion Detection System," *Security and Communication Networks*, vol. 2020, 2020, doi: 10.1155/2020/8875404.

[105] M. W. Nadeem, H. G. Goh, V. Ponnusamy, and Y. Aun, "Ddos detection in sdn usingmachine learning techniques," *Computers, Materials and Continua*, vol. 71, no. 1, pp. 771–789, 2022, doi: 10.32604/cmc.2022.021669.

[106] G. B. Gaggero, P. Girdinio, and M. Marchese, "Advancements and research trends in microgrids cybersecurity," *Applied Sciences*, vol. 11, no. 16, p. 7363, 2021.