A Taxonomy of IDS in IoTs: ML Classifiers, Feature Selection Models, Datasets and Future Directions

Hessah Alqahtani, Monir Abdullah*
College of Computing and Information Technology, University of Bisha, Bisha, Saudi Arabia

***Abstract*—The applications of the Internet of Things (IoT) are becoming increasingly popular nowadays. Network security and privacy are major concerns of the IoTs, as many IoT devices are connected to the network via the Internet, making IoT networks more vulnerable to various cyber-attacks. An Intrusion Detection System (IDS) is a solution to deal with security and privacy issues by protecting IoT networks from different types of attacks. In this paper, we provide a taxonomy of IDS in IoT. Different Machine Learning (ML) classifiers, feature selection models, and Datasets with high detection accuracy are presented. Our analysis indicates a heightened emphasis on ML-based IDS, with Support vector machines (SVMs) at 33% and RFs at 31% being the most widely used classifiers. Despite the diversity in the use of different datasets for IDS, the NSL-KDD is the most commonly used in 49% of studies. In the realm of feature selection, the K-means and SMO algorithms emerge with an impressive 99.33%, marking the highest percentage in previous research on feature selection for ML-based ID. Moreover, we addressed the future pathways and challenges of IDS detection.**

Keywords*—*Intrusion detection system; feature selection; support vector machine; random forest; decision tree; NSL-KDD

I. Introduction

Massive advancements in telecommunications networks and the introduction of the idea of the IoT are the results of incredible increases in the ordinary usage of electronic services and applications. Devices are objects, or "things," in the IoT, a developing communications paradigm that allows them to detect their surroundings, communicate with one another, and share data. Recently, the IoT paradigm has been used in the development of smart environments, including smart homes and cities, with a range of application areas and associated services. By resolving issues with the living environment, energy use, and industrial requirements, the development of such smart settings aims to improve human productivity and comfort. IoT offers a range of applications, including health monitoring, smart water, smart cities, smart environments, and smart homes. An enormous number of issues are emerging with the growth of IoT applications. IoT security is a concern that cannot be disregarded among many other difficulties. Because IoT devices may be accessible from anywhere over an untrusted network, such as the Internet, IoT networks are vulnerable to a wide range of malicious attacks. In the event that security flaws are not fixed, confidential data might leak at any time. As a result of the significant advancement in the realm of information technology, network security has emerged as one of the most challenging issues. The fundamental security guidelines governing network communication aim to restrict unauthorized users from accessing the network. There is still a lot of unstructured networking activity that follows different kinds of server assaults. These attackers sign on to the network as users to steal data from the server database. These dangerous actions might be prevented with the use of an IDS.

- Intrusion Detection Systems

Intrusion is an unnecessary or malicious activity that is dangerous to sensor nodes. A network's malicious traffic can be detected using an ID system, serving as an extra layer of security to keep hackers out of the network. IDS may be utilized as a hardware or software tool. IDS can scan and analyze user and machine behavior, identify patterns of known attacks, and classify harmful network traffic. IDS monitors networks and nodes, finding different types of network intrusions and notifying users of these intrusions. As a network observer or alarm system, the IDS prevents system harm by sounding a warning before an attacker launches an attack [1,14].

Both external and internal assaults can be detected by it. Whereas external attacks are started by outside networks and launched by third parties, internal assaults are started by malevolent or compromised network nodes. IDS scans the network packets to identify if they come from authorized users or attackers. ID is made up of three parts: alarm, analysis and detection, and monitoring. The monitoring component keeps an eye on resource use, traffic trends, and network traffic. The Analysis and Detection module of IDS detects intrusions according to a set of algorithms. If an intrusion is detected, the alarm section raises an alert. Originally, network attack detection and monitoring for this IDS were done manually. In the future, this ID system will be automated and fixed as a web application to identify malicious nodes before they infiltrate the network. There are two kinds of this kind of IDS [1].

1) Host-based IDSs: they are employed to identify irregularities in computer systems.

2) IDS-based on the network, which finds irregularities in the network environment.

The two types of network-based IDS are signature-based and anomaly-based. Anomaly-based Network NIDS is used to identify new attacks by identifying a user's typical network activity. In contrast, signature-based NIDS identifies attacks by comparing the payload of arriving packets to signatures stored in the signature database. A signature is a pattern or guideline used to identify known attacks, but it cannot identify unidentified assaults. More training data is needed for signature-based NIDS to distinguish attack types from regular data. A departure from typical behavior and the observed occurrence may be seen as invasive. One drawback of anomaly-based NIDS

*Corresponding Author.

is that it is difficult to establish typical behavior due to the diversity of network traffic [1].

II. Literature Review

Modern communication technologies, notably the Internet of Things (IoT), have surpassed traditional environmental sensing methods significantly. IoT technologies empower the collection, measurement, and understanding of surrounding environments, enabling advancements that enhance quality of life. This circumstance enables the realization of smart cities, facilitating novel forms of communication between objects and individuals. IoT stands as one of the most rapidly expanding sectors in computer history. The author postulates in study [2] that IoT technologies play a vital role in enhancing practical smart applications such as smart homes, transportation, healthcare, and education. However, the widespread and interconnected nature of IoT systems, along with their numerous components, has introduced additional security concerns. Ensuring security in IoT systems with extensive attack surfaces presents a significant challenge. As noted in study [2], IoT devices are predominantly deployed in uncontrolled environments, leaving them vulnerable to physical access by intruders. Additionally, IoT devices are typically interconnected via wireless networks, exposing them to potential eavesdropping and unauthorized access by hackers. Addressing security requirements necessitates comprehensive solutions. The author mentioned in study [3] emphasizes the importance of safeguarding the availability and integrity of these systems against diverse threats. Consequently, IoT security has become crucial for societal well-being. Moreover, ensuring security requires robust ID and Prevention Systems (ID/PSs) to identify security vulnerabilities effectively.

To comprehend ID/PSs, one needs to grasp the nature of the threats they aim to identify. An incursion denotes a type of assault on information assets, wherein the attacker seeks to infiltrate a system or disrupt its normal operations. In study [3], the authors specify that an intrusion refers to an effort to circumvent the security protocols of a computer system. Such actions encompass a range of activities that pose a risk to the availability, confidentiality, or integrity of both data and the information system. Confidentiality implies that data remains undisclosed and inaccessible to unauthorized parties, entities, or processes, while integrity ensures that data has not been illicitly altered or destroyed. Availability refers to the guarantee that a system with the necessary data will be available and useable when called upon by a legitimate system user. The author stated in study [3] that on occasion, an intrusion is brought about by an attacker using the operating system of the compromised device, the internet, the network, or any security hole in third-party (middleware) programs that control the information system. Outsider assaults are those that originate from outside sources. Unauthorized internal users trying to obtain and abuse non-authorized access rights are known as insider attacks. ID is the process of keeping an eye out on networks or PCs for any unwanted activity, entrance, or file alteration. The ID process can be automated with an IDS, which can be either hardware or software-based. IDSs have many options for handling suspicious events: they can log the occurrence, provide an alert, or even call an administrator. The process of detecting identified system threats in real time and stopping them from reaching their intended targets is known as intrusion prevention. It works well against brute force attacks, floods, and Denial of Service (DoS) attacks. A software or hardware tool with all the features of an IDS plus the ability to prevent potential occurrences is called an intrusion prevention system (IPS). When preventative mechanisms in IPS devices are disabled, they often behave as IDSs. Although both IPS and IDS scan network traffic for threats, IPSs and IDSs differ significantly. IPSs are thought of as an extension of IDSs. The study of [3] indicates both IDS and IPS may identify undesired or harmful traffic. They both respond differently, but they both try to do it as fully and properly as they can. As seen in Fig. 1, the main purpose of an IDS is to alert users to potentially harmful actions. In contrast, IPS is created to enhance the IDS and other conventional security solutions by promptly responding to halt or prevent intrusions with more proactive protection.

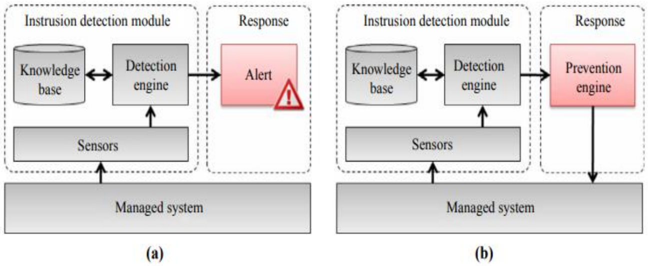


Fig. 1. ID and IP systems [3] (a) ID (b) IP.

A. Intrusion Detection System

The author defines in [4] that an IDS is a system that automatically conducts the ID process. ID itself refers to any sort of mechanism to identify such intrusive behavior. IDS monitors the network's data flow and scrutinizes any suspicious activity that poses a threat to network security. IDS is classified into two categories, which are Host-Based IDS and Network-Based IDS.

B. Host-Based IDS (HIDS)

The term HIDS refers to the detection of intrusions using data collected from a single or several host systems. The author mentioned in study [5] that the activities monitored by the HIDS operator include machine logs, host-based community traffic, document modifications, system integrity, and utility activity. Through the use of file timestamps, device logs, machine calls from visual displays, and frequent hashing tools, the local community interface provides the agent with information on the current state of the nearest host. A pop-up notification notifies the user of any unlawful changes or actions, and it can also notify the central management server, block the activity, or do all three at once. The policy that is put on the local system serves as the main basis for the option. These host-based strategies are considered passive elements.

C. Network-Based IDS (NIDS)

NIDSs serve the purpose of monitoring and scrutinizing users on a community site to safeguard a system against attacks originating from the community, where data is transmitted through a network. The study in [5] highlights NIDS' capability to discern malicious activities and reveal the network of assaults initiated by visitors. To monitor the movement of packets, NIDS employs various sensors. The system is designed to identify intrusion patterns by progressively analyzing packets, either in real-time or very close to it. In ID, the analysis of visitor patterns can be carried out using sensors, administrative servers, or a combination of both. The NIDS approach is considered an active component in this context.

D. Intrusion Detection Techniques

The study in [5] discusses the techniques for Detecting Intrusions. Anomaly-based detections and signature-based detections are the two main methods used by NIDSs to identify threats. Anomaly-based data is gathered from system traffic and compared to the gold standard for typical traffic and system behavior. An alarm is set off by the system exhibiting anomalous behavior. Its benefit is that it can identify intrusions that the system had not previously recognized, effectively identifying novel attack patterns. However, there are a lot of false alarms when employing this strategy because the system is alerted to any abnormality. It's also feasible for certain assaults to pass unnoticed because they conform to typical behavioral criteria. In contrast to anomaly-based detections, signature-based detections employ a variety of algorithms to identify attacks to produce more accurate findings. They base their warnings on established attack pattern signatures prior to notifying administrators.

E. Intrusion Prevention System (IPSS)

The author in study [4] defines the IPS as a system that detects both intrusions and takes responsive actions to mitigate such intrusions. IDS detects network intrusions at the host level, whereas preventive measure tools—frequently implemented through hardware—prevent the network from different types of assaults. Together, these components make up the IPS. As a result, the IPS not only recognizes attacks but also automatically counters them by implementing various actions, including logging users off of the system, terminating processes, shutting down the system, cutting the connection, etc.

III. Taxonomy of IoT Security Approaches

A. IDS Approaches

In the work [6], the main problem is network security. They proposed to offer ID where the data was labeled as normal or invasive using ML classifiers listed, including SVM, K-nearest neighbor (KNN), logistic regression (LR), NB, Multi-layer Perceptron (MLP), RF, Extra tree classifier (ETC), and DT. Using four distinct feature subsets taken from the NSLKDD dataset, the model's performance was investigated. Using RF, extra-tree, and DT classifiers on all four feature subsets, an accuracy of more than 99% was attained overall.

The author noted in study [7] that security poses a significant concern in the realm of IoT. A Deep Learning-based IDS (DLIDS) is proposed to detect security threats in IoT environments, aiming to overcome the challenges of IoT devices security. For higher detection accuracy, the Spider Monkey Optimization algorithm (SMO) is combined with the Stacked-Deep Polynomial Network (SDPN). SMO determines the optimal features for each dataset, while SDPN classifies them appropriately.

It evaluated the performance of DL-IDS using the NSL-KDD benchmark dataset and achieved a 99.02% accuracy rate.

The author assumed in study [8] that one crucial piece of equipment for network security defense is an IDS. Several ML techniques have been proposed to create Anomaly-based IDS (AIDS). It utilizes ten well-known supervised and unsupervised ML algorithms to find efficient and successful ML-AIDS in computers and networks.

An unbalanced multiclass dataset from CICIDS2017 is used to test the ML-AIDS models. They evaluated the performance of the tested ML-AIDS. Generally, KNN, DT, and NB algorithms are more capable of detecting web attacks than other algorithms.

In study [9], the authors identify breaches in IoT devices by presenting a hybrid model that combines shallow and DL. The suggested approach aims to identify the most significant characteristics first, using a spider monkey optimization feature selection technique. To improve data classification, a Siamese neural network-based model is then proposed. The suggested model used the NSL-KDD dataset to test to assess its performance. The accuracy of the proposed model, calculated with a RF classifier, is 94.69%.

The authors of study [10] investigated and employed efficient feature selection strategies to enhance ID through ML techniques. The proposed approach centers around a centralized IDS. Training the model to recognize malicious and unusual activities in network traffic involves utilizing deep feature abstraction, feature selection, and classification through artificial neural networks, SVM, DTs, and NB DL algorithms. The effectiveness of the suggested method is demonstrated on the Aegean Wi-Fi Intrusion Dataset through experimental results, showcasing a high detection accuracy of up to 99.95%.

In addition, the authors in study [11] provide a feature selection and KNN classifier-based network ID model for IoTs scenarios. To increase the accuracy (ACC) and detection rate (DR) of the IDS, they constructed the NIDS utilizing the KNN algorithm. Additionally, to enhance the quality of the data and identify the top 10 features, principal component analysis (PCA), univariate statistical tests, and Genetic Algorithm (GA) are applied independently for feature selection. The Bot-IoT dataset is used to assess the model's performance. The models have demonstrated encouraging results in terms of ACC, DR, false alarm rate (FAR), and prediction time after applying the feature selection. The suggested model has an accuracy of 99.99%.

The researchers proposed in study [12] an IoT network ID solution that utilizes a hybrid convolutional neural network model to identify various assault types. The suggested paradigm can benefit a variety of IoT applications. The model is assessed using the UNSW NB15 dataset.

The proposed model has been experimentally validated and compared to the traditional recurrent neural network, achieving a superior detection accuracy of 98%.

Authors hypothesize in study [13], that security methods for communication must progress. They suggest using DL architectures to create a robust and adaptable IDS that can identify and categorize network threats. The focus is on how DL and deep neural networks (DNNs) might enable adaptive IDS that can identify and eliminate unknown or zero-day network behavioral characteristics, therefore expelling system intruders and mitigating compromise risk.

To showcase the model efficacy, the UNSW-NB15 dataset was employed, yielding a model performance with an accuracy of 95.6%.

The author discussed in study [14] the problems in the realm of computer network security. The SVM model is proposed to identify malicious activity on short-range, low-power, and low-rate networks, particularly those seen in the IoT. Two SVM techniques were evaluated; the OC-SVM only observes normal behavior activity, while the C-SVM requires two classes of vector values—one for normal activity and one for aberrant activity. Both methods were applied as components of an IDS. The author's specialized network-layer assaults were utilized to generate and assess the SVM detection models using real network traffic. It is demonstrated that when assessed in an unknown topology, the C-SVM obtains an accuracy rate of 85.1%.

The researchers proposed in study [15] a new method for selecting and extracting features for anomaly-based IDS. Two methods based on entropy—information gain (IG) and gain ratio (GR)—are used in this method to choose and extract pertinent features in a range of ratios. To extract the best characteristics, one uses the union and intersection of mathematical sets theory.

In the IoT, the intrusion dataset 2020 (IoTID20) and the NSL-KDD dataset are used to train and evaluate the model using four ML algorithms: IBk, J48, Multilayer Perception, and Bagging. The model's classification accuracy is a very high 99.98%.

B. Machine Learning Approaches

The author stated in study [16] that the use of network security technology to identify new attacks is crucial. Two models, one for multi-class and another for binary classification, were introduced to incorporate DL techniques in the detection of network attacks. These models leverage a deep neural network algorithm for enhanced accuracy.

This experimental investigation focuses on multi-class classification and utilizes the KDD Cup 99 datasets. The excellent accuracy of the suggested approach (99.98% for both binary and multiclass classification) has yielded positive results.

The author mentioned in study [17] the problem with security related to bot attacks. The BoT-IoT dataset served as the training data for a model developed through various ML techniques, such as KNN, Naive Bayes (NB), and Multi-layer Perceptron Artificial Neural Network (MLP ANN).

A standard was set to determine the top-performing algorithm by assessing accuracy percentage and the area under the receiver operating characteristics curve (ROC AUC) score. ML methods were improved by incorporating feature engineering and integrating the Synthetic Minority Oversampling Technique (SMOTE). The suggested model attained an accuracy rate of 92.1%.

IoT devices are vulnerable to various security threats, including but not limited to DoS attacks, network intrusions, and data breaches. The study of [18] presents a novel security framework based on ML that automatically handles the growing security concerns associated with the IoTs. In order to mitigate risks, Network Function Virtualization (NFV) and Software Defined Networking (SDN) tools were employed. This AI framework incorporates anomaly-based ID into IoT systems utilizing a one-class SVM along with both a monitoring agent and an AI-driven response agent. The response agent utilizes ML models divided into network pattern analysis.

The evaluation of the framework based on the NSL-KDD dataset demonstrates the efficiency of the proposed scheme, achieving a 99.71% accuracy.

In study [19], the authors discussed the random access (RA) dilemma, in which massive machine-type communication (mMTC) applications are served by allocation algorithms that experience excessive signaling overhead and congestion. Consequently, a novel FUG resource allocation technique based on SVM and LSTM was presented. We apply the CMMPP traffic model with mixed alert and normal traffic to evaluate the suggested FUG allocation against other available allocation strategies. The model is employed in a denser network to assess the suggested method as well.

The proposed technique was tested using real-time measurement data gathered from the database of the Numenta Anomaly Benchmark (NAB). Furthermore, the evaluation results achieved an accuracy of 98%.

IoT management faces significant difficulties in terms of safety and confidentiality. The researchers proposed in study [20] an integrated approach, a combination of optimization-based and DL-based techniques called DCCNN-SMO, advocated for detecting software piracy using reference codes that have been stolen. The Hybrid Dual-Channel Convolution Neural Network (DCCNN) with Spider Monkey Optimization (SMO) is a DL technique designed to detect files that include malware and illegal software over the IoTs network.

In investigating software piracy, data for the study was collected from Google Code Jam (GCJ) for the dataset, while malware samples were sourced from the Leopard Mobile database for testing purposes. The proposed method yielded a higher detection accuracy rate of 98.12%.

In addition, to identify anomalies and intrusion attacks in IoT networks, the authors in study [21] suggested a unique CorrACC feature selection metric technique and used a bijective soft set for successful feature selection. To filter the features and choose the best features for a certain ML classifier using the ACC metric, a novel feature selection method called Corracc based on CorrACC is designed and developed. They employed four different ML classifiers on the BoT-IoT dataset to evaluate their suggested techniques. The experimental findings of the algorithms show an accuracy of more than 95%.

C. Machine Learning in IDS

The author in study [22] defines ML as a branch of AI. Without explicit programming, ML enables a system to learn from experience and enhance its autonomous capabilities. ML algorithms are more effective in quickly and reliably identifying assaults against large amounts of data in IDS. The three categories of ML algorithms are Supervised, Unsupervised, and Semi-supervised.

The study of [23] discussed the classifiers that can help IDSs based on anomalous progress in their development. This study's primary objectives are to encourage academics studying IoT security to create IDSs that use ensemble learning and to provide suitable techniques for statistically evaluating classifier performance. The statistical analysis of the noteworthy differences among classifiers is done using the Friedman and Nemenyi tests. Additionally, classifier response times on IoT-specific hardware are assessed using Raspberry Pi. Classifier performance is evaluated using widely used metrics and validation techniques. For classifier benchmarking, popular datasets such as CIDDS-001, UNSW-NB15, and NSL-KDD are utilized. The model uses the XGB classifier to obtain 98.77%.

IV. Classification Approaches

The process of classifying entails taking each and every instance of the dataset under examination and allocating it to one of two classes: normal or abnormal, where new examples are assigned to the known structure. Although it is more commonly used for abuse detection, it might be useful for anomaly detection as well. The datasets were grouped using classification into predefined sets. In [24], it is reported that IDS uses a variety of classification approaches, including SVM, NB classifiers, DTs, and K-nearest neighbor classifiers.

A. Classification Techniques for Intrusion Detection

The authors in [25] described a data mining framework for adaptively building ID models. Data mining techniques were employed to calculate abuse and anomaly detection models based on observed behavior in the data. Table I shows the following classification methods that are frequently used to categorize ID: KNN Classifiers, DT, Bayesian Classification, NNs, SVM, and RF.

TABLE I. Intrusion Detection Classification Techniques.

Classification Techniques	**Classification Task**		**Classifier Approaches**		**Algorithms category**
	Binary	**Multi-Class**	**Single**	**Hybrid**	
DT	Yes	Yes	Yes	Yes	Non-probabilistic
KNN	Yes	Yes	Yes	Yes	
NB classifier	Yes	Yes	Yes	Yes	
SVM	Yes	No	Yes	Yes	

B. Single and Hybrid Classifier Approaches

1) Naive bayes: The NB model is a probabilistic classifier that predicts the class based on the likelihood of membership. In [24], the investigation explores the correlation between independent and dependent variables to ascertain conditional probability. According to the Bayes Conjecture:

$$\mathrm{P(H/X)} = P(X/H) * P(H)/P(X)$$

Here, if H represents the hypothesis that pertains to data X and belongs to class C, and X denotes the data record, the posterior probability of H conditioned on X is P(H/X), the posterior probability of X conditioned on H is P(X/H), and the prior probability is P(H). Naive Bayes is straightforward to construct and does not necessitate complex iterative parameters. It can handle large datasets efficiently, although its complexity escalates over time.

2) In study [26], it was noted by the author that NB performs remarkably well in scenarios where there exist moderate dependencies in the data. The efficacy of the NB classifier is found to increase when employing a feature subset identified by CFS, albeit at the cost of time. A study in [27] conducted an empirical analysis on the KDD Cup '99 dataset, comparing the performance of NB and DT. Despite DT achieving higher accuracy (92.28% compared to 91.47%), NB achieved superior detection rates. Researchers in [28] proposed a network IDS framework established using NB. Through experiments conducted on a 10% subset of the KDD99 dataset, the system achieved a detection rate of 95% with a 5% error rate. The model was also built faster (1.89 seconds) and more efficiently.

3) Decision tree: A DT is a tree-like, recursive structure used to express classification rules. It divides based on attribute values using the divide and conquer strategy. Data is categorized starting at the root node and moving via leaf nodes, each of which indicates an attribute and its value as well as the class label of the data. Tree-based classifiers perform best when dealing with large datasets. In study [24], the authors discussed a variety of DT algorithms, which are explained below:

a) ID3 algorithm: It is a well-known DT algorithm that Quinlan created. The ID3 algorithm builds DTs based on training datasets primarily using attribute-based algorithms. The root of the tree is the characteristic with the biggest information gain.

b) C4.5 algorithm: It was created by Ross Quinlan and is based on the ID3 algorithm. Using information gain to build a DT, the characteristic with the highest information gain is chosen for decision-making. This algorithm's primary drawback is that it requires more CPU time and memory to run. An additional distinct tree-based classifier.

c) AD Tree: Alternating DT is used for categorization. AD Prediction nodes are found in both the leaf and root nodes of AD trees.

d) NB tree: DTs and NBs classifiers are both used by the tree algorithm. DT classifiers are used by the root node and NB classifiers by the leaf nodes.

e) RF: Lepetit et al. initially presented RF, an ensemble classification method made up of two or more DTs. Every tree in RF is created by selecting data at random from the dataset. Because RF is less susceptible to outlier data, it increases accuracy and predictive power. It can handle high dimensional data with ease.

4) K-Nearest neighbor: It's among the most basic methods of categorization. The author mentioned in [24] that the unlabeled data point is assigned to the nearest neighbor class once the distance between various data points on the input vectors is calculated. K is a crucial parameter. The item is placed in the class of its closest neighbor if k=1. When K is high, the prediction process takes a long time and affects accuracy by lessening the impact of noise.

5) SVM: In study [24] the authors define a supervised learning technique for categorization and prediction as SVM. Because it is a binary classification classifier, it uses a hyperplane to divide data points into two classes, +1 and -1. For regular data, the value is +1; for questionable data, it is -1. The expression for a hyperplane is: W. X + b = 0 where X={x1,x2,......,xn} are attribute values, b is a scalar, and W={w1,w2,.......,wn} are weight vectors for n attributes A={A1,A2,..........,An}. Finding a linear optimum hyper plane to maximize the margin of separation between the two classes is the primary objective of SVM. A subset of the data is used by the SVM to train the system.

C. Clustering

In study [29], the authors define Clustering methods function by grouping observed data into clusters using a designated similarity or distance metric. The commonly used process for this task involves selecting a representative point for each cluster. By using clustering algorithms, the amount of work needed to optimize the IDS is decreased since intrusion events can be found merely from the raw audit data. K Means, a nonhierarchical Centroid-based clustering method, is one of the most well-liked and often used clustering techniques. In [30], the authors discussed the partitioning approaches, density-based, model-based, search-based, and other types of methodologies may be used to broadly classify clustering techniques. Table II shows the ID Clustering techniques.

TABLE II. INTRUSION DETECTION CLUSTERING TECHNIQUES

Clustering Technique	Advantage	Disadvantage
Hierarchical clustering	- Unnecessary input parameters - Ease of implementation	- Interpretation issues - sensitive - Rollback problems
Based on Partitioning	- Simple, Powerful, Scalable - Understandable	- Difficulty predicting - sensitive
Based on Grid	- Divide space into a finite quantity. - Statistical information independently - Incremental and efficient update	- Poor locating performance. - Requires careful selection
Density based	- Random formation. - Ability to withstand noise and outliers.	- Work inefficiently with large and sparse data sets. - Not suitable for high-dimensional datasets
Model-based	- Easy to interpret. - Flexibility	- Requires more data. - Quality of predictions

1) Partitioning methods: Partitioning techniques divide the characteristics into subgroups and cluster the data using distance-based matrices. The author stated in [30], these matrices function based on the similarity of any unsupervised feature assessment standard. After one level of partitioning, this approach yields nonoverlapping spherical shaped clusters. There are three categories of partitioning methods: subspace clustering, relocation based, and grid based.

2) Hierarchical clustering or Connectivity based clustering: Using these approaches, clusters are represented as a dendrogram, which is a tree, as opposed to being shown as a circular, ovoid, C, or S shape. This clustering method is challenging. The author mentioned in [30], the hierarchical clustering is done using two different approaches: divisive (top-down) and agglomerative (bottom-up). Three types of linkages can serve as the foundation for hierarchical clustering techniques: average, full, and single links. One of the main drawbacks of hierarchical clustering is that a descriptor cannot be included in another hierarchy cluster once it has been included in one.

3) Model based clustering methods: These techniques group the data according to a certain mathematical model. The author assumed in [30] that the two model-based clustering techniques that are most commonly employed are "Decision Trees" and "Neural Networks."

4) Grid based: These methods quantize space by dividing the input data into a number of grids of equal size. These grids are used for all clustering operations. Grid-based techniques handle a grid's limited amount of features rather than a huge number of features, which reduces computational complexity and makes them quicker.

5) Density based: These techniques create clusters around densely populated locations within a subset of the chosen data. Round, concentric clusters are generated if all of the data subsets are concentrated around a single point; irregularly shaped clusters, such as S- or C-shaped clusters, When the densities of the data subsets match, clusters are created. While low density regions will keep data points from distinct clusters apart, dense regions will group data points together to create clusters.

D. Clustering Algorithms

1) K-Means Clustering algorithm: The study [24] also presented the K-Means clustering algorithm, proposed by James Macqueen, is a straightforward and widely employed clustering technique. By classifying occurrences into a predetermined number of clusters, the user specifies the number of clusters K in this process. Selecting k instances to serve as cluster centers is the first stage in the K-Means clustering process. Next, place each dataset instance in the closest cluster.

2) K-Medoids clustering algorithm: In [24], the K-Medoids clustering algorithm is discussed, which operates similarly to the K-means algorithm through a partitioning mechanism. However, instead of computing the mean value of objects in a K-Means cluster, the centroid of a cluster is determined by selecting the most centrally located instance within the cluster. The terms "reference point" and "medoid" refer to this centrally positioned item. By minimizing the squared error, it aims to reduce the distance between the centroid and the data points. In scenarios with a high number of data points, the K-Medoids method demonstrates superior performance compared to the K-Means algorithm. The medoid is less influenced by outliers, thus offering robustness against noise and outliers, albeit at the expense of increased computational complexity.

V. Intrusion Detection Dataset

The study in [31] discusses the datasets that have a significant impact on the assessment of NIDS, which is useful for testing and approving novel methods. Benchmark datasets were used by researchers to assess their findings. Nevertheless, the datasets that are publicly accessible lack actual features of contemporary network traffic. Furthermore, NIDS cannot adjust to the ongoing modifications in networks. Because networks are dynamic, relying just on historical datasets hinders the development of NIDS. The fact that the network is always changing should be taken into account while creating fresh datasets. The datasets are shown in Table III.

TABLE III. Intrusion Detection Dataset

Dataset	Realistic Traffic	Number of features	Number of attacks	Label data	Year
KDD cup 99	✓	42	4	✓	1999
NSL-KDD	✓	42	4	✓	2009
CICIDS2017	✓	86	14	✓	2017
UNSW-NB15	✓	49	9	✓	2015
CIDDS-001	✓	14	5	✓	2017

1) KDD99 dataset: The study described in [31] aims to introduce a tool utilized at MIT, developed for the KDD99 International Knowledge Discovery and Data Mining Tool Competition. The benchmark dataset utilized for IDS was the KDD99 dataset from DARPA. Despite being generated in 1999, the KDD99 dataset has remained the most commonly utilized dataset for assessing anomaly detection. It comprises 4,898,431 instances, each characterized by 42 features, as outlined in Table IV. The KDD99 dataset includes a single normal attack type along with 22 training attack types, with the testing data featuring an additional 17 types. Among the 41 features, there are labels distinguishing them as standard or specific attack types (DOS, U2R, R2L, and Probe). It is believed that by leveraging insights gained from documented attacks, it becomes possible to identify similar attacks.

2) NSL-KDD dataset: The KDD99 dataset was transformed into the public dataset known as NSL-KDD. A statistical examination of the KDD99 dataset uncovered significant issues that significantly affect ID accuracy and lead to an erroneous evaluation of IDS performance. In study [31], the authors assert that the primary issues stem from the abundance of duplicate packets in both the training and testing data, as well as from the analysis of the KDD99 dataset. It was found that 78% of network packets in the training set and 75% in the test set were duplicated. This prevalence of duplicate instances skews the training set towards normal cases in ML techniques, thereby shielding them from attacks that often pose greater threats to computer systems. Due to the lack of publicly available network-based IDS datasets, the updated KDD99 dataset still has certain issues and might not accurately reflect modern real networks. Nevertheless, it can still be used as a useful dataset to assist researchers in comparing various ID strategies.

3) UNSW-NB 15 dataset: In order to extract a combination of current normal and modern attack behaviors, the IXIA Storm tool was used in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) to build the UNSW-NB 15 dataset. The study of [31] describes one of the more recent datasets for analyzing NIDS; academics have had access to it since late 2015. To facilitate packet analysis, 100 Gigabytes (GB) of raw network data were captured using the tcpdump program. Each pcap file has 1000 MB. Twelve tools and algorithms, including Argus and Bro-IDS, were used in tandem on the UNSW-NB15 dataset. There are four CSV files with 2, 540,044 occurrences and 49 characteristics altogether. The UNSW-NB15 dataset categorizes its attributes into six main classes. These include thirteen fundamental features, eight content features, nine time-related features, seven connection-specific features, twelve supplementary features, and two features designated for class labeling. Each data instance within the dataset is characterized by a total of 49 attributes that detail various aspects of network connections.

4) CIDDS-001 dataset: A labeled flow-based dataset is the Coburg ID Dataset, or CIDDS-001. The goal of this dataset was to evaluate the effectiveness of an anomaly-based NIDS. The study in [31] discussed that the CIDDS-001 dataset is made up of unidirectional NetFlow data that is collected from an OpenStack environment that has external servers (web server and file synchronization) and internal servers (backup, mail, file, and web). These servers are deployed online to collect traffic that is current and in real time. Realistic normal and attack traffic is included in the CIDDS001 dataset, which makes it possible to test NIDS in a cloud context. It is made over the course of a week and is separated into four sections. There are 14 features total; the first 10 are NetFlow default features, while the latter 4 are extra features. There are 16 million flows in the CIDDS001 dataset. It was caught for two weeks at a time. The dataset contains assault flows for each of the four categories of attacks (suspicious, attacker, unknown, and victim).

5) CICIDS2017 Dataset: The relatively recent CICIDS2017 dataset was developed by the Canadian Institute for Cybersecurity IDS. CICIDS2017 represents an enhanced iteration of the ISCX2012 dataset, incorporating contemporary network attacks while fulfilling all criteria for real-world attack scenarios. Since its introduction, academics have been attracted to the CICIDS2017 dataset for the evaluation and development of new models and algorithms, as highlighted in [31]. This dataset comprises labeled network flows, encompassing complete packet payloads in PCAP format, accompanying profiles, labeled flows (contained in GeneratedLabelledFlows.zip), and CSV files tailored for ML and DL applications (MachineLearningCSV.zip), all of which are freely accessible to researchers. The ML CSV.zip file within the CICIDS2017 dataset contains eight CSV files illustrating network traffic profiles spanning five days, each encompassing both normal and attack traffic instances.

VI. Intrusion Detection Feature Selection

Four types of features are often present in complicated, multidimensional data: (i) high weighted features (most important and non-redundant) (ii) characteristics with a medium weight (not redundant, but somewhat relevant) less-weighted features (i.e., redundant and weakly relevant information) and zero-weighted features (i.e., noise or wholly irrelevant features). A study by [30] found Feature selection (FS), often referred to as variable or attribute selection, is the process of selecting the most pertinent characteristics from the data while removing less-weighted and unnecessary features. As a result, processing time and computing costs are decreased while prediction accuracy and extracted information validity are improved. A data set with "n" dimensions would have 2n -1 properties, and if "n" is too big, it could be computationally impossible to analyze the data. By choosing important characteristics, FS is helping to minimize data dimensions and end the "curse of dimensionality" associated with huge data. A review of the literature demonstrates that "classifications" using FS perform faster and more accurately than "classifications" using no FS. FS (FSAs) algorithms can be classified as Unsupervised data sets don't have labels applied to them, semi-supervised data sets have labels applied to certain parts of the data, and supervised data sets have labels applied to every component of the data. The study of [30] aims to provide four types of FSAs may be distinguished based on the techniques used for feature searching: Filter, Wrapper, Embedded, and Hybrid techniques.

1) Filter method: Four criteria are used by filter techniques to analyze the features: information theory, dependence, consistency, and distance. Without the use of algorithms, filter techniques use the intrinsic properties of the data to identify the most discriminative features out of all of them. The degree of association between the output class label and a selected feature is computed via filters. Correlation scores, or degrees of correlation, are used to rank characteristics, with the highest-ranking features being chosen. Filtering techniques need less computing power and are quicker.

2) Wrapper method: Using classification accuracy as the fitness function, subsets of the most pertinent features are chosen and assessed one at a time in wrapper approaches rather than individual features. These are closed-loop techniques that are used in algorithms for both classification and clustering. The techniques employed in wrapper methods include recursive feature removal, forward selection, and backward selection. Wrapper techniques are far slower and need more computing power than filter methods since they involve repeated assessment. Wrappers may be random or deterministic. While deterministic wrappers are used with sequential forward selection (SFS), Plus-L Minus-R selection (LRS), Smart Beam search (SBS) algorithms, and sequential backward elimination (SBE), randomized wrapper-based FSAs are used with genetic algorithms (GA), randomized hill climbing, simulation annealing (SA), and estimation of distribution (ED).

3) Embedded method: FS is carried out during the execution of clustering algorithms or clustering techniques that use embedded approaches. As the name suggests, these techniques utilize special "sparsity regularization algorithms," such LASSO, Ridge Regression, and Elastic Net (RREN), to eliminate the weight of particular characteristics. They are either integrated into the algorithm's regular or expanded capabilities. Among the classification algorithms utilized by embedded techniques of FS are DT, RF, ANN, NB, and SVM.

4) Hybrid method: Hybrid approaches are either altered versions of pre-existing FSAs or a mix of many FS techniques. In contrast to ensemble approaches, hybrid methods successively apply several FS algorithms throughout the whole dataset. Hybrid approaches minimize computing complexity by combining the high accuracy of wrapper techniques with the high efficiency of filters. Hybrid approaches employ filter techniques to initially decrease the size of the data, and then they apply wrapper techniques to choose the best candidate subset.

VII. Feature Selection Algorithm-Based IDS

The author in study [32] found a method for identifying pertinent features from KDD99 by employing a hybrid approach to find the best possible subset of features. This method effectively determines the type of assault that each register in the dataset alludes to. The evaluation's findings demonstrate that an optimal subset of attributes can enhance IDS performance.

In study [33], the authors introduced a technique aimed at selecting an optimal subset of features to address performance challenges. This approach incorporates PCA, GA, and Multilayer Perceptron (MLP). Evaluation is conducted using the KDD-cup dataset. Implementing this approach enables the decrease in feature count while maximizing the detection rate.

The study in [34] provided a hybrid approach that combines Enhanced Particle Swarm Optimization (EPSO) and Modified Artificial Bee Colony (MABC) to forecast ID issues. The 10-fold cross-validation approach is used to achieve the classification accuracies, and the methods are merged to discover superior optimization outcomes. The ID KDDCup'99 benchmark dataset is used to assess the effectiveness of the suggested approach.

The study in [35] introduces the classification of the KDD intrusion dataset, incorporating noise reduction, clustering, and feature selection. The application of the DBSCAN algorithm is employed to diminish noise in the KDD dataset. To select relevant features, a Genetic Algorithm (GA) is used after noise removal. A K-Means++ clustering method is used to cluster the dataset and a SMO-based classifier is used to test the resultant dataset. the proposed methods give 96.922% accuracy.

The author in [1] defines optimal feature selection method using SVM classifier. The model undergoes testing using the KDD99 benchmark dataset and produced better results.

In [36], the author introduced a hybrid model that incorporates Filter-based Attribute Selection to decrease the dataset's feature dimensionality. Detection of various attack categories is achieved through the utilization of K-Means Clustering and Sequential Minimal Optimization (SMO), applied to the KDD99 dataset.

The Studies mentioned in Table IV, has demonstrated how FS improves both classification and clustering accuracy; hence, any appropriate FS must be applied. Additionally, computational scientists have a lot of room to design new methods that need less processing time and computational complexity. Thus, the need for more advanced, quick, and precise data mining techniques remains.

TABLE IV. COMPARISON BETWEEN METHODOLOGY, AND EVALUATION FROM DIFFERENT STUDIES

Ref.	Algorithm	Methodology	Dataset	Evaluation
[32]	Hybrid approach	k-means	KDD99	All subsets surpass 99% rate.
[33]	GA	Principal Component Analysis (PCA)	KDD99	Accuracy is 99%
[34]	ABC and PSO	Tenfold cross-validation method	KDDCup'99	The highest accuracy 88.59%
[35]	combination of DBSCAN, and K-Means++	KMSVM (Simple K-mean with SVM classification)	KDD99	The methods give 96.922% accuracy.
[1]	Bat algorithm.	SVM classifier	KDD99	Achieved 94.12% accuracy
[36]	K-means and SMO algorithms	SVM classifier	KDD99	The model obtained 99.33 % accuracy

VIII. DISCUSSION AND ANALYSIS

Upon examining various IDS models and conducting a review, we identified challenges that inspire research into the utilization of ML for feature selection in IDS. In this paper, we discuss algorithms, dataset, and feature selection, as they are all factors that affect the detection accuracy of an IDS. They can help to compare the quality of different IDS. Therefore, we analyze previous literature in the last five years and found that the most widely used ML classifiers in ID are SVMs, NBs, DTs, RFs, and KNN classifiers. Based on the analysis in Fig. 2, we find that supporting devices are the most used with a rate of 33%, then RFs with a rate of 31% and DTs with a rate of 21% while both NB and KNN are the least used with a rate of 7%.

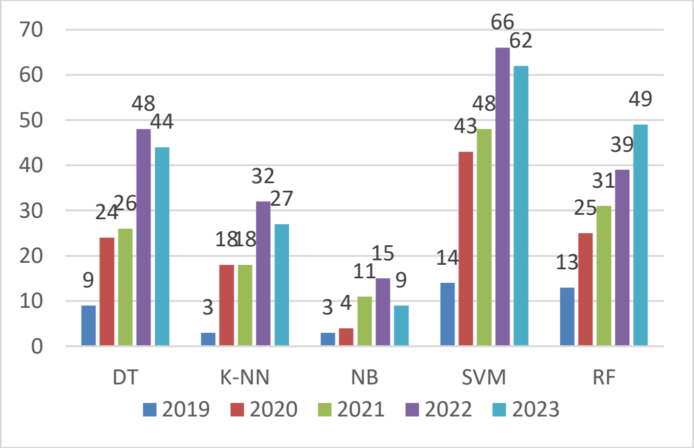


Fig. 2. Classification algorithm used for IDS.

The previously mentioned datasets were used in the research to evaluate the performance of ML-based IDS. Analyzing the public datasets available for IDS in in last five years is shown in Fig. 3. It is shown that the NSL-KDD dataset is the highest at 49% used to evaluate research over the past five years. We also find that UNSW-NB15 was used at 28%, CICIDS2017 at 15%, while the least used for evaluating research are KDD cup 99 at 6% and CIDDS-001 by 1%.

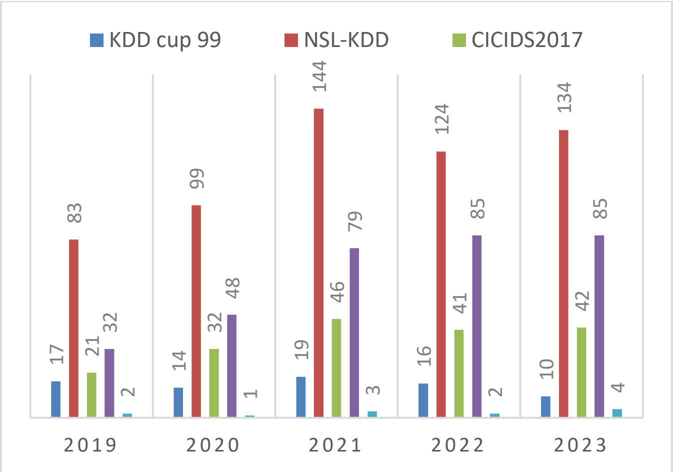


Fig. 3. Datasets used for ML-based IDS.

In IDS, selecting the appropriate features is crucial. In the learning phase, an expert machine may detect attacks in the testing phase with the assistance of a feature subset that has been properly selected. The goal of optimization-based feature selection is to identify the best subset of features from all features across various domains. Fig. 4 shows the accuracy of feature selection by different IDS given in Table IV. The algorithms with the highest detection accuracy were K-means and SMO, both achieving a rate of 99.33%, along with the Genetic Algorithm at 99%. In contrast, the least accurate in detection were Artificial Bee Colony and Particle Swarm Optimization, each with a rate of 88.59%. Overall, it has been demonstrated that feature selection using an ML classifier significantly impacts the detection accuracy of IDS.

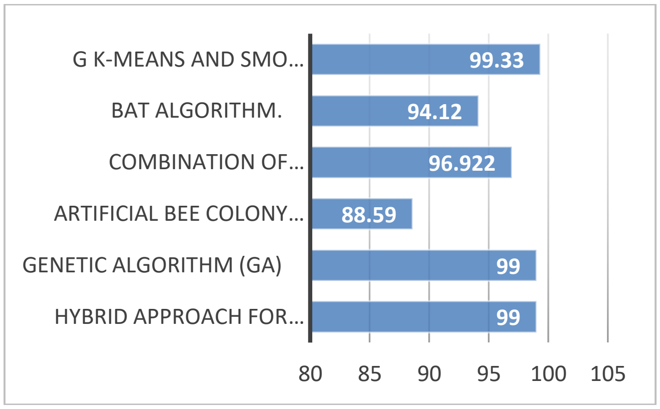
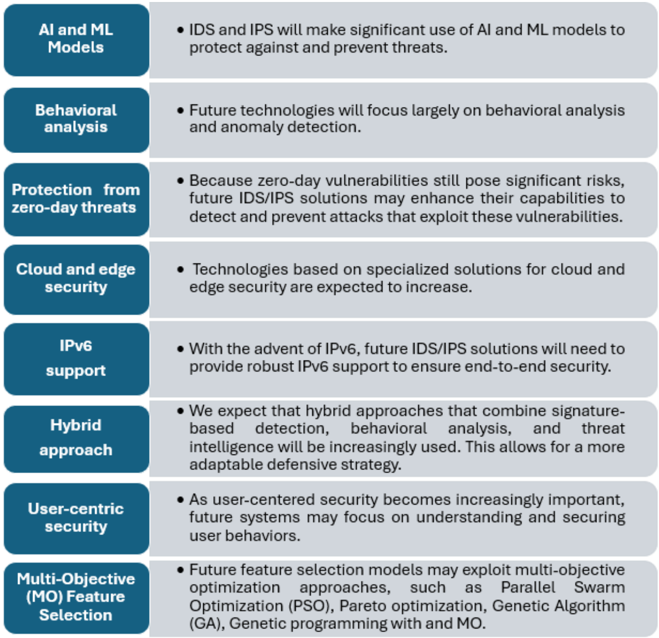


Fig. 4. Faature Selection for IDS.

Our analysis based on previous research indicates that the most widely used ML classifiers in ID are SVMs at 33% and RFs at 31%. Despite the diversity in the use of different data sets for IDS, the NSL-KDD is the most used in 49% of studies. In the realm of feature selection, the K-means and SMO algorithms emerge with an impressive 99.33%, marking the highest percentage in previous research on feature selection for ML based intrusion detection.

IX. Future Direction

Many important trends in the future of IDS research for IoT. These include, but are not limited to: AI utilization, behavioral analysis, IPv6 integration, Muli-Objective (MO) feature selection methods, etc. More details can be found in Fig. 5:



AI and ML Models	• IDS and IPS will make significant use of AI and ML models to protect against and prevent threats.
Behavioral analysis	• Future technologies will focus largely on behavioral analysis and anomaly detection.
Protection from zero-day threats	• Because zero-day vulnerabilities still pose significant risks, future IDS/IPS solutions may enhance their capabilities to detect and prevent attacks that exploit these vulnerabilities.
Cloud and edge security	• Technologies based on specialized solutions for cloud and edge security are expected to increase.
IPv6 support	• With the advent of IPv6, future IDS/IPS solutions will need to provide robust IPv6 support to ensure end-to-end security.
Hybrid approach	• We expect that hybrid approaches that combine signature-based detection, behavioral analysis, and threat intelligence will be increasingly used. This allows for a more adaptable defensive strategy.
User-centric security	• As user-centered security becomes increasingly important, future systems may focus on understanding and securing user behaviors.
Multi-Objective (MO) Feature Selection	• Future feature selection models may exploit multi-objective optimization approaches, such as Parallel Swarm Optimization (PSO), Pareto optimization, Genetic Algorithm (GA), Genetic programming with and MO.

Fig. 5. Future research trends on IDS for IoT.

Cybersecurity is dynamic, other challenges and attacks may emerge over time and we will need to develop innovative solutions to detect and prevent intrusions and maintain security.

X. Conclusion

As the IoT field expands, ensuring the security of IoT data becomes increasingly important. The increase in threats in the field of IoTs gives us the need to build an effective IDS by exploiting science and technology. To develop this field further, ML can be used to build effective IDS systems. In this review article, we outline IDS and give an overview of the various IDS and ML kinds. We also spoke about how important it is to apply ML classifiers in ID and gave a thorough explanation of the approaches employed. We reviewed research using ML classifiers for ID, its methods, and methodology. A review of each of these methods is also given, along with a comparison of the most popular ID datasets used for assessment. This comparison highlights the functions of the various feature selection algorithms employed, as well as the efficacy and accuracy of each method's detection. The examination reveals a notable focus on ML-based IDS, with SVM and RF techniques being the predominant classifiers, accounting for 33% and 31% respectively. Although various datasets are employed for IDS, NSL-KDD is the most prevalent, utilized in 49% of studies. In terms of feature selection, K-means and SMO algorithms stand out with an impressive 99.33%, representing the highest percentage reported in previous research on feature selection for ML-based IDS.

Acknowledgment

The authors are thankful to the Deanship of Graduate Studies and Scientific Research at University of Bisha for the financial support through the Graduate Students Research Support Program.

References

[1] Prashanth, S. K., Shitharth, S., Praveen Kumar, B., Subedha, V., & Sangeetha, K. (2022). Optimal feature selection based on evolutionary algorithm for intrusion detection. SN Computer Science, 3(6), 439.

[2] Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Communications Surveys & Tutorials*, *22*(3), 1646-1685.

[3] Patel, A., Qassim, Q., & Wills, C. (2010). A survey of intrusion detection and prevention systems. Information Management & Computer Security, 18(4), 277-290.

[4] Thapa, S., & Mailewa, A. (2020, April). The role of intrusion detection/prevention systems in modern computer networks: A review. In Conference: Midwest Instruction and Computing Symposium (MICS) (Vol. 53, pp. 1-14).

[5] Coulibaly, K. (2020). An overview of intrusion detection and prevention systems. *arXiv preprint arXiv:2004.08967*.

[6] Abrar, I., Ayub, Z., Masoodi, F., & Bamhdi, A. M. (2020, September). A machine learning approach for intrusion detection system on NSL-KDD dataset. In *2020 international conference on smart electronics and communication (ICOSEC)* (pp. 919-924). IEEE.

[7] Otoum, Y., Liu, D., & Nayak, A. (2022). DL-IDS: a deep learning–based intrusion detection framework for securing IoT. *Transactions on Emerging Telecommunications Technologies*, *33*(3), e3803.

[8] Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. *IEEE access*, *9*, 22351-22370.

[9] Hosseini, S., & Sardo, S. R. (2023). Network intrusion detection based on deep learning method in internet of thing. Journal of Reliable Intelligent Environments, 9(2), 147-159.

[10] Rahman, M. A., Asyhari, A. T., Wen, O. W., Ajra, H., Ahmed, Y., & Anwar, F. (2021). Effective combining of feature selection techniques for machine learning-enabled IoT intrusion detection. *Multimedia Tools and Applications*, 1-19.

[11] Mohy-eddine, M., Guezzaz, A., Benkirane, S., & Azrour, M. (2023). An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection. *Multimedia Tools and Applications*, 1-19.

[12] Smys, S., Basar, A., & Wang, H. (2020). Hybrid intrusion detection system for internet of things (IoT). *Journal of ISMAC*, *2*(04), 190-199.

[13] Ashiku, L., & Dagli, C. (2021). Network intrusion detection system using deep learning. *Procedia Computer Science*, *185*, 239-247.

[14] Ioannou, C., & Vassiliou, V. (2021). Network attack classification in IoT using support vector machines. *Journal of sensor and actuator networks*, *10*(3), 58.

[15] Albulayhi, K., Abu Al-Haija, Q., Alsuhibany, S. A., Jillepalli, A. A., Ashrafuzzaman, M., & Sheldon, F. T. (2022). IoT intrusion detection using machine learning with a novel high performing feature selection method. Applied Sciences, 12(10), 5015.

[16] Maithem, M., & Al-Sultany, G. A. (2021, February). Network intrusion detection system using deep neural networks. In *Journal of Physics: Conference Series* (Vol. 1804, No. 1, p. 012138). IOP Publishing.

[17] Pokhrel, S., Abbas, R., & Aryal, B. (2021). IoT security: botnet detection in IoT using machine learning. arXiv preprint arXiv:2104.02231.

[18] Bagaa, M., Taleb, T., Bernabe, J. B., & Skarmeta, A. (2020). A machine learning security framework for iot systems. IEEE Access, 8, 114066-114077.

[19] Eldeeb, E., Shehab, M., & Alves, H. (2021). A learning-based fast uplink grant for massive IoT via support vector machines and long short-term memory. IEEE Internet of Things Journal, 9(5), 3889-3898.

[20] Vijayalakshmi, P., & Karthika, D. (2023). Hybrid dual-channel convolution neural network (DCCNN) with spider monkey optimization (SMO) for cyber security threats detection in internet of things. Measurement: Sensors, 27, 100783.

[21] Shafiq, M., Tian, Z., Bashir, A. K., Du, X., & Guizani, M. (2020). IoT malicious traffic identification using wrapper-based feature selection mechanisms. *Computers & Security*, *94*, 101863.

[22] Saranya, T., Sridevi, S., Deisy, C., Chung, T. D., & Khan, M. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. *Procedia Computer Science*, *171*, 1251-1260.

[23] Verma, A., & Ranga, V. (2020). Machine learning based intrusion detection systems for IoT applications. Wireless Personal Communications,111,2287-2310.

[24] Chaudhari, R. R., & Patil, S. P. (2017). Intrusion detection system: classification, techniques and datasets to implement. *International Research Journal of Engineering and Technology (IRJET)*, *4*(2), 1860-1866.

[25] Amudha, P., Karthik, S., & Sivakumari, S. (2013). Classification techniques for intrusion detection-an overview. *International Journal of Computer Applications*, *76*(16).

[26] Mukherjee, S., & Sharma, N. (2012). Intrusion detection using naive Bayes classifier with feature reduction. *Procedia Technology*, *4*, 119-128.

[27] Amor, N. B., Benferhat, S., & Elouedi, Z. (2004, March). Naive bayes vs decision trees in intrusion detection systems. In *Proceedings of the 2004 ACM symposium on Applied computing* (pp. 420-424).

[28] Panda, M., & Patra, M. R. (2007). Network intrusion detection using naive bayes. *International journal of computer science and network security*, *7*(12), 258-263.

[29] Wagh, S. K., Pachghare, V. K., & Kolhe, S. R. (2013). Survey on intrusion detection system using machine learning techniques. *International Journal of Computer Applications*, *78*(16), 30-37.

[30] Neeraj, K. N., & Maurya, V. (2020). A review on machine learning (feature selection, classification and clustering) approaches of big data mining in different area of research. *Journal of Critical Reviews*, *7*(19), 2610-2626.

[31] Ghurab, M., Gaphari, G., Alshami, F., Alshamy, R., & Othman, S. (2021). A detailed analysis of benchmark datasets for network intrusion detection system. *Asian Journal of Research in Computer Science*, *7*(4), 14-33.

[32] Araújo, N., De Oliveira, R., Shinoda, A. A., & Bhargava, B. (2010, April). Identifying important characteristics in the KDD99 intrusion detection dataset by feature selection using a hybrid approach. In 2010 17th International Conference on Telecommunications (pp. 552-558). IEEE.

[33] Ahmad, I., Abdullah, A., Alghamdi, A., Alnfajan, K., & Hussain, M. (2011). Intrusion detection using feature subset selection based on MLP. *Sci. Res. Essays*, *6*(34), 6804-6810.

[34] Amudha, P., Karthik, S., & Sivakumari, S. (2015). A hybrid swarm intelligence algorithm for intrusion detection using significant features. *The Scientific World Journal*, *2015*.

[35] Shakya, V., & Makwana, R. R. S. (2017, May). Feature selection based intrusion detection system using the combination of DBSCAN, K-Mean++ and SMO algorithms. In *2017 international conference on trends in electronics and informatics (ICEI)* (pp. 928-932). IEEE.

[36] Chandra, A., Khatri, S. K., & Simon, R. (2019, February). Filter-based attribute selection approach for intrusion detection using k-means clustering and sequential minimal optimization techniq. In *2019 Amity International Conference on Artificial Intelligence (AICAI)* (pp. 740-745). IEEE.