Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 7, 2025

MACHINE LEARNING APPROACHES FOR INTRUSION DETECTION IN INTERNET OF THINGS

Samra Majeed[*1], **Gohar Mumtaz**[2]

[*1,2]*Faculty of Computer Science and Information Technology,The Superior University, Lahore, Pakistan*

DOI: https://doi.org/10.5281/zenodo.16678335

Keywords
Machine Learning; ML Approaches; Intrusion Detection Systems; IOT; Security

Article History
Received on 31 April 2025
Accepted on 16 July 2025
Published on 31 July 2025

Corresponding Author: *
Samra Majeed

Abstract
The internet of things' rapid and continuous development has created significant security issues that require immediate attention, especially in light with the rising frequency and sophistication of cyberattacks on linked networks and devices. As the trend continues to grow, it has become more and more important to have real-time and effective intrusion detection to protect the IoT systems' core principles of privacy, integrity, and availability. With the ultimate objective of greatly increasing intrusion detection effectiveness, particularly in IoT systems, this paper offers a thorough investigation of several machine learning methods. The overall thrust of this research is constructing and implementing lightweight machine learning models specifically designed to run effectively within the small computational resource's characteristic of IoT devices. Further, the study explores a wide variety of feature selection methodologies for dimension reduction and optimization of overall model efficiency. In addition, the detection accuracy, scalability, and power efficiency of numerous categorization models, such decision trees, random forests, and light-weight neural networks are also looked at to see how well they operate. This research also finds the promising potential of federated learning as a highly innovative future trend that not only seeks to maintain data privacy but also seeks to improve and improve existing security protocols. Finally, the conclusions of this research demonstrate that machine learning offers a highly promising avenue for developing intelligent and adaptive intrusion detection system in IoT networks.

INTRODUCTION

The Internet of Thing paradigm shift involves connecting billions of Internet gadgets. This networking facilitates the development of sophisticated healthcare applications, industrial automation, smart homes, and transportation, among others. IoT environments are open, distributed, and resource-constrained; therefore, their rapid growth poses major security risks. By connecting billions of things to the internet, the IoT is a paradigm change [1]. This connectivity allows sophisticated applications in healthcare, industrial automation, smart homes, and transportation. However, open, distributed, and resource-constrained IoT settings pose significant security risks. IoT technologies are convenient and efficient, but these security problems are dangerous [2]. This is true even when IoT solutions improve efficiency and convenience.

To stop these dangers, intrusion detection systems (IDS) are very important. These tools look through network data to find strange activity. Normal intrusion tracking won't work in IoT environments because they are spread out, complicated, and always changing. Anomaly-based systems often give false positives, but static signature-based models can't be changed. Machine learning can look at old data, find

complicated trends, and make systems smart enough to respond to new threats, which will completely change this field. Several types of security risks are lessened by intrusion detection systems. These tools look through network data to find strange things. Standard intrusion tracking methods often fail to find intrusions in IoT environments because they are decentralized, have many parts, and are still growing. Anomaly-based systems often give false positives, while static signature-based models can't be changed. Machine learning can sort through huge amounts of data, find complex trends, and teach systems how to act before problems happen, which will completely change this industry [3-4].

Machine learning is necessary for internet of things intrusion-detection since the number of connected devices and cyber threats are growing. As IoT spreads to more areas like smart cities, transportation, healthcare, and manufacturing, it creates a bigger attack surface that is hard for traditional security solutions to guard. These devices are often spread out and have limited resources, which leaves them open to data breaches, hacking, DoS attacks, and people who aren't supposed to be there. One important area of research is machine learning (ML) intruder detection in the IOT. This is mostly because of how quickly connected gadgets are getting better and more dangerous online. IoT is being used more and more in smart cities, transportation, healthcare, and industry [4]. This makes it easier for hackers to get in, but regular security measures can't protect it.

Role of Machine Learning in Modern IDS

Machine learning introduces the capability to recognize patterns, predict future anomalies, and adapt to evolving threats without being explicitly programmed. In the context of IOT, ML-enhanced IDS systems can[5]:

Detect Unknown Threats: Unlike signature-based IDS, ML models can identify zero-day attacks by learning normal behavior and spotting anomalies.

Improve Accuracy: ML models reduce false positives and negatives by training on large datasets.

Enable Real-Time Response: ML techniques like online learning allow IDS systems to adapt to new data on the fly.

Handle High-Dimensional Data: IOT generates massive amounts of data, which ML algorithms can process and analyses effectively.

The convergence of ML and IDS opens the door to next-generation security systems that are reactive but also predictive and self-evolving.

Challenges in securing IOT

Secure IoT networks is more complicated than regular networks. Main obstacles are:

Resource Constraints: IoT devices frequently possess constrained memory, computational capacity, and battery longevity. These constraints impede the deployment of complex device-based security systems.

Scalability Issues: Billion IoT devices worldwide will bring logistical and security challenges.

Resource Limitation: The Internet of. Things have many communication protocols and platforms, resulting in diverse security methods.

Physical Vulnerability: Due to their location in insecure regions, devices are vulnerable to physical interference.

Poor Patch Management: IoT devices are vulnerable to known dangers due to infrequent firmware updates.

Data privacy and integrity: are essential for sending accurate and confidential information. Because IoT networks are open and distributed, this objective is difficult.

To solve these problems, intelligent and adaptive security frameworks that can learn and evolve are needed, and machine learning can help.

Prevention and Adaptive Strategies

An integrated proactive and reactive strategy is needed to defend against IOT attacks. Machine learning provides a flexible security framework for new threats.

ML-based prevention methods:
Encryption & Authentication: Small cryptographic techniques secure all communication paths.

Access control policies: restricting people and devices by role and action.
Regular upgrades include fixing known vulnerabilities in the firmware.

ML-based adaptive methods:
Anomaly Detection: Recognizing unusual device activity.

Federation Learning: Training models on decentralized devices without sharing data protects privacy.

Reinforcement learning lets IDS learn the best replies from its environment.

Edge intelligence: running small ML models on edge devices for faster local decisions.

Research Problem Statement
Existing intrusion detection systems (IDS) provide basic security, but they aren't flexible enough to handle the unique and ever-changing nature of IoT contexts. Intricate attack methodologies pose a difficulty for static, rule-based systems, which also tend to produce elevated false positive rates. Moreover, several current machine learning-based intrusion detection system models are ill-suited for low-resource environments, hence constraining their effective implementation in the IoT. This work highlights the effective adaptation of machine learning for operation within resource-constrained Internet of Things systems.
• How well do machine learning algorithms work to solve the different problems that come up with Internet of Things security?
• What can we do to improve detection rates such that false alarms happen less often?
The goal of this research is to look at. machine learning techniques that may be used in Internet of Things environments for intrusion detection systems that are real-time, lightweight, and flexible.

Research Questions / Hypothesis
To guide the direction of this research and establish a focused investigation, the research questions (RQS) and hypotheses (H) have been framed:

Research Questions:
RQ1: What are IOT environments' most prevalent threats and intrusion patterns?
RQ2: How do machine learning techniques compare to traditional IDS detecting intrusions in IOT?
RQ3: Which ML algorithms balance detection accuracy and resource utilization in restricted IoT devices?
Hypotheses:
H1: ML-based IDS significantly outperforms signature-based IDS in detecting unknown threats in Iot systems.
H2: Supervised learning models detect more accurately than unsupervised models but require more computation.
H3: IoT-optimized lightweight ML algorithms can secure devices without affecting performance.

Objective of the studies
As the Internet of Things continues to grow in many areas, it is important to make sure that systems that are tied together are safe even though resources are limited. When IDS devices are used in traditional networks, their consistent signatures and frames might be a problem when they are used in IoT networks. It is very important for assault tracking systems to be flexible, able to quickly find threats, and work well. This study looks into how machine learning could be able to find intruders in Internet of Things devices. It is important to make them work better, cut down on the number of false findings, and run smoothly in areas with limited resources.
This study categories machine learning based internet of things attack detection and prevention. systems according to their methods of learning (supervised, unsupervised, and deep learning), finding objects, and implementing them. We evaluate these models' effectiveness, efficiency, accuracy, scalability, and usefulness in actual Internet of Things applications. When applied to the IoT, conventional intrusion detection system models cause problems, as this paper explains. Improve your

internet of Things intrusion monitoring system by applying machine learning.
• Find out about the interesting area of IoT security holes and machine learning applications that find intrusions.
• Give an IoT intrusion detection system a platform that can change and employs machine learning.
• See how successfully machine learning models balance the frequency of false alarms, the precision of the identification, and the amount of resources used.
This study aims to discover strong, smart, and adaptable ways to identify intrusions in Internet of Things systems.

Contribution Summary
Using ML-based intrusion detection, this study adds several new insights into the field of IoT security:
Detailed Review: It provides an extensive overview of existing IDS approaches, considering their limitations in IoT environments.

Taxonomic Structure: The research categorizes IDS based on methodologies, strategies, and environments of deployment and proposes a standardized taxonomy of them.

ML and IoT Needs: Align machine learning algorithms with IoT security challenges to identify the most effective solutions for constrained devices.

ML-based IDS Framework proposal: A hypothetical IoT architecture-oriented ML-based IDS model is evaluated using critical performance measures.

Gap Analysis: The research illuminates understudied literature and suggests future research.

Organization of the paper
This document provides a comprehensive analysis of the detection of IoT intrusions using machine learning. The principles, classifications, and security concerns of Intrusion Detection Systems are addressed in Section II. A comparative analysis of classical and deep learning intrusion detection algorithms is presented in Section III. The evaluation measures for IDS efficacy are delineated in Section IV, which employs widely recognized benchmark datasets.

Intrusion-Detection-system (IDS)
A detection system includes an audit data collection agent. This agent collects data regarding the system in issue. The detector subsequently transmits this information to the site security office (SSO) after either storing or processing it. Subsequently, additional procedures frequently commence with an investigation into the cause of the alert. IDS was first introduced by computer network security monitoring and surveillance, which employed audit trails to detect odd network activity. The two tasks of a typical IDS must be distinguished. Intrusion warning systems detect system anomalies first. Second, an SSO responds to the alert. Recognising that invasions take numerous forms is vital. Passwords let unauthorised individuals in. Masked invaders are hard to spot in the field. Real users abuse their rights and exploit the system via a network utilising online exploit script [6].
Motion-based side-channel attacks leverage smartphone accelerometer, magnetometer, and gyroscope vibration predictions to determine character types. Sybil attacks can harm wireless sensor networks with misidentified nodes. Academic and professional communities investigate computer system risks. Therefore, this list may be incomplete.

IDS Process and Terminology

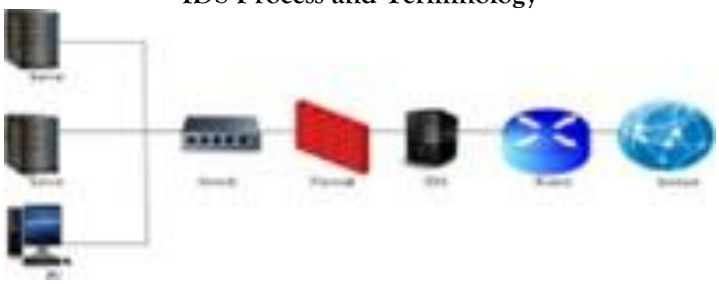

Figure 1Process of IDS:

Alert/alarm: This is a signal that goes off when someone tries to assault a system. This signal will be one of the following:

True positive: An IDS alarm goes off because of a bad action. A false positive is when an alert goes off but there is no assault.

False negative: When an assault is happening but no alert goes out.

True negative: This is a situation in which no alarm goes off and no bad things happen.

Site policy: A set of rules that regulate how an organization's intrusion detection systems are set up and what they can and can't do.

3. Site policy awareness: The ability of an IDS to change its rules and settings to find new incursions.

4. Confidence value: This is a number that is different from all the others and is used to measure how well IDS can find an attack.

5. Alarm filtering is a method that helps you figure out if an event is a false positive or a real assault.

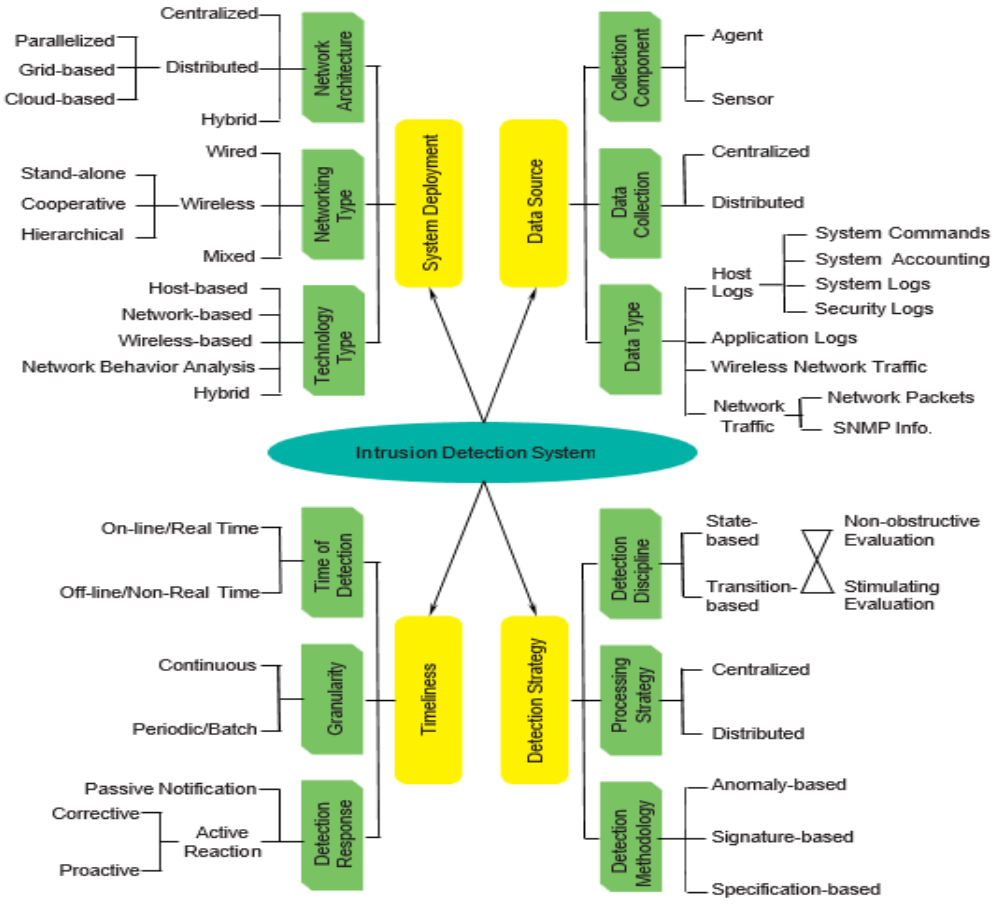

Figure 2: Taxonomy of IDS

Taxonomy of IDS

There are several ways to group IDS, such as by how they are deployed, how they find things, and what sorts they are. Fig. 3 gives a short overview of how IDS are classified [7].

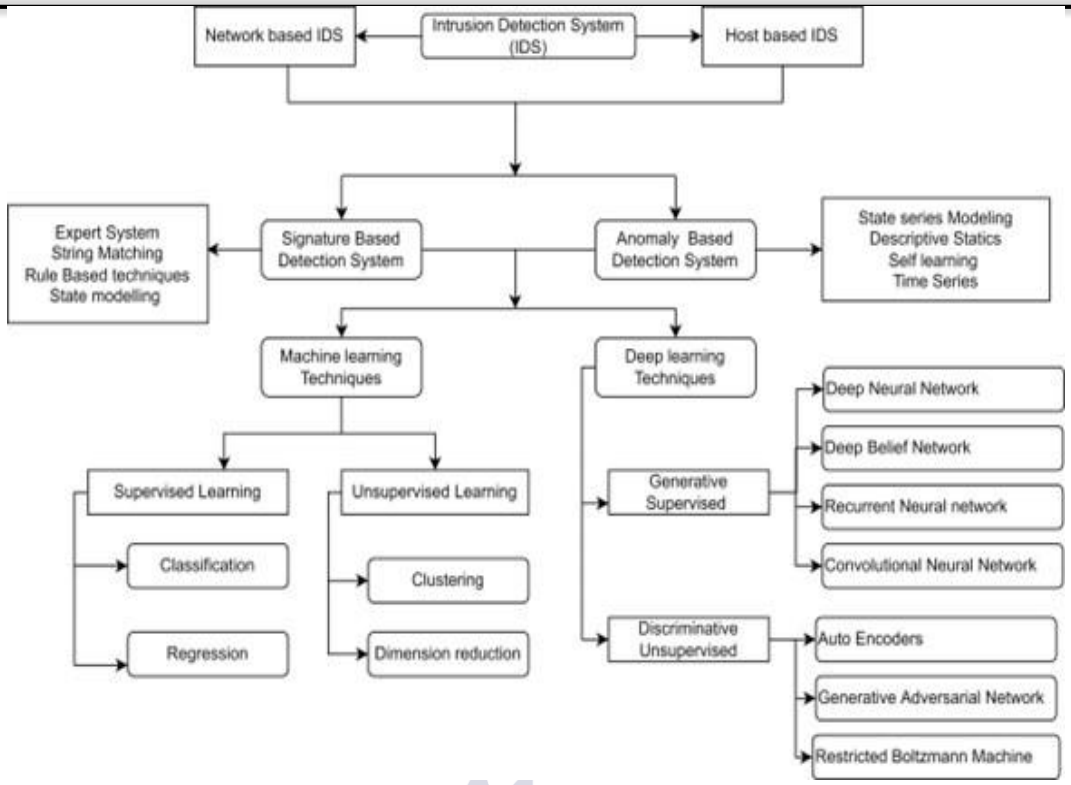

Figure 3: Classification of IDS

Types of IDS

Host-Based-IDS (HIDS)

HIDS is a complex detection system that acts like an agent to watch over a host device and report any unusual activity that happens. The main job of HIDS is to keep an eye on the system's changing behavior, state, storage space, internal configuration, targeted network packets, executed programs, and accessible resources at all times. In addition to this, the analysis of log files present on the host (including kernel, system, server, and network) is conducted, alongside monitoring file access and configuration changes in real-time. Finally, the system compares these activities with previous attacks stored on the server [7-8].

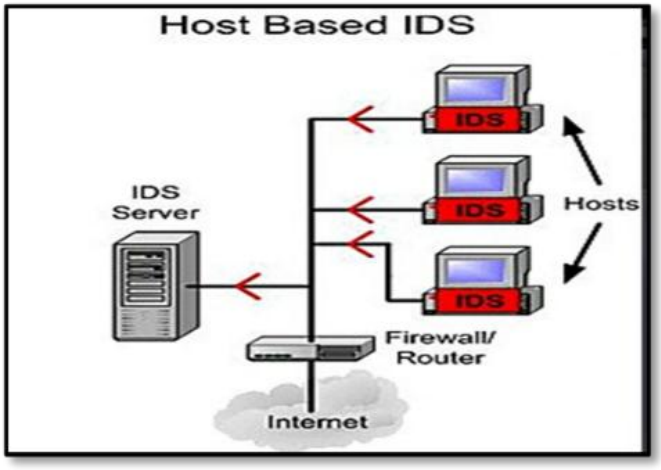

Figure 4: Host based IDS

Network-Based IDS (NIDS)

NIDS are frequently put adjacent to firewalls and employ special sensors to keep an eye on network traffic. This detects denial of service threats and scanned network ports from incoming packets. This solution integrates into network ports and works

with a firewall to protect against known threats. NIDS can be network-node-based or promiscuous-mode-based. Distributed agent-based node-based Network Intrusion Detection Systems for single-destination packet analysis are successful. Promiscuous-mode-based NIDS monitors all network packets and analyses suspicious attempts using one sensor per segment [8]. NIDS analyses and correlates incoming traffic in a subnetwork of the network. Then, it checks the agreements and alerts for violations. The sensors begin management, control, and alert reception interfaces and provide data to the central server [9]. Two network interfaces connect NIDS applications; one monitors network traffic and the other controls and creates activity reports.

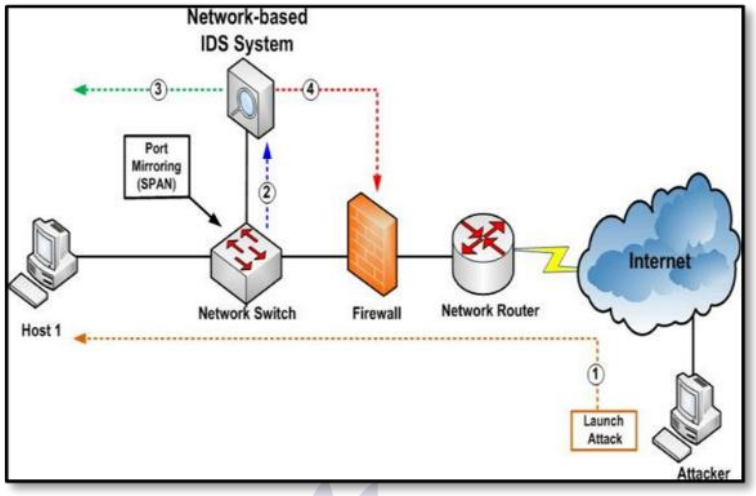

Figure 4: Network-based-IDS

Hybrid-Based-IDS or Mixed-IDS (MIDS)

Double Guard, which employs both host IDS and network IDS, is an example of how MIDS incorporates multiple varieties of IDS to optimize their capabilities and improve detection accuracy. However, the analysis of data necessitates a significant amount of time with MIDS [10].

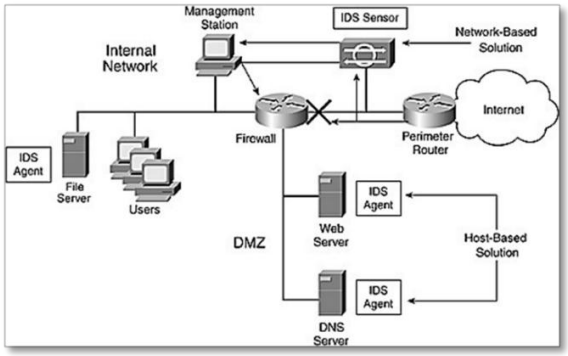

Figure 5: Hybrid based IDS

IDS	Presence	Basic of ID	Advantages	Limitations
NIDS	Set up on the computer that is a member of a network	Pattern comparison	Best for outsider detection	• Not good for insider's attacks • Ciphered data cannot be analyses • Damage tolerance capability is poor • Suspicious behavior pattern detection is not available. • Very few prosecutions capabilities
HIDS	Located on a	Configuration	Identifying the dataset outlier	• Extensive memory is required to

	particular computer, or "host," and regulating operations	of change	Ideal for identifying insiders Quickly determines the level of compromise Effective in identifying patterns of questionable conduct Capable of providing prosecution help	conduct accurate analysis. • Weakness in detecting outsiders Low responsiveness in real-time

Methods of IDS

Anomaly-based Method: Systems utilize historical data regarding system activity and the specifications of expected user behaviour to predict the types of data a user may require or need. An anomaly detector identifies deviations from the anticipated behaviour outlined in its profile. The IDS then looks for patterns of behavior that don't fit the profile it has already created. One big problem with this kind of IDS is that it sends out a lot of false alerts. On the plus side, it can be changed a lot and is fairly accurate compared to other IDS methods. People know that Anomaly Detection is good at finding security problems, even new or unexpected ones (novel attacks). To correctly describe the predicted behavior, large training sets are needed (Jyothsna et al., 2011). Another name for this technology is a profile-based intrusion detection system [11].

Signature or Misuse-based: Using the previous data, a collection of known facts is searched through the dataset that was given. The repository has hazardous or malicious signatures that might put the system at risk and lead to several types of attacks. This collection of signatures is very important for IDS to work well. The greater the number of signatures, the higher the probability of detecting intrusions. One can juxtapose the catalog of undesired patterns with the compilation of network traffic and alerts (García-Teodoro et al., 2008). This is sometimes referred to as misuse incursion detection [11-12].

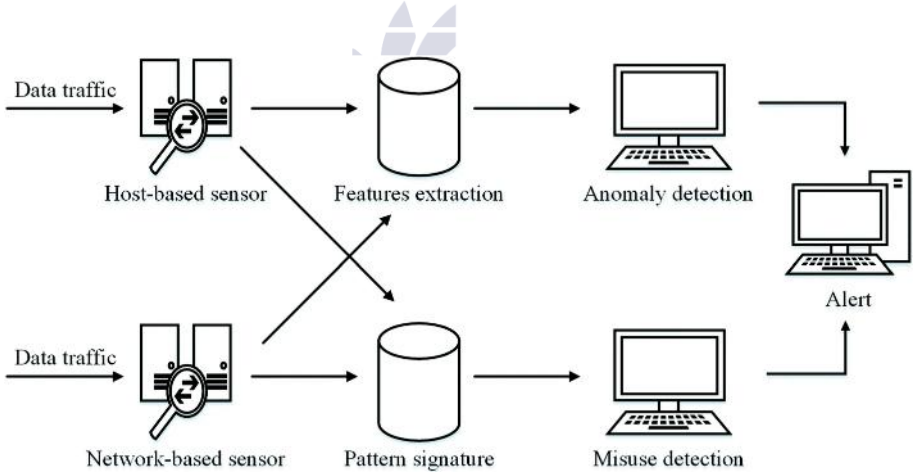

Figure 6: Anomaly and Signature-based-Methods

Pros and Cons

Anomaly /behavior-based	Signature/ knowledge-based
Pros	
• Capable of identifying novel and unanticipated vulnerabilities. • Reduced reliance on the operating system. • Facilitate the identification of privilege misuse.	• The most straightforward and efficient approach to identify recognized assaults. • Comprehensive contextual analysis.
Cons	
• Inaccurate profiles resulting from the continual alteration of observed events. • Not accessible for the reconstruction of behavioural profiles.	• Ineffective in identifying unknown assaults, evasion tactics, and versions of recognized attacks. • Limited comprehension of states and protocols. • Maintaining current signatures and patterns is

• Challenging to activate notifications promptly.	challenging. • Prolonged effort required to sustain knowledge

Techniques in IDS

The implementation of Intrusion Detection Systems is crucial for network security, particularly in the context of the Internet of Things. The intrusion detection system employs many methods for monitoring, analysing, and detecting potentially hazardous activities. There is a significant amount of variation in the success of various strategies, which may be attributed to the structure of the network, the accessibility of resources, and the level of difficulty of the attack. The following is a discussion of the techniques that are considered to be more important [10-11-12]:

Statistical Methods: These approaches are employed to analyse the data acquired from the network or sensors. These methods entail the establishment of a statistical baseline that is capable of representing anomalous behavior. Disturbances that are not consistent with this baseline are referred to as anomalies. These disturbances are often analyzed through the use of probability distributions or statistical tests. When applied to Internet of Things sensor stream time-series data, statistical algorithms may effectively identify outliers. However, high unpredictability or noise may make them more sensitive to false positives.

Rules-Based System: Also known as signature-based detection, identify recognized threats by utilizing defined rules or signatures formulated by experts. This allows them to identify risks that have been identified. It is difficult for these systems to cope with zero-day vulnerabilities or unknown exploits, and they require frequent rule adjustments in order to continue being successful. Despite the fact that these systems have a high detection accuracy for assaults that have already been acknowledged, they have difficulties dealing with these types of vulnerabilities.

Machine learning Techniques: Machine learning (ML) methodologies enable the identification of fresh invasions. These strategies allow intrusion detection systems to assimilate historical data and extrapolate trends to recognize novel intrusions. These approaches can identify new invasions. There are three distinct types of learning mechanisms: supervised learning, wherein the system is trained with labelled data; unsupervised learning, which identifies anomalies without labelled data; and reinforcement learning, where the system adapts dynamically through interaction with the environment. Supervised learning is the predominant form of learning process. Machine learning-based intrusion detection systems (IDSs) provide superior flexibility and accuracy compared to traditional IDSs, as the threat environment of Internet of Things networks is always evolving [13].

Various Methods Concerning the Mining of Data: The three methods utilized in data mining approaches to uncover significant patterns and correlations from vast datasets are clustering, association rule mining, and classification. These procedures are utilized in order to extract meaningful patterns and correlations. One use of these techniques that is particularly useful is the identification of long-term assault trends and complicated correlations within the traffic of the Internet of Things. On the other side, the enormous processing demands that they have could potentially make real-time deployment challenges more challenging [14].

Deep Learning: Using real-world data collected from the Internet of Things (IoT), deep learning models like convolutional and recurrent neural networks may autonomously build hierarchical representations. They have a knack for detecting both simple and complex issues, which allows them to reduce the amount of false positives. Algorithms for deep learning commonly need a large quantity of processing resources, regardless of how precise they are. Because of this, deploying these algorithms on limited-capability Internet of Things devices presents difficulties [15].

IDs in the Iot context

Security Challenges in IOT Department

IoT services and apps need to be safe and private in order to be used in business. There are many different types of modern Internet security threats, from simple hacking to big, planned attacks that have affected healthcare and business. IoT devices and the environments they work in have limitations that make it tougher to keep applications and devices safe. Researchers have looked at IoT security and privacy issues from a number of approaches, such as communication security, data security, privacy, architectural security, identity management, and virus analysis.

Gaps in Existing Security Solutions

For the Internet of Things to achieve momentum, it is essential to recognize and mitigate the issues related to security and privacy. The notion of the "Internet of Things" has lately garnered minimal interest from the IT sector [16]. Determining if issues regarding IoT security are innovative or consistent with those of prior platforms is essential. In their study, Fernandes and colleagues focused on the need of ensuring the safety of mobile devices and the Internet of Things. Each and every concern regarding confidentiality has been addressed. The vast majority of people examine the parallels and differences between computer hardware, computer software, computer networks, and computer computers. The Internet of Things (IoT) raises a number of security problems that are comparable to those that were associated with prior information technologies, which is why the general public should be concerned. These are few examples that illustrate the difficulties that we are now facing.

In spite of the fact that there are only a limited number of resources available, the Internet of Things group has set a primary target of improving network security. Innovative technological advancements that are safer and more secure may be developed more easily through collaborative efforts. In order to alleviate worries regarding privacy and security of the Internet of Things, it is vital to have efficient algorithms and cross-layer planning tools. For instance, alongside conventional data security measures, IoT devices may require a new form of strong encryption owing to their restricted processing capacity. The expansion of IoT devices, however, introduces new issues for security systems. The intricacy of several security challenges eliminates the likelihood of straightforward solutions.

ML solutions to IoT security Challenges

Machine learning is the process of intelligently learning to maximize performance criteria by utilizing past experience or example data. The precise way ML algorithms work is by applying mathematical techniques to large data sets in order to generate behavior models. With the help of ML, smart devices may also learn on their own, without the need for direct programming. These models form the basis for future forecasts that are formed from the data that was just entered. To name just a few [16], ML draws from a wide range of scientific and technical disciplines, including AI, optimization theory, information theory, cognitive science, and many more.

Robotics, voice recognition, etc. are all examples of areas where machine learning comes in handy when human knowledge is either unavailable or useless, such as while navigating a dangerous environment. Also, it's employed when the solution to a certain problem change over time, like finding malicious code in an application or rerouting a computer network. In addition, ML finds use in practical, cutting-edge systems; for instance, Google uses ML to investigate security flaws in Android-powered mobile devices and applications. Additionally, it aids in the detection and removal of malware on infected mobile devices. In a similar vein, Amazon's Macie service utilizes machine learning to organize data stored in the cloud [17].

Even while ML methods are really good at many things, they might nonetheless give you both true negatives and false positives. So, if ML methods make wrong predictions, they need help and model changes. The model can figure out how accurate its predictions are on its own in the new type of machine learning called Deep Learning. When it comes to classification and prediction tasks, DL models' self-service nature is more useful for creative Internet of Things apps that give tailored and contextual advice. There are a lot of Internet of Things components that depend on traditional methods, such as security. Some of them are

application, service, architecture, protocol, data aggregation, resource allocation, clustering, and analytics. However, the broad usage of IoT requires techniques that are smart, strong, and reliable. For a number of reasons, ML and DL are interesting technologies for IoT networks. For example, IoT networks give ML and DL algorithms the huge volumes of data they need to make systems smarter [16-17].

There are several security issues that come up while designing and running IoT networks. The problems with IoT devices' memory, CPU, and battery life are quite serious. This constraint makes elaborate intrusion detection systems and other security measures that need a lot of resources less effective. The extensive attack surface and significant security challenges in IoT environments stem from the diverse hardware configurations, operating systems, devices, and communication protocols involved. The widespread adoption of IoT systems renders centralized security platform-based solutions ineffective. This necessitates distributed, low-impact detection systems.

For defensive systems to effectively prevent attacks from propagating over networks, they must be able to detect and neutralize them instantly. Data security is of the utmost importance whenever a device transmits sensitive information, such as financial or medical records. Solutions must be innovative, practical, adaptable, and centered around the Internet of Things. Because they enable IoT systems to make intelligent decisions, ML and DL approaches also raise the value of the data produced by the IoT.

Security, privacy, threat detection, and malware analysis are all improved by using machine learning and deep learning. Deep learning techniques may be used to sophisticated sensing and identification tasks in IoT devices, enabling the creation of novel applications and services that take into consideration interactions between people, intelligent devices, and their physical environment in real time. The following are a few real-world security-related uses of machine learning:

- Forensic face recognition: posture, lighting, occlusion (beard, spectacles), hairstyle, makeup, etc.
- Character recognition: various handwriting styles for security encryption.
- Malicious code identification: locating malicious code within software and apps.
- Distributed Denial of Service (DDoS) detection: using behavior analysis to identify DDoS attacks on infrastructure.

On the other hand, applying ML approaches to IoT applications presents additional difficulties. These difficulties are complex. For example, creating an appropriate model to handle data from many IoT applications is difficult. In a similar vein, properly classifying input data is a difficult undertaking.

A further problem emerges when employing little labeled data in the learning phase. Implementing these models on resource-limited Internet of Things (IoT) devices introduces more hurdles, along with the necessity to minimize processing and storage overhead [23]. Likewise, critical infrastructure and real-time applications are susceptible to abnormalities in machine learning algorithms. A comprehensive evaluation of IoT security solutions utilizing machine learning is essential in the specified context.

IOt and machine learning

Modern intrusion detection systems (IDS) automatically recognize and comprehend fresh IoT infiltration patterns using machine learning (ML). The adaptability of machine learning enables IoT-specific intrusion detection system solutions.

ML Techniques used for IOT

Supervised learning: This method trains models with labeled datasets to discriminate normal and dangerous actions. SVMs, Decision Trees, and k-NNs are popular supervised learning algorithms. Supervised learning works, but IoT contexts may limit labeled data [18].

Unsupervised Learning: These methods find abnormal patterns in unlabeled data, detecting undiscovered assaults. K-Means clustering and PCA group similar trends and highlight outliers to find abnormalities. The methods benefit IoT contexts with sparse labeled data.

Semi-supervised learning: To address the labeled sample scarcity and increase detection accuracy, semi-supervised learning combines a limited amount of

labeled data with a large volume of unlabeled data. The difficulties in collecting complete labelled data make IoT semi-supervised learning a realistic option [19].

Reinforcement Learning: Reinforcement learning optimizes intrusion detection system settings based on environmental feedback through trial-and-error interactions. This adaptive technique has great potential for IoT security management [20].

ML Algorithms used in intrusion detection

All detection methods use machine learning techniques to train the intrusion detection system (IDS), except for specification-based detection. This part provides an overview of the several ways that intrusion detection systems use machine learning in an Internet of Things (IoT) setting. Table 4 gives a summary of several machine learning methods, including their pros and cons and links to relevant research. In the end, Table 5 below focuses on research that suggests using different machine learning methods in Intrusion Detection Systems. Figure 7 shows the most common machine learning methods used to build intrusion detection systems in Internet of Things networks.

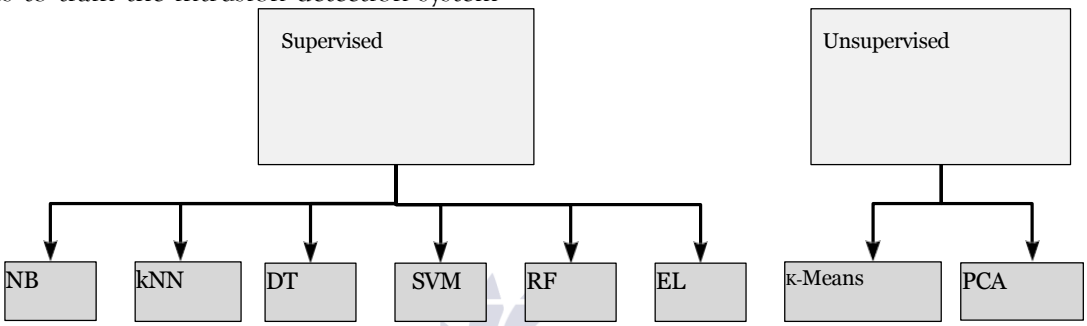

Figure 7: ML IOT-based Techniques

Naïve-Bayes (NB) Classifier

This host-network intrusion detection system by Double Guard shows how MIDS may improve precision and performance [20]. MIDS data analysis takes time. Many automated classification systems use the Naive Bayes classifier. NB employs its posterior probability to evaluate if unlabeled traffic is indicative. This method assesses the protocol, latency, and status indicators to ascertain the normality of the connection. Numerous Intrusion Detection Systems (IDSs) employ a Naive Bayes (NB) classifier for the detection of aberrant traffic, due to its simple design and minimal processing requirements [21, 22]. The training configuration, characterized by its simplicity, possesses the capability to classify data into many categories [23]. Nevertheless, it is imprecise as it fails to account for the interdependence of characteristics during categorization [24].

K-Nearest Neighbor (KNN)

KNN can operate independently based on the input that is supplied. The Euclidean distance is the standard by which the distance between friends is measured. New data is classified into established classes by the KNN classification algorithm, as demonstrated in figure 8, by assessing its proximity to existing classes. Given that the red triangles exhibit subpar performance and the green squares exhibit standard behavior, the maximum closest neighbor's method can be employed to classify each newly documented unknown case (blue hexagon). This unique occurrence represents a category that has been acknowledged. For classification purposes, the k-nearest neighbors' technique is implemented.

The categorization expands as k increases. The red hexagon is classified as normal when k is equal to 2 or 3, and it is designated as aberrant when k is equal to 1. The main thing that decides this method's accuracy is the optimal k value [26]. Several studies [27–28] that used KNN-based classification have looked at how to quickly find U2R and R2L assaults. The focus of this study has been on finding anomalies and intrusions, as well as finding intrusions on IoT-based networks. KNN is easy to use, however it might be wrong and becomes harder

to use when the value of k goes up and missing nodes are found [29].

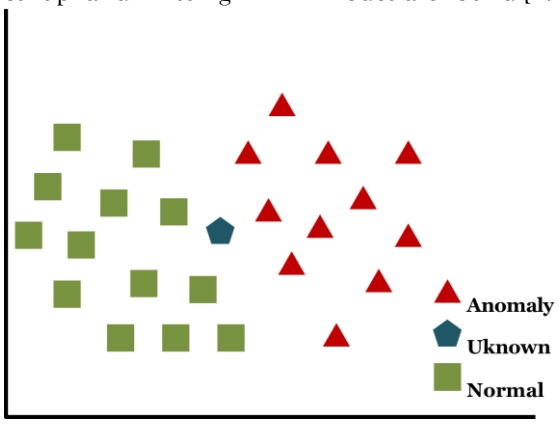

Figure 8: KNN principle

Decision Trees (DTs)

It may help you arrange and group dataset characteristics into hierarchical structures. The branches display the values of each feature, while the nodes reveal the traits themselves. Tree origin nodes, also known as feature nodes, split the tree into two halves [30]. The origin node that splits training datasets is dependent on a number of things, such as the Gini index [31] and Information Gain [32].

Fig. 9 presents nodes of a decision tree. DT methods derive and classify models by means of induction and inference [33]. At induction, nodes and branches are added to create a DT. Although these nodes are first empty, additional criteria and information acquired help to choose a feature that divides the samples from the training ground. This becomes the DT origin vertex.

Features root nodes are chosen to lower training dataset class overlap. Classifier accuracy in identifying class instances so gets better. At last, class helps each sub-DT to identify and classify their leaves. Following DT construction, the inference process iteratively compares unknown instances of classes with features to classify them. New sample classification is complete after finding a matching leaf node [33]. DTs may classify intrusion detection [34,35].

However, computational complexity and larger storage requirements must be considered [33]. In [36], DT was used to detect DDoS assaults in IoT by analyzing network data for malicious origins.

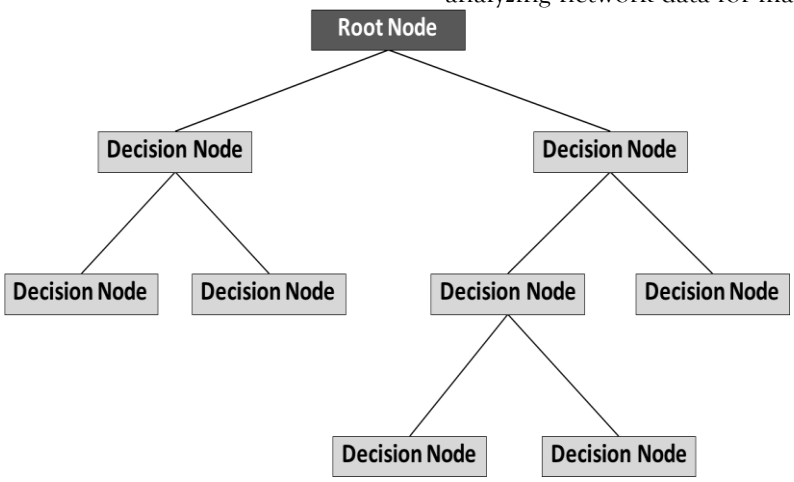

Figure 9: Decision Tree structure

Support-Vector-Machines (SVMs)

This classifier is that constructs a hyperplane inside the feature space of many classes. The splitting hyperplane is the point that is maximally far from the nearest data point of the comparison class, as illustrated in reference [37] (Fig. 10). SVMs are optimal when several features require classification but data samples are limited [13,38,39]. Statistical learning indicates that Support Vector Machines (SVMs) are effective at identifying outliers and categorizing data into normal and abnormal classifications. Support Vector Machines (SVMs) can proliferate because to their user-friendliness, capability to detect intrusions in real-time, and ability to acquire new knowledge online [39–40]. [41] proposes "Sec-IoV," an advanced multi-stage anomaly detection model utilizing support vector machines (SVM) for the identification of abnormal data within the Internet of Vehicles (IoV) network.

Another distinguishing feature of SVM is its minimal memory and storage requirements. Support Vector Machines outperformed Decision Trees, Naive Bayes, and Random Forest in the analysis of Internet of Things systems [41–43]. Simultaneously, achieving the appropriate classification speed while utilizing the correct kernel function in SVM to distinguish non-linearly separable data remains challenging.

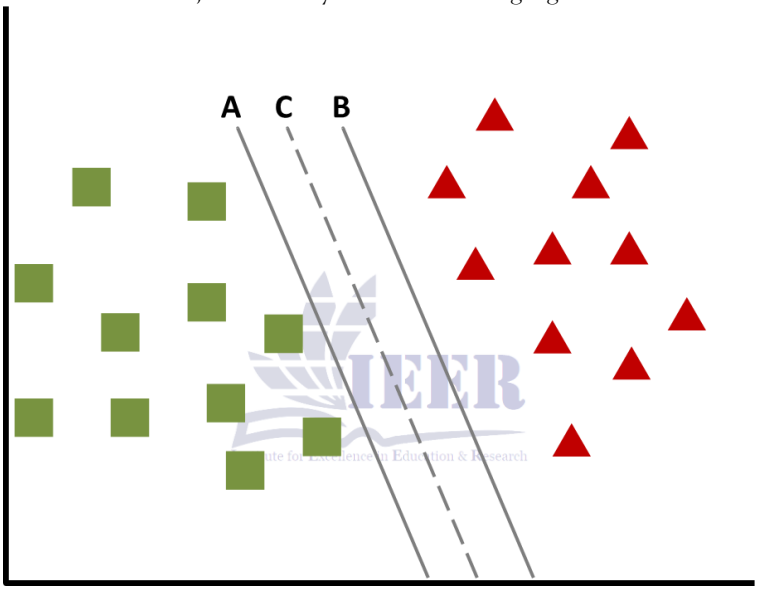

Figure 10: SVM Hyperplane Splitting

Ensemble-Learning (EL)

As shown in Figure 11, EL takes use of several classifiers' strengths, combines their outputs, and then generates a majority classification vote. Classification accuracy is enhanced by combining the outputs of homogeneous and heterogeneous classifiers [44,45]. The effectiveness of ML classification algorithms is application and data dependent, according to study [46], which is the basis of EL. Consequently, no machine learning method is generally applicable. In larger contexts, techniques like ensemble learning can enhance accuracy by decreasing variation and overfitting [47]. The accuracy of EL necessitates the simultaneous use of many classifiers, hence augmenting temporal complexity [48,49]. A lot of studies have looked at how well EL detects intrusions [50-51]. After investigating EL in resource-limited situations like the Internet of Things (IoT), a lightweight framework for online anomaly detection in IoT networks was presented. Based on the results of this experiment, the EL algorithm outperformed all classifiers in terms of accuracy and quality [52].

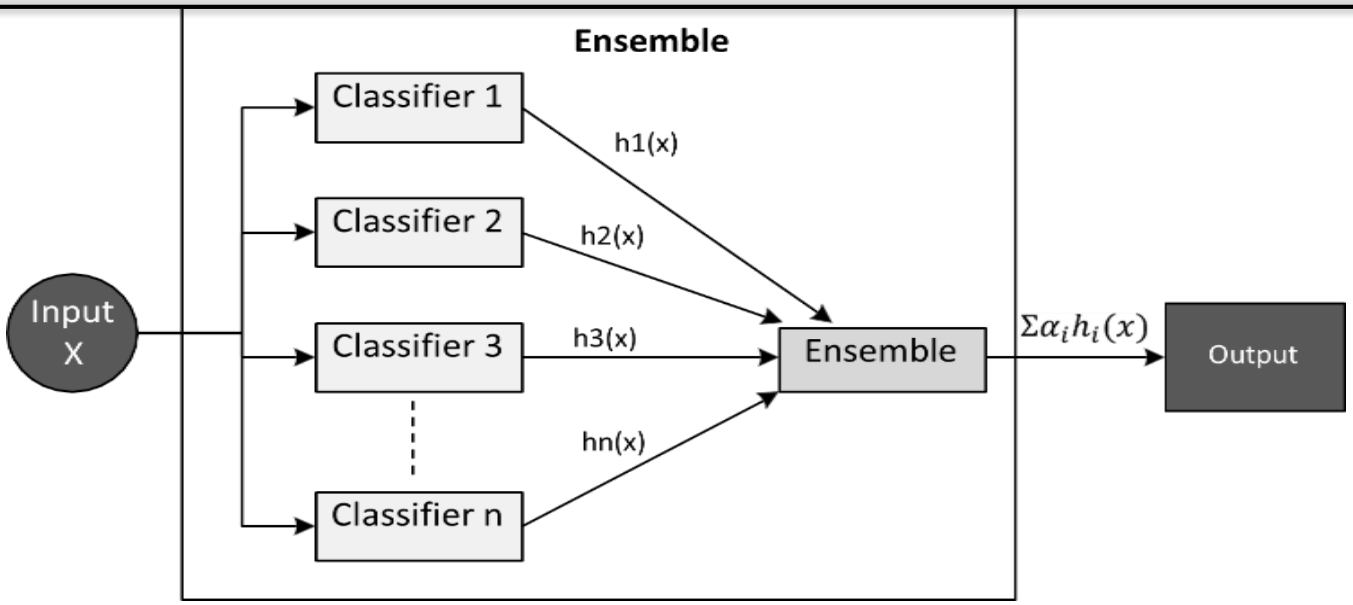

Figure 11: Ensemble Classifier

Random Forest (RF)

Recurrent Fuzzy Machine Learning employs supervised learning. A Random Forest with several Decision Trees [53,54] enhances classification accuracy while reducing mistakes. Use random decision trees to categorize majority votes [53]. The major distinction between the two classification systems is that decision trees (DTs) construct a rule set during training to categorize incoming data, whereas random forests (RF) generate a rule subset by combining all constituent decision trees. Thus, the output is more precise, requires fewer inputs, and resists overfitting [35]. Numerous studies have shown that RF can identify IoT abnormalities and intrusions [55,56]. Separate research found that RF detects IoT DDoS assaults better than KNN, ANN, and SVM [57]. This is because it requires few input characteristics and avoids costly feature selection calculations in real-time IDS.

k-Means-Clustering.

In unsupervised data analysis, k clusters are found. The properties of each sample data instance determine the grouping. The centroids are estimated using squared Euclidean distance to group the data into k clusters depending on their features. Figure 12 shows how to identify cluster centroids using the mean of data points inside each cluster. You do this until cluster modifications are hard [59,60]. K-means clustering presupposes a constant number of groups and homogeneous dataset sample distribution. Using feature similarity, K-means clustering may find outliers [61,62]. Decision trees and k-means clustering were advised for IoT anomaly detection [63].

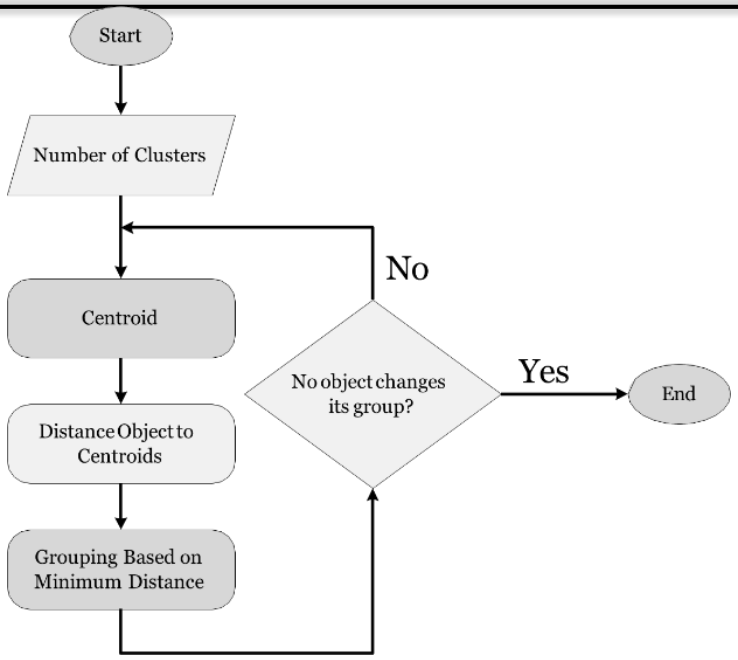

Figure 12: Illustration of k-means clustering

Principle-Component Analysis (PCA).

PCA is effective for feature selection or dimensionality reduction in extensive datasets; nevertheless, it is not intended for outlier detection. The aforementioned feature sets may be utilized in conjunction with other machine learning classifiers to detect anomalies in IoT networks. PCA may reduce a multitude of variables while preserving the majority of pertinent information. Anomalies in IoT networks have been detected by PCA and classifiers in several studies [64–65].

ML-Methods	Attack Handled	pros	cons
KB	HTTP attacks (Buffer overflow, Shell attacks), DoS, probe, r2l	It necessitates a minimal number of samples for training. It is capable of classifying in both binary and multi-label formats. It demonstrates resilience to extraneous elements.	It neglects to consider the interdependencies among features for categorization, hence impacting its accuracy.
KNN	Dos, DDos	Easy to utilize.	Identifying ideal values of K and detecting absent nodes presents significant challenges.
DT	ddos, U2R, R2L	Effortless and straightforward way for use.	It need more storage capacity and computationally complex. It is straightforward to utilize, provided that a limited number of decision trees are employed.
SVM	Scan, DDoS (tcp, udp, flood), Smurf, port sweep	Support Vector Machines exhibit significant scalability owing to their simplicity and are proficient in executing tasks like as anomaly-based intrusion detection in real-time, including online learning capabilities. Support Vector Machines (SVMs) are deemed appropriate for datasets with a high dimensionality, as they use fewer storage and memory resources.	The application of an ideal kernel function in Support Vector Machines (SVM), utilized for separating non-linearly separable data, continues to provide a problem in attaining the appropriate classification speed. Understanding and understanding SVM-based models is challenging.
EL	DoS, Probe, R2L, U2R attacks	Exhibits superior performance compared to an individual classifier. It diminishes variance. It is resilient to overfitting.	Augmented temporal complexity resulting from the concurrent utilization of many classifiers
RF	DoS, Probe, R2L, U2R	It generates a more resilient and precise output that is impervious to overfitting. It necessitates far fewer inputs and eliminates the need for feature selection.	Due to the construction of many decision trees, use of random forests maybe problematic in real time scenarios necessitating extensive datasets.
K-Means	DoS, Probe, R2L, U2R	Ooperates without the necessity of labeled data.	It is less successful than supervised learning techniques, particularly in recognizing known assaults.
PCA	Utilized in conjunction with other machine learning techniques	PCA is appropriate for datasets with a substantial number of variables, as it converts them into a diminished collection of features while preserving significant information. Can diminish the intricacy of data.	It is not an anomaly detection tool; it must be utilized in conjunction with other machine learning techniques to construct a security model.

Table 5. Comparison ML and DL techniques in IoT Security

Study	NBC	KNN	DT	SVM	EL	RF	K-Means	RNN	CNN	AE	RBM	DBN	EDLN	GAN	Dataset	Threat Detected
[110,111]	C	-	-	C	-	-	C	C	-	-	-	-	-	-	KDD99	Anomaly Detection
[116]	-	C	-	-	-	-	-	-	-	-	-	-	-	-	KDD99	apache2, udpstorm, processtable, mailbomb
[125]	-	-	-	-	-	-	-	C	C	-	-	-	-	-	ADFA-LD and ADFA-WD	Adduser, Meterpreter, Webshell
[129]	-	-	-	C	-	-	-	-	-	-	-	-	-	-	DARPA dataset	Probe attack, U2R attack
[143]	-	-	-	-	C	-	-	-	-	-	-	-	-	-	KDD99	Network Traffic anomaly detection
[150]	-	-	-	-	-	C	-	-	-	-	-	-	-	-	Boot-strapped	Worms, Buffer overflows
[157]	-	-	-	-	-	-	C	-	-	-	-	-	-	-	KDD99	-
[165]	-	-	-	-	-	-	-	C	-	-	-	-	-	-	ISCX2012	PROBE attacks or non-PROBE attacks
[166]	-	-	-	-	-	-	-	C	C	-	-	-	-	-	Android Malware Genome project	Malware
[167]	-	-	-	-	-	-	-	-	-	C	-	-	-	-	Outlier Detection Datasets	Anomaly detection
[168]	-	-	-	-	-	-	-	-	-	-	C	-	-	-	KDD	-
[169]	-	-	-	-	-	-	-	-	-	-	-	-	-	C	NSL-KDD	-
[170]	-	-	-	-	-	-	-	-	-	-	-	C	-	-	500 samples for dataset	Anomaly detection

Synopsis of Ml-based ids / ips

Machine Learning is now integral to Intrusion detection and prevention Systems, detecting threats with minimal human intervention. These techniques effectively identify security violations in network environments by rapidly scanning large and intricate datasets. Signature-based intrusion detection often employs machine learning methods, detecting attacks by matching patterns against previously stored data. They are skilled in behavior-based detection techniques, which analyze system activity to identify variations that may indicate potential threats, such as zero day attacks [63-64].

Adaptability enhances the overall reliability of machine learning-powered intrusion detection and prevention systems. One significant advantage of machine learning methods lies in their ability to effectively operate with very little computational resource, while maintaining high accuracy and timely detection capability. They are perfect for integration into various security systems, particularly in IoT environments, because to their versatility, learnability, and clarity.

An inclusive understanding of these attacks is necessary to design efficient IDS/IPS models. This research presents an exhaustively descriptive overview of machine learning-based intrusion detection processes, provides comprehensive information regarding relevant mitigation strategies, and presents a foundation for further research work. In addition, we review the existing literature in the area and present a set of research questions to guide future research into ML-based cybersecurity solutions [65].

materials and methods

Modern, accurate datasets are needed to assess an Intrusion Detection System (IDS). These datasets must accurately reflect common and unusual network occurrences. Since dataset quality influences threat system applicability and generalizability, choosing the right dataset is crucial. This study examined key datasets and their pros and cons, focusing on IoT security [66].

Dataset Available for IOT Security

Evaluated Datasets Overview Intrusion detection datasets have evolved as researchers have learned more about IoT problems. Early datasets were important, but they couldn't capture modern IoT security issues [64-65-66].

The KDD99 dataset, developed by Lincoln Laboratory at MIT to augment the DARPA98 dataset, dominated intrusion detection system (IDS) research for nearly two decades. Since there were no other options, it was utilized to evaluate classifier accuracy. Many restrictions characterize KDD99, including duplicated features, cyclical patterns, non-stationary training and testing datasets, and unbalanced objectives. IDS results are harmed by these limits [66].

NSL-KDD: To address KDD99's shortcomings, NSL-KDD resamples more evenly, highlighting cases that classifiers trained on the original KDD99 may miss. Despite these advances, its authors note drawbacks such its lack of low-footprint assaults.

The Defcon dataset: Erroneous packets, port scanning, buffer overflows, and FTP over Telnet are included in the DEFCON Dataset. It consists of DEFCON-8 (2000) and DEFCON-10 (2002). IDS assessments are irrelevant since most of its traffic is attack traffic, not background network activity. Competitive Capture the Flag (CTF) competitions provide this traffic. It mostly evaluates alert correlation methods [66-67].

The LBNL collection: This collection contains just header data from anonymized traffic from two edge routers at Lawrence Berkeley National Laboratory. Lack of tagging and other key features is a major drawback.

BoT-IoT: UNSW Canberra Cyber's Cyber Range Lab produced botnet and network traffic simulation. To address dataset issues, researchers offered network information, precise tagging, and the latest and most complex assaults. It labels original pcap, argus, and CSV files by attack type (OS, Service Scan, DoS, DDoS, Data exfiltration, Keylogging) and subcategory.

IoT PoT Dataset: Honeypots collected the data, eliminating the need for manual identification or

anonymization. Since it only logged honeypot attacks, the network's exposure was limited. Telnet assaults typically target MIPS, ARM, and PPC-powered IoT devices. It found 17,000 IP addresses that attempted to download malware over 39 days, totalling 76,000 attempts. Traditional Telnet honeypots cannot handle diverse inputs; therefore, they cannot recognize these binaries [68].

N-BaIoT Dataset: The IP camera video surveillance network (eight assaults on video uplink availability and integrity) and the IoT network with three PCs and nine IoT devices (one infected with Mirai) are used in N-BaIoT for online network IDS research. OS Scan, Fuzzing, Video injection, ARP MiTM, Active Wiretap, SSDP flood, SYN DoS, SSL Renegotiation, and Mirai feature vectors were generated.

From generic network datasets like KDD99 to IoT datasets like BoT-IoT and N-BaIoT shows a key field adaption. Although essential, KDD99 was criticized for its unbalanced aims and duplicated features, whereas NSL-KDD was an improvement but still did not adequately reflect minimal footprint assaults. BoT-IoT and N-BaIoT, explicitly designed with realistic network environments, botnet and typical traffic, proper labelling, and a variety of complex attacks, demonstrate that generic network traffic datasets cannot handle IoT's unique attack vectors and resource constraints. This development reflects the field's reaction to IoT threats' growing complexity and specificity.

Establishing reliable "ground truth" for benign vs malicious activities in IDS research, especially for IoT, is difficult. The comprehensive criticisms of datasets like the LBNL dataset missing tagging, the IoT PoT dataset having restricted network traffic visibility despite no human labelling, and DEFCON's traffic being different from real-world network traffic all point to this underlying problem. Labelling and characterizing varied, encrypted real-world IoT traffic make creating representative datasets difficult. This means that even with fresher datasets, researchers must consider how their data may not capture all IoT behaviours and dangers, which may affect their results' generalizability.

Comparing these datasets shows design trade-offs. Honeypot datasets like IoT PoT provide attack data but not regular traffic. Artificial datasets like BoT-IoT can regulate realism and labelling but may not capture all real-world complexity. Older datasets like KDD99 are large yet irrelevant to new threats. Choosing a dataset that balances realism, comprehensiveness, and labelling accuracy is difficult, thus researchers must justify their choice based on the study goals[69].

Dataset Name	Year(s) of Generation	Key Characteristics	Primary Focus/Attack Types	Advantages for IoT IDS	Limitations for IoT IDS
KDD99	1999	Refinement of DARPA98 dataset	Valid incoming connections and threats	Widely available, historical benchmark	Imbalanced goals, non-stationary, periodic patterns, redundant features, negatively impacts IDS results [1]
NSL-KDD	-	More balanced KDD-99 resampling	Examples missed by KDD-99 classifiers	Addresses some KDD-99 weaknesses	Absence of low-footprint attacks
The DEFCON dataset	2000 (Defcon-8), 2002 (defcon-10)	Generated during CTF competitions	Port sweeps, buffer overflow, malformed packets, telnet FTP, admin privilege	Useful for assessing alert correlation	Primarily attack traffic, dissimilar from real-world network traffic, limited applicability for comprehensive IDS evaluation
The LBNL Dataset	-	Header data from anonymized traffic	Outbound, inbound, routing traffic	Real-world traffic source	Absence of tagging process and other crucial features [1]
BoT-IoT	-	Realistic network environment with	OS, Service Scan, DoS, DDoS, Data	Fully labeled network, latest/complex attack	-

		botnet and typical traffic	Exfiltration, Protocol Keylogging	diversity, realistic environment	
IoT PoT Dataset	-	Collected using honeypots	Telnet-based attacks on MIPS, ARM, PPC CPUs (malware binary downloads)	No manual labeling/anonymization, authentic attack data	Only honeypot attacks recorded, limited network traffic visibility, traditional honeypots cannot identify varied binaries [1]
N-BaIoT Dataset	-	Traffic from IP camera and IoT networks (Mirai infected)	ARP MiTM, Active Wiretap, SSDP flood, SYN DoS, SSL Renegotiation, Mirai, OS Scan, Fuzzing, Video insertion,	Designed online network IDS evaluation, specific IoT attacks	-

Data Preprocessing

Any ML model, especially for IoT intrusion detection, requires data preparation. The research does not recommend data preparation techniques, however model efficiency and accuracy are critical in resource-constrained settings. Characteristic selection and dimensionality reduction are effective data preparation methods[68-69-70].

Importance of Data Preprocessing

Develop and use lightweight machine learning models for real-time intrusion detection to maximize IoT device CPU power. As mentioned in the conclusion, data simplification needs one approach for picking features. This accelerates app development, model performance, and object recognition. For ML algorithms to work with high-dimensional, chaotic IoT data, transformation is necessary. Because fewer data requires less computer resources for processing and inference, this aids in achieving "lightweight" and "real-time" performance requirements. Therefore, in order to fulfill the strict operating requirements of IoT IDS, data preparation is not only frequent but also crucial.

PCA helps you uncover significant qualities and makes it simpler to interpret large datasets. This strategy keeps the most significant components and gets rid of the less important ones. For future machine learning models, it's crucial that it doesn't utilize the key data to locate outliers. ML models have a hard time working on tiny devices since raw IoT data has so many dimensions and is so complicated. If you don't preprocess "lightweight" algorithms to make them easier to work with, they could be challenging to calculate. The IoT IDS performs better or worse depending on the data processing chain that comes before the ML algorithm. The pipeline organizes data so that certain machines can process it rapidly.

Model Implementation

Machine learning algorithms struggle to identify IoT network intrusions due to their distributed position and restricted resources. The study emphasizes identifying "distributed" and "lightweight" systems to overcome these limits.

Considerations for Implementation

The main objective is to create and apply lightweight machine learning models that work on IoT devices' limited processing capabilities. Model design and deployment must prioritize minimal memory, computational capacity, and battery life. The broad deployment of IoT devices makes centralized security platform-based solutions inefficient, requiring distributed, low-impact detection methods. The model deployment must be decentralized, with processing near the data source. Due to the magnitude and geographical dispersal of IoT devices, this marks a major architectural transition from centralized to distributed security architectures.

"Edge intelligence" uses modest machine learning models on devices at the edge to assist users make choices quickly. Processing should be done on the device or locally to reduce network load and latency. Devices that are part of the Internet of Things and don't have a lot of resources may have trouble running complex algorithms. This becomes evident

when there isn't much space or power on the PC. This means that the model's memory use, reasoning ability, and size need to be carefully looked at. An algorithm's performance in a high-resource environment doesn't guarantee IoT success if it's not set up for peripheral usage. The model's implementation must describe how it was built or optimized for low-resource IoT devices. The likelihood of implementation substantially influences algorithm selection.

Evaluation Metrics

In demanding IoT contexts, Intrusion Detection Systems (IDS) must be thoroughly tested. The suggested machine learning-based IDS models' important performance metrics are addressed below. It targets general and Internet of Things-specific detection accuracy indicators.

Terms Used in Core IDS Evaluation

Standard categorization metrics are usually used to assess how well IDS works:

True Positive: An IDS alarm goes off appropriately because of a bad activity.
False Positive: An alarm goes out, but there is no real attack happening.

False Negative: An assault is going on, but there is no alarm.
True Negative: There is no warning and no bad behavior.
More ideas on how IDS works and what it should do:

Site Policy: A collection of rules that control how an organization's IDS is set up and runs.
Site Policy Awareness: an IDS's capacity to change its rules and settings to find new intrusions.

Confidence Value: A number that shows how likely it is that an IDS will find an attack.
Alarm filtering is a way to tell the difference between false alarms and real attacks.

Evaluation Considerations for IoT

The research assesses the power efficiency, scalability, detection accuracy, standard accuracy, and other criteria of classification algorithms. Implementing the Internet of Things requires the following procedures. Minimizing false positives and enhancing system performance in resource-limited environments are primary objectives. In extensive IoT deployments, minimize false positives to prevent warning fatigue and resource depletion. This paper discusses balancing resource utilization, accuracy of identification, and false alarms in machine learning models. Balancing numerous goals along with performance is challenging.
While classic classification metrics (TP, FN, TG) are important, the Internet of Things (IoT) is more focused on scalability, power efficiency, and minimization of false positives due to resource constraints. An accurate IoT model becomes impractical if false positives result in elevated power usage or operational burdens. Consequently, operational stress and resource consumption metrics are as significant for IoT Intrusion Detection Systems as accuracy rates.
Models must "balance the frequency of false positives, the accuracy of identification, and the allocation of resources," as articulated. It illustrates that measures are interconnected and frequently work towards conflicting goals. More processing might be required for accuracy or result in false negatives. The measurement looks for the best compromise between IoT operational constraints and risk tolerance rather than just a number. This makes informed decision-making more important than reporting.

Metric	Definition	Significance in IoT Context
True Positive (TP)	An IDS alarm correctly indicates a malicious action.	Essential for effective threat mitigation; directly reflects successful attack detection.
False Positive (FP)	An alert is generated, but no actual attack is occurring.	High FP rates lead to alert fatigue, wasted resources, and potential disregard for genuine threats in large-scale IoT deployments. Minimizing FPs is critical for operational efficiency.

False Negative (FN)	Represents a missed attack; critically dangerous as it allows intrusions to persist undetected, compromising IoT system integrity and availability.	
True Negative (TN)	No alarm is generated, and no malicious activity is present.	Indicates the system is not generating unnecessary alerts for normal behavior, contributing to system stability and resource conservation.
Detection Accuracy	Overall correctness of classification (TP + TN) / Total.	Fundamental measure of model performance, but must be balanced with other IoT-specific factors.
Scalability	Ability of the IDS to handle increasing numbers of devices/data.	Crucial for IoT given the billions of connected devices; ensures the solution remains effective as the network grows.
Power Efficiency	Energy consumption of the IDS model/system.	Paramount for battery-powered IoT devices; directly impacts device longevity and maintenance costs.
Frequency of False Alarms	Rate at which false positives occur.	Directly impacts operational burden and trust in the IDS; high frequency can render the system unusable.
Precision of Identification	Proportion of positives among all positive alerts (TP / (TP + FP)).	Indicates the reliability of positive alerts; high precision reduces wasted investigative efforts.
Amount of Resources Used	Computational, memory, and network resources consumed.	Directly relates to the "lightweight" requirement for IoT devices; impacts deployability and operational cost.

CONCLUSION

Given the proliferation of Internet of Things devices, each possessing distinct privacy needs, standard security methodologies may prove inadequate in this context. The study results clearly indicate an urgent need for new intrusion detection systems specifically designed for this context. This research shows how machine learning may be used to find attacks, especially when it comes to making defenses that can change and adapt to deal with increasingly complex attacks. To reach this goal, we need to carefully look at a number of machine learning methods.

Many IoT devices have processing power that works with machine learning models to find intrusions in real time while keeping the system running as usual. The study's findings show that applying feature selection procedures is very important for making data less complex. This makes it easier to find things more accurately, makes the model work better, and allows for more applications. When setting up networks for the Internet of Things, you need to think about three main things.

The study shows that collaborative learning is a new and helpful way to go forward in the future. This one-of-a-kind technology makes open IoT devices safer and keeps user data safe. This method works well to safeguard people's privacy while still allowing them to adapt to new situations and learn.

To protect important aspects like privacy, availability, and integrity, intrusion detection systems must utilize strong and flexible machine learning techniques. This is especially true now that the Internet of Things business has expanded so rapidly. Findings of this inquiry will lead to improved methods for detecting assaults that protect privacy in the future. Their suggestions will also help us learn more about how to secure the Internet of Things. The Internet of Things is a novel concept with its own security issues. Modern machine learning technologies may be capable of solving these difficulties effectively.

REFERENCES

Ashton, K. That 'Internet Of Things' Thing. RFID J. **2009**, 2, 97–114.

Perera, C.; Liu, C.H.; Jayawardena, S.; Chen, M. A Survey on Internet of Things From Industrial Market Perspective. IEEE Access **2014**, 2, 1660–1679. [CrossRef]

Islam, N.; Farhin, F.; Sultana, I.; Kaiser, M.S.; Rahman, M.S.; Mahmud, M.; Hosen, A.S.M.S.; Cho, G.H. Towards Machine Learning Based Intrusion Detection in IoT Networks. Comput. Mater. Contin. **2021**, 69, 1801–1821. [CrossRef]

Ahmad, Z.; Khan, A.S.; Nisar, K.; Haider, I.; Hassan, R.; Haque, M.R.; Tarmizi, S.; Rodrigues, J.J.P.C. Anomaly Detection Using Deep Neural Network for IoT Architecture. Appl. Sci. **2021**, 11, 7050. [CrossRef]

Union Internationale des Télécommunications. Infrastructure Mondiale de l'Information, Protocole Internet et RÉSeaux de Prochaine Génération; UIT: Tromsø, Norway, 2012.

Corici, A.A.; Emmelmann, M.; Luo, J.; Shrestha, R.; Corici, M.; Magedanz, T. IoT inter-security domain trust transfer and service dispatch solution. In Proceedings of the 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, 12–14 December 2016; pp. 694–699. [CrossRef]

Sha, K.; Errabelly, R.; Wei, W.; Yang, T.A.; Wang, Z. EdgeSec: Design of an Edge Layer Security Service to Enhance IoT Security. In Proceedings of the 2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC), Madrid, Spain, 14–15 May 2017; pp. 81–88. [CrossRef]

Al-Sarawi, S.; Anbar, M.; Abdullah, R.; Al Hawari, A.B. Internet of Things Market Analysis Forecasts, 2020–2030. In Proceedings of the 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), London, UK, 27–28 July 2020; pp. 449–453. [CrossRef]

Görmüs , S.; Aydın, H.; Ulutas , G. Security for the internet of things: A survey of existing mechanisms, protocols and open research issues. J. Fac. Eng. Archit. Gazi Univ. **2018**, 33, 1247–1272.

Ibrahim, M.; Abdullah, M.T.; Abdullah, A.; Perumal, T. An Epidemic Based Model for the Predictions of OOFI in an IoT Platform. Int. J. Eng. Trends Technol. **2020**, 52–56. [CrossRef]

Aboelwafa, M.M.N.; Seddik, K.G.; Eldefrawy, M.H.; Gadallah, Y.; Gidlund, M. A Machine-Learning-Based Technique for False Data Injection Attacks Detection in Industrial IoT. IEEE Internet Things J. **2020**, 7, 8462–8471. [CrossRef]

Garg, S.; Kaur, K.; Batra, S.; Kaddoum, G.; Kumar, N.; Boukerche, A. A multi-stage anomaly detection scheme for augmenting the security in IoT-enabled applications. Future Gener. Comput. Syst. **2020**, 104, 105–118. [CrossRef]

Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.K.; Du, X.; Ali, I.; Guizani, M. A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security. IEEE Commun. Surv. Tutor. **2020**, 22, 1646–1685. [CrossRef]

Brun, O.; Yin, Y.; Yin, Y.; Gelenbe, E. Deep Learning with Dense Random Neural Network for Detecting Attacks against IoT-Connected Home Environments. In Security in Computer and Information Sciences: First International ISCIS Security Workshop 2018, Euro-CYBERSEC 2018, London, UK, February 26–27 2018, Revised Selected Papers 1; Springer International Publishing:

Berlin/Heidelberg, Germany, 2018; pp. 458–463.

Kim, J.; Shin, N.; Jo, S.Y.; Kim, S.H. Method of intrusion detection using deep neural network. In Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju, Republic of Korea, 13–16 February 2017; pp. 313–316. [CrossRef]

Kim, J.; Kim, J.; Kim, H.; Shim, M.; Choi, E. CNN-Based Network Intrusion Detection against Denial-of-Service Attacks. Electronics **2020**, 9, 916. [CrossRef]

Park, S.H.; Park, H.J.; Choi, Y.J. RNN-Based Prediction for Network Intrusion Detection. In Proceedings of the 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, Japan, 19–21 February 2020;pp. 572–574. [CrossRef]

Tang, T.A.; Mhamdi, L.; McLernon, D.; Zaidi, S.A.R.; Ghogho, M. Deep Recurrent Neural Network for Intrusion Detection in SDN-Based Networks. In Proceedings of the 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), Montreal, QC, Canada, 25–29 June 2018; pp. 202–206. [CrossRef]

Torres, P.; Catania, C.; Garcia, S.; Garino, C.G. An analysis of Recurrent Neural Networks for Botnet detection behavior. In Proceedings of the 2016 IEEE Biennial Congress of Argentina (ARGENCON), Buenos Aires, Argentina, 15–17 June 2016; pp. 1–6. [CrossRef]

Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. Neural Comput. **1997**, 9, 1735–1780. [CrossRef] [PubMed]

Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Evaluation of gated recurrent neural networks on sequence modeling. arXiv **2014**, arXiv:1412.3555.

Ferdowsi, A.; Saad, W. Generative Adversarial Networks for Distributed Intrusion Detection in the Internet of Things. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6. [CrossRef]

Liao, D.; Huang, S.; Tan, Y.; Bai, G. Network Intrusion Detection Method Based on GAN Model. In Proceedings of the 2020 International Conference on Computer Communication and Network Security (CCNS), Xi'an, China, 21–23 August 2020; pp. 153–156. [CrossRef]

Panda, M.; Patra, M.R. Network Intrusion Detection Using Naïve Bayes. IJCSNS Int. J. Comput. Sci. Netw. Secur. **2007**, 7, 258–263.

Buczak, A.L.; Guven, E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutor. **2015**, 18, 1153–1176.

D'Agostini, G. A multidimensional unfolding method based on Bayes' theorem. Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrometers Detect. Assoc. Equip. **1995**, 362, 487–498.

Panda, M.; Patra, M.R. Network intrusion detection using naive bayes. Int. J. Comput. Sci. Netw. Secur. **2007**, 7, 258–263.

Swarnkar, M.; Hubballi, N. OCPAD: One class Naive Bayes classifier for payload based anomaly detection. Expert Syst. Appl. 2016, 64, 330–339. [CrossRef]

Box, G.E.; Tiao, G.C. Bayesian Inference in Statistical Analysis; John Wiley & Sons: Hoboken, NJ, USA, 2011; Volume 40.

Ng, A.Y.; Jordan, M.I. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Advances in Neural Information Processing Systems; 2002; pp. 841–848. Available online: https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf (accessed on 13 July 2020).

Soucy, P.; Mineau, G.W. A simple KNN algorithm for text categorization. In Proceedings of the 2001 IEEE International Conference on Data Mining, San Jose, CA, USA, 29 November–2 December 2001; pp. 647–648.

Deng, Z.; Zhu, X.; Cheng, D.; Zong, M.; Zhang, S. Efficient kNN classification algorithm for big data. Neurocomputing **2016**, 195, 143–148.

Su, M.Y. Real-time anomaly detection systems for Denial-of-Service attacks by weighted k-nearest-neighbor classifiers. Expert Syst. Appl. **2011**, 38, 3492–3498.

Pajouh, H.H.; Javidan, R.; Khayami, R.; Ali, D.; Choo, K.K.R. A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. IEEE Trans. Emerg. Top. Comput. **2016**.

Li, W.; Yi, P.; Wu, Y.; Pan, L.; Li, J. A new intrusion detection system based on KNN classification algorithm in wireless sensor network. J. Electr. Comput. Eng. **2014**, 2014.

Kotsiantis, S.B.; Zaharakis, I.; Pintelas, P. Supervised machine learning: A review of classification techniques. Emerg. Artif. Intell. Appl. Comput. Eng. **2007**, 160, 3–24.

Du, W.; Zhan, Z. Building decision tree classifier on private data. In Proceedings of the IEEE International Conference on Privacy, Security and Data Mining; Australian Computer Society, Inc.: Sydney, Australia, 2002; Volume 14, pp. 1–8.

Quinlan, J.R. Induction of decision trees. Mach. Learn. **1986**, 1, 81–106.

Kotsiantis, S.B. Decision trees: A recent overview. Artif. Intell. Rev. **2013**, 39, 261–283.

Goeschel, K. Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive Bayes for off-line analysis. In Proceedings of the SoutheastCon 2016, Norfolk, VA, USA, 30 March–3 April 2016; pp. 1–6.

Kim, G.; Lee, S.; Kim, S. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Syst. Appl. **2014**, 41, 1690–1700.

Alharbi, S.; Rodriguez, P.; Maharaja, R.; Iyer, P.; Subaschandrabose, N.; Ye, Z. Secure the internet of things with challenge response authentication in fog computing. In Proceedings of the 2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC), San Diego, CA, USA, 10–12 December 2017; pp. 1–2.

Tong, S.; Koller, D. Support vector machine active learning with applications to text classification. J. Mach. Learn. Res. **2001**, 2, 45–66.

Vapnik, V. The Nature of Statistical Learning Theory; Springer Science & Business Media: Berlin, Germany, 2013.

Miranda, C.; Kaddoum, G.; Bou-Harb, E.; Garg, S.; Kaur, K. A collaborative security framework for software-defined wireless sensor networks. IEEE Trans. Inf. Forensics Secur. **2020**, 15, 2602–2615.

Hu, W.; Liao, Y.; Vemuri, V.R. Robust Support Vector Machines for Anomaly Detection in Computer Security. ICMLA. 2003; pp. 168–174. Available online: https://web.cs.ucdavis.edu/~vemuri/papers/rvsm.pdf (accessed on 13 July 2020).

Garg, S.; Kaur, K.; Kaddoum, G.; Gagnon, F.; Kumar, N.; Han, Z. Sec-IoV: A multi-stage anomaly detection scheme for internet of vehicles. In Proceedings of the ACM MobiHoc Workshop on Pervasive Systems in the IoT Era, Catania, Italy, 2 July 2019; pp. 37–42.

Torres, J.M.; Comesaña, C.I.; García-Nieto, P.J. Machine learning techniques applied to cybersecurity. Int. J. Mach. Learn. Cybern. **2019**, 10, 2823–2836.

Woz´niak, M.; Graña, M.; Corchado, E. A survey of multiple classifier systems as hybrid systems. Inf. Fusion **2014**, 16, 3–17.

Illy, P.; Kaddoum, G.; Moreira, C.M.; Kaur, K.; Garg, S. Securing fog-to-things environment using intrusion detection system based on ensemble learning. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–7.

Domingos, P.M. A few useful things to know about machine learning. Commun. ACM **2012**, 55, 78–87.

Zhang, H.; Liu, D.; Luo, Y.; Wang, D. Adaptive Dynamic Programming for Control: Algorithms And Stability; Springer Science & Business Media: Berlin, Germany, 2012.

Baba, N.M.; Makhtar, M.; Fadzli, S.A.; Awang, M.K. Current Issues in Ensemble Methods and Its Applications. J. Theor. Appl. Inf. Technol. **2015**, 81, 266.

Santana, L.E.; Silva, L.; Canuto, A.M.; Pintro, F.; Vale, K.O. A comparative analysis of genetic algorithm and ant colony optimization to select attributes for an heterogeneous ensemble of classifiers. In Proceedings of the IEEE Congress on Evolutionary Computation, Barcelona, Spain, 18–23 July 2010; pp. 1–8.

Aburomman, A.A.; Reaz, M.B.I. A novel SVM-kNN-PSO ensemble method for intrusion detection system. Appl. Soft Comput. **2016**, 38, 360–372.

Reddy, R.R.; Ramadevi, Y.; Sunitha, K. Enhanced anomaly detection using ensemble support vector machine. In Proceedings of the 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Chirala, India, 23–25 March 2017; pp. 107–111.

Bosman, H.H.; Iacca, G.; Tejada, A.; Wörtche, H.J.; Liotta, A. Ensembles of incremental learners to detect anomalies in ad hoc sensor networks. Ad Hoc Netw. **2015**, 35, 14–36.

Bosman, H.H.; Iacca, G.; Tejada, A.; Wörtche, H.J.; Liotta, A. Ensembles of incremental learners to detect anomalies in ad hoc sensor networks. Ad Hoc Netw. **2015**, 35, 14–36.

Breiman, L. Random forests. Mach. Learn. **2001**, 45, 5–32.

Cutler, D.R.; Edwards, T.C., Jr.; Beard, K.H.; Cutler, A.; Hess, K.T.; Gibson, J.; Lawler, J.J. Random forests for classification in ecology. Ecology **2007**, 88, 2783–2792.

Chang, Y.; Li, W.; Yang, Z. Network intrusion detection based on random forest and support vector machine. In Proceedings of the 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 21–24 July 2017; Volume 1, pp. 635–638.

Zhang, J.; Zulkernine, M. A hybrid network intrusion detection technique using random forests. In Proceedings of the First International Conference on Availability, Reliability and Security (ARES'06), Vienna, Austria, 20–22 April 2006; p. 8.

Doshi, R.; Apthorpe, N.; Feamster, N. Machine learning ddos detection for consumer internet of things devices. In Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 24 May 2018; pp. 29–35.

Jain, A.K. Data clustering: 50 years beyond K-means. Pattern Recognit. Lett. **2010**, 31, 651–666.

Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A k-means clustering algorithm. J. R. Stat. Society. Ser. C Appl. Stat. **1979**, 28, 100–108.

Bhuyan, M.H.; Bhattacharyya, D.K.; Kalita, J.K. Network anomaly detection: Methods, systems and tools. IEEE Commun. Surv. Tutor. **2013**, 16, 303–336.

Kanjanawattana, S. A Novel Outlier Detection Applied to an Adaptive K-Means. Int. J. Mach. Learn. Comput. **2019**, 9.

Muniyandi, A.P.; Rajeswari, R.; Rajaram, R. Network anomaly detection by cascading k-Means clustering and C4. 5 decision tree algorithm. Procedia Eng. **2012**, 30, 174–182.

Zhao, S.; Li, W.; Zia, T.; Zomaya, A.Y. A dimension reduction model and classifier for anomaly-based intrusion detection in internet of things. In Proceedings of the 2017 IEEE 15th International Conference on Dependable, Autonomic and Secure Computing, 15th International Conference on Pervasive Intelligence and Computing, 3rd International Conference on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), Orlando, FL, USA, 6–10 November 2017; pp. 836–843.

Zhang, B.; Liu, Z.; Jia, Y.; Ren, J.; Zhao, X. Network intrusion detection method based on PCA and Bayes algorithm. Secur. Commun. Netw. **2018**, 2018.