Archives of Computational Methods in Engineering (2021) 28:3211–3243
https://doi.org/10.1007/s11831-020-09496-0

ORIGINAL PAPER

A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges

Ankit Thakkar[1] · Ritika Lohiya[1]

Received: 11 February 2020 / Accepted: 8 September 2020 / Published online: 20 October 2020

Abstract

Internet of Things (IoT) is widely accepted technology in both industrial as well as academic field. The objective of IoT is to combine the physical environment with the cyber world and create one big intelligent network. This technology has been applied to various application domains such as developing smart home, smart cities, healthcare applications, wireless sensor networks, cloud environment, enterprise network, web applications, and smart grid technologies. These wide emerging applications in variety of domains raise many security issues such as protecting devices and network, attacks in IoT networks, and managing resource-constrained IoT networks. To address the scalability and resource-constrained security issues, many security solutions have been proposed for IoT such as web application firewalls and intrusion detection systems. In this paper, a comprehensive survey on Intrusion Detection System (IDS) for IoT is presented for years 2015–2019. We have discussed various IDS placement strategies and IDS analysis strategies in IoT architecture. The paper discusses various intrusions in IoT, along with Machine Learning (ML) and Deep Learning (DL) techniques for detecting attacks in IoT networks. The paper also discusses security issues and challenges in IoT.

1 Intusion Detection Systems for Internet of Things

The advancement in the technologies such as sensors, automation in object identification and tracking, communication between the inter-connected devices, integrated and distributed Internet services resulted in the increased use of smart devices in day-to-day activities. The combination of Internet services along with smart communication devices is referred to as Internet of Things (IoT) and the systems built using these devices are referred to as Cyber Physical Systems (CPS) [1]. According, to the infographics presented by Intel, IoT consist of a large varieties of smart sensors and devices that are making the web intelligent [2]. There were two billion inter-connected devices in 2006 which is expected to rise to 200 billion by 2020 as per the growth rate usage of IoT device, presented by Intel [2]. The applicability areas of IoT can be listed as industrial and logistic processing, automation, healthcare, securing the computing devices, and examining the environmental systems [3].

However, the demand of IoT devices with the real-world applications increased the risks for the Internet services as well as the devices [1]. The CPSs built with critical infrastructure are prone to security threats such as false alarms in the home appliances compromising the security and privacy of the individuals, faults in the power and transportation plants affecting the daily activities of the cities and countries. Thus, exposure to the vulnerabilities in system resources prone to breach the security requirements of the user as well as the system. For instance, experiments were performed with three smart devices namely Phillips Hue light-bulb, the Belkin WeMo power switch, and the Nest smoke-alarm in [4]. The experiments showed that security and privacy of the devices can be breached with ease because of the low power and computing capabilities of the connected devices that are connected in large number. Therefore, designing appropriate security solutions for the IoT networks is a challenging task for allowing users to take advantage of the opportunities offered by IoT devices while satisfying the security requirements [5].

The methods implemented for ensuring the security of IoT devices such as firewall and access control mechanism

✉ Ritika Lohiya
18ftphde30@nirmauni.ac.in

Ankit Thakkar
ankit.thakkar@nirmauni.ac.in

[1] Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat, India

focus on providing confidentiality and authenticity of the data, access control within the network of IoT devices, and developing security and privacy policies to build trust among the individuals [5]. Inspite of incorporating these security mechanisms, the IoT networks are still exposed to security threats. Therefore, IoT networks need a security mechanisms that can act as a line of defense for detecting intruders. An Intrusion Detection System (IDS) is therefore, used as the protection tool for information and communication systems [6].

An IDS is capable of examining the network activities between the connected devices and generate alert whenever any breach detected [6]. An IDS is considered to be an essential defense mechanism for the traditional IP networks due to its monitoring and alerting capability. Though, IDS performs well for the traditional networks, still developing IDS for IoT network is a challenging task. This is because of the characteristics of the IoT networks such as limited processing and storing capabilities of the IDS agent nodes of a network [7].

In traditional IP networks, security analyst assigns IDS agents that have the capability to perform in dynamically changing environment, whereas in IoT network nodes are digitally augumented with sensors, actuators, programming logic, and comuunication interface. Thus, to develop nodes with such capabilities and to ensure security is a challenge in IoT networks. In conventional IP networks, nodes exhibit reliable connectivity and forward the packets from source to destination. While, in IoT networks the devices aren't always connected. The devices exhibit intermittent connectivity that will connect periodically to save energy and bandwidth consumption. For instance, IoT-based home alarm systems have sensor nodes with intermittent connectivity and short range communication capabilities [4]. Therefore, multiple nodes are needed to forward message in the network [8–10]. The data collected from the node travel through the designated path identified by the sensors and delivers the information passing through gateway to the destination. This kind of infrastructure poses challenges for IDS to identify the intrusion. Another security challenge is related to the network protocols used by the IoT networks such as IEEE 802.15.4, IPv6 over Low-power Wireless Personal Area Network (6LoWPAN), IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), and Constrained Application Protocol (CoAP) [11]. Thus, different IoT protocols expose IDS to different vulnerabilities and requirements. Hence, such challenges should be addressed and mitigated by the IDS designed for IoT. The IDS model to be developed should be able to manage, classify, and correlate the generated alerts. Table 1 presents the protocol suite for TCP/IP networks and IoT networks.

1.1 Related Work

Research has been performed in the field of IDS considering various IoT technologies such as cloud computing [12–14], wireless sensor networks [15–17], and smart grid technologies [18–20].

In [12], a brief overview of security challenges in cloud computing environment is discussed. Here, the security threats and challenges are discussed by considering Platform as a Service and Software as a Service for cloud environment. The data life cycle of cloud computing environment is described and survey on security threats in data life cycle is presented in brief in [13]. The paper discusses the use of security threats to exploit the components of the cloud environment and reveals its effect on the different cloud entities such as providers and users. A survey on intrusion detection techniques based on cloud and attacks in cloud are presented in [14]. The paper discusses different types of intrusions in cloud environment, detection techniques used by IDS, and types of cloud computing-based IDS. The paper summarizes existing cloud-based IDS alongwith their types, postioning, data source, and detection time. The survey lists the advantages and disadvanatges of each system and also highlights deployment of IDS in the cloud network to handle security challenges of IDS.

A survey of various IDS strategies used for detecting attack in wireless sensor networks is presented in [15]. The paper lists various types of IDS and presents comparison between different types of IDS based on detection layer, detection rate, energy consumption, amount of computation needed, and magnitute of false alarms generated. The paper also discussed community-based IDS for home networks and workplaces. The paper characterized the IDS based on the detection technique used such as neural networks, fuzzy system, support vector machine, and embedded systems. The study presented in [16], is a comprehensive evaluation and analysis of communication standards along with security issues in wireless sensor networks. The paper describes

Table 1 TCP/IP Stack and IoT Network Protocol Stack

Layer	TCP/IP	IoT network
Application layer	HTTP, FTP, SMTP, POP3, SNMP, HTTPS	MQTT, CoAP, WebSocket, CoRE
Transport layer	TCP, UDP	UDP
Network layer	IPv4, IPv6	6LoWPAN, RPL, IPv6
Link layer	802.3 Ethernet, 802.11-Wireless, LAN	IEEE 802.15.4e, WirelessHART, Z-Wave, BLE

various applications of wireless sensor networks in smart grid infrastructure and also lists challenges of wireless sensor networks in smart grid applications. The paper presents comparative analysis of various communication standards used for wireless sensor networks based on protocol standard, spectrum type, frequency band, data throughput, and coverage range alongwith their advantages and disadvantages. Open research issues in the field of sensor networks and various IDS techniques implemented for mobile ad-hoc network and sensor networks are presented in [17]. The paper discusses various security threats in wireless networks based on availability, integrity, and confidentiality. The paper lists out various attacks identified mainly due to high energy consumption in wireless networks. The paper also describes components and framework for constructing IDS for wireless networks and discussed various anomaly-based IDS techniques implemented in literature.

A review on six different smart grid technologies is presented in [18] along with the emphasis to have the same communication protocol to ensure interoperability and security. The paper describes the integral components of smart grid infrastructure as well as an overview of smart grid applications. The paper highlights the need to have a common communication protocol standard to achieve interoperability. The paper also decribes challenges and critical skill gaps while deploying communication infrastructure in smart grid environment. A survey on critical challenges in smart grid technologies is presented in [19] considering various aspects such as information and communication, sensing, automation, security, control, and energy consumption. The paper discusses the evolution of smart grid technology over the years and how this evolution has given rise to security issues in the field of information and communication technology, sensing, automation and control technology, and power and energy storage technology. A brief study on cyber-physical smart grid testbeds is carried out in [20]. The objective of the study presented in [20] is to design a taxonomy and guidelines for developing and identifying features and design decisions of future smart grid testbeds. The survey describes a four step taxonomy based on smart grid technology, research goals, test platforms, and communication infrastructure. The paper also gives an overview about the existing smart grid testbeds in literature along with challenges and future research directions in the field of smart grid technology. The paper also evaluates existing testbeds on research support capacity, communication capability, security requirements, protocol support, and remote access capability. Summary of survey studied is presented in Table 2.

1.2 Motivation

Lately, various research contributions have been recorded in the field of IoT due to its potential applications in various fields. IoT presents a potential solution to ease and provide quality life to consumers with various technologies. Moreover, IoT technologies have gained recognition with the popularization of remote storage applications and big data. With easily accessible resources, new applications in the field of IoT have emerged. For instance, the prominent emerging applications of IoT are smart home, wearable technology like fitness bands, connected cars, industrial internet, smart cities, IoT in agriculture, IoT in healthcare, smart retail, and energy engagement [21]. As a consequence, this fast emerging applications with interconnected devices have raised the need for security. Furthermore, usage of the data obtained from IoT devices raise a concern regarding how and where this data can be used. This is one of the motivation of our study. However, we realize that an in-depth view of Machine Learning (ML) and Deep Learning (DL) for securing IoT networks against intrusions, is yet to be explored that ends up being the main contribution of our survey.

In our study, we intent to present an overview of the research contributions performed in the field of intrusion detection in IoT environments. The scope of this survey discusses ML and DL techniques applied for building IDS models for securing IoT networks. The goal of this study is to impart information from the current literature in the field

Table 2 Summary of survey studied

Referencs	Focus
[12]	Security challenges in cloud computing environment
[13]	Security threats in cloud computing environment
[14]	Survey different types of IDS for cloud computing environment
[15]	Survey of different types of attack in IoT-based home network and community-based IDS
[16]	Study on communication standards and security issues in wireless IoT networks
[17]	Study of attacks in IoT networks caused due to high energy consumption
[18]	Study on smart grid infrastructure and communication protocols for interoperability and security
[19]	Study on evolution in smart grid technology and its impact on other technologies along with security issues
[20]	Survey on cyber physical systems for IoT-based smart grid testbeds

of intrusion detection for IoT networks that can serve as source of knowledge for novice researchers interested in IoT and security issues. The paper also discusses different categories of intrusions for IoT networks and explores research challenges based on the study conducted. The contribution of our study is summarized as follows.

- Presenting taxonomy of IoT-based IDS, considering various IDS placement strategies in IoT networks, and IDS analysis strategies adopted for detecting attacks in IoT networks.
- Discussing various security threats by categorizing threats as Physical Intrusions, Network Intrusions, Software Intrusions, and Encryption Intrusions.
- Presenting different ML and DL techniques to build IDS for securing IoT networks.
- Based on the study conducted, paper derives various security issues and challenges in IoT-based IDS.

The roadmap of the paper is as follows: Sect. 2 gives a brief introduction of IoT along with the discussion on taxonomy of IDS in IoT networks, IDS placement strategies, and analysis strategies. Section 3 discusses various types of intrusions in IoT architecture. The Sect. 4 presents a review of various ML and DL techniques for attack detection in IoT networks. Section 5 lists out security issues and challenges in the field of IDS for IoT networks and Sect. 6 concludes the paper.

2 Introduction: Internet of Things

Internet of things (IoT) comprises of sophisticated devices embedded in the physical networks. These devices are connected together and share large amount of data with each other without any human intervention. IoT networks ensure ease and comfort in using home appliances to industrial machines [22]. The IoT systems are application specific with some general characteristics [22]. In general, a basic IoT architecture has three phases namely collection phase, transmission phase and processing phase [3]. The collection phase, aggregates the data collected from the sensor nodes deployed in the physical network. The sensor nodes capture the short ranged communication of the devices present within the limited range. To process the captured data, communication protocols are used, that have low data rates and storage capacity with limited coverage range. Therefore, the networks in this phase are often called as Low and Lossy Networks (LLN) [3]. The information collected during the collection phase is transmitted to the specific applications and users in the transmission phase. The data is transmitted using Ethernet, WiFi, Hybrid Fiber Coaxial (HFC), or Digital Subscriber Line (DSL) [3]. These technologies are grouped with IP protocols to construct a network that is capable of communicating between the devices located at the longer distances. The collection phase protocols are integrated with the IP protocols in the transmission phase through gateways. The data processed by the applications retrieve necessary details regarding the physical network in the processing phase. Thereafter, necessary actions are taken for controlling and managing objects present in the network. The processing phase is responsible for integrating and communicating between the physical devices and applications. Study have been conducted towards developing standardized communication protocols for IoT devices. The communication protocols demand for providing security to devices and applications [23]. Some of the protocols include IEEE 802.15.4, Bluetooth Low Energy (BLE), WirelessHART, Z-Wave, LoRaWAN, 6LoWPAN, RPL, CoAP, and Message Queue Telemetry Transport (MQTT) [24]. IEEE 802.15.4, 6LoWPAN, RPL, CoAP, and MQTT are standards designed to address specific layers of LLNs protocol stack. However, there are also IoT standards that specify vertically integrated architectures, such as BLE, WirelessHART, Z-Wave, and LoRaWAN [24].

IEEE 802.15.4 is a standard proposed by the Institute of Electrical and Electronics Engineers (IEEE) for physical and medium access control layers of low-rate personal area networks. CoAP and MQTT are two of the most widely discussed application protocols for IoT networks. Constrained RESTful Environments (CoRE) is an organization that developed data transfer protocols such as CoAP for LLN [25]. This protocol is similar to HTTP protocol in IP networks [24]. CoAP is responsible for handling request and response messages in LLN. Using this request and response mechanism of CoAP, devices and users communicate with each other [25]. MQTT is a message transfer protocol, that is based on publish and subscribe operations for exchanging information between the devices [24]. It is a lightweight protocol with low power usage and requires less data packets for exchanging information. This protocol is ideal for device-to-device communication because of its ease of implementation in IoT networks [24].

BLE was found by Bluetooth Special Interest Group [24]. It is based on bluetooth technology designed for low power devices. The layered architecture of BLE consists of physical layer that transfers and modulates the bits of information, and a link layer that connects the nodes present in network and provides access to the information. The architecture of BLE is similar to piconet that consists of nodes that act as either master or salve where salve nodes connected to one master node. The link layer functions using the Logical Link Control and Adaptation Protocol (L2CAP). This protocol belongs to the simplified version of BLE. It performs the task of multiplexing the data received from the upper layers. The upper layers of L2CAP contains the Generic Attribute Profile (GATT) that search for appropriate services and the

Generic Access Profile (GAP) that defines the operations to be performed for the devices [24].

WirelessHART was developed based on Highway Addressable Remote Transducer (HART) protocol for wireless networks. It is designed for controlling industrial processes in wireless networks [22]. It is a protocol that serves as a standard for wireless communication between the devices. It provides process management, control, and asset management capability to the wireless applications. It is compatible with the existing HART protocol and serves as a standard for industrial instrument communication. In this protocol each device in the network is connected in a mesh topology. The aim behind adopting mesh topology is to provide redundant pathways at the time of failure or change in the network settings [24]. For automating home devices, a low power protocol architecture named as Z-wave was developed [24]. It is a wireless communication protocol, developed for smart home networks. It allows the devices to connect and exchange information with each other through commands. It implements two way communication with mesh topology and message acknowledgment. It consists of smart devices and a primary controller, usually referred to as smart home hub. The hub is the only device that is connected to the Internet. When hub receives command from any of the home application, it forwards the command to the destination device in the network [22]. To address low power wide area networks, LoRa Alliance developed LoRaWAN wherein, physical objects interact with each other through gateway [26]. All communications are performed at the centralized network server. Physical devices are directly connected to gateways using wireless links whereas centralized servers are connected to gateways through traditional IP networks. It is possible that a physical object can communicate to more than one gateway at a given time and centralized server are responsible for handling the communication packets and destroying the redundant packets [26].

2.1 Taxonomy of Intrusion Detection Systems for Internet of Things

The taxonomy of IDS techniques is categorized based on various attributes such as placement strategy of IDS, type of the IDS analysis strategy, type of intrusions in IoT, and the attack detection method implemented for IoT [27]. The taxonomy of IDS for IoT is shown in Fig. 1.

2.1.1 IDS Placement Strategy

The architectures for IoT are based on the three phases that are collection, transmission, and processing [27–29]. The architecture of IoT presented in [27–29] might vary from functionality aspects, but developed considering three layers namely perception layer, network layer, and application layer as shown in Fig. 2. The perception layer is responsible for communication between the sensor nodes and devices in the LLN of the physical environment. The network layer is based on the transmission phase that forwards the collected information from the perception layer to the application and

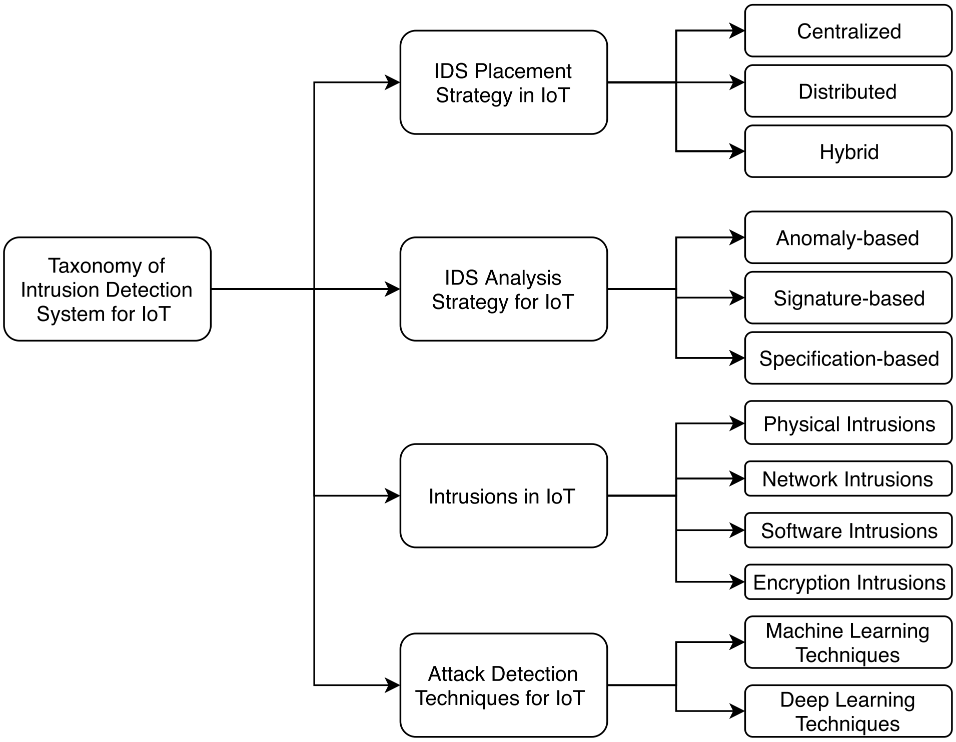

Fig. 1 Taxonomy of Intrusion Detection System in IoT

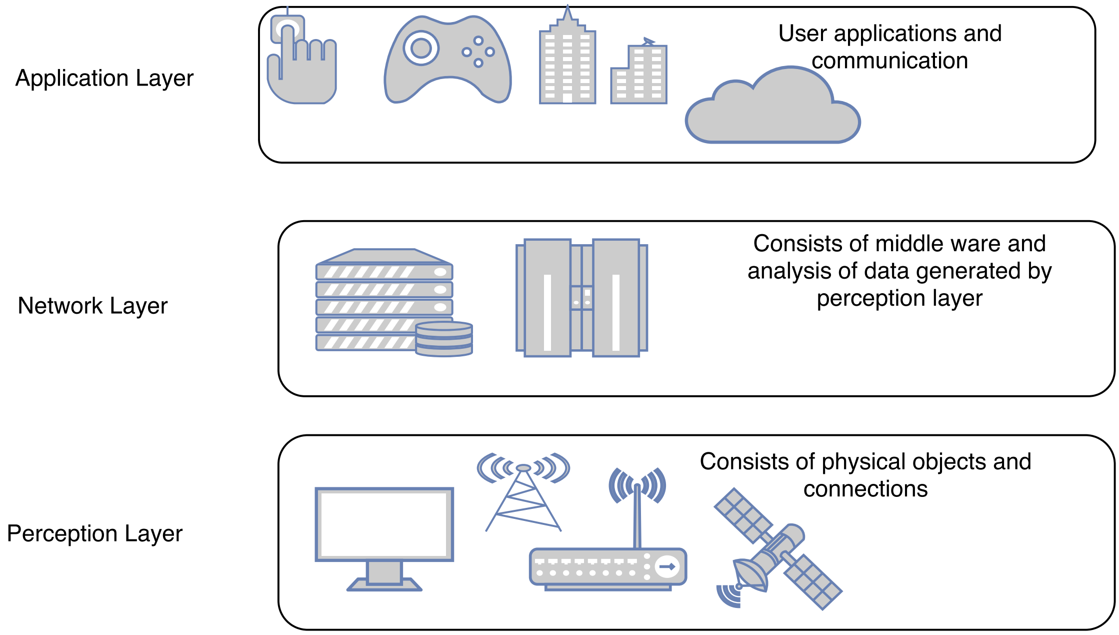
Application Layer
User applications and communication
Network Layer
Consists of middle ware and analysis of data generated by perception layer
Perception Layer
Consists of physical objects and connections

Fig. 2 Layers in IoT Architecture

users through communication protocols. Networking devices such as routers are used to connect the protocols of physical layer and network layer. The application layer is based on the processing phase that allows users to interact with the objects and applications placed in the physical environment.

While developing IDS for IoT networks, the IDS can be deployed either in the router connecting various devices or in dedicated smart devices and hosts. When the IDS nodes placed in the router, intrusions from the Internet targeted for the devices in the network can be detected easily. However, the placement of IDS nodes in the router results in communication and processing overhead because of the frequent alerts generated by the IDS [30]. Placement of the IDS nodes in the dedicated nodes of the network reduces the communication overhead but results in increased resource requirement for storing the alerts generated [30]. This limitation can be overcome by distributing the IDS nodes across the network. This results in more processing capability with less monitoring [30]. The various IDS placement strategies in IoT networks with their advantages and disadvantages are as follows.

1. *Distributed IDS* In distributed IDS placement strategy, IDS agents are deployed in every physical device of the IoT network. Therefore, these nodes need to be optimized for detecting intrusions with high efficiency. For instance, a framework for lightweight distributed IDS is proposed in [31] to examine the attack signatures and study the packet payload information for detecting intrusions. The approach was based on auxiliary shifting and early matching that resulted in reduce number of attempts needed for matching the attack patterns for detecting intrusions. The proposed work was compared with Wu-Manber (WM) algorithm, which is considered as the fastest pattern matching algorithm [32]. The experiments were conducted in the resource constraint environment. The results showed that the proposed algorithm is faster than the WM algorithm. Energy consumption based distributed IDS approach for detecting intrusions is proposed in [33]. Here, the computational overhead and resource-constraint are minimized by optimizing single node parameter. In a distributed IDS, the single node parameter refers to nodes keeping track of the neighbouring nodes by monitoring their activities. The nodes keeping track of the neighbours are called watchdogs. Based on the trust and privacy concepts of the watchdogs, an approach is proposed in [34] for detecting sinkhole attack on 6LoWPAN in IoT networks. Here, the nodes organized in the hierarchical structure and categorized as leader nodes and member nodes. Each node in the network have flexibility of changing its role at the time of configuring the network or detecting any intrusion. Each node in the network keeps record of the incoming and out going network traffic of its superior node. When any intrusion is detected, alarm is generated and message is forwarded to each node in the network to isolate the infected node.
2. *Centralized IDS* In centralized IDS placement strategy, IDS agents are situated at the centralized network entity such as router. The centralized node keeps the record

of each request send to other nodes in the network. The data transmitted between the nodes is also analyzed by the centralized node. Thus, the IDS agent deployed in the router examines all the data transmitted inbound and outbound of the network [30]. The centralized placement strategy poses challenges in terms of maintaining the one node structure and detecting attacks. Moreover, analysis of the network traffic at one node is not enough for identifying intrusions at the nodes and devices of the LLN [30]. Therefore, an IDS strategy must be developed that can examine the network traffic at the other low capacity nodes in the network. However, centralized IDS also faces challenge if the central node is compromised. For instance, an IDS approach is proposed in [35] that examines the network packets exchanged between the physical devices and the network. The proposed framework is designed to detect the botnet attacks in IoT networks by analyzing the traffic passing through the router. A centralized IDS placement strategy is implemented in [36, 37], with the aim of securing IDS from DoS attacks. Here, in both the proposed framework [36, 37] IDS analysis and IDS response agent are deployed in a dedicated host device and the rest of sensor nodes are deployed in LLNs. These sensor nodes capture the network traffic and forwards the information to the centralized node for analysis. These IDS sensor nodes connected through wires to the centralized node for data transmission while, the devices in LLN are arranged in the wireless network. Such arrangement of the IDS nodes do not affect underlying data transmission even when DoS attack occurs in the wireless network. This is because of the use of wired connection between the physical network and IDS. Moreover, the centralized arrangement of the network helps to overcome the resource constraints with low-power devices. A centralized approach is proposed in [38], where the IDS agent is placed within the router. The proposed work aims at identifying the attacks occuring in the physical network. To achieve the same, heartbeat protocol is used instead of analyzing the traffic passing through router. The heartbeat protocol continuously exchange messages between the nodes to ensure their availability and detect any node crash. In the proposed framework, a router broadcasts the ICMP messages to all the nodes present in the network at the regular intervals. The nodes respond to the router for ensuring their availability. However, exchanging of ICMP messages generates additional network traffic, but the paper showed that the minimum memory and energy are required to run the heartbeat protocol.

3. *Hybrid IDS* The hybrid placement strategy for the IDS combines the advantages of both centralized and distributed IDS strategy. In hybrid IDS placement strategy, the network is arranged in form of clusters and a central IDS node is assigned in each cluster. This central node manages and monitors the activities of other nodes in the network. For instance, network is organized in form of a cluster in [34], and each cluster assigned a cluster head to monitor the activities of its neighbours. In hybrid IDS, resource consumption is high compared to distributed IDS and the robust nodes are selected as cluster heads for the given cluster. A hybrid IDS approach is proposed in [39], wherein nodes are selected as IDS host node. These nodes eavesdrop the network packets in their neighbourhood for detecting intrusions. A set of rules is defined, using which the IDS nodes decides whether the neighbouring nodes are compromised or not. The set of rules defined based on the behaviour of the nodes, as each node would exhibit different behaviour. For instance, the router would experience high network traffic compared to the other nodes in the network. Thus, the proposed approach works efficiently based on the set of rules defined for each cluster of the network [39]. A hybrid IDS built using a backbone of monitor nodes [40], wherein the network is divided into regions. The monitor nodes capture the network traffic and analyze the same for identifying whether the nodes are compromised or not. The proposed approach results in less computational overhead as the monitor nodes analyze the traffic of their neighbourhood. In the hybrid approach presented in [41], network is divided into cluster having the same number of nodes and an IDS agent act as cluster head. The cluster head node is directly connected to all the members of the cluster. The network communication between the cluster nodes is captured by the cluster head as it, also acts as the IDS host agent. The cluster members forward information about their neighbours to the cluster head that helps in monitoring. In this approach the cluster head node is considered as center of all communication. A second method was also proposed in [41], wherein IDS agents were placed in the router. The only difference between the two approaches is the centralized IDS component. The study described in [41], identified and evaluated the potential internal threats given in [40]. The architecture developed in [41], decreases the storage and computational overhead for the IDS agents. The simulation results showed that the proposed approach with centralized IDS agent as cluster head can effectively identify topology attacks with a small amount of efforts. An IDS named SVELTE is proposed in [30], where router deployed in the network analyzes the RPL data for detecting intrusions. The nodes in the network perform activities such as forwarding the captured RPL network data to router and reporting the router about any intrusive information they may have recieved. The proposed framework consists of centralized nodes for analyzing data and distributed

mini-firewall for filtering the unwanted traffic entering the IoT network. Similarly, in [42], network nodes identify the changes in neighbouring nodes and forwards the information to the centralized node present in the router. The centralized node stores the information forwarded by the neighbouring routers and examines the data to detect malicious activities to track the intruders. The proposed approach showed that in resource constrained environment, adequate energy, sufficient network packets, and storage are needed. In [43], router and network nodes work in cooperation with each other to analyze the activities of the neighbouring nodes to detect intrusions. When an intrusion is identified, an alert for the same is forwarded to the router. The router consists of IDS modules that correlates the alerts generated by the different nodes in the network to identify and classify the intrusion. The paper states that, IDS approach presented in the paper is distributed IDS. However, the final decision is taken by IDS agent deployed in the router which makes the approach a hybrid IDS [43].

2.1.2 Analysis Startegies

An IDS is classified into four types based on the analysis strategy adopted for detecting intrusions, namely, anomaly-based IDS, signature-based IDS, specification-based IDS, and hybrid IDS. This section discusses IDS techniques developed for IoT networks.

1. *Signature-based IDS* In signature-based IDS, attack is detected by matching the attack signature of the analyzed network traffic with stored signature in the database. An alarm is generated if the signatures are matched. Signature-based IDS, work better with the known attacks and are easy to implement. However, it is not able to detect novel attacks and variants of known attack signatures as these attack signatures are not present in the attack signature database [44, 45]. A signature-based IDS is proposed for Artificial Immune System in [46]. Here, immune cells detect the intrusions based on the attack signatures. The network datagrams packets are classified as malicious and normal by these immune cells. Also, the proposed approach possessed the capability of adapting to changes in the network environment. However, storing the attack signatures and information of network packets is a challenge in the proposed approach. In [37], an integrated signature-based IDS is developed along with the EBBITS project [47]. Here, the proposed approach focused on detecting DoS attack in 6LoWPAN-based network. The experiments were performed using the Suricata signature-based IDS wherein generated alarm was sent to the DoS projection manager. The manager examines details such as channel interference rate and packet dropping rate to identify the intrusion. Verifying these details result in reduction in false alarm rate. To avoid the problems related to low capacity nodes, experiments performed on a Linux host. However, regular updation of signature database is not discussed in the paper [47]. For reducing the computational cost of comparing the attack signatures and packet payload, a multiple pattern detection approach is developed in [31]. Here, auxiliary shifting is applied to avoid unnecessary pattern matching. The experiments are performed by integrating the Rasberry Pi unit with omnivision 5647 sensor [31]. The devices provide, captured images to the main server. The proposed approach is compared with pattern sets from Snort and ClamAV IDS [48]. The results showed that proposed approach outperformed traditional pattern matching approaches.
2. *Anomaly-based IDS* Anomaly-based IDS detects intrusion by matching the attack patterns. An Anomaly-based IDS builds user profile after analyzing the system activities. Any deviation from the profile, is regarded as intrusion by anomaly-based IDS. This IDS is capable of detecting zero-day attacks and specifically the attacks associated with misuse of system resources. Thus, any action that does not match with normal behaviour is identified as malicious, and therefore, studying the user behaviour profile is crucial in case of anomaly-based IDS. Also, anomaly-based IDS, reports high positive rates [45, 49]. The user behaviour profile can be constructed using either statistical techniques or machine learning technique [49]. Anomaly-based IDS technique is proposed in [35] to detect botnet attacks using sensor nodes in 6LoWPAN network. The proposed method generates the network profile by computing the sum of TCP control field, packet length, and number of connected links of each sensor node in the network. The proposed framework monitors the network and generates an alarm for the nodes that produce average threshold more than computed threshold. In [50], computational intelligence techniques used for building user profile for network devices. Here, a separate network profile is created for each different IP addresses of the connected device for analysis and detecting intrusive activities in the network. In [33], energy consumption is considered for constructing behaviour profile for every node of a given network under routing scheme and route-over routing scheme. Every node in the network keeps the record of its energy consumption at the sampling rate of 0.5s. If the energy consumption value is higher than the expected threshold, it is classified as malicious and removed from the routing table. An anomaly-based IDS is proposed in [51] for resource constrained IoT network. The proposed work based on the assumption that IoT devices use less and simple protocols for performing network commu-

nication. Here, bit-pattern matching technique used for selecting features from network payloads. The network payload assumed to be a series of bytes. Features are selected by performing overlapping over the series of bytes called as n-grams. A match is found when all bits of n-gram match with corresponding bits of bit-pattern. The experiments performed on two IoT devices and results were recorded for worm propagation, tunneling, SQL injection, and directory traversal attack. Similarly, a distributed anomaly-based IDS is proposed in [43] to identify intrusions by examining attributes of network packets of neighbouring nodes such as packet size and data transfer rate. Here, the system generates the network profile by analyzing information flowing through the network. However, the specific method for building the network profile is not discussed in the paper including how to identify intrusions in the presence of the low capacity nodes in the network. The process of detecting wormhole attack in IoT networks is based on analysis of system resources [42]. This includes looking for the number of control packets exchanged between end devices or number of neighbours formed after execution of attack. Based on this ideology anomaly-based IDS is proposed for detecting wormhole attacks in IoT networks [42]. The results showed that 94% and 87% accuracy is achieved for detecting the attacker and the attack, respectively. The paper also presented analysis on power and energy consumption by the nodes in the network. The analysis revealed that the proposed framework is appropriate for IoT networks due to low power and memory consumption.

3. *Specification-based IDS* Specification-based IDS is based on the set of rules that builds behaviour profile of networking devices such as nodes, router, and server [17]. The behaviour profile consists of information related to protocol and routing table. The specification based IDS generates an alert when behaviour profile deviates from defined specifications. Thus, specification-based IDS works similar to anomaly-based IDS as they generate an alert when network profile deviates from normal behaviour. However, specification-based IDS differs from anomaly-based IDS in terms of building profile for devices. In specification-based IDS, profile is built manually by security expert by defining rules for each device of the network [17, 39, 49]. Specification based IDS report low false positive rates and they do not require prior learning for understanding the behaviour of the network devices and network flow. As these IDS, define specifications manually, they require more time to adapt to environmental changes in the network and are more vulnerable to errors [39]. A specification-based IDS is proposed in [52] for detecting DDoS attacks in IoT networks. Here, in the proposed work, maximum threshold for each middle-ware layer is specified. If resource requests exceeds the specified threshold, an alarm is generated. Another specification-based IDS proposed in [40], wherein specifications to detect RPL attacks are specified using finite state machine that examines the behaviour of the underlying network and generates an alert if intrusions are detected. Simulation trace files were used to develop finite state machine for RPL attack identification in [41]. The profile prepared using the finite state machine and converted to the set of specifications for examining the network traffic exchanged between the network nodes. The experimental results showed high detection accuracy with low false alarm rate with energy consumption overhead of 6.3%. Specification-based IDS depends on the skills of the network administrator, who defines specifications for detecting intrusions. The method proposed in [39] is based on this attribute of specification-based IDS. In [39], network administrator defines rules for detecting intrusions and an alert is generated if any defined rules is violated. Here, every node in the network is assigned to an Event Management System that performs correlation of alerts generated by all the nodes in the network. Thus, incorrectly defined specifications may result into high false alarm rate for the given network.
4. *Hybrid IDS* Hybrid IDS, combine concepts and advantages of signature-based, anomaly-based, and specification-based IDS to have high classification accuracy and detection rate. In [30], a hybrid IDS is developed to have trade-off between memory consumption of signature-based IDS and computational cost of anomaly-based IDS to detect RPL attacks in the network. In [53], experiments performed using signature-based IDS and anomaly-based IDS individually. The results showed that both IDS frameworks are inefficient in detecting certain kinds of attacks. Hence, a combined framework developed using both IDS approaches that could address more number of attacks. A hybrid IDS is developed using anomaly-based and specification-based IDS in [34], where anomaly-based IDS used for examining the network communication between the devices and specification-based IDS is used for extracting attributes for computing the reputation and trust values of the nodes. These values lies between 0 and 1. For instance, if the reputation and trust value is more than 0.5, node considered to be benign. The proposed IDS was built to detect sinkhole attacks in IoT networks. The results showed that the proposed system achieved detection accuracy of 92% in wired network and 75% in wireless network with low false alarm rate.

3 Intrusions in IoT Networks

An attack can be accomplished in the IoT network by exploiting the physical structure of the network for instance, hampering the network nodes, by exploring the vulnerabilities of the protocol used for network communication, or by injecting malicious code to circumvent the encryption policies. Based on the target of performing intrusions in IoT network and devices, types of intrusions are classified in four categories as physical intrusion, network intrusion, software intrusion, and encryption intrusion [22]. Taxonomy of types of intrusions in IoT network is presented in Fig. 3. Intrusions in physical devices and infrastructure affect the services and the processes performed on the devices. These intrusions stop the processes and are also capable of altering the data present in the system resources. Network intrusions related to routing of network packets in wired or wireless network are risky, as they can affect one or more nodes deployed in the network. It can also interrupt the network packets flowing in the network by altering the flow of packets, dropping the packets, or forwarding the packets to undefined routes. Software intrusions accomplished using malicious programs such as viruses and/or worms. These programs are destructive and dangerous form of malwares that exploit the vulnerabilities present in the hardware or software of the system. These programs are capable of stealing and altering confidential data such as passwords and deleting files from the system hardware. Intrusions related to encryption are performed by observing and decoding the side channel information passing through the channel.

3.1 Physical Intrusions

Physical intrusions are concerned with the hardware devices present in IoT networks. Different physical intrusions for IoT networks are as follows.

- *Compromised Nodes* A node is tampered by the intruder either by changing the configurations of the node or replacing it with other malicious node. An intruder can also steal the information and gain unauthorized access by altering sensitive data or routing data from the routing tables. The communication through compromised node affects processing of data at higher layers of the IoT network [54].
- *Disrupting RFID Signals* An attacker hampers the RFID systems by compromising the reader component of RFID. The reader component is compromised by sending unwanted noise signals instead of radio frequency signals. This results in interference in the communication between the devices present in wireless IoT network [55].
- *Node Jamming* Node jamming is one of the exploits performed in wireless networks that results in DoS attack, wherein authorized devices are denied access to the legitimate network traffic using tools for sending frequencies of illegitimate traffic. The frequencies of signals flowing through the network tampered in such a way that the network can no longer function properly [54].

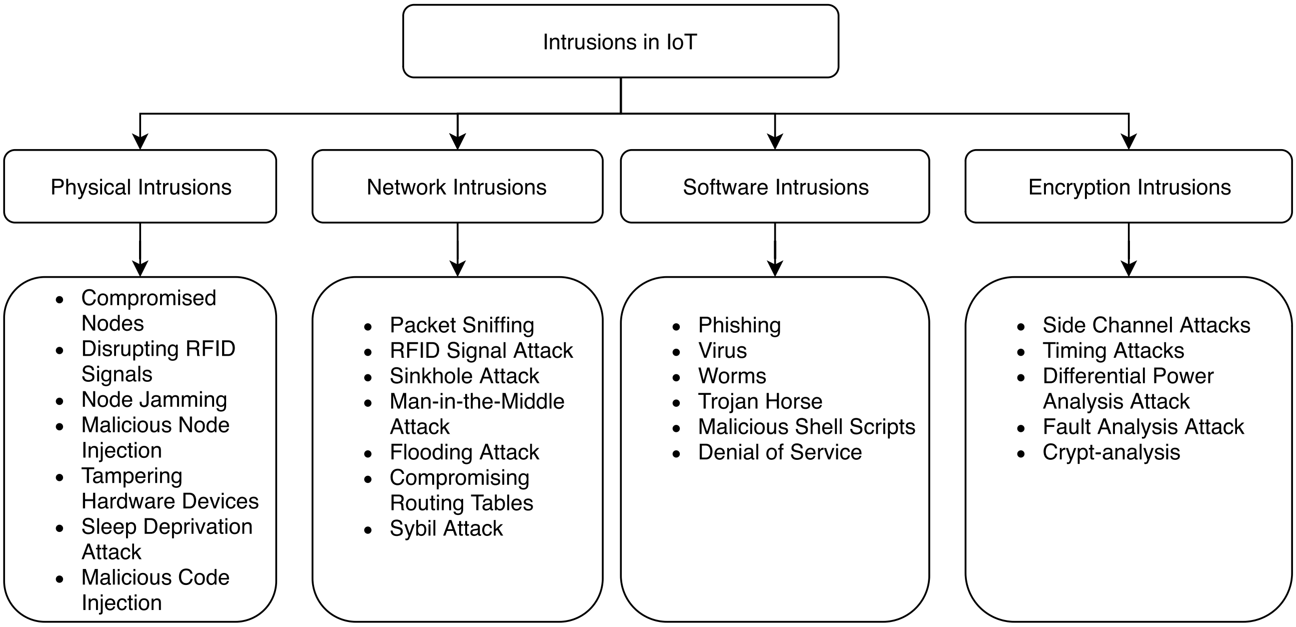

Fig. 3 Intrusions in IoT Networks

- *Malicious Node Injection* An attacker, physically inserts a malicious node in the network. This node intercepts the communication between the benign nodes to steal the data flowing in the network. The malicious node also capable of modifying the data flowing between the network. An attacker injects the malicious node by creating the replica of the node to perform an attack so as, the victim node cannot send or receive any packet in the network [56].
- *Hardware Component Tampering* An intruder can physically harm the devices and components of the IoT network resulting in DoS attack [56].
- *Social Engineering* An attack is performed by doing physical interaction to manipulate configurations of IoT devices. This leads to gaining access to sensitive information about the system [22].
- *Sleep Deprivation Attack* This attack is performed by an intruder with the aim to maximize the power consumption of nodes of a network that minimizes the lifetime of nodes [57].
- *Malicious Code Injection* An intruder injects malicious code in the system in order to gain access of the IoT system [57].

3.2 Network Intrusions

Network intrusions aim at harming the IoT network by disrupting the network activities or stealing the network information. Types of network attacks are described as follows:

- *Packet Sniffing Attacks* The intruder sniffs network packets flowing in the network using packet sniffing or packet capturing tools. These tools sniff the packets and extract the packet header and packet payload information. This information analyzed by the intruder for performing network intrusions [54].
- *RFID Signal Attacks* IoT networks consists of radio frequency identification technology known as RFID for tracking the objects in the network. The RFID consists of two components namely tags and signals for the communication between the IoT devices. The tags and signals vulnerable to attacks such as spoofing, cloning, and unauthorized access. In spoofing, an intruder pretends to send legitimate signals with the aim to retrieve information from the system [55]. In cloning attack, an intruder creates a clone of already existing RFID tag in order to inject malicious information or get control of information flowing in the network [7]. By compromising signals and tags of RFID, an intruder can gain information about credentials of the system. This results in unauthorized access to the information available at nodes of IoT network. An intruder can eavesdrop, modify, or remove the information from the system [7].
- *Sinkhole Attack* In this attack, intruder compromises a node in the network and executes the attack through the compromised node. The victim node then sends wrong routing information to its neigbours, indicating false path distance between the source and destination. In this way, it attracts network traffic towards itself and thereafter, it either alters the routing information or drops the packet. The sinkhole attack can be identified by computing the average hop count of each node of a network and comparing it with the average hop count [58]. If the minimum hop count is below the threshold value for the given node than that node may be considered as vulnerable to sinkhole attack [58].
- *Man-in-the-middle attack* An attacker eavesdrops the communication between the nodes and so this is called as man-in-the-middle attack [7]. Attacker tries to intercept the network communication between the victim nodes [7].
- *Denial of Service (DoS)* This attack is performed by overwhelming the nodes by flooding large number of service request. Thus, intended users are not able to get the legitimate resources [54].
- *Compromising Routing Tables* Here, routing tables are compromised by spoofing or altering the routing information. This results in increased packet dropping ratio, forwarding wrong route information, or dividing the network.
- *Sybil Attack* In this attack, the intruder sabotage the network service by creating large number of pseudonymous identities of the compromised node and uses them to gain influence [59].

3.3 Software Intrusions

The software intrusions tamper the data present in the system resources and even affect processing running in the system. The system intrusions includes malicious software programs in the form of virus, worms, malware, or Trojan horse. Different types of software intrusions are as follows.

- *Phishing* This attack is executed using email spoofing. Often malicious programs are forwarded as mail attachments. The aim of phishing attack is to steal credentials of users and their other confidential information such as bank details [59].
- *Malicious Software* This can be in the form of virus, worms, or trojan horse. An intruder can harm the system resources by using these malicious code. The code is injected in the system using email attachments and file downloads. Virus programs are activated when users click the file or attachment. Worms are malicious programs that replicate itself without even user interaction. These malicious programs are detected using

anti-virus software, firewall, and IDS. For instance, a hybrid IDS combining anomaly-based and signature-based IDS is developed with honeypot in [60] to detect worms in the system.

- *Malicious Shell Scripts* The malicious shell scripts are injected remotely in network environment with the aim of gaining access to local user system resources. This allows the attacker to take charge of all user processes and information remotely. The attack can also access and modify the data present on the system [59].
- *Denial of Service* This attack does not let legitimate users to use the services of the system. This attack occurs at the application layer. The applications are overwhelmed with flood of requests which keeps the users deprived of the services [54].

3.4 Encryption Intrusions

The encryption intrusions are concerned with compromising encryption process by tampering or stealing the public and private encryption keys. Some of the attacks performed on encryption techniques are as follows.

- *Side-Channel Attack* Here, side channel data of encrypted devices is used by the attacker to execute the attack or steal encryption keys. The side channel data consist details regarding power consumption, computational time, and fault frequency. These information are used for stealing encryption keys. There are various side channel attacks such as timing attack, differential power and fault analysis attacks [61]. For instance, in timing attacks, computational time required to perform an operation is used for executing the attack. This information can be utilized to steal the secret keys, destroy cryptosystems, or steal information regarding components of encryption protocols such as Diffie-Hellman exponents, RSA keys [61].
- *Cryptanalysis Attack* In this type of attack, an intruder uses plaintext or encrypted text to obtain the cryptographic keys. Based on the method used, there are different forms of cryptanalysis attack such as ciphertext only attack, known plaintext attack, chosen ciphertext attack, and chosen plaintext attack [61]. In ciphertext only attack, attacker uses the cipher text to obtain corresponding plaintext [61]. In known plaintext attack, an intruder is aware about some portion of plaintext, which is utilized to decipher the cipher text [61]. And in chosen plaintext and chosen ciphertext attack, attacker chooses plaintext and ciphertext to derive the encryption keys [61].

4 Review of Machine Learning and Deep Learning Techniques for IoT Security

Machine Learning and Deep Learning techniques are applied in various fields of IoT applications and systems. A schematic showing use of ML and DL techniques for IoT system is shown in Fig. 4. These techniques have characteristic of learning through experience, and therefore, these techniques are used for attack classification in IoT networks [62]. ML and DL techniques have gained significance because of development of new algorithms, generation of large amount of data, and low computational cost [62]. Over the years, ML and DL techniques have shown advancement in performing empirical analysis of various applications [62]. This paper discusses ML and DL techniques used for the attack identification and classification in IoT networks. In ML techniques, feature engineering is performed to extract relevant features for performing classification of attacks in IoT networks. Whereas, DL techniques uses various linear and non-linear processing layers for abstracting discriminative or generative features to perform pattern analysis [63]. The objective behind discussing ML and DL techniques is to provide an in-depth overview of these techniques used for securing IoT networks.

ML and DL algorithms address given application problem by using a dataset for learning. The dataset is divided in training and testing sets. The training set is used for learning and studying various features of dataset. For instance, given an intrusion detection dataset, the algorithms learn features from the training dataset to perform classification of a given sample as attack or normal. The task of ML and DL algorithm is to improve classification accuracy of the system by performing behvaioural analysis of normal and attack traffic scenarios in the network. ML algorithms are categorized in classification and clustering algorithms. Classification algorithms work with labeled data samples and build prediction model by analyzing input parameters and mapping them with expected output [62]. Thus, these methods builds relationship between input and output parameters. In the training phase of classification algorithm, learning model is trained using training set. The learning in the training phase is then utilized to predict and classify new data input [64]. In DL-based supervised techniques, multi-layer network built with an input layer, an output layer, and one or more hidden layers. These layers consists of nodes called as neurons with each layer connected through edges. The neurons are initialized with random weights and are multiplied with input parameters to return an output [65, 66].

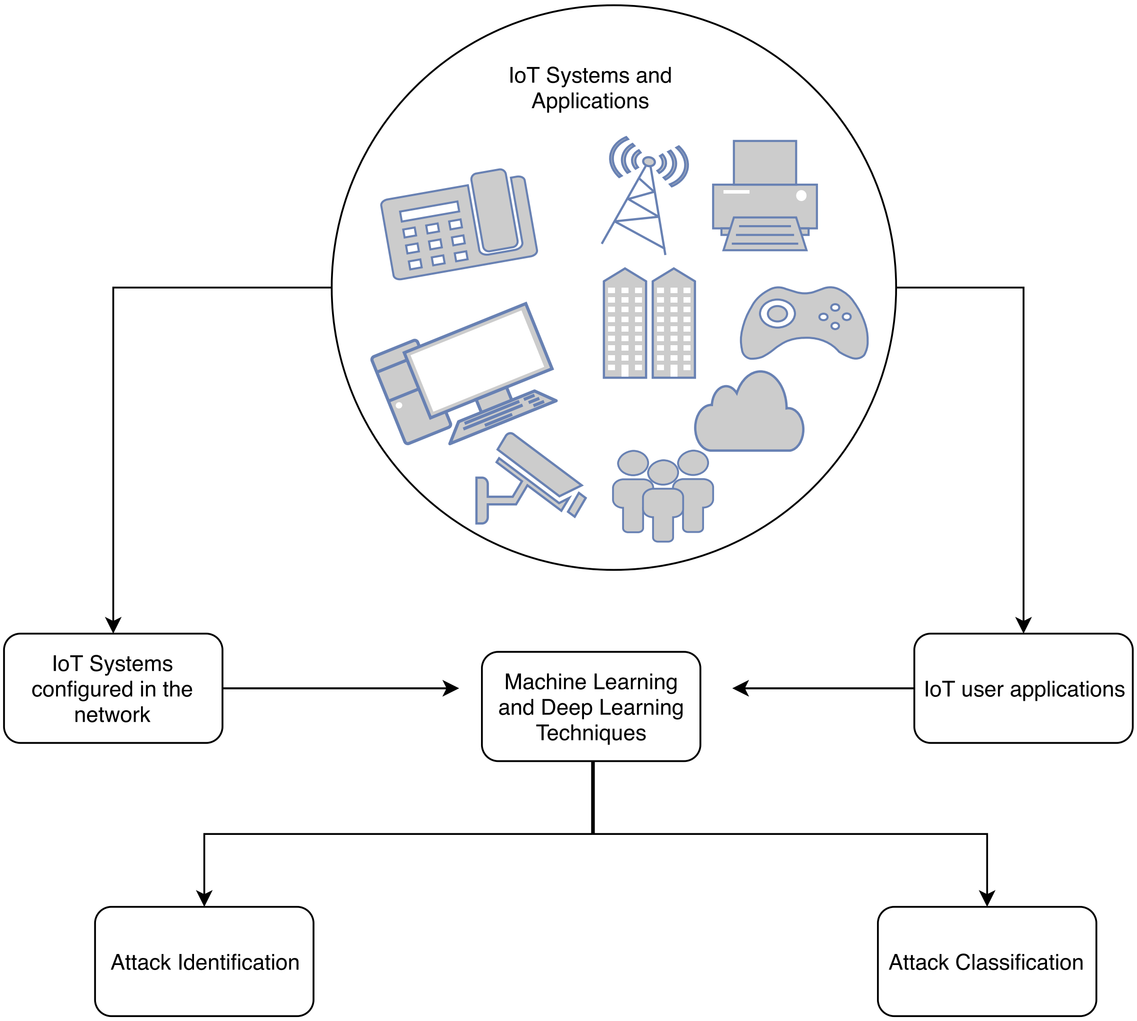
IoT Systems and Applications
IoT Systems configured in the network
Machine Learning and Deep Learning Techniques
IoT user applications
Attack Identification
Attack Classification

Fig. 4 Schematic of ML and DL techniques in IoT systems and applications

Classification methods are recognized for learning through data representation and labeling, whereas clustering methods are known for learning through unlabeled datasets [67]. Clustering techniques do not require pre-labeling of data for learning and form distinctive clusters of unlabeled data on the basis of similarity characteristics between them. Another common type of ML technique is Reinforcement Learning (RL) [62, 64]. RL techniques works in coordination with the environment for learning data. Its objective is to analyze the environment and derive the best method for agents present in the environment [68]. RL techniques are trial-and-error techniques. On the basis of environmental attributes, a set of actions are defined for the given set of input parameters. In this section, ML and DL techniques are discussed with their advantages, disadvantages, and applications in securing IoT networks.

4.1 Machine Learning (ML) Methods for IoT

The common ML algorithms i.e. decision trees (DT), support vector machines (SVM), Bayesian algorithms, k-nearest neighbour (KNN), random forest (RF), association rule (AR) algorithms, ensemble learning, k-means clustering and principal component analysis (PCA) are discussed with their advantages, disadvantages, and applications in IoT security.

4.1.1 Decision Trees

In Decision Tree (DT) classifier, data samples are classified based on their feature values. Here, data is organized in a tree like structure, where each node in a tree denotes a feature or an attribute of the dataset and each branch represents a decision rule that splits the data based on feature value. The aim of DT classifier is to develop a training model which can be used to derive class labels for target variable by learning decision rules learned from the training data [69]. In order to derive optimal feature set, two techniques are used namely information gain and gini index [70]. There are two main task in DT classifier namely building the decision tree and classification [71]. The root node of decision tree is constructed by computing the information gain or gini index of the features identified in the dataset. The class label with highest gain value or gini index is selected as root node. Gradually, gain values and gini index for other class labels is computed to build the decision tree. This process is performed until all class labels are set in the decision tree with their respective feature nodes. After the tree is constructed, feature set for the new samples is derived by traversing through the tree to reach to a leaf node. In this way, classes are predicted for the new input samples [71].

The decision tree models are commonly used for selecting variables, computing relative importance of each variables present in the dataset, managing the missing values, predicting class labels, and data manipulation [71]. The process to construct DT is explained as follows.

- *Construction of Nodes* A decision tree consists of three types of nodes such as root node, internal nodes, and leaf nodes. The root node and internal nodes represent the class labels of given dataset. These are formed by computing the gain value or gini index value of class labels.
- *Edge* An edge represent expected output or decisions from root node and internal nodes. Each path from the root node or internal nodes to the leaf nodes represent a set of decision rule for classifying the data.
- *Splitting* Input variable of the identified target variables are used for splitting the internal nodes into leaf nodes. The input variables can either be discreet or continuous variables. The splitting of nodes is performed until defined stopping criteria is satisfied.
- *Termination Condition* A termination condition is selected based on the analysis and attributes of the dataset being used. Some of the common parameters considered while deciding termination condition are minimum number of data samples, minimum number of records in node before splitting, and number of steps from the root node [72]. The termination criteria is defined in DT model to ensure robustness and simplicity.
- *Purning* Purning is performed to optimize the size of large DT, either by eliminating few nodes or by providing minimum information about the DT. There are two types of purning namely forward and backward [72].

DT along with different ML classifiers is used for securing the IoT networks and devices against intrusions [72, 73]. For instance, a fog-computing IDS is proposed in [74] for securing the IoT devices against DDoS attack in virtual private network. Here, DT is used to examine network traffic for identifying malicious traffic sources and detecting DDoS attack. In [75], DT is applied for building behavioural profile of IoT devices employed in the network to detect anomalies. An IDS for IoT-based systems developed in [76] using hybrid approach of inverse weight clustering and DT, where inverse weight clustering technique clusters the data based on similarities and DT classifies the clustered data. In IDS, computational power and valuable resources are not utilized properly when irrelevant data is processed and data flagged is disregarded. Therefore, in order to imporve the performance of IDS, false positive rate must be reduced. In an effort build an efficient IDS, SVM, DT, and NB are combined in [72] for reducing the false postive rate of the system. In [73], a hybrid approach is proposed for detecting network attack in IoT networks. Here, the framework proposed combine misuse-based detection method with anomaly-based detection. The misuse based model uses DT, for detecting and classifying the network attacks, whereas, SVM is used for anomaly-based intrusion detection.

4.1.2 Support Vector Machines

Support Vector Machine (SVM) classifies the data using hyper-plane for dividing the data between two or more classes. The hyperplane passes through the data points in such a way that distance between the nearest data points to the hyperplane is maximized [77]. SVMs have generalization characteristics, and therefore, this method works well with the dataset having large number of features and less number of data samples [78, 79]. Initially, SVM technique was formed from statistically learning [79], and was used to classify linearly separable data into classes on a two-dimensional plane. Although, SVM can also be used for separating non-linear data by using functions called kernel tricks [77]. These functions transform the non-linear data into linear data and then hyperplane is used, that divides the data into classes.

SVMs have been applied for detecting real-time attacks in IoT networks by learning through the attack patterns while training the data [80–82]. For instance, in [80], game theory is exploited for developing network intrusion detection system. Here, advantages of radial basis function kernel and polynomial kernel are integrated with the notion of game theory to build a novel kernel-based SVM algorithm for network-based intrusion detection. SVM are used for buidling IDS for mobile ad-hoc networks in [81], where IDS detects intrusions and removes the malicious nodes from the network. Here, the performance of the proposed approach is not dependent on the network routing protocol, node mobility, and network size. In [82], SVM is used with chi-square feature selection method for multi-class classification for detecting and classifying network attacks in IoT networks.

SVMs have been applied for different security applications and are also, efficient in storage capacity with a time complexity of $\mathcal{O}(N^2)$ where, N is the number of data points [77, 79].

A linear SVM based android malware detection framework is proposed in [83] for IoT networks to secure the system resources. The proposed work is compared with other ML techniques such as Naïve Bayes (NB), RF, and DT. The results showed that SVM performs better than other classifiers. An android malware detection framework is also proposed in [84], where automated learning method for android malware detection is designed for IoT devices. However, the performance of SVM technique depends on the type of datasets, environment used for creating the datasets, and different attack scenarios [84].

SVM are also used for securing smart grid technologies. In [85], empirical analysis of SVM-based smart grid technologies is performed. The study showed that SVM technique is efficient in identifying known as well as unknown attack attacks for smart grid networks. Moreover, SVM is also used for exploiting IoT device security [86, 87]. The experiments performed in [87] has shown that the ML techniques such as SVM can effectively break the security of machine for performing attacks such as template attacks.

4.1.3 Naïve Bayes

Naïve Bayes algorithm is based on Bayes theorem that works on computing posterior probability of an event based on the prior probability of class [88]. For instance, for detecting probe attack, various network traffic features of probe attack are analyzed. Thus, probabilities of occurrence of various attacks and normal traffic can be studied using the prior probabilities of the defined class for a given dataset.

The NB technique is a classification technique that computes the posterior probability of the given class using Bayes theorem. Thus, it evaluates the probability whether a particular feature set of given input sample categorizes into a specific class or not. For instance, for IDS, NB can classify the data samples of the network traffic as benign or anomalous based on the network traffic features. The traffic features are extracted from the network packet header and payload such as time duration of connection establishment, protocol used for communication, and connection flag status. Even though, these features might depend on each other, but NB technique considers all these features independently [89]. This is because, in NB classification technique, all features contribute independently for computing probability of the data sample. [89].

NB classifier has been used for intrusion detection in [89–91], along with other ML techniques for reducing false positives. Mainly, NB classifier is used for its simplicity and ease of implementation. It can perform binary as well as multi-class classification. It also requires less training data and is robust towards irrelevant features [91]. NB classifiers cannot build useful relationships between the features and input variables as it treats each feature individually. The relationship between the features and the input variables can contribute in increasing the efficiency of IDS for detecting attacks [89].

For protecting IoT infrastructure from DDoS attack, a framework is proposed in [92]. Here, NB classification algorithm is applied to IDS agents that are deployed in entire network for identifying malicious traffic and activities in the nodes. A novel two-tier classification module is proposed in [93] for detecting User to Root (U2R) and Remote to Local (R2L) attacks in IoT networks. Here, component analysis and linear discriminant analysis are used for feature extraction and thereafter, NB and certainty factor version of k-NN are used for identifying attacks. In [94], NB and SVM classification techniques are used for detecting image spam. Here, the proposed work focuses on extracting content and correcting content using the language model on the images before classification. The proposed method converts images into text and then statistical text localization methods are used to extract the text region for spam detection.

4.1.4 k-Nearest Neighbour

The k-nearest neighbours technique is a non-parametric technique applied for classification and regression problems, where k is a user-defined constant. Here, input set consists of k nearest training samples present in the feature space. A characteristic property of k-NN algorithm is that, it is vulnerable to the size of dataset [95]. Here, in k-NN algorithm, training samples are considered as vectors in multi-dimensional feature space with a designated class label [95]. In training phase, vectors and class labels of samples are stored. In classification phase, a distance metric is used for classifying the class labels. Generally for continuous variables, Euclidean distance is used and for discreet variables Hamming distance is used as distance metric. k-NN is also used with correlation coefficient such as Pearson and Spearman.

k-NN algorithm does not work well when the class distribution is skewed [96]. That is, samples of a more frequent class try to rule the classification of the new sample. To address this, weighted classification method can be used where every data point of k-NN is multiplied with a weight proportional to the inverse of the distance from that data point [97]. Also, the skewness can be removed by applying data abstraction techniques. The performance of k-NN also depends on the value of k. If the value of k is large than it minimizes the effect of noise in classification but would also affect classification accuracy. Thus, the value of k can be selected using heuristic techniques such as hyperparameter optimization. Also, performance of the algorithm is affected by the presence of irrelevant features. This issue can be addressed by using feature selection or feature extraction algorithms [98].

k-NN algorithm has been applied for intrusion detection in [95–98]. In IoT environment, to detect U2R and R2L attacks, a feature clustering based dimensionality reduction method is used for intrusion detection in [99]. Here, k-NN classifier is used for reducing the features for further classification. To address the issues of high dimensionality and anomaly mining, k-NN algorithm is used in [100] for IoT networks. An attack detection framework is proposed in [101] for IoT devices. Here, a data-centric approach is used for processing energy consumption data and attack classification for the monitored device. The experiments were performed on real hardware devices and based on energy used, the system detects cyber as well as physical attacks. To minimize the computational time, a two phase approach is proposed using k-NN and neural networks [101]. An IDS is also developed for mobile ad-hoc networks in [102] for detecting faulty nodes and reducing the routing overhead in the networks. The proposed system efficiently secures the network and improves the packet delivery ratio.

4.1.5 Random Forest

Random Forest technique is used for performing binary as well as multi-class classification [103]. Random forest tree is constructed using many decision trees for building a precise classification model for the given problem [104]. Thus, RF is constructed using many trees that are formed randomly and are trained to predict a class for given input sample. The class having highest importance is selected as output label [104]. Even though, RF classifier is developed using DT, it still differs from DT algorithm in terms of formulating rules for classification. In DT algorithm, a set of rules is formulated in the training phase for classifying the input. Whereas, RF formulates subset of rules for voting class using DT algorithm and therefore, it does not face the problem of overfitting the data. Moreover, it requires minimum parameters for feature selection. RF classifier has the capability of calculating the feature importance based on mutual information and selecting features with highest importance for classification [105, 106]. However, RF algorithm does not work well when the training data is very large as constructing large of decision trees is time consuming process [107].

The RF algorithm has been applied in IDS for selecting relevant features as well as attack classification [108]. In [109], RF, SVM, k-NN, and ANN algorithms are used for detecting DDoS attack in IoT networks. The results obtained from the experiments showed that RF algorithm outperformed the other ML techniques in detecting the DDoS attack when selected features were used. The features were selected to minimize computational complexity and enhance the performance of the algorithms for detecting attack. Also, RF algorithm is used for detecting unauthorized IoT devices in [110]. Here, RF was trained with features derived from the network traffic with the aim of correctly identifying IoT device categories. The paper listed manually extracted features from seventeen IoT devices that belonged to nine different categories. The results showed that RF algorithm precisely identifies unauthorized IoT devices. In [111], RF is used along with SVM for building IDS. Here, RF is used for selecting features for classification of attacks performed by SVM.

4.1.6 Association Rule Mining

Association rule mining algorithms are used for detecting unknown variable in the dataset by analyzing the relationship between other variables in the dataset [112]. The objective of the AR algorithm is to examine correlation between different variable of large dataset and simultaneously develop the prediction model [112]. The constructed model is used for deriving the class of the new input data samples. AR techniques form different sets of variable, that are groups of variable that appear often in various attack scenarios [105]. For instance, in [113], AR is applied for building associations among the TCP/IP variables and different attack types. Variable used for forming association are service name, source and destination port number, source and destination IP address, were used for identifying the attack type.

A pairwise fuzzy system is proposed in [114], where genetic algorithm is used alongwith fuzzy systems for IoT networks. Here, the proposed algorithm improves the detection accuracy of the system alongwith with improvement in precision for rare attack events. AR algorithm is also used for intrusion detection in [115], where fuzzy association rules are used for building the intrusion detection model. However, AR algorithms are rarely used for IoT environments compared to other ML techniques. Therefore, AR algorithm can be combined with other methods or optimized for improving the performance of the classifier. The major drawback of AR technique is high computational time. Moreover, association rules are increased rapidly with decrease in the frequency of variables [116]. Also, this techniques is based on the relationships among the variables and their occurrences, which might not be useful in security applications.

4.1.7 Ensemble Learning

Ensemble learning in ML groups the outputs of various classification algorithms to derive a collective output that results in improved classification accuracy and detection rate [117]. EL combines homogeneous and heterogeneous classifiers for building the prediction model [117]. EL methods work better with data stream classification. This is because this method is integrated with drift detection algorithms and can easily include dynamic updates such as removal of features or inclusion of different classifier [118]. The applicability of ML techniques differ from application to application as well as characteristics of dataset being used. Hence, the technique used for one application dataset might not work well with other application dataset [119]. Moreover, EL ensures better predictive performance compared to the single classifier model [117]. Therefore, EL is used for combining different classifiers to enhance the accuracy of the system. As EL includes various methods, it minimizes the variance and works well with over-fitting of data [119]. The EL-based system can deliver better results with the given set of hypotheses and exhibit adaptability [119]. However, these methods require high computational time compared to a single classifier system as they consist of several classifiers [120].

EL has been used for intrusion detection and malware detection [121–124]. For instance, in [121], an ensemble of tree classifiers is used for detecting network attacks in IoT networks. Here, in the proposed framework REPTree is used as the base classifier with the bagging ensemble for attack detection and classification. An ensemble of SVM, k-NN, and Particle Swarm Optimization (PSO) is proposed in [122]. Here, in the proposed framework PSO is used for assigning weights to the ensemble of SVM and k-NN for attack classification. SVM is also, ensembled with rough set theory for extracting useful features from the dataset that can enhance the detection process in [123]. Ensemble learning is used for Andriod malware detection in [124]. Here, ensemble of additive logistic regression and RF classifier is used for detecting malware from the McAfee's internal repository dataset. Moreover, to minimize the computational time in resource constrained wireless network with IoT devices, a lightweight, application independent, ensemble framework is proposed in [125] for identifying anomalies in IoT network. Here, two issues are addressed: i) building an automated approach for detecting intrusions in resource constrained networks and ii) evaluating the performance using real-time dataset. The results showed that ensemble based framework performed better than each of the individual classifiers.

4.1.8 k-Means Clustering

k-means algorithm is a clustering technique, that groups data into clusters based on their similarity characteristics. Here, k refers to a number of clusters to be generated by the technique. Every data sample in the dataset is iteratively assigned to one of the clusters based on their similarity index [126]. Thus, k-means algorithm works iteratively in order to predict the class label for the given new input sample. Here, in k-means algorithm, input to the training phase is the number of clusters and the data samples from the dataset. The procedure followed by the k-means algorithm is as follows: assuming the k centriods and computing Euclidean distance of each data sample from the centriod. The data samples are assigned to the cluster having the minimum distance from the centriod. After the data samples are grouped in clusters, the cluster centers are re-computed by taking the mean of all the samples assigned to that cluster. The process continuous until the termination criteria is satisfied i.e there is no data sample that can update the existing cluster centers [126].

The major drawback of k-means clustering algorithm is the selection of k before the commencement of the algorithm. Also, this algorithm assumes that all the clusters consists of equal number of data samples [127]. The k-means algorithm is also applied for intrusion detection by computing the feature similarity index [99, 128, 129]. An IDS framework was built using k-means and DT in [128] for monitoring the behaviour of the network and generating alerts for any malicious activity detected. Clustering algorithms work well with unlabeled dataset. For IoT system security, k-means clustering has been applied to wireless sensor networks [130]. A multi-kernel approach is proposed in [129] for detecting sybil attack in industrial wireless sensor networks. Here, channel vectors are clustered in order to distinguish sybil attackers from beingn sensors. To address data anonymization in IoT networks, a clustering based approach is proposed in [131]. The application of clustering algorithm improves the data exchange security of the network. The summary of ML techniques for IoT is summarized in Table 3.

4.2 Deep Learning Methods for IoT Security

The implications of DL techniques have become indispensable in IoT network systems [132]. The DL techniques are preferred over ML techniques, because of their characteristic performance with large datasets. IoT devices employed in the network generate a large amount of data, and therefore, DL techniques are appropriate for such networks. Moreover, DL techniques have capability of extracting features from the data and represent the data in better form [133]. Also, the layered architecture of DL techniques ensure better linking of IoT network devices that serves as a base for communication between IoT devices and applications without any human interaction [133]. For instance, building smart home with IoT devices ensures automation in devices where they interact with each other [132].

The DL techniques possess a computational structure that consists of multiple processing layers to learn data representations with many layers of data abstraction [63]. Unlike traditional ML techniques, DL techniques work on

Table 3 Machine Learning techniques for IoT

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[115]	Fuzzy association rulesets are exploited for attack classification	DoS, Probe, R2L, U2R	IoT network	Association rule mining	KDD CUP 99	DR: 91% , FPR: 3.3%
[130]	Clustering for refining the output of self organizing maps for attack classification	DoS, Probe, R2L, U2R	IoT network	SOM, k-means	KDD CUP 99	Detection rate and false alarm rate graphs are presented
[83]	Detection of android malware in IoT applications	Malware	IoT android applications	SVM	Simulated dataset	Accuracy: 99.5%
[114]	Using genetic fuzzy systems for attack classification by performing one versus one binarization	DoS, Probe, R2L, U2R	IoT network	Association rule mining	KDD CUP 99	Accuracy: 99%
[124]	Using ensemble learning for feature extraction and android malware classification	Malware	IoT android applications	RF	McAfee's internal repository	Accuracy: 97.5%
[125]	Implementing incremental learning using multiple classifiers for anomaly detection in ad hoc networks	Anomaly	WSN	ELM, RLS, FA	Intel lab, IndoorWSN, Sensorscope	Precision-Recall curve are presented for performance evaluation
[85]	Detection is unobservable attacks using statistical learning methods	Anomaly	Smart Grid network	k-NN, SVM	IEEE test systems	Performance graphs are presented
[86]	A lightweight boolean masking countermeasure to secure the implementation of AES	Side channel attack	Embedded IoT devices	SVM	DPAContest V4	Results in terms of number of traces and execution time are shown
[87]	Profiling of template attacks using ML techniques	Side channel attacks	Embedded IoT devices	SVM, LR, RF	Simulated dataset	Success rate graphs of profiling phase and attack phase are shown
[93]	Two-tier classification is performed by combining NB with certainty factor of k-NN for attack classification	DoS, Probe, U2R, R2L	IoT backbone network	NB, k-NN	NSL-KDD	DR: 84.82% and FAR: 4.86%
[102]	Integrated cluster with highest connectivity packet dropping algroithm is proposed for detecting malicious node	Anomaly	MANET	k-NN	Simulated dataset	Performance graphs of packet delivery ratio, end to end delay and throughput are presents for varying value of nodes

Table 3 (continued)

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[74]	Fog computing based security system is proposed to secure the communication between IoT devices and protect them against DDoS attack	DDoS	VPN	DT	Simulated dataset	Average response latency: 150 ms
[76]	A hybrid approach combining DT with inverse weight clustering is proposed to detect intrusions in IoT networks	Anomaly	WSN	DT-IWC	Intel lab IoT dataset	Accuracy: 97%
[100]	A fuzzy membership function is designed to handle large dataset and anomaly mining	DoS, Probe, R2L, U2R	IoT network	k-NN	NSL-KDD	RoC curve has been presented
[110]	Detection of unauthorized IoT devices for ensuring the security of the organization	Identify device type	Enterprise network	RF	Simulated dataset	Accuracy: 99.4%
[111]	Detecting intrusion in the network using host based statistical features	DoS, Probe, R2L, and U2R	IoT network	RF	KDD CUP 99	Detection rate: 93%, False alarm rate: 3%
[129]	A sybil attack detection scheme is proposed for wireless network by understanding the correlation between channel responses from various sensors	Sybil attack	WSN	Fuzzy c-means	Simulated dataset	Performance graphs of false negative rate and false positive rate are presented
[131]	Fuzzy clustering is used for protecting the data exchange between the IoT devices	Security of data exchanged	IoT devices	Fuzzy clustering	Intel berkley dataset	Loss rate analysis is presented
[92]	To secure the IoT infrastructure against DDoS attacks	DDoS	WSN	NB	Simulated dataset	Performance graphs of packet delivery ratio, end to end delay and throughput are presents for varying value of nodes
[81]	Detection of flooding attack in mobile adhoc network using AODV, DSR, and OLSR routing protocol	Flooding attack	MANET	SVM	Simulated dataset	Performance graphs of packet delivery ratio, end to end delay and throughput are presents for varying value of nodes
[75]	To perform behavioural fingerprinting of IoT devices using ML technique	–	IoT Home network	DT, Gradient Boost, k-NN	14 IoT devices	Accuracy: 99%

Table 3 (continued)

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[109]	Incorporating IoT network packets features for detecting malicious attacks	DDoS	Embedded IoT devices	RF	Simulated dataset	Accuracy: 99%
[84]	Detection of andriod malware in IoT applications	Malware	IoT android applications	DT, NN, LR, RF	DREBIN	Success rate: 99%
[101]	A two stage anomaly detection framework is proposed that detects anomaly is two stages namely short term and long term detection	Physical and cyber attacks	IoT embedded devices	k-NN	Simulated dataset	Short term detection accuracy: 90%, Long term detection accuracy: 99.5%

substantially enhanced applications [63]. DL techniques are a part of ML techniques, that work with several non-linear processing layers [133]. These processing layers perform generative or discriminative feature extraction and representation for learning patterns in data. DL technique built with deep architecture and therefore, they can represent the data in hierarchical form. Thus, DL methods are also referred as hierarchical learning [133]. The architecture of DL technique is inspired from that of a human brain where several neurons are connected with each other for processing signals [63]. DL techniques are classified in two categories namely discriminative techniques and generative techniques [63]. Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are discriminative DL techniques while Deep Autoencoder (AE), Deep Belief Network (DBN), Restricted Boltzmann Machine (RBM), Generative Adversial Network (GAN) are generative DL techniques. Combination of different DL techniques are also used, that is referred as Ensemble DL technique.

4.2.1 Convolutional Neural Networks (CNNs)

Convoluntional Neural Networks (CNN) were introduced to limit the data attributes used in a traditional neural network. The use of data attributes was minimized by using sparse interaction, parameter sharing, and equivariant representation [134]. This results in reduced connections between layers, improved computational processing and improved training time complexity [63]. In CNN architecture, it has two layers namely, convolutional layer and pooling layer. In the convolutional layers, data attributes are processed using various kernel functions of equal size [134]. The pooling layers are of two types max-pooling layers and average pooling layers [63]. These layers are used for performing sampling of the data to reduce the size of subsequent layers [135]. In max-pooling layer, input data is divided into non-overlapping clusters and maximum value for each cluster is selected [135]. Whereas, in average pooling layer, the data values are averaged for each cluster [135]. A CNN architecture also consist of activation layer that implies a non-linear activation function on each attribute present in the feature space. Activation functions are essential for learning and having non-linear functional mappings between the input parameters and response variables. There are different activation functions such as sigmoid, tanh, and Rectified Linear Unit Activation (RELU) function [136]. The benefit of using CNN is that, it exhibits automatic learning of features from the dataset that result in high accuracy. However, CNN approaches requires high computational cost and therefore, applying these methods in resource constrained networks is challenging. This issue can be addressed by building a distributed architecture. In [137], a lightweight deep neural network is built for a distributed network using subset of output classes. The training of the network is performed at cloud level for achieving deep classification. CNN approaches are widely accepted in image recognition applications. In [138], a large public resource image dataset named Imagenet [139] used for performing effective image classification and recognition using CNN. In [140], CNN is applied for classifying remote sensing data for ensuring security in wireless sensor networks. Moreover, in [141], CNN is used for android malware detection to provide security in IoT devices. Here, CNN approach examines the raw data related to malware and automatically learns features for detecting malware. Hence, the need for implementing feature engineering process is removed in CNN approach. Thus, CNN during the training learns the features automatically and classifies the input data simultaneously. Hence, eliminating the need to implement feature selection and feature extraction techniques for building the efficient classification model [141]. Moreover, this learning process of CNN is also used by intruders as one of the means to break the security. For instance, in [142], CNN is used for tampering cryptographic functions and procedures that have been implemented for securing the network.

4.2.2 Recurrent Neural Networks (RNNs)

A Recurrent Neural network (RNN), is a part of DL techniques that forms connections between the nodes in a directed graph structure. It exhibits dynamic behaviour with temporal sequential data. Unlike other neural network structures, RNN possess an internal memory for processing inputs and storing their previous state [143]. RNN approaches have been applied for various application data such as handwriting recognition [144, 145], text categorization [146, 147], and wireless sensor networks [148, 149].

The term "recurrent" in RNN is used to refer to the networks having structure with finite impulse and infinite impulse [143]. Both types of networks are dynamic in nature. The only difference between the networks is that, network having finite impulse form directed acyclic graph and network having infinite impulse form directed cyclic graph of the data sequences [149]. Moreover, both the types of networks has additional space for storing the state of data sequences. A feedback loop or time delay strategy can also be incorporated with the storage space and such RNN is referred as Long-Short Term Memory (LSTM) and Gated Recurrent Units (GRU), respectively [147].

RNN is useful in applications where output is obtained by analyzing the previous data. Here, output of the network is dependent on the previous input variables. Thus, a feed-forward neural network cannot be applied to such data as there is no dependency between the input and output layer [150]. In RNN architecture, a temporal layer is integrated to capture and learn different variations in multifaceted sequential data. This data is hidden in the units of the recurrent nodes [151]. The hidden nodes are altered in coordination with the data present in the network and continuously updated with respect to the network operations. RNN is used extensively because of its characteristic of handling sequentially data efficiently. This property of RNN is useful in detecting threats, where patterns of threats are dependent on time. Thus, using RNN can improve the performance of IDS. However, RNN network shows limitations of vanishing and exploding gradients [152].

RNN are used for securing IoT network and devices [153–155]. The devices in the IoT networks generate sequential data in large quantity. The data collected through network traffic flows and communication between the devices serve as the key features for identifying potential flaws in the network. For instance, in [156], RNN is used for detecting botnet attacks by analyzing network behaviour. In [157], faulty node is detected using RNN in wireless sensor networks. In [154], dense RNN is used for detecting attacks in IoT home environment. Whereas, malwares in IoT devices are detected using RNN in [153]. RNN and its variants can be used for providing security to IoT systems and devices, specifically protecting them from time-series attacks [155].

4.2.3 AutoEncoders

An autoencoder is a type of DL technique that learns the data in an unsupervised manner [63]. The main objective of AE is to perform encoding of the given data, and simultaneously reduce the size of the dataset by eliminating noisy signals. Along with reducing the dimension of the data, it also tries to reproduce the representation that is similar to the original input [63]. There are various types of autoencoders such as sparse, denoising, recursive, stacked, and contractive that are used for efficient classification tasks [158]. An AE network consists of two components namely, encoder function and decoder function. The encoder function takes data sample as input and transforms the data sample into an abstraction known as code. In the same way, decoder function takes, the code generated by the encoder function as input and tries to reproduce the original output. Thus the training phase in AE is completed with reduced reconstruction error [159]. However, AE has the limitation of perfectly replicating the input. They are capable of constructing similar input based on training data by prioritizing the characteristics of the input data [159]. AE are widely used for extracting features from the dataset [160]. They are useful for performing representation learning of the feature set instead of manually engineering the features for classification. Thus, they are capable of reducing the dimension of the dataset without even having any prior information about the dataset [160]. However, AEs require high computational time for processing the data. Even though, AEs are used for effective representation of data, they still complicate the training process [160].

AEs are extensively used for IoT system security and devices [161]. AE are also capable of detecting malwares from smart IoT devices [162]. Here, AEs were trained to learn and represent feature vectors extracted from cyber systems. The experiments were performed and compared with SVM and k-NN algorithms. The results showed that AE outperformed to the used ML algorithms in terms of detection accuracy [162]. AEs are also used for feature extraction in [163] where in, a hybrid approach is proposed by combining AE with Deep Belief Networks (DBN) to develop a malware detection system. The AEs are used for extracting features and DBN is used for classifying and detecting the malicious code. In [164], AEs are used for detecting botnet attacks in IoT networks and for classification and prediction of attacks in 5G and IoT network [165].

4.2.4 Restricted Boltzmann Machines

Restricted Boltzmann Machines (RBM) are deep generative models that are developed for performing shallow learning of probability distribution of a set of data inputs [63]. RBMs consists of two layers: hidden layer and visible layer, with no link between any two nodes in the same layer. The visible layer are fed with the input variables, while hidden layer consists of multiple nodes of latent variables [166]. RBMs build undirected models that learn features from data and the features learned in the previous layers are considered as latent variables for the next layer [166]. The development of intrusion detection model with RBM has inherent challenges. These challenges are due to the network traffic dataset that is multi-part and irregular [166]. Also, the type of anomalies also change over time. Therefore, to address these issues, a network anomaly detection model is built in [167] using discriminative RBM. The detection model exhibits scalability to group different generative models with appropriate classification accuracy to detect attack. However, the experimental results showed that the model did not perform well when different network dataset was used. Thus, a genralised classifier is needed to detect anomaly in different network environments. Also, single RBM has limited feature learning capability, and therefore, two or more RBMs could be stacked to form DBN to address this limitation [167].

4.2.5 Deep Belief Networks

A DBN is a deep generative model built by stacking RBMs [168]. The DBN model performs unsupervised training in a hierarchical manner to improve the performance of model. The DBN layers are actually trained RBM layers that are stacked on the top of each other for pre-training phase. Gradually, after the pre-training phase, DBN becomes a general feed forward network for tuning the weights with contrastive convergence [151]. In pre-training phase, features are trained using the greedy layered approach and later, softmax layer is used for the fine-tuning phase that tunes the features based on the labeled samples [169].

DBN have been applied for anomaly detection using secure mobile edge computing in [170]. Here, the proposed work was compared with other ML technqiues, and the results showed that DBN-based model performed well in detecting anomalies with high accuracy. A self adaptive model to address the network infrastructure with different types of anomalies is developed in [171]. Here, the proposed framework uses genetic algorithm along with DBN. The proposed framework generates optimal number of hidden layers and number of neurons in each layer adaptively through multiple iterations of genetic algorithm. This leads to high detection rate of the intrusion detection model. A secure architecture is developed for IoT-based SCADA network in [172], to protect intelligent devices placed in the network. Here, a hybrid approach based on DBN and SVM is built that uses network traffic features and payload features to detect anomalies. DBN technique is also applied for attack classification and analysis of IoT networks [173, 174]. Thus, DBNs are unsupervised techniques that iteratively learn data and features.

4.2.6 Generative Adversarial Networks

Generative Adversial Networks (GANs) are DL techniques that trains two models simultaneously namely generative and discriminative [175]. GAN trains the model through an adversarial procedure. The generative model generates the data samples by studying the distribution of the data while, discriminative model is used for evaluating the generated data samples. The generative model maps the feature space with data distribution, whereas discriminative network differentiates the generated data samples from the actual data distribution. The main aim of the generative network is to maximize the error rate of discriminative network [175].

In the initial training phase of the discriminator, a known dataset is given as input. Training involves, presenting GAN with data from the dataset until it gives acceptable accuracy. The input to the generator is randomized multivariate normal distribution data with a predefined feature space. The synthesized data samples generated in the generative model are then evaluated by the discriminator. Both the models, use back propagation algorithm for producing better results [175]. Thus, generator model works as deconvolutional neural network while discriminative model works as a convolutional neural network [63].

GANs are applied for securing IoT networks in [176]. Here, an architecture is proposed for protecting the cyber space of IoT systems. The proposed architecture implements

GAN technique for assessing the normal and abnormal behaviour of the system. A distributed GAN-based IDS is proposed for IoT network for detecting intrusions in IoT networks [177]. The proposed framework is a distributed IDS without any dependence on any centralized entity. Here, every device in the network monitors its neighbouring device for detecting internal as well as external attacks. The experimental results showed that GAN based distributed IDS gives higher accuracy compared to standalone IDS. Based on GAN, a novel image stegnographic techniques is proposed in [178]. Here, the proposed algorithm can secretly embed data in the foreground of the image and can also effectively address steganalysis.

GAN technique also has the potential to detect zero-day attack in IoT networks [179] by learning different attack scenarios. GANs also has the capability of producing samples more rapidly compared to other DL techniques such as DBN [179, 180]. However, training model developed using GAN is often unstable and complex and therefore, training high dimensional data is a challenging task [180].

4.2.7 Ensemble of DL Networks

An Ensemble Learning model can be developed by combining different generative and discriminative DL techniques [181]. The methods are combined to ensure improved performance compared to individual DL algorithms. The EL methods can manage complex data with high uncertainties and high-dimensional features. EL model developed by stacking either homogeneous or heterogeneous DL classifiers for improving variability, accuracy, generalization, and performance. These techniques have been applied for variety of applications such as human action recognition [182], image recognition [183], and also IoT security [184]. The EL based on DL classifiers can be implemented in distributed IoT network for detecting and classifying attacks and addressing issues related to the computation complexity [181]. The summary of DL techniques for IoT is summarized in Table 4.

5 Security Issues and Challenges

IoT have been applied in variety of vertical domains however, there is a need to cater fundamental security issues and challenges for the better momentum and growth of IoT. The security issues and challenges have been observed in the field of networking, software development and distributed systems, and enterprise networks. In networking, a large number of IoT devices are connected with each other for performing various network activities. Hence, networking challenges raise concern in terms of scalability, multitenancy, open network interface, limited resources, and security of the devices. In developing appropriate software for IoT networks, various tools and data abstraction methods are required that can easily adapt changes in the system. Hence, software development raises concern in understanding the code as well as data, configuring the devices, debugging and self-diagnosing devices, handling new and complex dependencies, addressing the semantic gap, and performing real-time analysis of the data. Cyber Physical Systems (CPS) are concerned with developing interconnected network of smart computing devices that communicate with each other in the physical network. Here, establishing human interaction with the CPS network and controlling the network in presence of attacks is a challenging task.

The research work carried out in the field of IoT security focus on different attacks in hardware devices as well as software processes. However, a standard dataset is needed for understanding the behaviour and nature of attacks as well as for comparing different techniques for attack identification and classification. The datasets used for analysis of the IoT network are conventional and needs to be updated. These datasets are either synthesized or developed by performing simulations for wired and wireless environment [185]. A detailed discussion on security issues and challenges for IoT networks is highlighted with potential future directions.

Using appropriate intrusion detection and IDS placement strategy Prior to building an IDS for IoT networks, it is important to investigate placement strategy and detection method that are suitable for a particular network. For instance, in [53], experiments were conducted with different detection method with an aim of finding detection method that has better attack detection capability. Here, the paper concluded that, building a hybrid IDS is more appropriate for detecting attacks in IoT networks. In [51], a concern was raised for detecting zero-day attacks alongwith a need to have robust devices for IoT networks. The proposed approach concluded that signature-based IDS does not work well with increase in the size of attack database in resource constrained environment. Moreover, anomaly-based IDS with a small network and limited number of resources resulted in executing few and less complex protocols. Hence, anomaly-based IDS can detect and generate alert for even a slight deviation from the normal network traffic.

Applying anomaly-based IDS for resource constrained IoT network results in high computational requirements. This issue is addressed in [42], where effect of IDS on energy consumption of nodes is computed for wireless sensor networks. Thus, for investigating appropriate detection method for IoT applications and devices, a strong empirical analysis is needed for the underlying detection methods. The empirical analysis of detection methods would reveal the impact of detection methods in identifying security threats. The characteristic property of IDS can be detection accuracy, computational time, energy consumption of nodes, and

Table 4 Deep Learning techniques for IoT

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[167]	An intrusion detection technique is proposed by combining the generalization capabilities of neural networks for improving the classsification accuracy	DoS, Probe, R2L, U2R	IoT network	RBM	KDD CUP 99	Accuracy: 88%
[137]	A distributed neural network architecture is proposed for detecting anomalies in IoT embedded devices	Anomaly	IoT embedded devices	DNN	MNIST	F-score: 97.5%
[163]	A hybrid model is proposed for detecting malicious code in IoT applications	Malicious code	IoT applications	AE, DBN	KDD CUP 99	· True Positive Rate: 92.20% · False Positive Rate: 1.58% · Accuracy: 92.10%
[142]	Building profiling approaches for breaking cryptographic implementation	Template attack	IoT network	CNN	Simulated dataset	Success rate: 100%
[156]	Behavioural models of malicious connections are studied for detecting botnet in IoT network	Botnet attack	IoT network	RNN	UNCuyo and CVUT	· Attack Detection Rate: 97% · False Alarm Rate: 1.8%
[141]	Android malware detection and classification is performed by considering the statistical analysis of the opcode sequence and dissembled programs of the android application	Malware	Android application	CNN	Android Malware Genome project, McAfee Labs dataset	· Accuracy (Android Malware Genome): 98% ·Accuracy(McAfee Lab dataset): 87%
[162]	An unsupervised feature learning approach is proposed for malware detection and classification	Malware	IoT network	AE	Microsoft Malware Classification Challenge dataset	Log loss: 0.0748
[170]	A deep learning based approach is proposed for detecting security threats in mobile cellular networks	Anomaly	IoT mobile network	DBN	Simulated dataset	Accuracy: 94%
[173]	A deep learning based approach is proposed for detecting network attacks in IoT network	Network attcks	IoT network	DBN	UNSW-NB15, CIDIDS-01, GPRS	· Accuracy for UNSW-NB15: 94.04% ·Accuracy for CIDIDS-01: 99.9% ·Accuracy for GPRS: 92.48%
[155]	Detecting distributed network attacks in social IoT networks	DoS, Probe, R2L, U2R	IoT network	RNN	NSL-KDD	Accuracy: 99.27%

Table 4 (continued)

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[184]	Deep-Q-Networks have been proposed to minimize the malware attacks while handling health information	Malware	IoT health network	Learning based Deep-Q-Network	Simulated dataset	Energy consumption, lifetime of nodes, throughput, error rate, and accuracy of malware detection with varying number of nodes has been presented.
[153]	IoT application's operation codes are analysed to detect malware	Malware	IoT application	RNN	IoT application dataset	Accuracy: 98.18%
[154]	Threats against IoT-connected home networks are analyzed and design principles for detecting network attacks are presented.	Network attacks	IoT home network	RNN	Simulated dataset	Graphical representation of attack prediction probability is shown at different time stamps.
[164]	A behvaioural model is proposed to detect malicious traffic generated from compromised IoT devices	Botnet attack	Embedded IoT devices	AE	Simulated dataset	Graphical representation of detection time and accuracy is presented for various IoT devices considered in the network.
[172]	A secure architecture is proposed for detecting malicious attacks in SCADA network traffic	Reconnai-ssance attack, Injection attack, DoS	IoT-SCADA network	DBN	SCADA network dataset	Accuracy: 95.60%
[165]	An efficient model is proposed for intrusion detection and classification in 5G and IoT networks	Flooding, Impersonation, Injection	5G and IoT network	AE	AWID dataset	Detection Accuracy: 99.9%
[171]	A self adapative model for intrusion detection is proposed for attack detection and classification	DoS, Probe, R2L, U2R	IoT network	DBN	NSL-KDD	·Classification accuracy for DoS: 99.45% ·Classification accuracy for Probe: 97.78% · Classification accuracy for R2L: 99.37% ·Classification accuracy for U2R: 98.68%
[174]	A deep learning model is build for detecting cyber attacks in IoT networks	DDoS	IoT network	DBN	CICIDS2017	Classification accuracy: 97.16%

processing overhead [1]. The selection of IDS placement strategy is also an essential issue. This is because placement of IDS in the network is related to monitoring of network traffic. An IDS should be able to monitor the network traffic transmitted from the physical devices in the network and the traffic exchanged within the devices and the hosts. The nodes in the network are generally organized in a mesh topology for performing routing in the network [42]. Hence, examining these nodes is important for identifying routing attacks. IoT devices communicate to users through application layer. Hence, identifying attacks in the traffic between the physical device and Internet is also essential. Thus, based on these scenarios, proper empirical analysis should be performed with advantages and limitations of using a particular IDS placement strategy for IoT applications.

Attack detection capability Another security concern in IoT networks and wireless sensor networks is increasing attack detection range. In both the systems, IDS are developed for identifying specific attack types such as routing attacks and DoS attacks [36]. Thus, there is a need to develop an IDS for detecting wide range of attacks. For instance, it is suggested in [30] that the framework proposed for detecting DoS attack in 6LoWPAN can be combined with SVELTE for detecting more attacks in [37]. This can be achieved by constructing specific modules for Suricata IDS [37]. Thus, there is a need to evaluate different detection methods using the same network environment and settings. The evaluation of different detection schemes would yield the amount of energy consumption, interoperability between the devices, and scalability of the system.

Detecting attack variants Another challenging task is to extend the network for detecting variants of known attacks and unknown attack. This issue is addressed in [30, 34]. Moreover, most IoT networks implement detection method to identify DoS attack, Man-in-the-middle attack, and routing attacks. Apart from these, there are many physical attacks concerned with IoT devices employed in the smart home network that should be considered [4]. An IDS should be built with the objective to detect different attack categories for IoT applications. This is because, the security level and type of attacks differ from application to application. For instance, the intensity of security and attacks in healthcare IoT application would differ from that of a IoT smart home.

Securing different IoT technologies Generally, while developing IDS for IoT network, 6LoWPAN networks are considered [22]. Hence, IDS addressing different types of IoT technologies needs to be developed. IoT is used with variety of applications and these applications are built with different IoT technologies. Therefore, to ensure the security of IoT systems with different technologies, an IDS addressing security issues of different IoT technologies should be developed. Apart from 6LoWPAN, IoT technologies such as BLE, CoAP, and Z-wave are also used for building IoT

Table 4 (continued)

References	Summary of Work	Attack	Network	Technique	Dataset	Results
[177]	A distributed IDS is proposed for protecting the user information in IoT network such as health monitoring systems and protecting network against internal as well as external attack	Anomaly	IoT network	GAN	Daily activity recognition dataset	Graphical representation of accuracy and false positive rate is presented for varying attack to signal power ratio
[178]	A deep learning approach is proposed that can resist steganalysis effectively	Steganalysis	IoT mobile network	GAN	CUB bird image set	Graphical representation of stegno images is presented

systems. These IoT technologies are also susceptible to vulnerabilities. For instance, in CoAP, physical devices are used for delivering services to the individual applications on the Internet [53]. For instance, in [53], intrusion detection for applications using CoAP is addressed. Also, smart home designed using IoT devices and systems widely used WiFi, NFC, and Bluetooth technologies for their communications [4]. Hence, an IDS should be developed for providing security to the users against attacks for such applications.

Validation strategies The validation strategies used for evaluating IDS needs to be improved. An ideal method for evaluating IDS is by considering real-time labeled network data for evaluation which consists of sufficient instructions about the network traces [186]. One of the earliest research attempt to build a standard IDS dataset was carried out by MIT Lincoln Laboratory. This dataset was named as DARPA, which contains extracted network traffic features from the network packets. This dataset contained four attacks classes namely DoS, Probe, U2R, and R2L. These attacks were injected in Windows NT audit data, processes, and file systems. Though this dataset has been widely accepted for evaluating IDS, still its precision and absence of real-world attack scenarios were criticized in [186–188]. Therefore, to create an ideal dataset for IDS, a systematic approach was adopted in [189], where a set of requirements for an ideal dataset is discussed. These set of requirements are listed as having realistic network configuration and traffic scenario, labeled data samples, and capturing entire network communication with attack scenarios. With these requirements, various concepts of network architecture such as periphery of the network, internal and external attacks, source of attacks should be clearly defined.

Need of representative dataset The datasets developed for traditional networks cannot be applied to IoT network environment. This is because set of features might be different for detecting intrusions in IoT networks. For instance, a physical test scenario is required for capturing the behaviour of network nodes connected in wired or wireless networks. For IoT systems designed for automating human activities, need an operational environment where user behaviour can be captured by mimicking user activities. Therefore, network testbeds for IoT network need to be created for evaluation. Initiative has been taken for creating testbeds for IoT systems such as SmartSantander, is an initiative for developing testbed for building smart cite for securing alerts and traffic management [191, 192]. These testbeds aim at developing efficient techniques to validate IDS in IoT networks.

Securing IDS communications An important security challenge while dealing with IDS, is securing the IDS communications. In traditional IP networks, various network protocols and virtual local area networks are used for securing the communication between the nodes and devices in the network. However, in IoT networks, certain characteristics of the devices and nodes, pose challenges for protecting the communication between them. For instance, if weak security techniques are used for protecting the communication between the sensor nodes and devices, then an intruder can easily sniff the network traffic and extract the information. Intruders can implement sniffing techniques to identify channels that are not being monitored for executing attacks. To address this issue, a wired connection between the IDS sensors is suggested in [36, 37]. In [30, 38], and [40], IDS for IoT network is built on the assumption that the communication between the nodes is secured. Also, in [100] and [39], encryption and authentication methods are used for building lightweight IDS for IoT networks. Thus, in order to ensure security of IDS communications, privacy preserving schemes should be developed for underlying IDS placement strategy as well as IoT applications.

Processing IDS alerts Addressing the security issues for implementing IDS in IoT networks is also a challenging task for the network admistrator as well as the users. In traditional IP networks, a huge amount of alerts are generated that includes a large number of false positives and low priority alerts. Thus, it becomes a tedious task for the network administrator to analyze the alerts for identifying attacks, attack source, and take timely necessary actions for building responses for attack occurrences. Thus, to address such issues in IoT networks as well, a post processing strategies for processing IDS alerts should be built such as alert correlation techniques, reducing the false positive, feature engineering, and data abstraction techniques [93, 164]. This can help the network administrators to extract useful information from the generated alerts [193, 194].

IDS Management Managing the IDS is also a challenging task for network administrators. Managing IDS is concerned with installing IDS in the network, configuring the network devices, maintaining the infrastructure, and handling complexity of processes in execution. Managing IDS in IoT networks is even more difficult, as IoT networks consist of smart devices that are ubiquitous and are deployed on a large scale. Therefore, IoT systems cannot be managed by only human interaction. This issue can be addressed by incorporating automation in managing IDS activities. Thus, by adding automation in IoT systems, tasks such as configuration of devices, adapting to network environment changes, and repairing from attacks could be accomplished with minimum human interaction. A survey on autonomic threat mitigation schemes for IoT networks is presented in [190]. A summary of security issues and challenges is presented in Table 5.

Table 5 Summary of Security Issues and Challenges in IoT networks

Challenge	Description
Using appropriate intrusion detection and IDS placement strategy	Placement strategy and detection methodology are important characteristics of IDS. The performance IDS depends on the topology of the network and available resources. Therefore, choosing an appropriate detection and placement strategy is a challenging task [51]
Attack detection capability	An IDS developed for wired and wireless networks can detect specific attacks such as routing attacks and DoS. Hence, there is need to develop an IDS that is capable of detecting wide range of attacks for the given network environment [36]
Detecting attack variants	An IDS should be developed to detect different attack categories for IoT-based networks and applications. This is because, security level and nature of attacks differ from network to network and application to application [30]
Securing different IoT technologies	IDS addressing different types of IoT technologies needs to be developed. IoT is used with variety of applications and these applications are built with different IoT technologies. Therefore, to ensure the security of IoT systems with different technologies, an IDS addressing security issues of different IoT technologies should be developed [53]
Validation strategies	Researchers have used datasets such as DARPA and KDD CUP 99 for evaluating the performance of IDS in IoT networks. These datasets have been developed with realistic network configuration and traffic scenario, labeled data samples, and capturing entire network communication with attack scenarios. However, these datasets can be applied for traditional networks, where details like network periphery and external/internal attacks could be clearly defined. The same can not be applied for IoT networks [186]
Need of representative dataset	Representative datasets are needed for comparing performance of IDS for IoT-based networks. This is because set of features might be different for detecting intrusions in IoT networks compared to traditional networks [185]
Securing IDS communications	In IoT networks, certain characteristics of the devices and nodes, pose challenges for protecting the communication between them. For instance, if weak security techniques are used for protecting the communication between the sensor nodes and devices, then an intruder can easily sniff the network traffic and extract the information [37]
Processing IDS alerts	A huge amount of alerts are generated in IoT networks that includes a large number of false positives and low priority alerts. Thus, to address such issues, post processing strategies for processing IDS alerts should be built such as alert correlation techniques, reducing the false positive, feature engineering, and data abstraction techniques [93]
IDS Management	Managing IDS is concerned with installing IDS in the network, configuring the network devices, maintaining the infrastructure, and handling complexity of processes in execution. Managing IDS in IoT networks is challenging, as IoT networks consist of smart devices that are ubiquitous and are deployed on a large scale [190]

6 Conclusion

IoT has been used widely because of its ability to interact with the physical devices of various application domains to users through Internet. However, the interconnected structure of IoT and the ability of devices to communicate with each other give rise to security issues in IoT networks. Therefore, a proper security mechanism for securing IoT networks and devices need to be developed such as Intrusion Detection System (IDS). In this paper, we have presented a survey of IDS strategies implemented for IoT networks. We begin with a general introduction of IoT technologies and IDS followed by the taxonomy of various IDS placement strategy and analysis strategy in IoT architecture along with different intrusions categories in IoT. Thereafter the paper, discusses various Machine Learning (ML) and Deep Learning (DL) techniques for IoT and presents security issues and challenges for IoT networks. From the study it can be inferred that, detection methods for IoT do not address wide range of attacks. Moreover, issues such as IDS administration, securing IDS communication between the devices, use of standardized dataset, and building techniques for correlating alerts need to be addressed. For addressing these issues, potential future research directions can be investigating advantages and disadvantages of the methods used for building IDS in IoT networks and developing new ensemble or hybrid methods to overcome the limitations of the existing methods, to address wide range of attacks for detection, to consider different types of IoT technologies used for different application domains, to develop methods with high alert correlation and low false alarm rates, to develop automatic threat mitigation methods for IoT-based applications.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Miorandi D, Sicari S, De Pellegrini F, Chlamtac I (2012) Internet of things: vision, applications and research challenges. Ad Hoc Netw 10(7):1497
2. Intel (2019) A guide to Internet of Things infographics. https://www.intel.in/content/www/in/en/internet-of-things/infographics/guide-to-iot.html (Accessed 23 July 2019)
3. Borgia E (2014) The Internet of Things vision: key features, applications and open issues. Comput Commun 54:1
4. Notra S, Siddiqi M, Gharakheili HH, Sivaraman V, Boreli R (2014) An experimental study of security and privacy risks with emerging household appliances. In: 2014 IEEE conference on communications and network security, IEEE, 2014, pp 79–84
5. Sicari S, Rizzardi A, Grieco LA, Coen-Porisini A (2015) Security, privacy and trust in Internet of Things: the road ahead. Comput Netw 76:146
6. Thakkar A, Lohiya R (2019) Role of swarm and evolutionary algorithms for intrusion detection system: a survey. In: Swarm and evolutionary computation, p 100631
7. Andrea I, Chrysostomou C, Hadjichristofi G (2015) Internet of Things: security vulnerabilities and challenges. In: 2015 IEEE symposium on computers and communication (ISCC), IEEE, pp 180–187
8. Ożadowicz A, Grela J (2017) Energy saving in the street lighting control system-a new approach based on the EN-15232 standard. Energ Effi 10(3):563
9. Elejoste P, Angulo I, Perallos A, Chertudi A, Zuazola I, Moreno A, Azpilicueta L, Astrain J, Falcone F, Villadangos J (2013) An easy to deploy street light control system based on wireless communication and LED technology. Sensors 13(5):6492
10. Wang W, Liu AX, Shahzad M (2016) Gait recognition using wifi signals. In: Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing, ACM, pp 363–373
11. Al-Fuqaha A, Khreishah A, Guizani M, Rayes A, Mohammadi M (2015) Toward better horizontal integration among IoT services. IEEE Commun Mag 53(9):72
12. Hussein NH, Khalid A (2016) A survey of cloud computing security challenges and solutions. Int J Comput Sci Inf Secur 14(1):52
13. Kazim M, Zhu SY (2015) A survey on top security threats in cloud computing
14. Chiba Z, Abghour N, Moussaid K, El Omri A, Rida M (2016) A survey of intrusion detection systems for cloud computing environment. In: 2016 international conference on engineering and MIS (ICEMIS), IEEE, pp 1–13
15. Mittal NK (2016) A survey on wireless sensor network for community intrusion detection systems. In: 2016 3rd international conference on recent advances in information technology (RAIT), IEEE, pp 107–111
16. Chhaya L, Sharma P, Bhagwatikar G, Kumar A (2017) Wireless sensor network based smart grid communications: cyber attacks, intrusion detection system and topology control. Electronics 6(1):5
17. Can O, Sahingoz OK (2015) A survey of intrusion detection systems in wireless sensor networks. In: 2015 6th international conference on modeling, simulation, and applied optimization (ICMSAO), IEEE, pp 1–6
18. Emmanuel M, Rayudu R (2016) Communication technologies for smart grid applications: a survey. J Netw Comput Appl 74:133
19. Colak I, Sagiroglu S, Fulli G, Yesilbudak M, Covrig CF (2016) A survey on the critical issues in smart grid technologies. Renew Sustain Energy Rev 54:396
20. Cintuglu MH, Mohammed OA, Akkaya K, Uluagac AS (2016) A survey on smart grid cyber-physical system testbeds. IEEE Commun Surv Tutor 19(1):446
21. Lee S, Bae M, Kim H (2017) Future of IoT networks: a survey. Appl Sci 7(10):1072
22. Ahamed J, Rajan AV (2016) Internet of Things (IoT): application systems and security vulnerabilities. In: 2016 5th international conference on electronic devices, systems and applications (ICEDSA), IEEE, pp 1–5
23. Meddeb A (2016) Internet of things standards: who stands out from the crowd? IEEE Commun Mag 54(7):40
24. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M (2015) Internet of things: a survey on enabling technologies, protocols, and applications. IEEE Commun Surv Tutor 17(4):2347
25. Shelby Z, Hartke K, Bormann C (2014) The constrained application protocol (CoAP)
26. Alliance L (2015) White Paper, A technical overview of LoRa and LoRaWAN, November
27. Gómez J, Huete JF, Hoyos O, Perez L, Grigori D (2013) Interaction system based on Internet of Things as support for education. Procedia Comput Sci 21:132
28. Bandyopadhyay D, Sen J (2011) Internet of things: applications and challenges in technology and standardization. Wireless Pers Commun 58(1):49
29. Aazam M, Khan I, Alsaffar AA, Huh EN (2014) Cloud of things: integrating Internet of Things and cloud computing and the issues involved. In: Proceedings of 2014 11th international bhurban conference on applied sciences and technology (IBCAST) Islamabad, Pakistan, 14th–18th, IEEE, pp 414–419
30. Raza S, Wallgren L, Voigt T (2013) SVELTE: real-time intrusion detection in the Internet of Things. Ad Hoc Netw 11(8):2661
31. Oh D, Kim D, Ro W (2014) A malicious pattern detection engine for embedded security systems in the Internet of Things. Sensors 14(12):24188
32. Zhang W (2016) An improved Wu-Manber multiple patterns matching algorithm. In: 2016 IEEE international conference on electronic information and communication technology (ICEICT), IEEE, pp 286–289
33. Lee TH, Wen CH, Chang LH, Chiang HS, Hsieh MC (2014) A lightweight intrusion detection scheme based on energy consumption analysis in 6LowPAN. In: Advanced technologies, embedded and multimedia for human-centric computing, Springer, pp 1205–1213
34. Cervantes C, Poplade D, Nogueira M, Santos A (2015) Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things. In: 2015 IFIP/IEEE international symposium on integrated network management (IM), IEEE, pp 606–611
35. Cho EJ, Kim JH, Hong CS (2009) Attack model and detection scheme for botnet on 6LoWPAN. In: Asia-Pacific network operations and management symposium, Springer, pp 515–518
36. Kasinathan P, Pastrone C, Spirito MA, Vinkovits M (2013) Denial-of-Service detection in 6LoWPAN based Internet of Things. In: 2013 IEEE 9th international conference on wireless and mobile computing, networking and communications (WiMob), IEEE, pp 600–607
37. Kasinathan P, Costamagna G, Khaleel H, Pastrone C, Spirito MA (2013) An IDS framework for internet of things empowered by 6LoWPAN. In: Proceedings of the 2013 ACM SIGSAC conference on computer and communications security, ACM, pp 1337–1340
38. Wallgren L, Raza S, Voigt T (2013) Routing attacks and countermeasures in the RPL-based internet of things. Int J Distrib Sens Netw 9(8):794326

39. Amaral JP, Oliveira LM, Rodrigues JJ, Han G, Shu L (2014) Policy and network-based intrusion detection system for IPv6-enabled wireless sensor networks. In: 2014 IEEE international conference on communications (ICC), IEEE, pp 1796–1801
40. Le A, Loo J, Luo Y, Lasebae A (2011) Specification-based IDS for securing RPL from topology attacks. In: 2011 IFIP wireless days (WD), IEEE, pp 1–3
41. Le A, Loo J, Chai K, Aiash M (2016) A specification-based IDS for detecting attacks on RPL-based network topology. Information 7(2):25
42. Pongle P, Chavan G (2015) Real time intrusion and wormhole attack detection in IOT. Int J Comput Appl 121(9):6989
43. Thanigaivelan NK, Nigussie E, Kanth RK, Virtanen S, Isoaho J (2016) Distributed internal anomaly detection system for Internet-of-Things. In: 2016 13th IEEE annual consumer communications and networking conference (CCNC)
43. Gupta B, Agrawal DP, Yamaguchi S (2016) Handbook of research on modern cryptographic solutions for computer and cyber security. IGI global
44. Vacca JR (2012) Computer and information security handbook. Newnes
45. Liu C, Yang J, Chen R, Zhang Y, Zeng J (2011) Research on immunity-based intrusion detection technology for the internet of things. In: 2011 seventh international conference on natural computation, IEEE, vol 1, pp 212–216
46. Vajda V, Furdík K, Glova J, Sabol T (2011) The EBBITS Project: an interoperability platform for a real-world populated Internet of Things domain. In: Proceedings of the international conference Znalosti (Knowledge), Technical University of Ostrava, Czech Republic, pp 317–320
47. Miretskiy Y, Das A, Wright CP, Zadok E (2004) Avfs: an on-access anti-virus file system. In: USENIX security symposium, pp 73–88
48. Mitchell R, Chen IR (2014) A survey of intrusion detection techniques for cyber-physical systems. ACM Comput Surv 46(4):55
49. Gupta A, Pandey OJ, Shukla M, Dadhich A, Mathur S, Ingle A (2013) Computational intelligence based intrusion detection systems for wireless communication and pervasive computing networks. In: 2013 IEEE international conference on computational intelligence and computing research, IEEE, pp 1–7
50. Summerville DH, Zach KM, Chen Y (2015) Ultra-lightweight deep packet anomaly detection for Internet of Things devices. In: 2015 IEEE 34th international performance computing and communications conference (IPCCC), IEEE, pp 1–8
51. Misra S, Krishna PV, Agarwal H, Saxena A, Obaidat MS (2011) A learning automata based solution for preventing distributed denial of service in internet of things. In: 2011 international conference on Internet of Things and 4th international conference on cyber, physical and social computing, IEEE, pp 114–122
52. Krimmling J, Peter S (2014) Integration and evaluation of intrusion detection for CoAP in smart city applications. In: 2014 IEEE conference on communications and network security, IEEE, pp 73–78
53. Uke S, Mahajan A, Thool R (2013) UML modeling of physical and data link layer security attacks in WSN. Int J Comput Appl 70(11):1099
54. Li H, Chen Y, He Z (2012) The survey of RFID attacks and defenses. In: 2012 8th international conference on wireless communications, networking and mobile computing, IEEE, pp 1–4
55. Kandah F, Singh Y, Zhang W, Wang C (2013) Mitigating colluding injected attack using monitoring verification in mobile ad-hoc networks. Secur Commun Netw 6(4):539
56. Muhammad MF, Anjum W, Mazhar KS (2015) A critical analysis on the security concerns of Internet of Things (IoT). Int J Comput Appl 111(7):198
57. Shafiei H, Khonsari A, Derakhshi H, Mousavi P (2014) Detection and mitigation of sinkhole attacks in wireless sensor networks. J Comput Syst Sci 80(3):644
58. Leloglu E (2016) A review of security concerns in Internet of Things. J Comput Commun 5(01):121
59. Jain P, Sardana A (2012) Defending against internet worms using honeyfarm. In: Proceedings of the CUBE international information technology conference, ACM, pp 795–800
60. Genkin D, Pachmanov L, Pipman I, Shamir A, Tromer E (2016) Physical key extraction attacks on PCs. Commun ACM 59(6):70
61. Jordan MI, Mitchell TM (2015) Machine learning: trends, perspectives, and prospects. Science 349(6245):255
62. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT press, New York
63. Hothorn T (2019) CRAN task view: machine learning and statistical learning
64. Schmidhuber J (2015) Deep learning in neural networks: an overview. Neural Netw 61:85
65. Gunning D (2017) Explainable artificial intelligence (xai). In: Defense Advanced Research Projects Agency (DARPA), nd Web, vol 2
66. Sutskever I, Jozefowicz R, Gregor K, Rezende D, Lillicrap T, Vinyals O (2015) Towards principled unsupervised learning, arXiv preprint arXiv:1511.06440
67. Sutton RS, Barto AG (2018) Reinforcement learning: an introduction. MIT press, New York
68. Breiman L (2017) Classification and regression trees. Routledge, New York
69. Gupta B, Rawat A, Jain A, Arora A, Dhami N (2017) Analysis of various decision tree algorithms for classification in data mining. Int J Comput Appl 163(8):15
70. Song YY, Ying L (2015) Decision tree methods: applications for classification and prediction. Shanghai Archiv Psychiatry 27(2):130
71. Goeschel K (2016) 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: SoutheastCon 2016, IEEE, pp 1–6
72. Kim G, Lee S, Kim S (2014) A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Syst Appl 41(4):1690
73. Alharbi S, Rodriguez P, Maharaja R, Iyer P, Subaschandrabose N, Ye Z (2017) Secure the internet of things with challenge response authentication in fog computing. In: 2017 IEEE 36th international performance computing and communications conference (IPCCC), IEEE, pp 1–2
74. Bezawada B, Bachani M, Peterson J, Shirazi H, Ray I, Ray I (2018) Iotsense: Behavioral fingerprinting of iot devices, arXiv preprint arXiv:1804.03852
75. Alghuried A (2017) A model for anomalies detection in internet of things (IoT) using inverse weight clustering and decision tree
76. Suthaharan S (2016) Support vector machine. In: Machine learning models and algorithms for big data classification, Springer, pp 207–235
77. Xiao H, Biggio B, Nelson B, Xiao H, Eckert C, Roli F (2015) Support vector machines under adversarial label contamination. Neurocomputing 160:53
78. Ratner B (2017) Statistical and machine-learning data mining: techniques for better predictive modeling and analysis of big data. Chapman and Hall/CRC, Oxford
79. Liu Y, Pi D (2017) KSII Trans Internet Inf Syst 11:8
80. Shams EA, Rizaner A (2018) A novel support vector machine based intrusion detection system for mobile ad hoc networks. Wireless Netw 24(5):1821

81. Thaseen IS, Kumar CA (2017) Intrusion detection model using fusion of chi-square feature selection and multi class SVM. J King Saud Univ Comput Inf Sci 29(4):462
82. Ham HS, Kim HH, Kim MS, Choi MJ (2014) Linear SVM-based android malware detection for reliable IoT services. J Appl Math 56:9999
83. Liu X, Du X, Zhang X, Zhu Q, Wang H, Guizani M (2019) Adversarial samples on android malware detection systems for IoT systems. Sensors 19(4):974
84. Ozay M, Esnaola I, Vural FTY, Kulkarni SR, Poor HV (2015) Machine learning methods for attack detection in the smart grid. IEEE Trans Neural Netw Learn Syst 27(8):1773
85. Lerman L, Bontempi G, Markowitch O (2015) A machine learning approach against a masked AES. J Cryptogr Eng 5(2):123
86. Lerman L, Poussier R, Bontempi G, Markowitch O, Standaert FX (2015) Template attacks versus machine learning revisited (and the curse of dimensionality in side-channel analysis). In: International workshop on constructive side-channel analysis and secure design, Springer, pp 20–33
87. Jadhav SD, Channe H (2016) Comparative study of K-NN, naive Bayes and decision tree classification techniques. Int J Sci Res 5(1):1842
88. Aziz ASA, Sanaa E, Hassanien AE (2017) Comparison of classification techniques applied for network intrusion detection and classification. J Appl Logic 24:109
89. Aljawarneh S, Aldwairi M, Yassein MB (2018) Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. J Comput Sci 25:152
90. Ashraf N, Ahmad W, Ashraf R (2018) A comparative study of data mining algorithms for high detection rate in intrusion detection system. Ann Emerg Technol Comput 2(1):512
91. Mehmood A, Mukherjee M, Ahmed SH, Song H, Malik KM (2018) NBC-MAIDS: Naïve Bayesian classification technique in multi-agent system-enriched IDS for securing IoT against DDoS attacks. J Supercomput 74(10):5156
92. Pajouh HH, Javidan R, Khayami R, Ali D, Choo KKR (2016) A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. In: IEEE transactions on emerging topics in computing
93. Khakurel N, Bhagat N (2019) Natural language processing technique for image spam detection. In: Advanced engineering and ICT–convergence 2019 (ICAEIC-2019), p 22
94. Li L, Zhang H, Peng H, Yang Y (2018) Nearest neighbors based density peaks approach to intrusion detection. Chaos Solitons Fractals 110:33
95. Serpen G, Aghaei E (2018) Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms. Intell Data Anal 22(5):1101
96. Saleh AI, Talaat FM, Labib LM (2019) A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers. Artif Intell Rev 51(3):403
97. Syarif AR, Gata W (2017) Intrusion detection system using hybrid binary PSO and K-nearest neighborhood algorithm. In: 2017 11th international conference on information and communication technology and system (ICTS), IEEE, pp 181–186
98. Kumar GR, Mangathayaru N, Narsimha G (2017) A feature clustering based dimensionality reduction for intrusion detection. IADIS Int J Comput Sci Inf Syst 12(1):65
99. Gunupudi RK, Nimmala M, Gugulothu N, Gali SR (2017) CLAPP: a self constructing feature clustering approach for anomaly detection. Future Gener Comput Syst 74:417
100. Shi Y, Li F, Song W, Li XY, Ye J (2019) Energy audition based cyber-physical attack detection system in IoT
101. Selvi E, Shashidara M (2016) Enhanced packet dropping algorithm and neighbour node cluster strategy for intrusion detection in MANET. Int J Comput 5(3):150
102. Resende PAA, Drummond AC (2018) A survey of random forest based methods for intrusion detection systems. ACM Comput Surv 51(3):48
103. Biau G, Scornet E (2016) A random forest guided tour. Test 25(2):197
104. Buczak AL, Guven E (2015) A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun Surv Tutor 18(2):1153
105. Hasan MAM, Nasser M, Ahmad S, Molla KI (2016) Feature selection for intrusion detection using random forest. J Inf Secur 7(03):129
106. Farnaaz N, Jabbar M (2016) Random forest modeling for network intrusion detection system. Procedia Comput Sci 89:213
107. Ahmad I, Basheri M, Iqbal MJ, Rahim A (2018) Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. IEEE Access 6:33789
108. Doshi R, Apthorpe N, Feamster N (2018) Machine learning ddos detection for consumer internet of things devices. In: 2018 IEEE security and privacy workshops (SPW), IEEE, pp 29–35
109. Meidan Y, Bohadana M, Shabtai A, Ochoa M, Tippenhauer NO, Guarnizo JD, Elovici Y (2017) Detection of unauthorized iot devices using machine learning techniques, arXiv preprint arXiv:1709.04647
110. Chang Y, Li W, Yang Z (2017) Network intrusion detection based on random forest and support vector machine. In: 2017 IEEE international conference on computational science and engineering (CSE) and IEEE international conference on embedded and ubiquitous computing (EUC), IEEE, vol 1, pp 635–638
111. Feng F, Cho J, Pedrycz W, Fujita H, Herawan T (2016) Soft set based association rule mining. Knowl-Based Syst 111:268
112. Hussain J, Kalita P (2015) Understanding network intrusion detection system using OLAP on NSL-KDD dataset. IUP J Comput Sci 9(3):105
113. Elhag S, Fernández A, Bawakid A, Alshomrani S, Herrera F (2015) On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. Expert Syst Appl 42(1):193
114. Tajbakhsh A, Rahmati M, Mirzaei A (2009) Intrusion detection using fuzzy association rules. Appl Soft Comput 9(2):462
115. Fürnkranz J, Kliegr T (2015) A brief overview of rule learning. In: International symposium on rules and rule markup languages for the semantic web, Springer, pp 54–69
116. Zhou ZH (2015) Ensemble learning. In: Encyclopedia of biometrics, pp 411–416
117. Gomes HM, Barddal JP, Enembreck F, Bifet A (2017) A survey on ensemble learning for data stream classification. ACM Comput Surv 50(2):23
118. Ren Y, Zhang L, Suganthan PN (2016) Ensemble classification and regression-recent developments, applications and future directions. IEEE Comput Intell Mag 11(1):41
119. Witten IH, Frank E, Hall MA, Pal CJ (2016) Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann
120. Gaikwad D, Thool RC (2015) Intrusion detection system using bagging ensemble method of machine learning. In: 2015 international conference on computing communication control and automation, IEEE, pp 291–295
121. Aburomman AA, Reaz MBI (2016) A novel SVM-kNN-PSO ensemble method for intrusion detection system. Appl Soft Comput 38:360
122. Reddy RR, Ramadevi Y, Sunitha K (2017) Enhanced anomaly detection using ensemble support vector machine. In: 2017

international conference on big data analytics and computational intelligence (ICBDAC), IEEE, pp 107–111

123. Yerima SY, Sezer S, Muttik I (2015) High accuracy android malware detection using ensemble learning. IET Inf Secur 9(6):313
124. Bosman HH, Iacca G, Tejada A, Wörtche HJ, Liotta A (2015) Ensembles of incremental learners to detect anomalies in ad hoc sensor networks. Ad Hoc Netw 35:14
125. Awasthi P, Charikar M, Krishnaswamy R, Sinop AK (2015) The hardness of approximation of euclidean k-means, arXiv preprint arXiv:1502.03316
126. Arora P, Varshney S et al (2016) Analysis of k-means and k-medoids algorithm for big data. Procedia Comput Sci 78:507
127. Muniyandi AP, Rajeswari R, Rajaram R (2012) Network anomaly detection by cascading k-Means clustering and C4.5 decision tree algorithm. Procedia Eng. 30:174
128. Li Q, Zhang K, Cheffena M, Shen X (2017) Channel-based sybil detection in industrial wireless sensor networks: a multi-kernel approach. In: GLOBECOM 2017-2017 IEEE global communications conference, IEEE, pp 1–6
129. Wang HB, Yuan Z, Wang CD (2009) Intrusion detection for wireless sensor networks based on multi-agent and refined clustering. In: 2009 WRI international conference on communications and mobile computing, IEEE, vol 3, pp 450–454
130. Xie M, Huang M, Bai Y, Hu Z (2017) The anonymization protection algorithm based on fuzzy clustering for the ego of data in the internet of things. J Electr Comput Eng 10:83
131. Li H, Ota K, Dong M (2018) Learning IoT in edge: Deep learning for the Internet of Things with edge computing. IEEE Netw 32(1):96
132. Fadlullah ZM, Tang F, Mao B, Kato N, Akashi O, Inoue T, Mizutani K (2017) State-of-the-art deep learning: evolving machine intelligence toward tomorrow's intelligent network traffic control systems. IEEE Commun Surv Tutor 19(4):2432
133. Sze V, Chen YH, Yang TJ, Emer JS (2017) Efficient processing of deep neural networks: a tutorial and survey. Proc IEEE 105(12):2295
134. Scherer D, Müller A, Behnke S (2010) Evaluation of pooling operations in convolutional architectures for object recognition. In: International conference on artificial neural networks, Springer, pp 92–101
135. Ramachandran P, Zoph B, Le QV (2017) Searching for activation functions, arXiv preprint arXiv:1710.05941
136. De Coninck E, Verbelen T, Vankeirsbilck B, Bohez S, Simoens P, Demeester P, Dhoedt B (2015) Distributed neural networks for Internet of Things: the Big-Little approach. In: International Internet of Things Summit, Springer, pp 484–492
137. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
138. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, IEEE, pp 248–255
139. Zhang L, Zhang L, Du B (2016) Deep learning for remote sensing data: a technical tutorial on the state of the art. IEEE Geosci Remote Sens Mag 4(2):22
140. McLaughlin N, Martinez del Rincon J, Kang B, Yerima S, Miller P, Sezer S, Safaei Y, Trickel E, Zhao Z, Doupé A, et al. (2017) Deep android malware detection. In: Proceedings of the seventh ACM on conference on data and application security and privacy, ACM, pp 301–308
141. Maghrebi H, Portigliatti T, Prouff E (2016) Breaking cryptographic implementations using deep learning techniques. In: International conference on security, privacy, and applied cryptography engineering, Springer, pp 3–26
142. Zhang Y, Yi C (2011) Zhang neural networks and neural-dynamic method. Nova Science Publishers, Inc.,
143. Voigtlaender P, Doetsch P, Ney H (2016) Handwriting recognition with large multidimensional long short-term memory recurrent neural networks. In: 2016 15th international conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 228–233
144. Tolosana R, Vera-Rodriguez R, Fierrez J, Ortega-Garcia J (2018) Exploring recurrent neural networks for on-line handwritten signature biometrics. IEEE Access 6:5128
145. Lai S, Xu L, Liu K, Zhao J (2015) Recurrent convolutional neural networks for text classification. In: Twenty-ninth AAAI conference on artificial intelligence
146. Liu P, Qiu X, Huang X (2016) Recurrent neural network for text classification with multi-task learning, arXiv preprint arXiv:1605.05101
147. O'Shea TJ, Clancy TC, McGwier RW (2016) Recurrent neural radio anomaly detection, arXiv preprint arXiv:1611.00301
148. Lopez-Martin M, Carro B, Sanchez-Esguevillas A, Lloret J (2017) Network traffic classifier with convolutional and recurrent neural networks for Internet of Things. IEEE Access 5:18042
149. Al-Jumeily D, Hussain A, Fergus P (2015) Using adaptive neural networks to provide self-healing autonomic software. Int J Space-Based Situated Comput 5(3):129
150. Nweke HF, Teh YW, Al-Garadi MA, Alo UR (2018) Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges. Expert Syst Appl 105:233
151. Pascanu R, Mikolov T, Bengio Y (2013) On the difficulty of training recurrent neural networks. In: International conference on machine learning, pp 1310–1318
152. HaddadPajouh H, Dehghantanha A, Khayami R, Choo KKR (2018) A deep recurrent neural network based approach for Internet of Things malware threat hunting. Fut Gener Comput Syst 85:88
153. Brun O, Yin Y, Gelenbe E (2018) Deep learning with dense random neural network for detecting attacks against iot-connected home environments. Procedia Comput Sci 134:458
154. Diro AA, Chilamkurti N (2018) Distributed attack detection scheme using deep learning approach for Internet of Things. Fut Gener Comput Syst 82:761
155. Torres P, Catania C, Garcia S, Garino CG (2016) An analysis of recurrent neural networks for botnet detection behavior. In: 2016 IEEE biennial congress of Argentina (ARGENCON), IEEE, pp 1–6
156. Atiga J, Mbarki NE, Ejbali R, Zaied M (2018) Faulty node detection in wireless sensor networks using a recurrent neural network. In: Tenth international conference on machine vision (ICMV 2017), vol 10696, p 106962
157. Chen J, Xie B, Zhang H, Zhai J (2019) Deep autoencoders in pattern recognition: a survey. In: Bio-inspired computing models and algorithms, p 229
158. Mohammadi M, Al-Fuqaha A, Sorour S, Guizani M (2018) Deep learning for IoT big data and streaming analytics: a survey. IEEE Commun Surv Tutor 20(4):2923
159. Du B, Xiong W, Wu J, Zhang L, Zhang L, Tao D (2016) Stacked convolutional denoising auto-encoders for feature representation. IEEE Trans Cybern 47(4):1017
160. Alhajri R, Zagrouba R, Al-Haidari F (2019) Survey for anomaly detection of IoT botnets using machine learning auto-encoders. Int J Appl Eng Res 14(10):2417
161. Yousefi-Azar M, Varadharajan V, Hamey L, Tupakula U (2017) Autoencoder-based feature learning for cyber security applications, In: 2017 International joint conference on neural networks (IJCNN), IEEE, pp 3854–3861

162. Li Y, Ma R, Jiao R (2015) A hybrid malicious code detection method based on deep learning. Int J Secur Appl 9(5):205
163. Meidan Y, Bohadana M, Mathov Y, Mirsky Y, Shabtai A, Breitenbacher D, Elovici Y (2018) N-BaIoT-Network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Comput 17(3):12
164. Rezvy S, Luo Y, Petridis M, Lasebae A, Zebin T (2019) An efficient deep learning model for intrusion classification and prediction in 5G and IoT networks. In: 2019 53rd annual conference on information sciences and systems (CISS), IEEE, pp 1–6
165. Hinton GE (2012) A practical guide to training restricted Boltzmann machines. In: Neural networks: Tricks of the trade, Springer, pp 599–619
166. Fiore U, Palmieri F, Castiglione A, De Santis A (2013) Network anomaly detection with the restricted Boltzmann machine. Neurocomputing 122:13
167. Hinton GE (2009) Deep belief networks. Scholarpedia 4(5):5947
168. Zhang Q, Yang LT, Chen Z, Li P (2018) A survey on deep learning for big data. Inf Fusion 42:146
169. Chen Y, Zhang Y, Maharjan S (2017) Deep learning for secure mobile edge computing, arXiv preprint arXiv:1709.08025
170. Zhang Y, Li P, Wang X (2019) Intrusion detection for IoT based on improved genetic algorithm and deep belief network. IEEE Access 7:31711
171. Huda S, Yearwood J, Hassan MM, Almogren A (2018) Securing the operations in SCADA-IoT platform based industrial control system using ensemble of deep belief networks. Appl Soft Comput 71:66
172. Tama BA, Rhee KH (2017) Attack classification analysis of IoT network via deep learning approach. In: Research briefs on information and communication technology evolution (ReBICTE), vol 3, p 1
173. Roopak M, Tian GY, Chambers J (2019) Deep learning models for cyber security in IoT networks. In: 2019 IEEE 9th annual computing and communication workshop and conference (CCWC), IEEE, pp 0452–0457
174. Jahanian A, Chai L, Isola P (2019) On the"steerability" of generative adversarial networks, arXiv preprint arXiv:1907.07171
175. Hiromoto RE, Haney M, Vakanski A (2017) A secure architecture for IoT with supply chain risk management. In: 2017 9th IEEE international conference on intelligent data acquisition and advanced computing systems: technology and applications (IDAACS), IEEE, vol 1, pp 431–435
176. Ferdowsi A, Saad W (2019) Generative adversarial networks for distributed intrusion detection in the Internet of Things, arXiv preprint arXiv:1906.00567
177. Cui Q, Zhou Z, Fu Z, Meng R, Sun X, Wu QJ (2019) Image steganography based on foreground object generation by generative adversarial networks in mobile edge computing with Internet of Things. IEEE Access
178. Liran M, Yan H, Chunqiang H, Wei L (2019) Security and privacy for smart cyber-physical systems
179. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training gans. In: Advances in neural information processing systems, pp 2234–2242
180. Lakshminarayanan B, Pritzel A, Blundell C (2017) Simple and scalable predictive uncertainty estimation using deep ensembles. In: Advances in neural information processing systems, pp 6402–6413
181. Lee I, Kim D, Kang S, Lee S (2017) Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks. In: Proceedings of the IEEE international conference on computer vision, pp 1012–1020
182. Codella NC, Nguyen QB, Pankanti S, Gutman D, Helba B, Halpern A, Smith JR (2017) Deep learning ensembles for melanoma recognition in dermoscopy images. IBM J Res Dev 61(4/5):5
183. Shakeel PM, Baskar S, Dhulipala VS, Mishra S, Jaber MM (2018) Maintaining security and privacy in health care system using learning based deep-Q-networks. J Med Syst 42(10):186
184. Thakkar A, Lohiya R (2020) A review of the advancement in intrusion detection datasets. Proced Computer Sci 167:636–645
185. Taivalsaari A, Mikkonen T (2017) A roadmap to the programmable world: software challenges in the IoT era. IEEE Softw 34(1):72
186. Brugger ST, Chow J (2007) An assessment of the DARPA IDS Evaluation Dataset using Snort, UCDAVIS Department of Computer Science, 1, 22
187. McHugh J (2000) Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory. ACM Trans Inf Syst Secur 3(4):262
188. Sharafaldin I, Lashkari AH, Ghorbani AA (2018) Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSP, pp 108–116
189. Ashraf QM, Habaebi MH (2015) Autonomic schemes for threat mitigation in Internet of Things. J Netw Comput Appl 49:112
190. Sanchez L, Galache JA, Gutierrez V, Hernandez JM, Bernat J, Gluhak A, Garcia T (2011) Smartsantander: the meeting point between future internet research and experimentation and the smart cities. In: 2011 future network and mobile summit, IEEE, pp 1–8
191. Sanchez L, Muñoz L, Galache JA, Sotres P, Santana JR, Gutierrez V, Ramdhany R, Gluhak A, Krco S, Theodoridis E et al (2014) SmartSantander: IoT experimentation over a smart city testbed. Comput Netw 61:217
192. Lavrova D, Pechenkin A (2015) Applying correlation and regression analysis to detect security incidents in the internet of things. Int J Commun Netw Inf Secur 7(3):131
193. Arshad J, Abdellatif MM, Khan MM, Azad MA (2018) A novel framework for collaborative intrusion detection for M2M networks. In: 2018 9th international conference on information and communication systems (ICICS), IEEE, pp 12–17

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.