Machine Learning in IoT Security: Current Solutions and Future Challenges

Article *in* IEEE Communications Surveys & Tutorials · April 2020

DOI: 10.1109/COMST.2020.2986444

CITATIONS
87

READS
3,175

4 authors:

Fatima Hussain
Ryerson University
46 PUBLICATIONS **406** CITATIONS
SEE PROFILE

Rasheed Hussain
Innopolis University
139 PUBLICATIONS **1,767** CITATIONS
SEE PROFILE

Syed Ali Hassan
National University of Sciences & Technology
222 PUBLICATIONS **1,902** CITATIONS
SEE PROFILE

Ekram Hossain
University of Manitoba
549 PUBLICATIONS **26,213** CITATIONS
SEE PROFILE

Machine Learning in IoT Security: Current Solutions and Future Challenges

Fatima Hussain, Rasheed Hussain, Syed Ali Hassan, and Ekram Hossain

***Abstract*—The future Internet of Things (IoT) will have a deep economical, commercial and social impact on our lives. The participating nodes in IoT networks are usually resource-constrained, which makes them luring targets for cyber attacks. In this regard, extensive efforts have been made to address the security and privacy issues in IoT networks primarily through traditional cryptographic approaches. However, the unique characteristics of IoT nodes render the existing solutions insufficient to encompass the entire security spectrum of the IoT networks. Machine Learning (ML) and Deep Learning (DL) techniques, which are able to provide embedded intelligence in the IoT devices and networks, can be leveraged to cope with different security problems. In this paper, we systematically review the security requirements, attack vectors, and the current security solutions for the IoT networks. We then shed light on the gaps in these security solutions that call for ML and DL approaches. We also discuss in detail the existing ML and DL solutions for addressing different security problems in IoT networks. We also discuss several future research directions for ML- and DL-based IoT security.**

***Index Terms*—Internet of Things (IoT), IoT Applications, Security, Attacks, Privacy, Machine Learning, Deep Learning**

I. Introduction

IoT is considered as an interconnected and distributed network of embedded systems communicating through wired or wireless communication technologies [1]. It is also defined as the network of physical objects or *things* empowered with limited computation, storage, and communication capabilities as well as embedded with electronics (such as sensors and actuators), software, and network connectivity that enables these objects to collect, sometimes process, and ex6uchange data. The *things* in IoT refer to the objects from our daily life ranging from smart house-hold devices such as smart bulb, smart adapter, smart meter, smart refrigerator, smart oven, AC, temperature sensor, smoke detector, IP camera, to more sophisticated devices such as Radio Frequency IDentification (RFID) devices, heartbeat detectors, accelerometers, sensors in parking lot, and a range of other sensors in automobiles etc. [2]. There are a plethora of applications and services offered by the IoT ranging from critical infrastructure to agriculture, military, home appliances, and personal health-care [3].

The domains covered by the IoT services include, but not limited to, energy, building management, medical, retail, transportation, manufacturing, and so on. The huge scale of IoT networks brings new challenges such as management of these devices, sheer amount of data, storage, communication, computation, and security and privacy. There have been extensive researches covering these different aspects of IoT (e.g. architecture, communication, protocols, applications, security and privacy) [3]–[17]. However, the cornerstone of the commercialization of IoT technology is the security and privacy guarantee as well as consumer satisfaction [7], [12], [18]. The fact that IoT uses enabling technologies such as Software-Defined Networking (SDN), Cloud Computing (CC), and fog computing, also increases the landscape of threats for the attackers [19].

IoT devices generate a sheer amount of data and therefore, traditional data collection, storage, and processing techniques may not work at this scale. Furthermore, the sheer amount of data can also be used for patterns, behaviors, predictions, and assessment. Additionally, the heterogeneity of the data generated by IoT creates another front for the current data processing mechanisms. Therefore, to harness the value of the IoT-generated data, new mechanisms are needed. In this context, Machine Learning (ML) is considered to be one of the most suitable computational paradigms to provide embedded intelligence in the IoT devices [20].

ML can help machines and smart devices to infer useful knowledge from the device- or human-generated data. It can also be defined as the ability of a smart device to vary or automate the situation or behavior based on knowledge which is considered as an essential part for an IoT solution. ML techniques have been used in tasks such as classification, regression and density estimation. Variety of applications such as computer vision, fraud detection, bio-informatics, malware detection, authentication, and speech recognition use ML algorithms and techniques. In a similar manner, ML can be leveraged in IoT for providing intelligent services. In this paper, however, we focus on the applications of ML in providing security and privacy services to the IoT networks. In the following, we discuss the existing surveys that are already published in the literature covering different aspects of security in IoT networks through ML and DL techniques.

A. Existing Surveys

IoT has a rich literature and to date, many surveys have been published that cover different aspects of the IoT security. In

F. Hussain is with API Operation and Delivery Squad, Royal Bank of Canada, Toronto, Canada (email: fatima.hussain@rbc.com).

R. Hussain is with Networks and Blockchain Laboratory, Innopolis University, Innopolis, Russia (email: r.hussain@innopolis.ru).

S. A. Hassan is with the School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Pakistan (email: ali.hassan@seecs.edu.pk).

E. Hossain is with Department of Electrical and Computer Engineering at University of Manitoba, Winnipeg, Canada (email: Ekram.Hossain@umanitoba.ca).

TABLE I
EXISTING SURVEYS ON SECURITY, ARCHITECTURE, AND SERVICES IN IOT NETWORKS.

Year	Paper	Topic(s) of the survey	Related sections in our paper	Enhancements in our paper
2017	[21]	IoT authentication and access control	Sec. III	Detailed security and privacy solutions through ML and DL in IoT
2015	[4]	Communication security protocols	Sec. III	Security requirements, threats, vulnerabilities, and ML- and DL-based solutions
2017	[5]	Security in the edge layer of IoT	Sec. III	Coverage of entire IoT from security, privacy standpoint
2018	[7]	Security of IoT framework architectures	Sec. III	In-depth coverage of security and privacy in generic IoT with focus on state-of-the-art ML and DL techniques
2018	[8]	IoT-enabled attacks on different sectors and assess different attacks in critical infrastructure	Sec. III	In-depth coverage of security issues, threats, attacks, and solutions in generic IoT
2018	[9]	Security threats in IoT	Sec. III	Enhanced threats, attacks, and solutions in IoT
2019	[11]	Data security in IoT and data lifecycle	N/A	Covering in-depth security issues and their ML- and DL-based solutions in IoT
2018	[12]	Blockchain and SDN solutions for IoT	N/A	Generic IoT and in-depth review of the ML- and DL-based solutions in multiple domains
2018	[22]	Resource scheduling techniques in IoT	N/A	Detailed investigation of security in IoT and state-of-the-art techniques based on ML and DL
2017	[13]	Threats and vulnerabilities in IoT applications, architecture and possible attacks	Sec. III	Enhanced threat landscape, requirements, attacks, and their respective solutions in generic IoT security spectrum
2017	[14]	IDS in IoT, detection methods, placement and validation strategies	Sec. III	Detailed coverage of security and privacy issues and state-of-the-art based on ML and DL
2019	[23]	Security of IoT applications in different domains	Sec. III	Coverage of generic IoT applications with solutions, independent of particular domains
2019	[15]	Current development in IoT security, challenges, simulators, and tools	Sec. III and V	In-depth and more detailed survey of the security requirements, threats, attacks, and solutions in IoT
2016	[24]	Secure routing protocols in IoT	N/A	Focus on the applications security and privacy
2015	[16]	Security protocols and key distribution	Sec. III	Focus on security in a holistic way
2018	[17]	Security challenges in IoT and sensor networks	Sec. III	Detailed security challenges, attacks, and solutions in IoT
2017	[25]	Trust models for service management in IoT	N/A	Coverage of different aspects of security with solutions based on ML and DL
2017	[26]	Communication standards in IoT	N/A	ML- and DL-based generic security solutions in IoT
2017	[6]	System architecture, and security, privacy in edge-/fog-based IoT	Sec. III	Enhanced coverage of state-of-the-art ML- and DL-based security and privacy in generic IoT
2016	[27]	Network types, technologies and their characteristics in IoT	Sec. II-A	Focus on state-of-the-art solutions in IoT and coverage of resource management, security, and privacy
2017	[28]	Health-care communications standards in IoT	N/A	Focus on generic IoT and in-depth coverage of the security solutions
2018	[29]	Architecture, scheduling, network technologies, and power management of IoT operating systems	N/A	Focus on the current solutions for generic IoT applications and future research directions
2018	[30]	Open issues and challenges in IoT and enabling technologies	Sec. VI	Enhanced state-of-the-art and research challenges in ML-driven IoT security
2018	[31]	IoT data analytics through DL	N/A	In-depth review of ML- and DL-based security solutions in IoT
2019	[32]	data fusion in IoT applications through ML	N/A	In-depth coverage of ML and DL in different aspects security in IoT

this section, we summarize the existing surveys and compare them with our work. To the best of our knowledge, most of the surveys in the literature do not focus on the ML techniques used in IoT. Furthermore, the existing surveys are either application-specific or do not encompass the full spectrum of the security and privacy in the IoT networks. We have organized the existing surveys in two tables. Table I summarizes the existing surveys in the literature that cover different aspects of the IoT such as security, architecture, and services. We outline the topics covered in these surveys and the respective enhancements in our survey. In Table II, we cover the surveys that discuss the role of ML and DL in security and IoT networks. Although, there is a rich literature available on ML- and DL-based techniques in IoT networks, but we focus only on the security aspects of IoT networks and the role of ML and DL in addressing security challenges in IoT networks. It is important to note that the reason of dividing the existing surveys into two tables is to increase the readability and highlight the comprehensiveness of the existing surveys on ML- and DL-driven IoT networks.

The current literature covers the security in IoT by investigating the existing traditional solutions and the solutions provided through the new emerging technologies. However, surveys covering ML- and DL-based solutions are scarce. One of the most relevant papers to our survey is [33]. In fact, our paper is complementary to [33] but we take a different approach to discuss the IoT security from the ML standpoint. Specifically, the mentioned paper focuses on ML and DL methods to discuss their applicability in security at different layers, whereas we take the existing security problems in

different functional domains and discuss the ML and DL solutions for the security issues such as authentication, authorization, DDoS, malware, and so forth, regardless of the layers. In our paper, we shed light on the existing security mechanisms and establish a precedent for the need of ML techniques and the gaps in the existing security solutions for IoT networks. We believe that these are two different perspectives, and while the existing paper can be more suited for ML/DL enthusiasts, our work targets the security enthusiasts. Moreover, in our paper, we have followed the natural and intuitive flow by pointing out the security issues, discussing the existing solutions, and then identifying the gaps that lead us to ML- and DL-based solutions in IoT networks. Also, our paper is more up-to-date in terms of surveying the most recent references. Moreover, we have also discussed emerging ML/DL techniques such as Adversarial Machine Learning (AML), Generative Adversarial Networks (GANs), Federated learning, and transfer learning, that increase the soundness and depth/breadth of this survey.

In nutshell, although ML and DL have been covered in few surveys but the overall information on the comprehensive usage of ML and DL is scarce. To fill the gaps, we conduct a comprehensive survey of the ML and DL techniques used in IoT security.

B. Scope of This Survey and Contributions

The pictorial illustration of the scope and taxonomy of this survey is shown in Fig. 1. As shown in the figure, first we discuss the motivation of using ML in IoT security in the presence of the existing traditional security approaches. Then we discuss the security requirements of IoT applications, threats and attacks in IoT networks. After that we discuss the role of ML and DL in IoT and discuss different ML and DL techniques that are actively leveraged for IoT applications and services. To focus more on the functional side of the IoT, we dive deeper into the ML- and DL-based security solutions in IoT. To this end, we also discuss the existing research challenges and future directions for more research on the ML and DL for IoT networks. Our aim is to bridge the gap between the requirements of the IoT security and the capabilities of the ML and DL which will help addressing the current security challenges of the IoT networks.

The main contributions of this paper can be summarized as follows:

1) We present an in-depth systematic and comprehensive survey of the role of machine and deep learning mechanisms in IoT.
2) We describe state-of-the-art results on ML and DL in IoT network with a focus on security and privacy of the IoT networks.
3) We describe the limitations of the existing security solutions of the IoT networks that lead to using ML and DL techniques.
4) We also present an in-depth review of different research challenges related to the application of ML and DL techniques in IoT that need to be addressed.

In essence, we investigate in detail, the security requirements, attack surface in IoT, and then discuss the ML and DL-based solutions to mitigate the security attacks in IoT networks. It is worth mentioning that we have covered the existing surveys till 2019. Furthermore, our survey contains the recent works carried out in the fields of ML and DL for addressing security issues in IoT networks.

The rest of the paper is organized as follows: Table III lists all the acronyms used in the paper. In Section II, we discuss the motivation for using ML in IoT security. We discuss threat model of IoT networks in Section III. In Section IV, we discuss the role of ML in the IoT networks and briefly review different ML and DL techniques. We survey the existing ML- and DL-based solutions for IoT networks in Section V. The future research directions are discussed in Section VI and Section VII concludes this paper.

II. Motivation of Using ML in IoT Security

In this section, we discuss the motivation of using ML in IoT security in the presence of existing security solutions currently used in IoT networks. We first put light on the unique characteristics of IoT networks that are pertinent to security and then touch upon the security challenges that hinder IoT deployment as well as discuss the gaps in the existing security solutions. After that, we establish the motivation for using ML to address security challenges in IoT networks.

A. Characteristics of IoT Networks

In the following, we discuss some unique characteristics of IoT networks.

Heterogeneity: In an IoT network, a multitude of different devices with different capabilities, characteristics and different communication protocols communicate with each other. More precisely, the devices could use different standards for communication, and different communication paradigms (such as cellular or Ethernet) and variable constraints on the hardware resources. Such heterogeneity on one hand enables cross-platform communication among different devices, but on the other hand introduce new challenges to the IoT network.

Massive deployment: It is speculated that the billions of devices connected with each other and through Internet will likely surpass the capabilities of the current Internet. The deployment of IoT on massive scale also brings challenges. Some of these challenges include design of networking and storage architecture for smart devices, efficient data communication protocols, proactive identification and protection of IoT from malicious attacks, standardization of technologies, and devices and application interfaces [34], [35] etc.

Inter-connectivity: IoT devices are expected to be connected to global information and communication infrastructure and can be accessed from anywhere and anytime. The connectivity depends on the type of service and application provided by the IoT service provider(s). In some cases, the connectivity could be local (such as in case of connected car technology or swarm of sensors) whereas in other cases it could be global such as in case of smart home access through mobile infrastructure and critical infrastructure management.

Communication in close proximity: Another salient feature

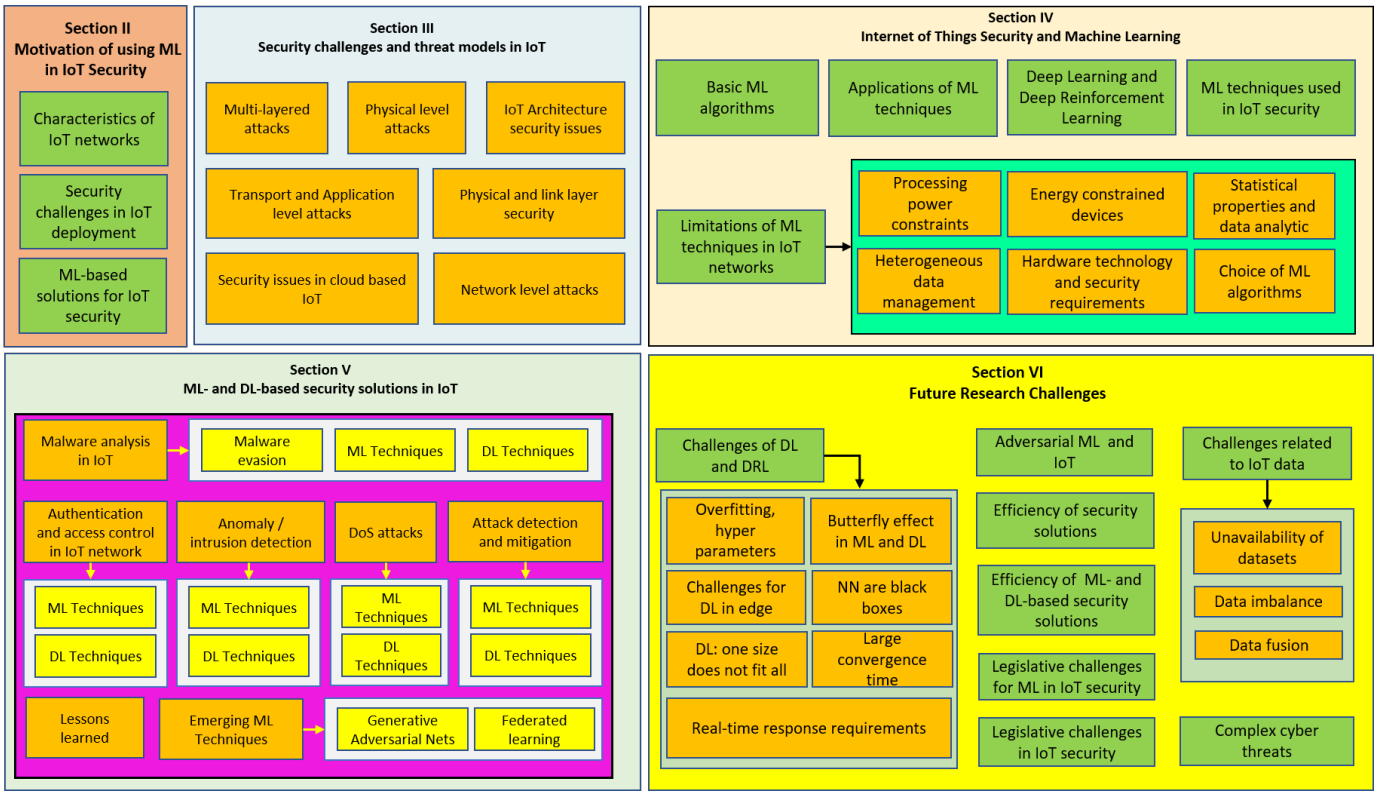

Fig. 1. Taxonomy of the survey.

of IoT is the communication in close proximity without involving the central authorities such as base stations. Device-to-Device communication (D2D) leverages the characteristics of point-to-point communication such as Dedicated Short Range Communication (DSRC) and similar technologies. The architecture of traditional Internet is more inclined towards network-centric communication whereas decoupling of networks and services enables device-centric as well as content-centric communication which enriches the IoT service spectrum.

Ultra-Reliable and Low Latency Communication (URLLC): This property of IoT networks is required in critical real-time applications such as industrial process automation, remote surgery, and intelligent traffic transport system, where the major performance constraints are both delay and reliability.

Low-power and low-cost communication: Massive connectivity of IoT devices requires ultra low-power and low cost solutions for efficient network operations.

Self-organization and self-healing characteristics: These are required for urgent and contemporary IoT communication that includes emergency or disaster situations. In such situations, reliance on the network infrastructure is not an option and therefore, self-organizing networks should be deployed.

Dynamic changes in the network: IoT consists of massive number of devices that need to be managed in an efficient way. These devices will act dynamically, for instance the sleep/wakeup time of devices will depend on the application, when do these devices use Internet and when do they communicate directly, and so on.

Safety: Safety is considered for both the consumers and devices because the large number of IoT devices connected to Internet may jeopardize the personal data that is shared through these devices. Furthermore, privacy and the security of device itself is also an important factor.

Intelligence: One of the most intriguing characteristics of IoT is the intelligence through which timely and informed decisions are made based on the processed data.

B. Security Challenges in IoT Deployment

Security and privacy are two of the main factors in the commercial realization of the IoT services and applications. Current Internet is the luring playground for security attacks ranging from simple hacks all the way to corporate level well-coordinated security breaches that have adversely affected different industries such as health-care and business. The limitations of the IoT devices and the environment they operate in, pose additional challenges for the security of both applications and the devices. To date, security and privacy issues have been extensively researched in the IoT domain from different perspectives such as communication security, data security, privacy, architectural security, identity management, malware analysis, and so on [4]. Detailed discussion on security challenges and threat model follows in Section III.

Fernandes et al. [53] focused on similarities and differences of the security issues in IoT and the traditional IT devices. Furthermore, they also focused on the privacy issues. The

TABLE II
EXISTING SURVEYS ON USING MACHINE AND DEEP LEARNING IN SECURITY AND IOT NETWORKS.

Year	Paper	Topic(s) of the survey	Related sections in our paper	Enhancements in our paper
2018	[33]	Machine and Deep Learning in IoT security	Sec. V	[33] focuses on ML/DL aspects and is suited for ML/DL enthusiasts whereas our paper is from security perspective whereas we identify security problems in IoT and discuss ML/DL solutions to address these problems
2019	[10]	ML-based techniques for IDS in IoT	Sec. V	Coverage of security and privacy and ML-based techniques in IoT
2016	[36]	ML-based data mining for intrusion detection through cyber analytics	N/A	Encompassing general security in IoT through ML techniques
2016	[37]	ML-based malware analysis for detecting the command and control channel	Sec. V-E	Focus on malware analysis in IoT
2018	[38]	Security vulnerabilities and threats in ML algorithms, and their corresponding countermeasures	N/A	Attack detection and mitigation through ML in IoT
2009	[39]	ML-based malicious code detection through static features extraction	Sec. V-E	Focus on general security solutions based on ML in IoT
2016	[40]	ML techniques in IDS, malware, and security policy management with a focus on general security domains	N/A	Coverage of security solutions in IoT through ML and DL techniques
2017	[41]	ML techniques for malicious URL detection in Internet	N/A	Focusing on IoT security challenges and solutions
2016	[42]	Intrusion detection in MANET through ML techniques	N/A	Focusing on the security issues and ML solutions in generic IoT networks
2018	[43]	ML-based network intrusion detection solutions	Sec. V-D	Covering IDS as well as other security challenges and ML-based solutions in IoT networks
2019	[44]	Analysis of ML techniques for intrusion detection	Sec. V-D	IDS as well as other security challenges and ML-based solutions in IoT networks
2017	[45]	ML techniques and their limitations for stealth malware detection	Sec. V-E	Holistic approach towards solving security problems through ML techniques in IoT
2014	[46]	Finding insider attackers in different setups through ML techniques	N/A	Generic attack detection and mitigation in IoT through ML algorithms
2019	[47]	Botnet detection in IoT through DL algorithms	Sec. V-C	Coverage of generic security solutions through ML and DL in IoT including Botnet
2018	[48]	Adversarial attacks on the DL algorithms for computer vision and defense mechanisms against such attacks	N/A	ML-based security solutions in generic IoT networks
2018	[9]	Security issues in IoT and solutions through ML and SDN	N/A	Focus on generic IoT security and ML-based solutions
2019	[49]	Review of ML algorithms security in adversarial settings, attacks, and defense mechanisms	N/A	Review of ML-based security solutions in IoT networks
2019	[50]	ML-based techniques for malware analysis	Sec. V-E	ML-based security solutions including malware analysis in IoT networks
2019	[51]	DL-bsed IDS, malware, phishing, and web defacement mechanisms	Sec. V-E and V-D	Coverage of more IoT security solutions through ML
2018	[52]	Review of random forest-based IDS solutions	Sec. V	Review of security solutions including IDS based on ML algorithms

main driving factors to argue on the similarities and differences include software, hardware, network, and applications.

Based on these classifications, there are fundamental similarities between the security issues in traditional IT domain and the IoT. However, the primary concern of the IoT is the resource-constraints that hinder the adoption of already available sophisticated security solutions in IoT networks. Furthermore, solutions to the security and privacy issues in IoT require cross-layer design and optimized algorithms. For instance due to computational constraints, IoT devices may need new breeds of optimized cryptographic and other algorithms to cope with security and privacy. A holistic security and privacy approach towards IoT will have nominations from the existing security solutions as well as development of new intelligent, robust, evolutionary, and scalable mechanisms to address security challenges in IoT.

C. Machine Learning: A Solution to IoT Security Challenges

Machine learning (ML) refers to intelligent methods used to optimize performance criteria using example data or past experience(s) through learning. More precisely, ML algorithms build models of behaviors using mathematical techniques on huge data sets. ML also enables the ability to learn without being explicitly programmed. These models are used as a basis for making future predictions based on the newly input data. ML is interdisciplinary in nature and inherits its roots from many disciplines of science and engineering that include artificial intelligence, optimization theory, information theory, and cognitive science [54].

Machine learning is utilized when human expertise either do not exist or cannot be used such as navigating a hostile place where humans are unable to use their expertise, for instance robotics, speech recognition etc. It is also applied in situations where solution to some specific problem changes in time

(routing in a computer network or finding malicious code in a software or application). Furthermore, it is used in practical smart systems, for instance Google uses ML to analyze threats against mobile endpoints and applications running on Android. It is also used for identifying and removing malware from infected handsets. Likewise, Amazon has launched a service **Macie** that uses ML to sort and classify data stored in its cloud storage service. Although ML techniques perform well in many areas; however, there is a chance of false positives and true negatives. Therefore, ML techniques need guidance and modification to the model if inaccurate prediction is made. On the contrary, in Deep Learning (DL), a new breed of ML, the model can determine the accuracy of prediction by itself. Due to self-service nature of DL models, it is rendered as more suitable for classification and prediction tasks in innovative IoT applications with contextual and personalized assistance.

Although traditional approaches are widely used for different aspects of IoT (e.g. applications, services, architectures, protocols, data aggregation, resource allocation, clustering, analytics) including security, the massive scale deployment of IoT however, advocates for intelligent, robust, and reliable techniques. To this end, ML and DL are promising techniques for IoT networks due to several reasons, e.g. IoT networks produce a sheer amount of data which is required by ML and DL approaches to bring intelligence to the systems. Furthermore, the data generated by the IoT is better utilized with the ML and DL techniques which enable the IoT systems to make informed and intelligent decisions. ML and DL are largely used for security, privacy, attack detection, and malware analysis. DL techniques can also be used in IoT devices to perform complex sensing and recognition tasks to enable the realization of new applications and services considering real-time interactions among humans, smart devices and physical surroundings. Some of the security related real-world applications of ML are as follows:

- Face recognition for forensics: pose, lighting, occlusion (glasses, beard), make-up, hair style, etc.
- Character recognition for security encryption: different handwriting styles.
- Malicious code identification: identifying malicious code in applications and software.
- Distributed Denial of Service (DDoS) detection: detecting DDoS attacks on infrastructure through behavior analysis.

Using ML and DL techniques in IoT applications on the other hand bring multi-faceted challenges. For instance, it is challenging to develop a suitable model to process data from diverse IoT applications. Similarly, labeling input data effectively is also a cumbersome task. Another challenge is using minimum labelled data in the learning process. Other challenges stem from the deployment of these models on resource-constrained IoT devices where it is essential to reduce the processing and storage overhead [55]. Similarly, critical infrastructure and real-time applications cannot withstand the anomalies created because of ML or DL algorithms. In the above context, it is imperative to systematically review the security solutions of IoT that leverage ML and DL techniques.

III. Security Challenges and Threat Models in IoT

IoT uses different communication technologies such as, but not limited to, IPv6, Zigbee, 6LoWPAN, Bluetooth, Z-Wave, WiFi, and Near Field Communications (NFC), to name a few [56]. Technologies such as information-centric networking (ICN) and software-defined networking (SDN) have been utilized to serve as underlying communication infrastructures for IoT [57], [58]. These aforementioned communications technologies have their own shortcomings and limitations from security standpoint and these limitations are inherited in the IoT domain as well. The pervasive deployment of large number of devices increases the attack surface in an IoT system. Since the IoT devices are (usually) resource-constrained, it is not feasible to use sophisticated security mechanisms against notorious attacks. Here, we provide the summary of the threats and attacks faced by IoT. Without loss of generality, security attacks in IoT can be abstractly divided into physical, network, transport, application, and encryption attacks.

A. Physical Attacks

In physical attacks, the attackers have direct access to the devices and manipulate different aspects of the devices. To get access to the physical devices, social engineering is one of the most prominent methods where the attackers access the devices and perform real attack that ranges from physical damage to the device to eavesdropping, side-channels, and other related attacks [59], [60]. Despite the fact that different technologies are used at the physical layer for IoT, the nature of physical attacks mostly resemble and need social engineering-like approaches. Furthermore, to launch physical attacks, the attackers must be in the close proximity of the devices/hardware with different intentions such as physically destroying the hardware, limiting its lifetime, endangering the communication mechanism, tampering with the energy source, and so on.

It is also worth noting that physical attacks maybe stepping stone for other attacks, for instance disabling an alarm in a home could lead to a burglary or other related damage in smart home environment. Similarly, a replacement of sensor with a malicious sensor would lead to sensitive data leakage. Injection of malicious node into the network can also cause man-in-the-middle attack that enables that attacker to escalate privileges and launch other attacks. Furthermore, such tampering with devices may also enable the attackers to make changes in the routing tables and security keys that will affect the communication with upper layers [61]–[63]. Other physical attacks include jamming radio frequencies which denies the communication in IoT environment. Among many other repercussions, jamming causes denial of service in IoT thereby adversely affecting the functionality of IoT applications [64], [65]. As has been mentioned, the attackers also use different social engineering approaches to have physical access to hardware/devices for different purposes such as the attacks we already mentioned. Through social engineering, the

attackers may manipulate users to gain physical access to the devices [66], [67].

Social engineering attacks are related to the physical perimeters of the networks and are hard to mitigate; however, better awareness and strict access control mechanisms help in alleviating such attacks. Other physical attacks are aimed at draining the energy of resource-constrained devices in IoT through different attack vectors. For instance, sensors are programmed to have predefined sleep mode where they conserve their energy when there is no data to send. Attackers could manipulate the configuration of these nodes to keep them awake all the times so that to drain the battery. This attack is referred to as sleep deprivation attack [59].

B. Physical (PHY) and Link Layer Security Issues

IoT combines various communication technologies at the lower layers of TCP/IP protocol stack and thus-forth provides a complex heterogeneous network. These technologies include, but not limited to, ZigBee, WSN, MANET, WiFi, RFID, NFC, and so forth and furthermore these technologies have their own security issues. In this subsection, we will shed light on the security issues in the physical and data link layers of IoT. Yang et al. [21] and Granjal et al. [4] discussed in detail, the security issues in IoT at different layers and their existing solutions. As has been mentioned before, the heterogeneity is introduced at physical layer of the IoT and then different amendments are made at data link layer, for instance special channel design and so forth, depending on the underlying physical layer technology. To this end, the security mechanisms of IoT must encompass the heterogeneity at the physical and data link layer.

Furthermore, detection of malfunction in the hardware is also of paramount importance and must be handled to avoid anomalies at upper layer. Intrusion is another physical security issue that needs efficient countermeasures from both detection and prevention standpoint. To this end, several mechanisms have been developed in the literature to efficiently detect faulty nodes and intrusion in IoT and separate them from the network. Zarpelao et al. [14] conducted a thorough survey of intrusion detection systems in IoT with current trends, architectures, implementations, and future challenges. It is also very important to place the intrusion detection system (IDS) at the right place to increase the probability of intrusion detection and at the same time increase the efficiency and decrease false positives. Raza et al. [68] proposed a real-time intrusion detection mechanism for IoT in the network layer. They target IPv6 connected devices in IoT through 6LoWPAN for which there are not any known solutions to detect and prevent intrusion in real time.

It is worth noting that there could be many attack vectors even for intrusion detection at upper layers, for instance routing attack [68]. From fault detection standpoint, it is important at par to detect the faulty nodes in IoT because it directly affects the quality of service (QoS) of the IoT application [69], [70]. Jadav et al. proposed a mechanism to detect faulty nodes in wireless sensor networks by using fuzzy logic. In principle, they employed fuzzy inference system (FIS) to determine different fault conditions of the sensors such as transmitter circuit condition, receiver circuit condition and battery conditions. In another work, Nishiguchi et al. [70] proposed a fault management platform for IoT.

As a prominent IoT technology standard, the IEEE 802.15.4 incorporates security only at the data link layer. The fact that in IoT, higher layers use low power protocols such as 6LoWPAN and Constrained Application Protocol (CoAP), a mechanism is needed at lower layers to enable these protocols work seamlessly. In this context, IEEE 802.15.4 provides the necessary amendments at the lower layers for these protocols. The security provided by the IEEE 802.15.4 at the MAC layer does not only make sure that the node level data transmission is secure, but also complements the security of upper layers. In this regard, symmetric cryptography algorithms such as AES are proved to be fast and efficient when implemented on chip, therefore such implementation in IEEE 802.15.4 hardware will complement the lower layer security [71], [72]. IEEE 802.15.4 standard implements AES algorithm and different implementations have been proposed to deal with the resource constraints of the devices [73]. The standard supports different security modes at the link layer, for instance the data may not be encrypted with only integrity check, or the data may be encrypted along with the integrity check. The selection of a particular mode is solely dependent on the security guarantees provided by the application and/or service provider. At the link layer, different security requirements such as confidentiality, data authenticity and integrity, semantic security and security against different attacks such as replay attacks, and access control are supported by the IEEE 802.15.4. Confidentiality is provided through different modes of AES algorithms, data authenticity and integrity are also guaranteed by using the cipher block chaining (CBC) mode of AES with mandatory message integrity code. Similarly, other modes of AES algorithm such as counter mode and feedback modes are also employed for this purpose. Details about different modes of AES and their applicability in the IEEE 802.15.4 hardware can be found in [4].

C. Network Layer Attacks

At the network level, the attacks are aimed at routing, data and traffic analysis, spoofing, and launching man-in-the-middle attack. Besides, sybil attacks are also possible at the network layer where fake identities/sybil identities are used to create illusions in the network [74], [75]. Routing is one of the main functions provided at the network layer and requires up-do-date routing information in the routing tables. If the attackers are able to either spoof, or alter any routing information in IoT, it will not only jeopardize the normal functionality of the application, but may also cause sensitive data leakage. Furthermore, change in forwarding nodes also compromises quality of service in IoT applications. Similarly spoofing and cloning at network layer (for instance RFID cloning) at network layer enables the attacker to transmit its own data instead of legitimate data. This phenomenon also leads to unauthorized access access, for instance in RFID, the lack of authentication process could lead to the leakage of

important sensitive data where attacker can read and/or even write data from/to the RFID nodes, respectively. Furthermore, at this stage, the attackers usually target network intrusion using small-scale affordable attacks as afore-mentioned. On the other hand, intrusion through different means, provides a way-in for the attacker to the system where the attackers can launch plethora of other attacks and therefore, securing the network is essential to contain the attacks at early stages. At the network layer, the attacker can also leverage a compromised node to use it as fake forwarding node and create a sinkohole. This type of attack, usually associated with sensor networks and mobile ad hoc networks, is equally dangerous in IoT environment [76].

With these attacks, the possibility of launching a collaborative DDoS attacks increases, and thereby disrupting the whole IoT network. At the network layer, the attacker can achieve this by bombarding the network with more traffic through compromised nodes than the network can handle. Compromising IoT nodes and masquerading identities will have catastrophic consequences on the network because with such fake nodes (either non-existent fake identities or existing compromised identities) enable the attackers to launch sybil attacks where sybil nodes (fake nodes) give the illusion to the core network as if real nodes were sending data. This phenomenon can spray fake data in the network as well as forward false data to applications. If there is a decision-support system where it relies on the incoming data (for instance voting-based mechanism for route selection or any other service) would easily got compromised through sybil nodes [75]. To summarize, the attack vectors in network layer target the communication aspects of the IoT and exploit the resource constraints and lack of sophisticated authentication and authorization schemes.

D. Transport and Application Layer Attacks

Transport layer is responsible for process to process delivery where transport protocols enable the processes to exchange data. In the context of IoT, the traditional transport layer security issues still persist. The most serious attack at this layer is the DoS attack that chokes the network and results in denial of services to the applications. It is worth mentioning that due to the nature of IoT, traditional TCP and UDP protocols do not scale with resource-constrained devices, and therefore lightweight versions of transport protocols have been proposed in the literature [77]–[79]. However, the security of these protocols is of primary importance to alleviate the DoS and DDoS attacks in IoT.

IoT applications are relatively lucrative targets for the attackers because applications level attacks are relatively easy to launch. Some of the well-known attacks include, but not limited to, buffer overflow attacks, malware attacks, denial of services, phishing, exploiting the WebApp vulnerabilities, cryptographic attacks, side channel attacks, and man-in-the-middle attacks. Buffer overflows are one of the mostly used attack vectors in different applications [80]. Existing techniques to mitigate buffer overflow mechanism include static and dynamic code analysis, and other sophisticated mechanisms such as symbolic debugging; however, these techniques cannot be used with IoT due to resource constraints. IoT applications are also prone to malicious code injection as a result of buffer overflow and other vulnerabilities such as SQL injection, cross-site scripting, object referencing, and so forth. Open Web Application Security Project (OWASP) identified the top 10 vulnerabilities that causes different attacks on applications. The latest list of most widely found vulnerabilities was compiled by OWASP in 2017[1]. These vulnerabilities lead to range of other attacks that could be launched, for instance, injection of malicious code, phishing, access control, privilege escalation, and so on. Furthermore, through malicious code injection attacks, the attacker can steal sensitive information, tampering with information, and range of other malicious activities [81]. Furthermore, botnets are other serious threats to IoT infrastructure and the applications [82].

Mitigation of attacks launched through intelligent botnets is a serious challenge for IoT because such botnets intelligently scan and crawl through the network looking for known vulnerabilities and exploiting them to launch different attacks such as massive DDoS. It is also worth mentioning that due to resource constraints, sophisticated cryptographic protocols are not feasible for IoT devices which leaves them at the mercy of capable attackers to launch cryptographic attacks. The attackers may use different techniques for such attacks, such as side-channel attacks and sniffing the network traffic, having known ciphertext, and then carrying out cryptanalytic attacks. The use of vulnerable or outdated cryptographic protocols also enables attackers to break the encryption schemes. Similar to previous layers, man-in-the-middle attack can be one of the feasible choice for attackers in the presence of the aforementioned vulnerabilities to launch different attacks. In a nutshell, the attack surface at application layer of the IoT infrastructure is wide and relatively computationally expensive to mitigate.

E. Multi-Layered Attacks

In addition to the aforementioned attacks, there are multi-layered attacks that could be launched on IoT infrastructure. These attacks include traffic analysis, side-channel attacks, replay attacks, man-in-the-middle attacks, and protocol attacks. Some of these attacks have already been discussed. Traffic analysis attacks are passive attacks where the attackers passively listen to the traffic and try to make sense out of it. These attacks are very hard to mitigate because the communicating parties usually have no idea that their traffic is being monitored. The attackers look for interesting information in the internet traffic such as users' personal information, business logic details, credentials, and other information that is of any value to the attacker. Besides, data transfer security is also of paramount importance in IoT. The data produced in the IoT environment is used for decision-making purpose; therefore, it is essential to guarantee the quality of the data.

SDN has been leveraged to achieve a range of benefits for IoT applications and IoT security [83]. The rich functionality provided by SDN control plane enables organizations to

[1]https://www.owasp.org/index.php/Top_10-2017_Top_10

TABLE III
ACRONYMS AND THEIR EXPLANATIONS

Acronym	Explanation	Acronym	Explanation
ML	Machine Learning	RFID	Radio Frequency IDentification
UURLLC	Ultra-Reliable and Low Latency Communication	DL	Deep Learning
RL	Reinforce Learning	SVM	Support Vector Machine
SVR	Support Vector Regression	KNN	K-Nearest Neighbour
NB	Naive Bayes	NN	Neural Network
DNN	Deep Neural Network	CNN	Convolutional Neural Network
PCA	Principal Component Analysis	RNN	Recurrent Neural Network
MLP	Multi-Layer Perception	ELM	Extreme Learning Machine
ESFCM	ELM-based Semisupervised Fuzzy C-Means	ANN	Artificial Neural Network
LSTM	Long-Short Term Memory	DRL	Deep Reinforcement Learning
QoS	Quality of Service	CSI	Channel State Information
SDN	Software-Defined Network	D2D	Device-to-Device Communication
NFC	Near-Field Communication	ICN	Information-Centric Networking
DDoS	Distributed Denial of Service	OWASP	Open Web Application Security Project
RBAC	Role-Based Access Control	CWAC	Context-Aware Access Control
PBAC	Policy-Based Access Control	ABAC	Attribute-Based Access Control
UCAC	Usage Control-based Access Control	CAC	Capability-based Access Control
OAC	Organizational-based Access Control	XACML	eXtensible Access Control Markup Language
OAuth	Open Authentication	UMA	User-Managed Access
CoAP	Constrained Application Protocol	FCM	Fuzzy C-Means
MCA	Multivariate Correlation Analysis	SINR	Signal-to-Interference Noise Ratio
CPS	Cyber Physical System	IRG	Influential Relative Grade
LDA	Linear Discriminant Analysis	RaNN	Random Neural Networks
IMA	Illegal Memory Access	XSS	Cross-Site Scripting
DEL	Deep Eigenspace Learning	WMS	Wireless Multimedia System
BAN	Body Area Network	AML	Adversarial Machine Learning

efficiently control millions of sensors and things in the IoT paradigm. However, despite all the aforementioned advantages of SDN, the open interfaces in SDN open pandora of new attacks on already vulnerable IoT devices, infrastructure, and applications [84], [85]. Therefore, the security of IoT is directly dependent on the security of SDN as well.

F. Security Issues in Cloud-Based IoT

Another very important enabler for IoT is the cloud computing which is leveraged for processing massive data generated by the IoT subsystems. Security consideration in IoT is of prime importance from cloud tenants perspective as well as from service providers perspective [86]. It is worth mentioning that cloud can be an essential part of the IoT infrastructure due to several-fold reasons: handling big data, storing and processing huge amount of data from IoT, and producing end-results to the respective applications in IoT environment. Additionally, cloud platform also provides a range of services such as device management, resource management, data processing, analysis, and management. The sheer scale of IoT devices poses a serious challenge for achieving the required security and privacy goals for cloud-based IoT. Furthermore, the IoT applications are designed from a single domain in mind and thus-forth do not encompass the whole range of other domains that may use the data originated from one particular IoT domain. Therefore, the number of devices added to IoT networks and the data produced by these devices, and then stored, processed, and analyzed by the cloud need viable, efficient, and scalable security and privacy considerations. To this end, Singh et al. [86] outlined 20 security considerations for cloud-based IoT. Some of the most important security considerations include access control and rights, data management security, secure data sharing, and identity management.

G. Security Issues Across IoT Architectures

IoT uses a diversified range of different communication technologies with different protocols, standards, and security requirements. In order to focus on the security issues of IoT, the underlying applications and architectures [5] need to be considered because they provide the context for necessary countermeasures.

Granjal et al. [4] reviewed and analyzed the existing protocols and solutions that address the security and privacy issues in IoT. The authors covered different communication protocols and routing in IoT and identified security challenges. Furthermore, it is important to consider the underlying reference architectures of IoT while addressing security issues and attacks on IoT. Mosenia et al. [5] discussed security attacks on IoT from the standpoint of three different reference architectures. First architecture consists of three layers consisting of sensor networks, servers, and applications. The second architecture employes edge computing and consists of edge nodes, intermediate nodes, and application whereas the third architecture employes edge layer, cloud layer, and user layer. These layers further consist of functional and design elements. The authors discussed the security issues and attacks in details and outlined the solutions to these attacks.

IV. IOT SECURITY AND MACHINE LEARNING

In this section, we discuss various machine learning algorithms and their applicability in IoT applications.

A. Basic Machine Learning Algorithms

The ML algorithms can be classified into four categories; supervised, unsupervised, semi-supervised, and reinforcement learning algorithms (Fig. 2).

Supervised Learning: Supervised learning is performed when specific targets are defined to reach from certain set of inputs. For this type of learning, the data is first labeled followed by training with labeled data (having inputs and desired outputs). It tries to identify automatically rules from available datasets and define various classes, and finally predict the belonging of elements (objects, individuals, and criteria) to a given class.

Unsupervised Learning: In unsupervised learning, the environment only provides inputs without desired targets. It does not require labeled data and can investigate similarity among unlabeled data and classify the data into different groups.

Semi-supervised Learning: In the previous two types, either there are no labels for all the observation in the dataset or labels are present for all the observations. Semi-supervised learning falls in between these two. In many practical situations, the cost to label is quite high, since it requires skilled human experts to do that. So, in the absence of labels in the majority of the observations but present in few, semi-supervised algorithms are the best candidates for the model building.

Reinforcement Learning: In Reinforcement Learning (RL), no specific outcomes are defined, and the agent learns from feedback after interacting with the environment. It performs some actions and makes decisions on the basis of the reward obtained. It is greatly inspired by learning behaviors of humans and animals. Such behaviors make it an attractive approach in highly dynamic applications of robotics in which the system learns to accomplish certain tasks without explicit programming [87]–[91]. It is also very important to choose the suitable reward function because the success and failure of the agent depends on the accumulated total reward [92]. RL techniques are generally used in the following scenario:

- When historical data and prior examples are not known for model training.
- Exact right and wrong values for the given scenario is not known a priory.
- Overall goal is known and environment can be sensed to maximize both immediate and long term rewards.

B. Applications of Basic ML Techniques

Supervised and unsupervised learning techniques mainly focus on data analysis problems while RL is preferred for comparison and decision-making problems. This categorization and the choice of ML technique depends on the nature of available data. When the type of input data and the desired outputs (labels) are known, supervised learning is used. In this situation, the system is only trained to map inputs to the desired outputs. Classification and regression are the examples of supervised learning techniques where regression works with continuous and classification works with discrete outputs. Various regression techniques such as Support Vector Regression (SVR), linear regression, and polynomial regression are commonly used techniques [93]–[95].

On the other hand, classification works with discrete output values (class labels). Common examples of classification algorithms include K-nearest neighbor, logistic regression, and Support Vector Machine (SVM). Some algorithms such as neural networks can be used for both classification and regression. When outputs are not well-defined and the system has to discover the structure within the raw data, unsupervised learning methods are used to train the system. Unsupervised learning includes clustering which groups the objects based on established similarity criteria such as K-means clustering. The degree of precision of the predictive analytics depends on how well the respective ML technique has used past data to develop models and how well is it able to predict the future values. Algorithms such as SVM, neural networks, and Naive Bayes are used for predictive modeling.

The fundamental limitation of basic ML techniques is that it mostly needs rich dataset for training a model. Afterwards, this trained model is applied to the real application data and prediction or classification is done. However, it is worth mentioning that the entire process may not encompass the full range of features and properties of the data. In this regard, DL techniques have been employed to address the limitations of the ML techniques. DL is capable of handling large amounts of data and DL algorithms are scalable with increasing amount of data which can be beneficial to the model training and may improve prediction accuracy. DL can automatically and hierarchically extract high-level features and corresponding correlations from the input data. IoT applications mostly generate unlabeled or semi-labeled data. DL has the capability to exploit the unlabeled data to learn useful patterns in an unsupervised manner. On the other hand, the conventional ML algorithms work effectively only when sufficient labeled data is available [96].

C. Deep Learning (DL) and Deep Reinforcement Learning (DRL)

Deep Learning: DL is a machine learning technique originated from ANN. The neural network is comprised of neurons (considered as variables) connected through weighted connections (considered as parameters). To achieve the desired set of outputs, supervised or unsupervised learning technique is associated with the network. The learning is carried out by using labeled and unlabeled data from supervised or unsupervised learning techniques, respectively followed by the iterative adjustment of the weights among each pair of neurons. Therefore, while discussing about DL, we refer to large deep neural networks where the term deep refers to the number of layers in that network [97], [98].

DL is known for distributed computing and, learning and analysis of sheer amount of unlabeled, un-categorized, and unsupervised data. It develops a hierarchical model of learning and feature representation motivated by layered learning process in the human brain [99]. DL models contribute to various ML applications such as speech recognition, computer vision and NLP by providing improved classification modeling

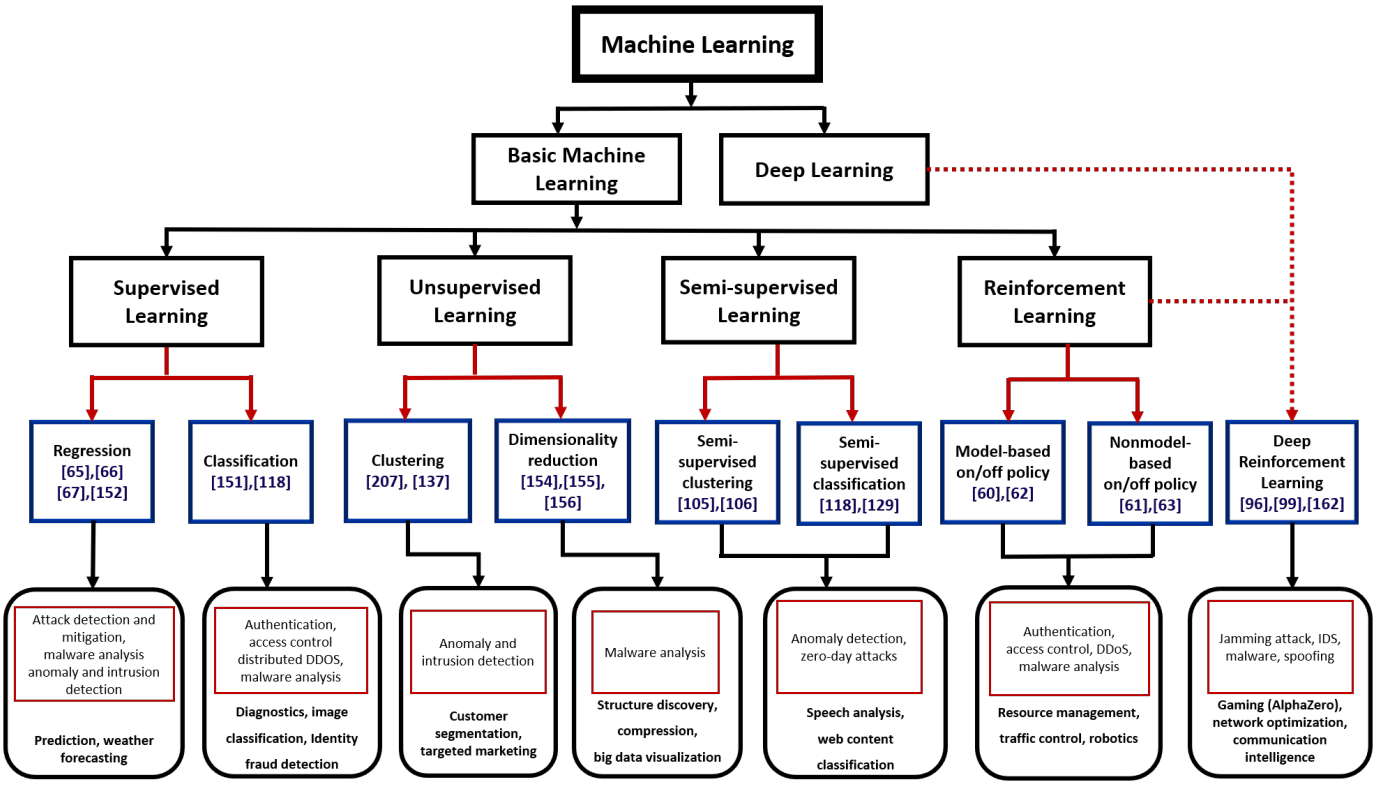

Fig. 2. Machine learning classes and typical applications.

and generating better data samples. Furthermore, these models also benefit the data compression and recovery, both in time and spatial domains because of its effectiveness in extracting patterns and features from large amounts of data and extracting relationships within time-dependent data. DL also find itself useful in Edge computing and for scheduling of IoT devices and mobile crowd sensing in Fog computing environment [100], [101]. The different DL architectures available in literature include, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Boltzmann Machine (BM), Long-short Term Memory (LSTM) Networks, Generative Adversarial Networks (GAN), Feed forward Deep Networks (FDN) and Deep Belief Networks (DBN). CNN (used in spatially distributed data) and RNN (used in time series data) are the most widely used deep learning architectures.

Deep Reinforcement Learning: DL is one type of ML techniques used for function approximation, classification, and prediction whereas RL is another type of ML techniques used for decision making in which a software agent learns about optimal actions by interacting with an environment over various states. DL and RL come into play together in situation when the number of states and data dimensionality are very large and the environment is non-stationary. Therefore, traditional RL is not efficient enough. By combining DL and RL, agents can learn by themselves and come up with a good policy to obtain maximum long-term rewards. In this approach, RL obtains help from DL to find the best policy and DL performs action values approximation in order to find the quality of an action in a given state.

DL is capable of learning from complex patterns but is prone to mis-classification. In this situation, RL has a powerful capability to automatically learn from environment without any feature crafting and helps DL in efficient classification [102]–[104]. DRL integrates RL's decision making and DL's perception. This combination has been used in game playing program "AlphaGo" developed by Google [105]. It can help solving tasks with high dimensional and raw data with some policy controls [106]. Most recent work in DRL can be found in [107], [108].

DQN is one example of such integration in which Q-learning is combined with a deep NN (DNN). The DQN agent learns policies using RL when high-dimensional data is presented at the inputs, whereas a deep CNN is used for approximation of the action and value function. Formally, the integration of both is know as DRL which is essentially a combination of RL and DL. DL extracts features from training data and in RL (inspired by psychology), an agent performs series of actions by interacting with an environment and tries to maximize the cumulative reward [102].

D. Machine Learning Techniques Used in IoT Security

In the following, we discuss various ML algorithms focusing on the underlying security and privacy problems in IoT networks, as shown in Table IV. More precisely, we consider authentication, attack detection and mitigation, Distributed Denial of Service (DDoS) attacks, anomaly and intrusion detection, and malware analysis.

Supervised learning algorithms work with labelled data and are utilized in IoT networks for spectrum sensing, channel

estimation, adaptive filtering, security, and localization problems. This category holds two distinct types of techniques: classification and regression. Classification under supervised machine learning is used for predication as well as modeling of the available data sets. Regression is used for predicting continuous numeric variables. SVM, Naive Bayes, Random Forest, Decision Tree are few of the widely used classification algorithms. While, Nearest neighbors and logistic regression are to famous regression algorithms. These algorithms are also known as "instance-based", that make predictions for each new observation by searching for the most similar training data.

The family of unsupervised learning algorithms deals with unlabeled data and utilize input data in a heuristic manner. These are used in anomaly, fault, and intrusion detection, cell clustering, and load balancing. Clustering under unsupervised learning category is used for data groupings based on some inherent similarities and dissimilarities. The clustering is unsupervised, and therefore, there are no right or wrong answers. To evaluate the accuracy of the results, data visualization is used. If there is a possible right or wrong answer, then the clusters can be pre-labeled in datasets and in this scenario, classification algorithms are preferred.

RL techniques learn by exploiting various stages and develop the reward and action relationship between agent and the environment. This relationship of action reward is very useful in solving various IoT problems [109]. It does not require extensive training data set; however, the agent is required to have the knowledge of the state transition function.

Table V lists different security challenges along with utilized ML and DL techniques. In this table, we also provide a comparison of different ML and DL techniques to solve particular security issues in IoT networks.

Most of the IoT applications befit the use of unsupervised learning approaches with very less initial information about the environment. For instance, the zero-day attacks on IoT networks have little or no information to start with. Therefore, the unsupervised learning class of ML can be a promising learning technique in the IoT networks to combat such security attacks.

Supervised ML algorithm such as SVM, DT and Naive Bayes are also used in IoT security. For instance, SVMs are able to model non-linear decision boundaries; however, it becomes difficult to use with large datasets. Therefore, random forests are usually preferred over SVM. Random forest algorithms are easier to implement and are also adaptive to the size of the available dataset. It achieves a higher degree of accuracy and takes less time for prediction. However, it takes longer time to train as compared SVM and NB. NB is suitable for problems such as text classification and spam detection. Logistic regression and Nearest neighbour algorithms are memory-intensive and perform poorly for high-dimensional data. Considering the unsupervised ML algorithms, K-means and hierarchical clustering are two popular clustering algorithms. K-means clustering is most popular because it is a simple and flexible algorithm that forms clusters based on geometric distances between data points. Clusters are grouped around centroids resulting into globular with the same size. However, the number of clusters has to be specified before clustering starts and it is not always possible and efficient to do. Also, if clusters are not globular, it results into poor cluster formation.

RL techniques are computationally simple but require significant time to converge to a steady state. This slow convergence and knowledge of the state transition function or optimal policy are the key challenges in using RL algorithms in dynamic environments of IoT networks. The optimal policy is determined by trial and error and is obtained after many transitions.

DL also relies on strong function approximation, estimation and the learning capabilities thus providing more efficient solutions in various problem areas of IoT domain including security and privacy. Due to their resource constraints, IoT devices may not be able to host or run complex computational algorithms for any type of task such as communication, analytic and prediction. Therefore, DL-based algorithms show better performance with lower latency and complexity compared to conventional theories and techniques [99]. Additionally, DNNs are good in locating and defining low dimensional representations from any type (text, image, audio) of high dimensional data patterns. DRL and its variants are used for authentication and DDoS detection in heterogeneous IoT networks. The major DRL algorithms used for security and privacy are; deep deterministic policy gradient, continuous DQN, prioritized experience replay, asynchronous N-step Q-learning, deep SARSA, and dueling network DQN.

E. Complexity Analysis of Machine Learning Algorithms

Machine learning is considered as a combination of art and science and there is no single solution or approach that befits all kinds of applications in different environments and contexts. There are various factors associated with selection of specific ML algorithm for specific scenarios. Some problems are very specific and require a unique approach, for instance, a recommender system is a common type of ML algorithm but it is used to solve a very specific type of problem such as, product recommendations (finding rating or preference of any item or product for online shopping experience). While some other problems are open and not specific (such as anomaly detection, fraud detection, etc.) and therefore, require a *trial and error* approach. In this situation, supervised learning, classification and regression etc., are used. Another very important criterion that affects the choice of the ML algorithm is its complexity.

In essence, ML and DL are considered as powerful tools to address many problems face by IoT networks. However, ML- and DL-based solutions can introduce complexity to the already complex IoT system. This complexity builds up over time and evolves into large technical and computational cost. The complexity of ML and DL solutions is caused by the additional steps of feature engineering, model tuning, continuous training, and/or model deployment. Therefore, here discuss the complexity associated with ML and DL algorithms. The complexity of a model depends on a couple of factors, for instance, the utilization of more features to learn and predict (e.g., using two features versus 10 features to predict a target).

It will eventually increase the complexity if it relies on more complex feature engineering such as using polynomial terms, principal components etc. Another factor is the increased computational overhead, for instance a single decision tree has less computational cost versus a random forest of 100 trees. This leads to a question that how can we define complexity? The complexity of learning is measured by information and computation complexity. Information complexity is related to generalization performance of learning algorithm, such as, how many training examples are needed? how fast learning take place?, etc. While computational complexity is related to the computation resources utilized for training the model and to extract data from model's predictions. Generally speaking, if a model training on n points take x amount of time, how much will be the training time for kn points? This time can be linear or quadratic. It encompasses space as well as computational complexity. It is harder to evaluate the complexity of a ML and DL algorithms because it can be characterized by various factors including number of features in a model, as well as whether the chosen model is linear or nonlinear. It may refer to algorithmic learning complexity, model training time, computational complexity, and implementation complexity. Furthermore training time often depends on some input parameters that are passed to the algorithm, and sometimes it also depends on the properties of data. It might be the case that one ML/DL algorithm is too complex and rely on other algorithms. It is also worth mentioning that the complexity of ML and DL algorithms is not static and depends on situations where these algorithms are used. For a supervised learning model, the "complexity" refers to the complexity of training the model. In essence, such complexity mainly depends on the size of the training data set. Since in many cases, the training can be done offline, the "complexity" therefore, is not a limiting factor for the algorithm. The space complexity on the other hand, depends on the size of the training and testing data set. The training and testing data set are different for different solutions and even change in different circumstances for the same type of algorithm. Furthermore, the implementation of model also plays a pivotal role in the complexity of the algorithm. If the DL or DRL model is implemented in hardware, the complexity will depend on the number of hidden layers, number of neutrons per layer, etc. For DRL or Deep Q-Learning (DQL), the learning is a continuous process and the computational complexity depends on the specific algorithm used (for example, the most common algorithm used for DQL is "Experience Replay" algorithm and the complexity is a function of the "batch size" used for training.

We usually refer to computational complexity when we discuss the efficiency of any ML or DL algorithm; however, various factors are involved in crafting the true complexity of the algorithm. Following are the few reasons that are considered to develop any ML or DL model and in turns ML/DL algorithm complex:

- An ML system is considered to be too complex if there are many different ways to perform similar tasks. Complexity is increased two-folds and more time is consumed to figure out the correct method to perform training and prediction. Also, maintenance overhead is increased as model implementation is done in various different ways.
- When enough explanation and insight to a system or model is not provided, it is generally considered complex. Such system is considered as "black box", and is hard to interpret from the outside. The actual implementation might not be very complicated; however, undocumented and hard-to-understand functionality creates complexity.
- When the functionality of a system is non-reusable and cannot be used in different contexts, then the complexity increases. Furthermore, complex systems require many steps to perform simple tasks.

Complexity of an algorithm or a model is defined using the Big O notation, where O is the function and n is the input data size. We have added a separate column in Table IV that contains the computational complexity of various ML algorithms.

F. Limitations of Applying Machine Learning in IoT Networks

Most of the traditional ML techniques are not inherently efficient and scalable enough to manage IoT data and thus need considerable modifications [54]. In the following, we discuss some of the common limitations of using ML techniques in IoT networks.

1) Constraints on processing power and energy: IoT devices are small and typically have energy constraints with limited processing power. Therefore, direct application of conventional ML techniques is not suitable in such resource-constrained environments.

2) Analytics of heterogeneous data: Wireless data can be generated from different sources including networked information systems, and sensing and communication devices [110]. The data generated in IoT networks is diverse in nature with different types, formats and semantics, thus exhibiting syntactic and semantic heterogeneity. Syntactic heterogeneity refers to diversity in the data types, file formats, encoding schemes, and data models. Semantic heterogeneity refers to differences in the meanings and interpretations of the data. Such heterogeneity leads to problems in terms of efficient and unified generalization, specifically in case of big data and various datasets with different attributes. The data require pre-processing and cleaning before fitting to a specific model.

ML-based networks are developed assuming that the entire data set is available for processing during training phase. However, this may not be true for the IoT data. Also, the prediction ability of an algorithm decreases with the increase in the dimensionality of data [111].

Selecting an appropriate ML algorithm for a particular scenario is also a challenging task. IoT application can generate combination of structured (and relational), semi-structured or unstructured data. If we have labelled data, classification (supervised) algorithms can be used and if the data is unlabeled, clustering (unsupervised) algorithm can be used for grouping and aggregating of the available data. If we have hybrid data, a combination of both types will be required.

The preceding discussion is, at par, applicable for the security-related functions in the IoT where real-time data is processed for possible attack vectors such as intrusion.

3) Hardware technology and security requirements: IoT applications involve combination of versatile devices and processing units, ranging from high performance cloud servers to ultra-low-power edge devices. These heterogeneous devices demand advanced ML capabilities and stronger security features. Emerging IoT applications such as autonomous vehicles, wearable devices and drones require higher performance and security with minimum energy consumption. However, such high performance and energy-efficient multi-core microprocessors, special circuit and chip design for ML and DL neural networks is scarce, and also required to unleash the real potential of ML-enabled security. New paradigm shifts in terms of neuromorphic computing circuit design is required for efficient integration of special ML accelerators and hardware security processors. Furthermore, energy constraint nature of IoT devices calls for design and development of ultra-low-voltage logic and memory circuits and ultra-lightweight encryption engines.

V. Survey of the Existing Machine Learning- and Deep Learning-based Solutions for IoT Security

In this section, we survey the existing ML-based solutions addressing different security issues in IoT.

A. Authentication and Access Control in IoT

Authentication is one of the primary security requirements in IoT. The users must be authenticated in order to use IoT applications and/or services. Typically, IoT applications and services are based on data exchange across different platforms. The data retrieved from the IoT devices is pre-processed, processed, and then passed through a decision-support system to make sense out of it. These processes may vary depending upon the underlying IoT architecture; however, data flow may be identical in these systems. Without loss of generality, when an application and/or a user needs some data from an IoT device, the entity (user or application) must be authenticated to IoT network and it should be made sure that the requester has required access rights for the data. Otherwise, the request to access such data will be denied. Like other networks, access control is also of paramount importance in IoT networks and equally challenging due to, but not limited to, network heterogeneity, volume of network, resource constraints of the devices, network (in)security, and attacks vulnerabilities, to name a few. Furthermore, it is also important to grant and revoke the access of certain users to the critical data of IoT applications and services. Before, we discuss ML-based access control mechanisms in IoT, it is important to put light on different categories of access control.

Taylor et al. divided the access control mechanisms in IoT into three categories, Role Based Access Control (RBAC), Context Aware Access Control (CWAC) and Policy Based Access Control (PBAC) [112]. Whereas Ouaddah et al. extended this classification into more categories that include Attribute-based Access Control (ABAC), Usage Control-based Access Control (UCAC), Capability-based Access Control (CAC), and Organizational-based Access Control (OAC) [113]. Most of the existing work fall under one of these categories. Ouaddah et al. conducted a comprehensive survey on the currently used access control mechanisms in IoT and identified challenges faced by access control in IoT [113]. The authors discuss the existing mechanisms based on the underlying IoT applications. IoT applications can broadly be divided into two major classes, personal and enterprise. Personal applications include smart homes, health-care, smart office, body area networks, sensors networks, whereas enterprise applications include smart cities, smart industries, critical infrastructure, and so on. Access control mechanism can be used both at application layer and at the architecture layer. At the architecture layer, the service providers have variety of choices such as defining an access control markup language such as Extensible Access Control Markup Language (XACML) [114], [115], Open Authorization (OAuth) [116], and User-Managed Access (UMA) [117]. These architectural level access control mechanisms are already being used by many existing protocols of IoT, for instance OAuth is implemented over existing IoT protocols such as MQTT[2] and Constrained Application Protocol (CoAP)[3]. Furthermore, services providers such as Google, Facebook, and Microsoft, Instagram, Flikr, Netflix, and many more, have billions of user accounts that use OAuth. In the following, we discuss some of the proposed schemes based on the aforementioned access control classes.

Pereira et al. [118] proposed a CoAP-based service level access control mechanism in IoT applications driven by Service-Oriented Architecture (SOA). The existing security mechanisms in the TCP/IP protocol stack such as IPSec, TLS, DTLS, and SSL are essential for SOA; however, these protocols do not provide access control on data for SOA-enabled protocols such as CoAP which is largely used in IoT applications. The proposed scheme uses Kerberos and RADIUS mechanisms with CoAP to enable access control. Extending the Kerberos and RADIUS with CoAP provides fine-grained access control for the IoT services. In addition to the general access control in IoT, application-specific access control is also important. For instance, health-care applications, and other private data consuming applications exhibit stringent privacy requirements and thus-forth require efficient and effective access control. In this context, Yang et al. [119] proposed an adaptive access control mechanism for personal health records data stored in a cloud in encrypted form. This access control mechanism has multi-faceted features such as the ability of cross-platform data sharing, handling emergency scenarios, and definition of sharing policies based on access rights. Furthermore, this scheme also supports smart deduplication where redundant data is removed to save the storage space. In short, this scheme is based on cryptographic primitives such as secret sharing, bilinear pairing, ciphertext re-encryption, and so on. It is also worth noting that Ciphertext Policy Attribute-based Encryption (CP-ABE) is traditionally used for access control in different domains; however, CP-ABE cannot be used directly in IoT

[2]http://mqtt.org/
[3]http://coap.technology/

because of its native characteristics that force CP-ABE to send the related access policy along with the ciphertext. This characteristic of traditional CP-ABE renders it inappropriate for privacy-sensitive applications such as patients health-care records because the receivers of the ciphertext can know the access policies that may contain sensitive information. Therefore, such IoT applications need optimized CP-ABE where access policies do not reveal any secret information. In this spirit, Zhang et al. [120] proposed a new access control mechanism for healthcare application where large number of attributes and hidden policies make sure that CP-ABE preserves user privacy. Similarly, Yeh et al. [121] proposed a modified CP-ABE with other cryptographic ingredients such as dual encryption and Merkle Hash Trees (MHTs) are used to provide access control in the cloud-stored data environment. Furthermore, this scheme also allow the users to delegate access rights to different entities and revoke the rights are as well. Similarly, Huang et al. [122] propose an attribute-based cryptographic solution for access control through outsourcing the cryptographic functions to the edge computing platform. First the data of the owner is encrypted using attribute-based encryption with multiple policies and outsourced to the cloud/edge. The access control is implemented using the attributes and the policies. The aim of this work is to reduce the computation overhead at the user end and outsourced to the cloud.

In addition to the healthcare sector, access control in industrial sector is also very important. To date, many researchers have worked on providing access control in industrial IoT. For instance, Liu et al. [123] proposed a role-based resource access control mechanism for industrial IoT applications. The authors define the access control mechanism as route optimization problem and propose an efficient algorithm for it. Li et al. [124] proposed an access control mechanism on top of a new efficient certificateless signcryption method to provide access control for industrial IoT applications. This work is extension of the previous work [125]. The authors moved the computational complexity from sensor nodes to the gateway as part of providing efficiency to the applications. Blockchain has also been leveraged to guarantee access control in IoT. Novo et al. [126] proposed a fully distributed blockchain-based access control mechanism for IoT. To reduce the processing overhead incurred by the blockchain technique, the authors employ a single smart contract where the access rights are saved. During the transactions, the smart contract is triggered and the access right is either given or denied.

Next, we discuss the existing ML- and DL-based authentication and access control mechanisms in IoT networks.

1) ML-based authentication and access control in IoT: Xiao et al. [127] proposed a physical layer authentication mechanism for IoT. The proposed authentication mechanism uses physical channel properties such as signal strength. In essence, Xiao et al. used two revolutionary methods for physical layer authentication, game theoretic approach and machine learning technique, to isolate spoofers from benign IoT users. The authentication mechanism is formulated as a zero-sum game where the spoofing nodes try to increase their utility through maximizing the attack frequency, whereas the channel frequency responses are used to establish Nash Equilibrium (NE). The packets received through radio interface have certain channel state which could be leveraged to test for a particular threshold based on which authentication decision is taken. For this purpose the authors used reinforcement learning (Q-learning) and Dyna-Q which helps in understanding the channel state without having detailed information about the channel. The experimental results show that Q-learning helps reducing the misdetection of spooring by 61.72% and false positive by 93.33%. On the other hand, the misdetection rate with Q-learning is 7.9% and with Dyna-Q is 6.9%. The false positive rate of Dyna-Q is also less than 5%. Hence, the utility of Dyan-Q is 1.28% higher than that of Q-learning. Therefore, the authors concluded that the detection accuracy and performance of Dyna-Q is better than Q-learning technique. Another similar physical-layer authentication scheme based on distributed Frank-Wolf (dFW) algorithm is proposed in [128].

Besides, ANN has also been leveraged in addressing the authentication problem in IoT. Chatterjee et al. [129] proposed an ANN-based authentication mechanism in IoT networks. In essence, Physically Unclonable Function (PUF)-based authentication can be effective in IoT where the physical properties of the transmitter (communicating device) are analyzed (which are otherwise discarded as impurities of the communication). In the same spirit, Chatterjee et al. accumulated these non-idealities of the communication and applied in-situ ML algorithm to classify the transmitters. The ML algorithm is applied at the receiver end and entropy is extracted to successfully identify and/or classify the transmitters. The authors measured the inaccuracy in detection, or false detection of their scheme. With 4800 transmitters, the detection error was $< 10^{-3}$ and slightly increased (about 10^{-2}) with 10,000 transmitters. From efficiency standpoint, there is no overhead incurred by the transmitter whereas the receiver need two neural networks that will incur about 3%-5% additional power.

2) DL-based authentication and access control in IoT: Shi et al. [130] proposed a user authentication technique for IoT based on human physiological activities through WiFi signals. The proposed authentication scheme is based on both activity recognition and human identification. Activity recognition can be performed with coarse-grained data and a smaller number of features. For this purpose, the channel state information in WiFi signals generated by IoT devices is used to derive characteristics of different features of the activities. Shi et al. use 3 layer Deep Neural Network (DNN) to learn the human physiological and behavioral characteristics that are used in authentication. In the three layers, the DNN extracts the type of activity (whether walking or stationary), then at the second layer, details of the activity are learnt, and at the third layer, high-level features based on which, the user is authenticated. Through experiments the authors evaluated their scheme from identification accuracy and spoofer detection accuracy standpoints. The proposed scheme achieved an average 91.2% accuracy in user authentication and 89.7% accuracy in spoofer detection. Additionally, the authors also measured activity recognition accuracy which is around 98% on average. It is, however, worth noting that various features such as sampling rate affect the accuracy. Furthermore, the training size also

affect the accuracy of the DNN-based authentication. Similarly Das et al. [131] proposed an LSTM-based authentication mechanism in low-power IoT networks. LSTM is leveraged to learn about the hardware imperfections with different carrier frequencies of the hardware that affect the signal strength. The developed DL model learns these imperfections and identify the users based on these features. It is also important to evaluate such solution in the presence of adversaries. The authors experimented their solution with a test-bed consisting of low-power devices and radios and with one legitimate node with 29 adversaries. The authors reported that they found 2-layer LSTM to be the optimal network setting to achieve 99.58% accuracy of classification. Similarly in case of Non Line-of-Sight (NLoS), 3-layer LSTM performs best and gives the classification accuracy of 88.10%.

Similarly, Chauhan et al. [132] proposed an RNN-based authentication mechanism that uses acoustics and voice commands in smart home applications. The authors focused on a lightweight RNN solution for the acoustic-based authentication. Furthermore, the authors compared the SVM-based solution and LSTM (a varient of the RNN) for authentication in a smart home environment. The authors also reported that RNN outperforms the other ML and DL algorithms not only in terms of accuracy but also complexity and efficiency. The accuracy of the system was above 90% and the authors reported that LSTM slightly outperformed SVM which explains the fact that DL algorithms can peform well in authentication than the mainstream ML algorithms.

From the above discussion, we can see that LSTM outperforms the DNN and RNN methods in terms of accuracy in the discussed experimental settings. However, this comparison is only valid in the respective environment. Additionally, each DL method has its limitation and shortcoming as well which could affect the performance of the method such as training size in DNN, Line of Sight in LSTM, and so on.

B. Attack Detection and Mitigation

The attacks launched on IoT system at different layers vary from a low-profile hacking into a device to a massive scale ransomware attacks such as WanaCry and even more sophisticated attacks such as Mirai and Dyn. The traditional attack detection and mitigation mechanisms are based on cryptographic primitives and sometimes suffer from lack in accuracy, and also cause false positives. Therefore, techniques including SVM, DL, autoencoders, K-NN, and unsupervised learning have been used.

1) ML-based attack detection and mitigation in IoT: Rathore et al. [133] proposed a semi-supervised learning-based attack detection mechanism for IoT. In essence, the proposed scheme is based on Extreme Learning Machine (ELM) algorithm and leverages Fuzzy C-Means (FCM) methods [134], collectively referred to as ESFCM. ESFCM is also implemented in fog infrastructure. One distinct feature of ESFCM is that it deals with labeled data, therefore it increases the detection rate of distributed attacks. However the detection accuracy of ESFCM is less than the previous two DL-based mechanisms but it outperforms the traditional ML algorithms for attack detection. Nevertheless, the semi-supervised learning mechanism harnesses the features of both supervised and unsupervised learning and makes it more efficient than its counterparts.

As has been mentioned before, IoT has many breeds ranging from personal networks (body area network) to more sophisticated critical industry infrastructure such as smart grid. Attack detection is important at par in these infrastructures. For instance, in smart grid, the measurements are critical and must be genuinely retrieved and not tampered with as a result of an attack. In this direction, Ozay et al. [135] studied in detail, different ML algorithms for attack detection in smart grid. The authors particularly investigated the role of supervised learning, semi-supervised learning, feature space fusion, and online learning algorithms for attack detection. The authors divided the networks into small and large networks and found out through experiments that in small networks, k-NN performs better whereas in large networks, SVM performs better in terms of the attack detection accuracy. To evaluate the performance of online learning methods for real-time attack detection, the authors focused on the computational complexity which tends to be lower than the batch learning algorithms. However, both families of the aforementioned algorithms performed equally.

2) DL-based attack detection and mitigation in IoT: Diro et al. [136] proposed a DL-based attack detection mechanism in IoT by leveraging fog ecosystem. In essence, the attack detection mechanism is implemented at the edge near the smart infrastructure. The distributed attack detection mechanism takes into account different parameters from the learning mechanism and decides upon the output of the learning architecture on a given data. The rationale for using fog infrastructure is the resource constraints and the application nature of IoT. In case of critical infrastructure the learning mechanism must be as near as possible to the data-generating nodes in order to take timely and informed decision in case of potential attacks. Similarly Abeshu et al. [137] proposed a distributed DL-based attack detection mechanism in IoT. They used fog computing architecture (which is one of the favored architecture to realize IoT applications) to implement DL techniques for attack detection. The proposed technique focused on fog-to-thing communication where the learning module is implement at the fog layer which is the optimal point for detection mechanism because it both reduces the latency for communication and utilizes the resources. The proposed 3 layer stacked autoencoder achieves 99.2% accuracy.

C. DoS and Distributed DoS (DDoS) Attacks

Vlajic et al. [138] referred to the IoT as the "Land of opportunities for DDoS attackers". The year of 2016 witnessed an unprecedented growth in attacks against IoT infrastructure with a massive scale. Mirai[4] was one of these attacks that almost brought down the Internet where household devices such as babycams, printers, and webcams etc. were used as

[4]https://www.csoonline.com/article/3258748/security/the-mirai-botnet-explained-how-teen-scammers-and-cctv-cameras-almost-\\brought-down-the-internet.html

bots to launch DDoS attacks on many organizations. Similarly, other Mirai-like bots have been reported in [139]. The Mirai is a family of malware that caused serious disruptions in Internet services due to its sophisticated spreading mechanism in the IoT network. A detailed surgery of Mirai malware can also be found in [140]. To date, noteworthy research results have been yielded through different mechanisms to mitigate DDoS attacks in IoT; however, different architectures make it really hard to devise a unified mechanism to combat DDoS attacks in different IoT platforms [141].

Traditional DDoS detection and mitigation mechanisms in IoT networks are applied at the gateways, routers, and the entry points for IoT networks with the help of both intrusion detection and prevention mechanisms. As mentioned before, MQTT and CoAP are two of the most widely used telemetry protocols in IoT. Pacheco et al. [142] evaluated the effect of DDoS attack on generic IoT network that uses CoAP. This work used only reflection attack [143] to assess the withstanding capabilities of IoT network. In addition to detection and mitigating DDoS attacks in IoT from inside, other enabling technologies such as fog and cloud computing have also been used to aid the DDoS detection mechanism in IoT. For instance, Alharbi et al. [144] used fog computing based approach to secure IoT communications and mitigate malicious attacks.

Among other flavors of IoT networks, industrial and critical infrastructure-aided IoT networks must exhibit strong resilience to DDoS attacks. Similarly, a fog and cloud computing-based DDoS mitigation framework is proposed in [145]. In this multi-level DDoS detection framework, traditional mechanisms are used at multiple layers of industrial IoT infrastructure. The lowest level uses edge computing with SDN gatweways, and then network traffic data is gathered at the fog computing level which consists of SDN controllers, and analyze the data for possible DDoS. Furthermore, honeypots are also utilized at this level. Finally at the cloud computing level, applications-generated data is analyzed at the cloud platform to detect any threat of potential DDoS attack.

There is no single solution for all IoT systems to detect and mitigate the DDoS attacks. Despite the recent advancements in mitigating such types of attacks, it is still essential to work on intelligent mechanisms that not only take into account the amount of traffic, but also the behavior of the attackers. In this context, machine learning is a suitable candidate to be leveraged for DDoS detection in IoT networks. The solution methods should consider the fact that, false positives can occur due to which genuine requests might be blocked.

1) ML-based techniques to address DoS and DDoS attacks in IoT: Doshi et al. carried out a detailed comparison of the existing ML mechanisms for DDoS detection in IoT networks [146]. They leveraged the distinctive features of the IoT traffic where IoT devices usually engage in a finite communication with end-points rather than back-end servers. Doshi et al. compared K-nearest neighbors, decision trees, neural networks, random forest, and SVM to detect DDoS in IoT. Due to computational overhead, the authors have to limit the number of features; however, they successfully detected DDoS attack with 99% accuracy. Similarly, Ye et al. [147] proposed an SVM-based DDoS detection mechanism in SDN environment which complements the IoT applications [148], [149]. This solution is applicable to SDN-driven IoT networks. The authors achieved the accuracy of 95.24% for DDoS detection with low amount of flow data which could be improved with more data. The authors focused on the traffic data collected by SDN controller. Another such SVM-based DDoS detection in SDN is proposed by Kolika et al. [150] with almost same accuracy as that of Ye et al.'s scheme [147]. Kolika et al. also compared other techniques such as Naive Bayes, Random Forest, Bagging, and Radial Basis Function (RBF). According to their experiments, SVM outperforms in terms of detection accuracy. Like [147], this scheme is also applicable to SDN-driven IoT networks.

Besides SVM, other ML techniques have also been used for DDoS detection. For instance, Tan et al. [151] leveraged Multivariate Correlation Analysis (MCA) for DDoS detection. MCA-based DDoS detection mechanism focuses on the server side and in the context of the IoT, this mechanism will detect DDoS attack as a result of data flow between the back-end servers for data gathering, processing, and decision-making. In essence, MCA techniques is based on behavioral analysis of the traffic where normal behavior is isolated from abnormal. Features are generated from network traffic and then MCA is applied on those features. MCA determines and extract the correlation between the features obtained from the first step. Ideally, the intrusions cause disruption in the correlation among features which could be an indication for possible intrusive activity. Finally, any known anomaly detection technique is used for decision making. The overall accuracy of the proposed MCA-based approach is above 99%.

A Signal-to-Interference-plus-Noise-Ratio (SINR)-based DoS attack detection mechanism for Cyber-Physical System (CPS) is proposed by Li et al. [152]. The authors formulated the DoS attack as a game between the sensor and attacker with multiple energy levels. Furthermore, in this work, the authors focus on the transmission power consumption for the sensor and the interference power consumption for the attacker. To establish a balanced equilibrium between the players of the game, the authors use Nash Q-learning algorithm.

2) DL-based techniques to address DoS and DDoS attacks in IoT: Hodo et al. [153] used supervised Artificial Neural Networks (ANNs) technique to thwart DDoS in IoT. The ANN technique used by the authors showed about 99.4% accuracy during the experimental evaluation in a limited dataset. Similarly, Kulkarni et al. [154] used Multi-Layer Perception (MLP) mechanism to detect DoS attack in sensor networks which is mainly used in IoT.

D. Anomaly/Intrusion Detection

Here we will briefly review the existing techniques to both detect and mitigate intrusion in IoT and then focus on the ML-based intrusion detection mechanisms in IoT. To date, a number of ML-based techniques have been used to detect anomalies and intrusions in the IoT networks and their different breeds [155], [156]. Traffic filtering is one of the most widely used mechanisms for intrusion detection where per packet analysis or batch analysis is performed to isolate

legitimate packets from the malicious ones. However, despite the effectiveness of traffic classification, a higher number of false positives produced as a result of such classification render this method to be less reliable. On the other hand, behavior-based models are also used to detect intrusion in the network. In the context of IoT, both traffic classification and behavior-based models are used. Meng et al. [157] discussed a trust-based approach for intrusion detection in IoT networks. Unlike the traditional mechanisms, the trust-based approach takes into account the level of confidence in a device in combination with the type and class of traffic. The authors proposed a combination of trust management and traffic classification to detect intrusion in the IoT networks.

However, it is important to note that traditional signature-based and behavior-based schemes fail to detect zero-day intrusions. Therefore, artificial intelligence and its breeds are employed in intrusion detection system (IDS). Li et al. [158] proposed an Artificial Intelligence- (AI) based mechanism for intrusion detection in SDN-driven IoT. This scheme is based on network traffic flow where the intrusion detection component of the network captures the flow and applies two algorithms for features extraction, i.e. Bat algorithm (with swarm division) and differential mutation. Afterwards random forest technique is used to classify the traffic flow to identify potential intrusion to the system. The aim is to improve the detection accuracy and reduce the false positives. Intrusion may occur in the IoT network through different types of attacks, the likes of which can be found in the normal networks such as spoofing, masquerading, distributed denial of service, hijacking control, and so forth [159].

The security mechanisms, access control, and protection techniques are not homogeneous across different IoT platforms. In this context, the intrusion detection techniques need special attention depending on the underlying technologies. To address the issue of intrusion detection in different technologies, Gendreau et al. [160] conducted a survey of intrusion detection techniques in different networking paradigms. Similarly, in [161], [162], the authors focused on IDS in wireless sensors networks. As discussed earlier, a number of intrusion detection techniques are proposed in the literature to detect and/or mitigate network intrusion in IoT. Colom et al. [163] proposed a distributed framework for intrusion detection in IoT based on task distribution among different nodes depending on the security requirements and the state of available resources. A controller component in the proposed framework administers the intrusion detection in an attack scenario.

Finally, an architecture-focused survey on IDS in IoT is carried out by Benkhelifa et al. [164]. This survey highlights the existing protocols, detection methods, and points out the future research challenges in this direction.

1) ML-based IDS in IoT: Shukla et al. [165] proposed ML-based lightweight IDS for low-power IoT networks running 6LoWPAN. They used the IDS mechanism to detect wormhole attacks in IoT networks. The proposed IDS mechanism uses three ML techniques, i.e. K-means clustering (unsupervised learning), decision tree (supervised learning), and a hybrid technique combining the aforementioned techniques. These ML techniques are centralized in architecture and the authors showed the adequacy of these techniques. K-means IDS achieve a varying detection rate (70-93%) depending on the size of the network and the decision tree method achieve 71-80% detection rate. Finally, the hybrid IDS mechanism achieve 71-75% detection in IoT networks. Despite its low detection rate, the hybrid mechanism is more accurate than the previous two mechanisms. On the other hand, Canedo et al. [166] leveraged two ML techniques to detect intrusions at the IoT gateways. The authors used ANN and genetic algorithms for IoT security. During the experiments, the authors investigated the anomalies in the data received from the edge devices. The anomalies were predicted over 99% of the times according to the training samples and with the two input neurons for the ANN. The authors also experimented on 3 input neurons and the prediction rate was still above 99% with 1% false negative.

Other ML techniques used for intrusion and anomaly detection include outlier detection, naive Bayes, RNN, decision tree, and DL. In [167], the authors used an outlier detection mechanism to deal with the unhealthy data in IoT networks. Traditional outlier detection methods include regression analysis, statistical methods that are known to require a sheer amount of data to draw conclusions about the outliers in the data. Nesa et al. [167] used non-parametric approach that is suitable for IoT because it does not require large storage to store the incoming data. The authors leverage sequence-based supervised learning based on Influential Relative Grade (IRG) and Relative Mass Function (RMF) which efficiently detects the outliers. The experimental results show that the error detection rate is 99.65% and 98.53% on different datasets. Viegas et al. [168] aim at energy-efficient and hardware-friendly implementation of the intrusion detection systems in IoT. The authors leveraged three classifiers, Decision Tree (DT), Naive Bayes (NB), and Linear Discriminant Analysis (LDA) during their experimentation for intrusion detection. In this work, the authors first analyzed the effect of single classifier among the aforementioned classifiers and then used the combination of these classifiers to see the effect on the detection. The experimental results showed that with the baseline testing, all the classifiers showed phenomenal accuracy above 99%; however, the accuracy was reduced by more than 30% in case of new attacks. This phenomenon shows that the ML approach still needs to reach maturity in order to detect new kinds of intrusions and other attacks. Similarly, Sedjelmaci et al. [169] also focused on the balance between energy consumption and intrusion detection in IoT networks. The authors used game theoretic approach for the detection of new types of intrusion in IoT networks.

2) DL-based IDS in IoT: Deep learning is also leveraged for IDS in heterogeneous IoT networks. For instance, Recurrent Neural Network (RNN) is used by Kim et al. [170] to train the IDS model which is based on Long Short Term Memory (LSTM) architecture. The authors performed experiments to find the optimal hyper-parameter to find the optimum false alarm and detection rate. The average accuracy was recorded to be 96.93% with hyper-parameter. Similarly, Saeed et al. [171] used Random Neural Networks (RaNN) for the realization of efficient and fast anomaly-based intrusion detection in low-

power IoT networks. The authors proposed a two-layer model where at the first layer, normal behavior is learnt by the system and at the second layer, different kinds of Illegal Memory Access (IMA) bugs and data integrity attacks on the network are detected. The proposed solution is centralized where the results are sent to a central server. Experimental results demonstrate that the detection accuracy on average is 97.23% with 10.45% overhead in the form of energy consumption in the experimental setup of this solution. However, these experiments do not encompass the full range of attacks and the zero-days attacks.

E. Malware Analysis in IoT

One of the most notorious attack domains is malicious code injection and execution in IoT devices by exploiting the existing vulnerabilities in IoT devices. The vulnerabilities that could be used for malware injection could be related to application security, authentication, and authorization. Apart from these methods, tampering the IoT devices physically for software modification and misconfiguration of security parameters could also enable attackers for the injection of malicious code. Before diving into the details of the malware, it is important to understand the types of the malware that endanger the IoT security. A malware is a threat that persists as a result of the aforementioned vulnerabilities and executed through a number of attacks. The common types of malware include, but not limited to, bot, spyware, ransomware, adware, trojan, and virus, to name a few. In this section, first we summarize the classification of malware that affect the IoT devices and then discuss the existing solutions including ML-based techniques that can keep IoT devices safe.

It has been found out through many exploratory studies that there are enormous smart devices that are connected to internet without any proper security protection that do not only pose threats to the device itself, but also enable the attackers to tap resources for the attacks at massive scale such as DDoS. For instance Moos et al. [172] tested his music device for vulnerability and exploited it to launch malicious-code attack which was very successful. Another project namely Insecam[5] lists the webcams that are potentially vulnerable to attacks and connected to Internet. These web-based cameras cover residential, public places, offices, and restaurants and can be used by attackers for malicious purposes. The types of Malware that have been successful in disrupting the normal functionality of the organization, application, or target entities include, but not limited to, NotPetya, Stuxnet, Cryptlocker, Red October, Night Dragon, and so on.

Makhdoom et al. [173] provided a detailed taxonomy and working principles of these malwares. These are the generic malware attacks whereas there are optimized families of malware attacks that particularly target the IoT devices. Such attacks include WanaCry, Cryptlocker, Mirai, Stuxnet, and so on. These are the malware attacks that have costed the industry staggering amount of money and other loss such as public image of the company. The details of these attacks are out of the scope of this paper. However, it is important to know the generalized approach of attackers to launch attack through malware [173]. First of all, the attackers gather knowledge about the potential target, for instance sensor networks, through reconnaissance. There are many methods/tools to carry out reconnaissance, for instance, tools such as Nmap, Metasploit, and Wireshark as well as social engineering.

The first step gives a clear idea to the attackers as to what kind of vulnerability to use against a specific class of devices. To date, standard application security assessment methods such as Open Web-Application Security Project (OWASP)[6] provides the major sources of vulnerabilities among which the attacker can choose appropriate vulnerability. In principle, OWASP provides the sources of exploitations that could be used by the attackers such as SQL injection, security misconfigurations, broken authentication, Cross-Site Scripting (XSS), and so on. Furthermore, depending on the type of the device, the attackers can send the payload to the target through several means such as phishing, updates, rootkit, and so on. The malware families vary from a simple single-task malware to a more complex, intelligent, dormant and multi-purpose evolving malware. Today's intelligent malwares are also adaptive according to the IoT environment where they can assess the network, and adapt its execution according to the underlying network. For instance, some malwares can evade the detection mechanism and stay dormant for sometime and do not execute the malicious code until it is safe for them not to compromise their intent. In this regard, there are a number of countermeasure techniques for malware to evade the detection mechanisms [45], [174], [175].

1) Malware evasion techniques: You et al. [174] outlined different approaches for malwares to evade the detection techniques. Encrypted malwares refer to an encrypted malicious code with respective decryptor to bypass the signature-based antivirus. However, since the decryptor remains the same for different versions of the same malware, it can still be detected. To overcome this problem, the decryptor can be mutated and thus bypasses the detection mechanism. This type of malware is referred to as oligomorphic malware. Similarly, polymorphic malware generates countless decryptor which makes it even harder for detection engines to detect malware. Finally, metamorphic malware is the most sophisticated among the malware groups where it evolves to a new generation where it is different from the previous one and very hard to be detected. You et al. [174] discussed the obfuscation techniques for polymorphic and metamorphic malwares that include dead-code insertion, register reassignment, instruction substitution, subroutine recording, and code integration. Furthermore, Rudd et al. [45] outlined in detail, the stealth malware and its mitigation techniques. To date, a number of malware detection techniques have been proposed in the literature for different breeds of the IoT networks. For instance Guerar et al. [176] considered the health-care domain of IoT and proposed malware detection technique for mobile devices. Mobile devices are used as gateway between back-end servers and the sensors in the IoT environment and therefore vulnerable to malware due to inherent mobile operating system characteristics. Guerar

[5]www.insecam.org

[6]https://www.owasp.org/

et al. proposed an invisible CAPTCHA mechanism for smart phone that uses a trusted sensor on mobile phone. This designated sensor records the user behavior on the phone and grants access to the IoT services based on the user behavior that is stored on the phone. Thus there is no need for user to enter the content of CAPTCHA. Similarly Abawajy et al. [177] investigated the malware in mobile IoT applications by analyzing the permissions set. The results of their investigation revealed that the security of mobile application is totally dependent on the vendors that host these applications, for instance Google and Apple. Therefore, abusing certain characteristics of application and targeting different features of the applications through malware, is not out of question. As it can be inferred from the preceding discussions, the traditional malware detection techniques might not be effective against sophisticated malwares; therefore, behavior-based intelligent malware analysis techniques are essential for IoT networks. In this context, ML and DL-based malware analysis techniques have been developed in the literature. Here we discuss these techniques in detail. ML and DL techniques such as RNN, random forest, Deep Eigenspace Learning (DEL), SVM, PCA, CNN, and ANN have been leveraged for malware analysis in IoT.

2) ML-based malware analysis in IoT: Alam et al. [178] used ensemble supervised learning technique with random forest classifier to detect android-based malware. The classifier is checked for the detection accuracy of the malware samples in android applications. In the experiments, the authors checked different parameters of the random forest such as tree size, and number of trees etc. The experiments showed that the classifier is able to achieve the detection accuracy of above 99% and the misclassification rate of random forest is less than its other counterparts such as BayesNet, NaiveBayes, Decision Stump, and so on. Furthermore, in the experimental setup of the authors, 16 was the optimal depth of the tree at which better detection accuracy was achieved. In [179], Zhou et al. investigated the malware detection and propagation in IoT-based Wireless Multimedia System (WMS). The authors proposed a cloud-based approach where SVM is leveraged to detect the potential malwares and their propagation, and used state-based differential game to suppress the malwares. After achieving the Nash Equilibrium, the authors try to find optimal strategies for WMS to defend against malwares. Similarly, Ham et al. [180] proposed a linear SVM-based technique for malware classification in android-based IoT. The authors also compared the results with other classifiers. Although SVM incurs more classification time due to the removal of unnecessary features; however, it is favorable due to its less complexity and better accuracy. To assess the detection accuracy of the detection model, the authors considered different types of malwares and their features. The results reported by the authors show that SVM performs relatively better than other classifiers for most of the investigated malwares where true positive is above 99%.

An et al. [181] proposed ML-based malware detection techniques to secure home routers again DDoS attacks. They used PCA, one-class SVM and an anomaly detector based on n-grams. Home routers in a smart home scenario are the luring targets for malware-based attacks where known vulnerabilities will provide a perfect playground for the attackers. The authors focused on anomaly detection through semi-supervised approaches and the experimental results reported in the paper demonstrate that these classifiers achieve higher detection rate and accuracy. The authors tested the well-known malware such as Mirai and its different variations. The results showed 100% detection rate with PCA and no false negatives. Similar results were reported for the other two classifiers with the well-known malware samples.

Similarly, Esmalifalak et al. [182], [183] use SVM and PCA to detect false data injection and stealthy attacks in smart grid. The application of this technique could be easily incorporated into IoT. The authors used two methods. In the first method, SVM is leveraged where labeled data is used for supervised learning for training the SVM, whereas in the second method, no training is used. Additionally, the authors used unsupervised learning. These ML techniques are used to isolate the tampered data from the normal data and thus-forth detecting the attacks. The results showed the effectiveness of the ML methods for bad data detection that could be the result of either malware or other kinds of attacks.

3) DL-based malware analysis in IoT: Pajouh et al. [184] proposed an RNN-based DL approach for malware analysis technique in IoT. The authors considered Advanced RISC Machines (ARM)-based applications in IoT. The authors train their models with different existing malware datasets and then test their framework with the new malware. Through experiments, the authors concluded that LSTM classifiers deliver best results among other classifiers. More precisely, the RNN-based DL technique achieved 98% accuracy in detecting new malware inside IoT application. Similarly, in another work, Azmoodeh et al. [185] aimed at a breed of IoT known as Internet of Battlefield Things (IoBT) and used DL technique to analyze the Operational Code (OpCode) sequence of the devices. The authors leveraged deep eigenspace learning and deep convolutional networks techniques to classify the malware in ARM-compatible IoT applications. In their analysis, the authors used Class-Wise Information Gain technique for features selection where both benignware and malware samples were selected for training. The authors used the OpCode sequences of the selected applications for the classification. The authors reported that the used classifiers achieve an accuracy of 99.68% in detecting malware with 98.59% precision and 98.37% recall.

Karbab et al. [186] proposed MalDozer, a DL-based malware analysis tool for android application framework. The detection framework is based on ANN and tested with both benign and malware applications in the android platform. In essence, MalDozer is based on sequences such as API method calls in android, resource permissions, and raw method calls. Furthermore, the proposed technique also automatically engineers features during training. The experimental results achieved detection rate of 96% to 98% android malware with the correct malware family. On the other hand, Su et al. [187] proposed an image recognition-based DDoS malware detection mechanism in IoT networks. In this solution, the authors first collect and classify two major families of malware, i.e. Mirai and Linux.Gafgyt and then convert the program binaries of

the IoT applications to grey-scale images. After that a small-scale CNN is applied to classify the images to goodware and malware. The experimental results reported by the authors show that the CNN-based method achieves 94% accuracy in classifying goodware and DDoS malware whereas it achieves 81.8% accuracy in detecting the afore-mentioned two families of the malware. It is also worth mentioning that ANN achieves better accuracy than CNN; however, the network size must be considerably larger than with the current solution incurring more processing overhead.

Meidan et al. [188] used deep autoencoders to detect Botnet attacks in IoT. In this solutions, the authors extract the network behavior and then use deep autoencoders to isolate the anomalous network behavior. For evaluation, the authors also used Mirai and demonstrated the effectiveness of using deep autoencoders.

Similarly, Shakeel et al. [189] used Deep Q networks with Q learning technique to address security issues such as authentication, access control, and malware analysis in health-care applications of the IoT networks. The authors leveraged Q-learning technique to analyze the patients' data through layered Deep Q-networks for both authentication and malware analysis. The authors reported that Deep Q networks consume less energy as compared to MLP and Learning Vector Quantization (LVQ). Furthermore, the mean error ratio for the proposed Learning-based Deep Q Network (LDQN) is the smallest (0.12) and the accuracy is the highest (98.79%) among LVQ, MLP, and Back Propagation Neural Network (BPNN).

F. Lessons Learned

In this section, we discuss the lessons learned as a result of the reviewed ML- and DL-based security solutions in IoT networks. We derive the following lessons from the preceding discussion on the security solutions based on ML and DL mechanisms in IoT networks. Since we already discussed the solutions in detail, we focus on the features of ML and DL mechanisms that need further attention despite their widespread use in both industry and academia.

1) ML-based security solutions in IoT networks have certain limitations: ML is used to create models, that are used to design, test, and train the datasets. These ML algorithms are used to identify possible patterns and similarities in large datasets and can make predictions in the new obtained data. However, we note that the fundamental limitation of ML approaches is that, it mostly needs dataset to learn from, and then the learned model is applied to the real data. This phenomena may not encompass the whole range of features and properties of the data. Furthermore, the data for training a model pose other security challenges and cyber attacks. Details of the attacks and their solutions are already discussed in the previous subsection. In this regard, DL techniques have been employed to address the limitations of the ML techniques. Addressing the limitations of ML, DL algorithms became the cornerstone of its success in the current industry. We also note that DL models have been used by a number of tech-giants such as Apple uses DL mechanism in its Siri project, Microsoft uses DL mechanisms in Cortana, Amazon uses DL algorithms in Alexa and similarly Google Photos, Spotify and Grammarly, are all driven by the DL algorithms. Furthermore, DL is also used in industrial domains such as financial industry for predicting stock price, health-care industry for re-purposing tested drugs for diseases, and for governmental institutions to get real-time predictions in food predication and energy infrastructure.

2) Breeds of DL algorithms are gaining momentum but bring along new limitations: RL, and DRL are some of the promising research areas and breeds of DL, leveraged for automated extraction of complex features from large amounts of high dimensional unsupervised data. Despite the fact that recent research in these domains has shown tremendous performance, there are still some areas where more optimized, focused, and IoT-specific improvements are needed. The current concrete theoretical and analytic foundations are still missing. It is also worth mentioning that the combined computational capabilities of RL and DL incur computational and storage overhead. Therefore, despite of their performance, these methods may not be well-suited for the resource-constrained IoT devices. Also, like other ML techniques, DL is susceptible to mis- and over-classification [190], [191]. Additionally, RL suffers from instability and divergence when all possible action-reward pairs are not used in a given scenario [102].

3) Noisy data is a challenge for ML and DL mechanisms: Most of the real-world data is embedded with noise that negatively affects the learning models used for classification. DL algorithms have better classification capabilities as compared to the traditional ML algorithms; however, these algorithms undermined by the noisy data.

4) Misclassification degrades the performance of ML and DL algorithms: Misclassification may occur due to the selection of characteristics and features that are not suitable for classification and training of a model. It happens when all the variables, classes, and groups have similar probability of being mis-classified. The cost of misclassification becomes disastrous in real world problems such as business and health-care applications. For instance, it can lead to wrong diagnosis in under-classified disease groups, incorrect credit scoring for identification of defaulters etc. [192], [193]. Unbalanced classes and the cost of misclassification are highly related as it means that for some of the samples, we have only little training data as well as making mistakes is even more risky. For example, in a medical disease diagnostics, it is very critical to diagnose a disease for a person who does not have it and even more so if the person has a disease and not diagnosing it. In both cases, it could have dire consequences for the subject of diagnosis because it could impair other normal functioning of the body and even endanger the life in some cases.

In order to minimize the effect of misclassification (under and over), it is better to collect more data from the rare classes. In medical diagnosis scenario, we try to focus on collecting images of patients who have a certain disease (the subject of diagnosis). Also, distribution of training labels should be carefully adjusted as it will affect the model's inference and prediction capabilities. For instance, if we increase the number of sick patients in our training data set, the model will also

predict the disease more often.

5) DL has limitations despite its widespread use, and therefore, we need emerging techniques instead of traditional DL mechanisms: It is evident from previous discussion that DL is a promising technique; however, it is still exposed to various challenges such as, but not limited to, need for massive data? and compute-intensive environment (for DL model building), centralized data storage and model development, adversarial effects, and smooth integration of ANN to existing wireless networks and IoT networks, are few most prominent to mention. These challenges are currently being addressed by new emerging approaches such as; Federated Learning (FL) and Generative Adversarial Network (GAN). Training of ML/DL models puts a massive strain on the network infrastructure due to huge data requirement and powerful computing resources after aggregation of data from various sources. On the other hand, FL trains models on distributed endpoint devices and simplifies the model training process. While GANs are also promising techniques where training of models takes place in a competing adversarial settings and thus increasing the accuracy of the classification.

As a result of these learned lessons, we discuss the emerging DL techniques for IoT security in the next section.

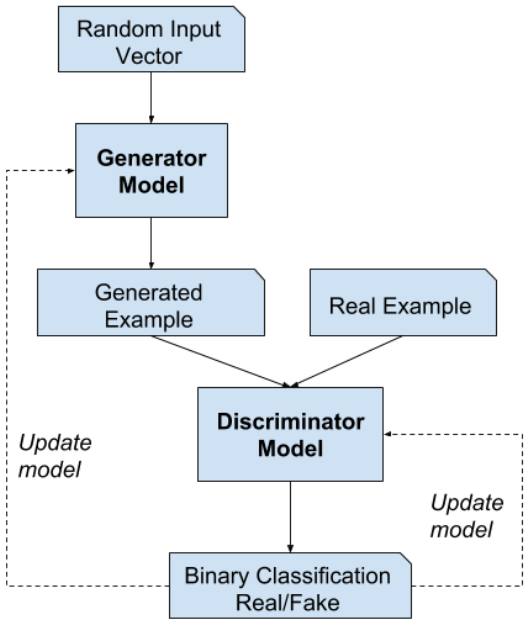

Fig. 3. General architecture of Generative Adversarial Network (GAN) [198]

G. Emerging ML Techniques in IoT Security

In the light of the existing ML and DL solutions for IoT security, we note that it is imperative to employ the current emerging ML techniques for addressing the IoT security issues. In essence, Generative Adversarial Networks (GANs) [194] and distributed learning (Federated Learning) [195]–[197] have recently gained a lot of attention from research community. In the following, we put light on these emerging technologies in the context of IoT security.

1) Generative Adversarial Network (GAN)-based IoT security: GANs have revolutionized the ML methods by using them in an adversarial setting [194]. In essence, two neural networks are deployed in a setting where they compete with each other in a zero-sum game and eventually come to an equilibrium. The two component neural networks include a generator and a discriminator. There is a training set on which a discriminator is trained. Then the generator network generates the candidate data whereas the discriminator evaluates the data to be either correctly classified or not (for instance mapping certain features to their respective labels). The aim of the generator is to fool the discriminator whereas the aim of the discriminator is to make itself stronger against the generator resulting eventually in a convergence of the network. In other words, the generator tries to make discriminator believe that the input data came from the sample rather than from the generator whereas the discriminator tries to figure out whether the data came from a real sample or a generator. The general layout of a GAN is shown in Fig. 3.

In the context of IoT security, GANs have produced remarkable results to solve different security problems in IoT networks. For instance, as already mentioned, intrusion detection is crucial for IoT networks and GANs have proved to be effective in detection intrusion in IoT. Belenko et al. [199] proposed a GAN-based intrusion detection system in a large-scale Cyber-Physical System (CPS).

Recently, Ferdowsi et al. [200] proposed a distributed GAN-based intrusion detection scheme for IoT networks. Unlike the traditional GANs, the authors employed distributed GANs to avoid the communication cost incurred by the centralized GAN. Furthermore, in case of a centralized GAN, the access to all the data points from the IoT devices should be granted. Therefore, it could also compromise the privacy. Whereas in the distributed GAN, the data is not shared by the IoT devices, rather the weights are shared among the IoT nodes. The authors reported that the distributed GANs could achieve about 20% higher detection accuracy, 25% higher precision, and 60% lower false positive as compared to the centralized GAN. Similarly, Intrator et al. [201] argued that GANs do not necessarily generate new valid samples and thus, may not provide robustness in anomaly detection. Therefore, the authors proposed a Multi-Discriminator GAN (MDGAN) which uses two discriminators where one is a dense network for ensuring the quality of the samples, and the second is an autoencoder for detecting intrusion. Furthermore, the generator in MDGAN aims at conflicting goals which include generating sufficiently 'good' samples to fool the discriminator and at the same time, help the autoencoder to generate the same samples. It is also reported in the results that GAN autoencoder outperforms the baseline autoencoder. Another simple GAN-based intrusion detection mechanism is proposed by Caminero et al. [202]. In this work, the authors use classifier to correctly predict the samples and then another adversarial configuration makes the prediction harder which is a typical GAN model. The authors reported that they achieved accuracy of more than 80% which is lower than the distributed GANs.

Apart from intrusion detection, GANs have also been leveraged for malware analysis, and D(DoS) attack detection.

In [203], the authors show that the existing ML classifiers for malware detection can easily be sabotaged by polluting the training data and the famous tools such as Drebin, DroidAPIMiner, and MaMaDroid can easily be fooled through specially crafted input data. Therefore, it is essential, to make the input data accurate and intelligent against such pollution. In this context, Chen et al. [203] proposed an AML-based scheme called KaufuDet, where in an offline phase, the features are extracted from the training set and a classifier that is trained in the previous phase, is used for detection. These two phases work in an adversarial setting where a camouflaged detector filters the potential false negatives in the first phases and give it again as an input to the training phase in order to strengthen the training phase. The authors reported that this setting the accuracy is increased by about 15%. Similarly, Chen et al. [204] also used API calls extracted from portable executable to evade malware detection mechanisms and employed an adversarial setting to make the malware classifier intelligent through classifier retraining like that of [203]. However, it is important that in an adversarial setting, the malware generated (adversarial examples) must possess their malicious features that will help the classifiers in strengthening their capabilities against malware. Because adversarial examples play vital role in the misclassification, which is the primary aim of adversarial attacks [205]. Dujaili eta al. [206] proposed a saddle-point formulation-based methods to produce a malicious-preserved malware samples for the training and test these samples on the portable executables. Similarly, McDaniel et al. [207] also investigated the causes of ML methods used in adversarial settings and their consequences.

Yalin et al. [208] investigated the adversarial attacks on DL-based spectrum sensing in IoT can easily degrade the wireless transmission of IoT devices by misleading them during spectrum sensing. They also propose a defense mechanism against such attacks where the IoT devices intentionally make some mistakes so that to mislead the attackers. This is a trade-off between the security and the efficiency. Samangouei et al. [209] also proposed a defensive GAN approach to mitigate the consequences of perturbations that cause disruptions in the learning process. The authors encourage the generator of GAN to produce samples close to the training data, which means that the legitimate samples will fall in the range of generator and the adversarial ones will be farther away. Although this defensive GAN is a generic solutions, but could be crafted for IoT applications.

In a nutshell, ML is a double-edged sword where on one hand, its features in adversarial settings could strengthen the accuracy, but on the other hand, it could endanger the whole learning process. It is important to make the positive side stronger and reduce the effect of adversarial settings that could manipulate the inputs for the training phases.

2) Federated leaning and IoT security: Federated Learning is a new AI model development framework, distributed over edge/mobile devices. It provides highly personalized and secure models, maintaining client/ user privacy. Essentially, in FL model development, training, and evaluation is done without direct access to user data [18].

With recent advancement in chip design and cellular technology, smartphones (Samsung S9, Apple X) have significant computational capability and are equipped with AI features. Therefore, most of the ML models are able to run on these mobile smart devices. These device (as a part of FL computing architecture) can download a model, which runs locally on these devices and model is further improved by learning from the local data stored in those devices. These updates in the improved model are summarized typically in the form of model parameters and corresponding weights [195]. These updates are encrypted and sent to master device (or to the cloud/ central server). Similarly, all devices part of the FL architecture will send the updates. Afterwards, all of these updates are averaged to improve the shared model. This distribution of heavy analytic and computations over smart devices, in contrast to centralized computing system, will result in various benefits. For instance, faster model deployment and time saving (faster response to continuously changing client behaviour) in the development of huge recommendation engines (personalized) and e-commerce pricing engines. It also enhances user privacy (as there is no direct access to raw data during development and training of models) and individual updates are unidentified in the cloud and central server during model updates.

These salient features make FL an excellent choice for mobile and distributed networks (e.g IoT networks) in terms of privacy preservation, and improved efficiency; however, it brings along few challenges. Adequate communication bandwidth is the most important requirement and it is very challenging to achieve the faster and in-time contributions from participating devices in a pervasive environment. Thus, the bottleneck is shifted from computation to communication. Insufficient communication bandwidth can incur latency and eventually longer time for FL model to converge. Scheduling of participating devices is also challenging and it is a problem in itself to decide *who* will send the updates and *when*? Another pressing issue is to ensure the reliability of participating devices. Practical and real-time IoT networks might not be static and network configuration keeps on changing. As a result, all the devices may not be fully participating till the convergence of the FL model. Since FL models learn iteratively and rely on the participating devices, learning quality of FL can be jeopardized if few devices are dropped out in the middle of the learning process.

VI. Future Research Challenges

In this section, we discuss the challenges faced by and future research opportunities in ML and DL techniques for the security of IoT networks.

A. Challenges and Limitations of DL and DRL

1) DL – one size does not fit all: DL techniques are very much application-specific where a model trained for solving one problem might not be able to perform well for another problem in the similar domain. The models usually need to be retrained with respective data to be used for other similar problems. This might not be a problem for some static

networks; however, for the real-time IoT applications, such models will be difficult to use. We believe that more insights are needed for DL techniques to be optimized and used for particular IoT applications.

2) Neural networks are black boxes: Deep neural networks act like a Blackbox, as we do not know how does any DL model reach a conclusion by manipulating the input data using the neurons at the intricately interconnected layers. Similarly in DNN, it is impossible to see how complex is the process of decision-making from one layer is transported to the next. Therefore, it becomes unsuitable for those applications in which *interpretability* is important.

3) Longer convergence time: Most of the RL algorithms The longer convergence time of RL algorithms may make them unsuitable for real-time applications. Furthermore, in case of safety-critical systems, time is essence, and the system cannot accommodate delays. Therefore, more research is needed to improve the convergence rate.

4) Butterfly effect of ML and DL: Butterfly effect is a phenomenon where a minute change in the input of a system creates chaos in the output. In this regard, ML and DL are also susceptible to this effect where a slight change in the input data to the learning system will create enormous change in the output which is the learned model. This phenomenon expose the ML and DL techniques used in IoT to security attacks where the attackers deliberately change the input data to make the system unstable. Such attacks are more dangerous since these attacks do not need an access to the system itself. More investigation is needed in this direction to devise integrity mechanisms for different IoT application domains.

5) Challenges for DL in the edge: IoT will leverage the advantages of edge computing which will increase the IoT applications and services space. However, due to the sheer amount of data generated by IoT devices, it will be hard to implement DL techniques in the edge devices. Furthermore, the time required for training a deep network also plays an important role. Therefore, real-time and time-critical applications might not able to take the advantage of DL in the edge. The stability of DL models is also important where newly available information will affect the already trained model. Therefore, more investigation is needed in this direction.

6) Over-fitting requirements and hyper parameters: Training off-line from the fixed data logs (specified with external behavior policy) and learning from limited samples on the real system greatly affect the credibility of decision making of DL models.

In essence, the efficacy of an ML model is judged by its ability to perform well on a new data set and not by its performance on the training data fed to it. Due to the difference in training and test distributions of data sets, ML classifiers usually fail when employed in real-world applications. Typically ML model is trained on a specific training data set and memorizes the training examples, but does not learn to generalize it for new datas ets and for new situations. As a result, errors occur in unseen new data set and during training data set, specifically in complicated models with too many parameters as compared to the number of observations.

Almost every ML algorithm has hyper parameters whose value is defined prior to the learning process and these parameters influence the behavior of the learning system. These are selectively or randomly selected and can invoke large change in models performance by even slight change in these parameters. Supervised learning is considered as stable due to fixed data sets whereas, RL and DRL are not stable at all [210].

7) Real-time response requirements: Real-time mission-critical IoT applications such as, autonomous vehicles, e-health, online banking etc. perform continuous sensing and information gathering from their surroundings. Therefore, model updates, interferences from surrounding knowledge sources, and predictions are based on live-streaming data. In all of these scenarios, the systems are stochastic and non-stationary, and have strong safety constraints. In contrast, training on a simulated environment has unlimited training data and deterministic system dynamics, and thus does not mimic the real-time behavior. DRL and RL suffer from large and/or unknown delays in the system in calculating rewards and computation due to real-time streaming of data in real-time applications. This phenomenon requires the design of new system architecture that can support flexible, programmable data pipelines (capable of handling variable volume, velocity and variety of real-time data) and algorithms capable of making decisions in real-time. Furthermore, real systems do not only have delays in the sensation of the state, the actuators, or the reward feedback, but also experience inference (at the control frequency of the system) in real-time [211].

The existing developed frameworks are capable of dealing with heterogeneous but static data. We need frameworks for dynamic data with stringent latency requirements. These new frameworks must guarantee real-time intelligence and incur extremely small latency.

B. Challenges Related to IoT Data

For data-driven ML and DL techniques, there are challenges related to unavailability of appropriate and enough data sets as discussed below.

1) Unavailability of training datasets: Efficient use of ML and DL solutions need datasets. Authentic datasets from real physical environment are required to analyze and compare the performances of various DL and RL algorithms. The data can contain personal and critical information that would not only identify the users but also their behavior and lifestyle. For instance, the data generated by BAN and other health-care related applications might compromise the user privacy and the data from smart home might result in exposing personal lifestyle as well as behavior. Therefore, it is important to make sure that the data used by ML and DL techniques does not put the user privacy at stake. To date, many anonymization techniques have been used that anonymize the data before using it for analytics; however, researches have also shown that the anonymization techniques can be hacked and the training models can be compromised by injecting false data. Collection of data while preserving the privacy and anonymity could be challenging. Also, questions such as how to apply ML and DL

algorithms to such data and what level of privacy should be preserved by the ML and DL algorithms need to be answered. It is, therefore important to investigate data protection and user privacy preservation techniques in ML and DL-based analytics for IoT networks.

Note that the data generated via simulations may not fully represent real IoT scenarios. Also, generation of synthetic data for training and testing DL models can be computationally very expensive.

2) Data imbalance: For an IoT system, the collected data sets for ML or DL is very likely to be imbalanced when the attacks are rarely events. These imbalanced data can significantly impact the performance of attack classifiers or IDS methods.

3) Data fusion: Fusion of data from different IoT devices and network elements will need to be done for construction of ML and DL models. However, this can be challenging since data from multiple sources are characterized by different modality and granularity, and also, there could be ambiguity and spuriousness.

C. Adversarial Machine Learning and IoT

Machine learning is a double-edge sword where on one hand it nourishes the value of the data, but on the other hand, can be used by the attackers for malicious purposes. Such branch of ML is called Adversarial Machine Learning (AML). In AML, the attackers use the features of the ML to attack the system. For instance, much research has been done by playing with the training parameters and misleading the learning system to learn the opposite of what it is supposed to do. More precisely, DL methods are prone to adversarial enhancements in the input data to launch attacks such as intrusion, DoS and so on. In this context, perturbation has been used in object recognition applications where changes into the classifiers causes the system to identify the wrong object. Recently, a one-pixel perturbation was used to fool a DNN [212]. Another sub-class of AML is called Generative Adversarial Network (GAN) which is leveraged by both attackers and the security experts to launch attacks and combat the security issues, respectively. For instance, GANs have been used for anomaly detection and intrusion detection in IoT networks [199]–[201], [213], [214] as well as DoS attack detection [215]. However, on the other hand, GANs as well as some other AML methods such as perturbation have also been used as attack vectors against IoT networks [216]–[218]. The recent research results show that DL mechanisms could sometime backfire if not safeguarded properly in already-vulnerable IoT networks. It is, therefore, extremely important to investigate the role and effects of AML in the IoT networks and address these challenges.

D. Efficiency of Security Solutions

The degree of sophistication of a security mechanism depends on the capabilities of the device and the system where it is used. The limitations of the IoT devices is a major challenge in applying sophisticated security mechanisms. In the previous section, we discussed the ML- and DL-based security mechanisms; however, the resource constraints create a set-back where a trade-off is needed between the level of security and the capabilities of the IoT devices. Sophisticated security solutions need considerable amount of computing, storage, and communication resources. Furthermore, it is also important to determine where to put the logic of ML and DL techniques in the network. Therefore, in-depth investigation is needed for the efficiency of the security mechanisms that use ML and DL techniques. In this context, low-cost and highly efficient security mechanisms must be investigated for IoT where they can harness the benefits of the ML and DL as well.

E. Complex Cyber Threats

IoT networks use resource-constrained devices ranging from home-appliances to personal gadgets. These devices are usually easy targets for cyber attacks. As aforementioned, a sheer amount of data is generated by these devices which might be used by ML and DL techniques for different applications. The compromise of these devices will have dire consequences on the outcomes of the applications. It is worth noting that for the applications such as smart home, the consequences of compromise might not be that critical as compared to critical-infrastructure and medical applications. These applications will not be able to withstand the results from the ML and DL systems as a result of compromised data. It could even endanger human lives. Therefore, it is essential to make sure the device safety and the health of the data that is input to the ML and DL systems. Furthermore, the compromised devices could also be used as bots by the attackers as launching pads for other attacks. Therefore, for ML and DL systems to work in a safe way, it is essential to focus on the security aspects of the IoT devices and on the health of the generated data. Similarly, the fairly recent cyber attacks (such as Mirai [82]) on low-power and resource-constrained devices is also alarming for the IoT networks. There is an increased interest in the evolved machine learning techniques such as distributed learning and generative adversarial learning-based games; however, in the wake of afore-mentioned cyber threats and active attacks, the ML and DL models will favor the attackers.

F. Legislative Challenges for ML in IoT Security

The influx of IoT services and applications in various domains has spurred the legislative discussion among the research community and the industry. Some domains of the IoT are still struggling with efficient and acceptable legislative policies. For instance, autonomous car technology is going to benefit from IoT in the customization of user experience; however, there is no clear legislation available for commercialization of the autonomous car technology as well as using the data generated by such technologies for training and analysis. Insurance is another challenge for such technology where it is hard to decide whom to insure. Hussain et al. [219], [220] discussed the policy challenges for autonomous car in detail. These challenges will equally affect the technologies supported by the autonomous car including IoT. Furthermore, the data generated by these technologies will be required for ML mechanism to learn and model different behaviors. However,

some data that might be either critical for business or too personal to use (for instance data from health-care applications and users' data from financial institutions), will be challenging for using with ML-based solutions.

The validation and certification of different components of IoT, for instance in Body Area Network (BAN) is also a challenge that is currently keeping the investors at the bay from investing in these technologies and it will equally affect the ML-based solutions. The implementation of General Data Protection Regulation (GDPR) [221], [222] and different regulations on the import and export of cryptographic algorithms also pose significant challenges on the IoT security. Furthermore, different legislations on different IoT applications such as smart home, smart e-health, and so on are shadowed by different regulations in different countries, therefore, one security solution (both traditional and ML-based) might not work for different regions. To date, USA and European, and some Asian governments are working on legislations that are viable and acceptable to the consumers and service providers (both in services and security). It is believed that the legislative policies will define the course for the success and adaptation of these new technologies among the consumers.

VII. Conclusion

IoT security and privacy are of paramount importance and play a pivotal role in the commercialization of the IoT technology. Traditional security and privacy solutions suffer from a number of issues that are related to the dynamic nature of the IoT networks. ML and more specifically DL and DRL techniques can be used to enable the IoT devices to adapt to their dynamic environment. These learning techniques can support self-organizing operation and also optimize the overall system performance by learning and processing statistical information from the environment (e.g. human users and IoT devices). These learning techniques are inherently distributed and do not require centralized communication between device and controller. However, the datasets needed for ML and DL algorithms are still scarce, which makes benchmarking the efficiency of the ML- and DL-based security solutions a difficult task.

In this paper, we have considered the role of ML and DL in the IoT from security and privacy perspective. We have discussed the security and privacy challenges in IoT, attack vectors, and security requirements. We have described different ML and DL techniques and their applications to IoT security. We have also shed light on the limitations of the traditional ML mechanisms. Then we have discussed the existing security solutions and outlined the open challenges and future research directions. In order to mitigate some of the shortcomings of machine learning approaches to IoT security, the theoretical foundations of DL and DRL will need to be strengthened so that the performances of the DL and DRL models can be quantified based on parameters such as computational complexity, learning efficiency, as well as parameter tuning strategies. Furthermore, new hybrid learning strategies and novel data visualization techniques will be required for intuitive and efficient data interpretation.

References

[1] O. Novo, N. Beijar, and M. Ocak, "Capillary Networks - Bridging the Cellular and IoT Worlds ," *IEEE World Forum on Internet of Things (WF-IoT)*, vol. 1, pp. 571–578, December 2015.

[2] F. Hussain, *Internet of Things; Building Blocks and Business Modles*. Springer, 2017.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, pp. 2347–2376, Fourthquarter 2015.

[4] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys Tutorials*, vol. 17, pp. 1294–1312, thirdquarter 2015.

[5] A. Mosenia and N. K. Jha, "A comprehensive study of security of internet-of-things," *IEEE Transactions on Emerging Topics in Computing*, vol. 5, pp. 586–602, Oct 2017.

[6] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, vol. 4, pp. 1125–1142, Oct 2017.

[7] M. Ammar, G. Russello, and B. Crispo, "Internet of things: A survey on the security of iot frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8 – 27, 2018.

[8] I. Stellios, P. Kotzanikolaou, M. Psarakis, C. Alcaraz, and J. Lopez, "A survey of iot-enabled cyberattacks: Assessing attack paths to critical infrastructures and services," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 3453–3495, Fourthquarter 2018.

[9] F. Restuccia, S. DOro, and T. Melodia, "Securing the internet of things in the age of machine learning and software-defined networking," *IEEE Internet of Things Journal*, vol. 5, pp. 4829–4842, Dec 2018.

[10] K. A. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147 – 157, 2019.

[11] J. Hou, L. Qu, and W. Shi, "A survey on internet of things security from data perspectives," *Computer Networks*, vol. 148, pp. 295 – 306, 2019.

[12] D. E. Kouicem, A. Bouabdallah, and H. Lakhlef, "Internet of things security: A top-down survey," *Computer Networks*, vol. 141, pp. 199 – 221, 2018.

[13] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of things security: A survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10 – 28, 2017.

[14] B. B. Zarpelo, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in internet of things," *Journal of Network and Computer Applications*, vol. 84, pp. 25 – 37, 2017.

[15] M. binti Mohamad Noor and W. H. Hassan, "Current research on internet of things (iot) security: A survey," *Computer Networks*, vol. 148, pp. 283 – 294, 2019.

[16] K. T. Nguyen, M. Laurent, and N. Oualha, "Survey on secure communication protocols for the internet of things," *Ad Hoc Networks*, vol. 32, pp. 17 – 31, 2015. Internet of Things security and privacy: design methods and optimization.

[17] K. Sha, W. Wei, T. A. Yang, Z. Wang, and W. Shi, "On security challenges and open issues in internet of things," *Future Generation Computer Systems*, vol. 83, pp. 326 – 337, 2018.

[18] M. Tao, K. Ota, M. Dong, and Z. Qian, "Accessauth: Capacity-aware security access authentication in federated-iot-enabled v2g networks," *Journal of Parallel and Distributed Computing*, vol. 118, pp. 107 – 117, 2018.

[19] I. Farris, T. Taleb, Y. Khettab, and J. Song, "A survey on emerging sdn and nfv security mechanisms for iot systems," *IEEE Communications Surveys Tutorials*, vol. 21, pp. 812–837, Firstquarter 2019.

[20] M. at. el, "Machine Learning for Internet of Things Data Analysis:A Survey ," *Journal of Digital Communications and Networks, Elsevier*, vol. 1, pp. 1–56, Feburary 2018.

[21] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet of Things Journal*, vol. 4, pp. 1250–1258, Oct 2017.

[22] A. Chowdhury and S. A. Raut, "A survey study on internet of things resource management," *Journal of Network and Computer Applications*, vol. 120, pp. 42 – 60, 2018.

[23] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of things applications: A systematic review," *Computer Networks*, vol. 148, pp. 241 – 261, 2019.

[24] D. Airehrour, J. Gutierrez, and S. K. Ray, "Secure routing for internet of things: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 198 – 213, 2016.

[25] J. Guo, I.-R. Chen, and J. J. Tsai, "A survey of trust computation models for service management in internet of things systems," *Computer Communications*, vol. 97, pp. 1 – 14, 2017.

[26] V. Gazis, "A survey of standards for machine-to-machine and the internet of things," *IEEE Communications Surveys Tutorials*, vol. 19, pp. 482–511, Firstquarter 2017.

[27] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-things-based smart environments: state of the art, taxonomy, and open research challenges," *IEEE Wireless Communications*, vol. 23, pp. 10–16, October 2016.

[28] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26521–26544, 2017.

[29] F. Javed, M. K. Afzal, M. Sharif, and B. Kim, "Internet of things (iot) operating systems support, networking technologies, applications, and challenges: A comparative review," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 2062–2100, thirdquarter 2018.

[30] A. olakovi and M. Hadiali, "Internet of things (iot): A review of enabling technologies, challenges, and open research issues," *Computer Networks*, vol. 144, pp. 17 – 39, 2018.

[31] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for iot big data and streaming analytics: A survey," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 2923–2960, Fourthquarter 2018.

[32] W. Ding, X. Jing, Z. Yan, and L. T. Yang, "A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion," *Information Fusion*, vol. 51, pp. 129 – 144, 2019.

[33] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, and M. Guizani, "A survey of machine and deep learning methods for internet of things (iot) security," *CoRR*, vol. abs/1807.11023, 2018.

[34] D. B. J. Sen, "Internet of Things - Applications and Challenges in Technology and Standardization," *IEEE Transactions in Wireless Personal Communication*, May 2011.

[35] U. S. Shanthamallu, A. Spanias, C. Tepedelenlioglu, and M. Stanley, "A Brief Survey of Machine Learning Methods and their Sensor and IoT Applications ," *IEEE Conference on Information, Intelligence, Systems and Applications*, March 2018.

[36] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 1153–1176, Secondquarter 2016.

[37] J. Gardiner and S. Nagaraja, "On the security of machine learning in malware c&c detection: A survey," *ACM Comput. Surv.*, vol. 49, pp. 59:1–59:39, Dec. 2016.

[38] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE Access*, vol. 6, pp. 12103–12117, 2018.

[39] A. Shabtai, R. Moskovitch, Y. Elovici, and C. Glezer, "Detection of malicious code by applying machine learning classifiers on static features: A state-of-the-art survey," *Information Security Technical Report*, vol. 14, no. 1, pp. 16 – 29, 2009. Malware.

[40] H. Jiang, J. Nagra, and P. Ahammad, "Sok: Applying machine learning in security - A survey," *CoRR*, vol. abs/1611.03186, 2016.

[41] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL detection using machine learning: A survey," *CoRR*, vol. abs/1701.07179, 2017.

[42] L. Nishani and M. Biba, "Machine learning for intrusion detection in manet: a state-of-the-art survey," *Journal of Intelligent Information Systems*, vol. 46, pp. 391–407, Apr 2016.

[43] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.

[44] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, "A detailed investigation and analysis of using machine learning techniques for intrusion detection," *IEEE Communications Surveys Tutorials*, vol. 21, pp. 686–728, Firstquarter 2019.

[45] E. M. Rudd, A. Rozsa, M. Gnther, and T. E. Boult, "A survey of stealth malware attacks, mitigation measures, and steps toward autonomous open world solutions," *IEEE Communications Surveys Tutorials*, vol. 19, pp. 1145–1172, Secondquarter 2017.

[46] A. Azaria, A. Richardson, S. Kraus, and V. S. Subrahmanian, "Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data," *IEEE Transactions on Computational Social Systems*, vol. 1, pp. 135–155, June 2014.

[47] N. Koroniotis, N. Moustafa, and E. Sitnikova, "Forensics and deep learning mechanisms for botnets in internet of things: A survey of challenges and solutions," *IEEE Access*, vol. 7, pp. 61764–61785, 2019.

[48] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[49] X. Wang, J. Li, X. Kuang, Y. an Tan, and J. Li, "The security of machine learning in an adversarial setting: A survey," *Journal of Parallel and Distributed Computing*, vol. 130, pp. 12 – 23, 2019.

[50] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," *Computers & Security*, vol. 81, pp. 123 – 147, 2019.

[51] S. Mahdavifar and A. A. Ghorbani, "Application of deep learning to cybersecurity: A survey," *Neurocomputing*, vol. 347, pp. 149 – 176, 2019.

[52] P. A. A. Resende and A. C. Drummond, "A survey of random forest based methods for intrusion detection systems," *ACM Comput. Surv.*, vol. 51, pp. 48:1–48:36, May 2018.

[53] E. Fernandes, A. Rahmati, K. Eykholt, and A. Prakash, "Internet of things security research: A rehash of old ideas or new intellectual challenges?," *IEEE Security Privacy*, vol. 15, no. 4, pp. 79–84, 2017.

[54] J. Qiu, Q. Wu, G. Ding, Y. Xu, and S. Feng, "A survey of machine learning for big data processing," *EURASIP Journal of Advance Signal Process*, May 2016.

[55] S. et al., "Deep Learning for the Internet of Things," *IEEE Journal of Computer*, vol. 51, pp. 32–41, May 2018.

[56] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of things (iot) communication protocols: Review," in *2017 8th International Conference on Information Technology (ICIT)*, pp. 685–690, May 2017.

[57] J. Chen, S. Li, H. Yu, Y. Zhang, D. Raychaudhuri, R. Ravindran, H. Gao, L. Dong, G. Wang, and H. Liu, "Exploiting icn for realizing service-oriented communication in iot," *IEEE Communications Magazine*, vol. 54, pp. 24–30, December 2016.

[58] Y. Liu, Y. Kuang, Y. Xiao, and G. Xu, "Sdn-based data transfer security for internet of things," *IEEE Internet of Things Journal*, vol. 5, pp. 257–268, Feb 2018.

[59] I. Andrea, C. Chrysostomou, and G. Hadjichristofi, "Internet of things: Security vulnerabilities and challenges," in *2015 IEEE Symposium on Computers and Communication (ISCC)*, pp. 180–187, July 2015.

[60] S. Benzarti, B. Triki, and O. Korbaa, "A survey on attacks in internet of things based networks," in *2017 International Conference on Engineering MIS (ICEMIS)*, pp. 1–7, May 2017.

[61] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Commun. ACM*, vol. 47, pp. 53–57, June 2004.

[62] P. Li, L. Sun, X. Fu, and L. Ning, *Security in Wireless Sensor Networks*, pp. 179–227. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[63] A. G. Finogeev and A. A. Finogeev, "Information attacks and security in wireless sensor networks of industrial scada systems," *Journal of Industrial Information Integration*, vol. 5, pp. 6 – 16, 2017.

[64] H. B. Salameh, S. Almajali, M. Ayyash, and H. Elgala, "Spectrum assignment in cognitive radio networks for internet-of-things delay-sensitive applications under jamming attacks," *IEEE Internet of Things Journal*, pp. 1–1, 2018.

[65] L. Hu, H. Wen, B. Wu, F. Pan, R. F. Liao, H. Song, J. Tang, and X. Wang, "Cooperative jamming for physical layer security enhancement in internet of things," *IEEE Internet of Things Journal*, vol. 5, pp. 219–228, Feb 2018.

[66] A.-R. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial internet of things," in *Proceedings of the 52Nd Annual Design Automation Conference*, DAC '15, (New York, NY, USA), pp. 54:1–54:6, ACM, 2015.

[67] H. K. Patil and R. Seshadri, "Big data security and privacy issues in healthcare," in *2014 IEEE International Congress on Big Data*, pp. 762–765, June 2014.

[68] S. Raza, L. Wallgren, and T. Voigt, "Svelte: Real-time intrusion detection in the internet of things," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2661 – 2674, 2013.

[69] P. Jadav and V. K. Babu, "Fuzzy logic based faulty node detection in wireless sensor network," in *2017 International Conference on Communication and Signal Processing (ICCSP)*, pp. 0390–0394, April 2017.

[70] Y. Nishiguchi, A. Yano, T. Ohtani, R. Matsukura, and J. Kakuta, "Iot fault management platform with device virtualization," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pp. 257–262, Feb 2018.

[71] C. Yicheng, Z. Xuecheng, L. Zhenglin, H. Yu, and Z. Zhaoxia, "Energy-efficient and security-optimized aes hardware design for ubiquitous computing," *Journal of Systems Engineering and Electronics*, vol. 19, pp. 652–658, Aug 2008.

[72] G. Bansod, N. Raval, and N. Pisharoty, "Implementation of a new lightweight encryption design for embedded security," *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 142–151, Jan 2015.

[73] S. Oukili and S. Bri, "High speed efficient advanced encryption standard implementation," in *2017 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–4, May 2017.

[74] R. Hussain and H. Oh, "On secure and privacy-aware sybil attack detection in vehicular communications," *Wireless Personal Communications*, vol. 77, pp. 2649–2673, Aug 2014.

[75] W. Dong and X. Liu, "Robust and secure time-synchronization against sybil attacks for sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 11, pp. 1482–1491, Dec 2015.

[76] G. Han, X. Li, J. Jiang, L. Shu, and J. Lloret, "Intrusion detection algorithm based on neighbor information against sinkhole attack in wireless sensor networks," *The Computer Journal*, vol. 58, pp. 1280–1292, June 2015.

[77] S. Katsikeas, K. Fysarakis, A. Miaoudakis, A. V. Bemten, I. Askoxylakis, I. Papaefstathiou, and A. Plemenos, "Lightweight secure industrial iot communications via the mq telemetry transport protocol," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1193–1200, July 2017.

[78] A. Mondal and S. Bhattacharjee, "A reliable, multi-path, connection oriented and independent transport protocol for iot networks," in *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, pp. 590–591, Jan 2017.

[79] A. Haroon, S. Akram, M. A. Shah, and A. Wahid, "E-lithe: A lightweight secure dtls for iot," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, Sept 2017.

[80] T. Buddhika and S. Pallickara, "Neptune: Real time stream processing for internet of things and sensing environments," in *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 1143–1152, May 2016.

[81] T. Heer, O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar, and K. Wehrle, "Security challenges in the ip-based internet of things," *Wireless Personal Communications*, vol. 61, pp. 527–542, Dec 2011.

[82] E. Bertino and N. Islam, "Botnets and internet of things security," *Computer*, vol. 50, pp. 76–79, Feb 2017.

[83] T. H. Szymanski, "Security and privacy for a green internet of things," *IT Professional*, vol. 19, no. 5, pp. 34–41, 2017.

[84] S. Scott-Hayward, G. O'Callaghan, and S. Sezer, "Sdn security: A survey," in *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*, pp. 1–7, Nov 2013.

[85] M. C. Dacier, H. Konig, R. Cwalinski, F. Kargl, and S. Dietrich, "Security challenges and opportunities of software-defined networking," *IEEE Security and Privacy*, vol. 15, pp. 96–100, Apr. 2017.

[86] J. Singh, T. Pasquier, J. Bacon, H. Ko, and D. Eyers, "Twenty security considerations for cloud-supported internet of things," *IEEE Internet of Things Journal*, vol. 3, pp. 269–284, June 2016.

[87] D. Ambedkar, "Reinforcement Learning Algorithms: Survey and Classification," *Indian Journal of Science and Technology*, vol. 10, pp. 1–8, January 2017.

[88] S. Bansal, R. Calandra, S. Levine, and C. Tomlin, "MBMF: model-based priors for model-free reinforcement learning," *CoRR*, vol. abs/1709.03153, 2017.

[89] V. Feinberg, A. Wan, I. Stoica, M. I. Jordan, J. E. Gonzalez, and S. Levine, "Model-based value estimation for efficient model-free reinforcement learning," *CoRR*, vol. abs/1803.00101, 2018.

[90] D. Ha and J. Schmidhuber, "World models," *CoRR*, vol. abs/1803.10122, 2018.

[91] W. Dabney, M. Rowland, M. G. Bellemare, and R. Munos, "Distributional reinforcement learning with quantile regression," *CoRR*, vol. abs/1710.10044, 2017.

[92] C. Wirth, R. Akrour, G. Neumann, and J. Frnkranz, "A Survey of Preference-Based Reinforcement Learning Methods," *Journal of Machine Learning Research*, vol. 18, pp. 1–46, January 2017.

[93] S. Mukkamala, A. H. Sung, A. Abraham, and V. Ramos, "Intrusion detection systems using adaptive regression spines," in *Enterprise Information Systems VI* (I. Seruca, J. Cordeiro, S. Hammoudi, and J. Filipe, eds.), (Dordrecht), pp. 211–218, Springer Netherlands, 2006.

[94] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, "Iot security techniques based on machine learning: How do iot devices use ai to enhance security?," *IEEE Signal Processing Magazine*, vol. 35, pp. 41–49, Sep. 2018.

[95] A. O. Prokofiev, Y. S. Smirnova, and V. A. Surov, "A method to detect internet of things botnets," in *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pp. 105–108, Jan 2018.

[96] R. H. E. H. Fatima Hussain, Syed Ali Hassan, "1Machine Learning for Resource Management inFuture Cellular and IoT Networks: Potentials,Current Solutions, and Open Challenges," *https://arxiv.org/submit/2772922*, 2019.

[97] L. et al., "Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices," *ACM international conference on Mobile computing and networking*, vol. 1, pp. 237–248, 2014.

[98] L. et al., "DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning," *ACM International Conference on Pervasive and Ubiquitous Computing*, vol. 1, pp. 283–294, 2015.

[99] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep Learning for Wireless Physical Layer:Opportunities and Challenges," *IEEE China Communication*, vol. 14, pp. 92–111, October 2017.

[100] H. Li, K. Ota, and M. Dong, "Deep Reinforcement Scheduling for Mobile Crowdsensing in Fog Computing," *ACM Transactions on Internet Technology*, vol. 19, pp. 26–35, April 2019.

[101] H. Li, K. Ota, and M. Dong, "Learning IoT in Edge:Deep Learning for the Internet of Things with Edge Computing," *IEEE Network*, vol. 32, pp. 96–101, April 2018.

[102] M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli, "Applications of deep learning and reinforcement learning to biological data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 2063–2079, June 2018.

[103] M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J. Oh, "Semisupervised deep reinforcement learning in support of iot and smart city services," *IEEE Internet of Things Journal*, vol. 5, pp. 624–635, April 2018.

[104] N. D. Nguyen, T. Nguyen, and S. Nahavandi, "System design perspective for human-level agents using deep reinforcement learning: A survey," *IEEE Access*, vol. 5, pp. 27091–27102, 2017.

[105] D. et al., "Review on the Research and Practice of Deep Learning and Reinforcement Learning in Smart Grids," *CSEE Journal of Power and Energy Systems*, vol. 4, pp. 362–370, September 2018.

[106] S. at. el, "Machine Intelligence Techniques for Next-Generation Context-Aware Wireless Networks," *AAAAI*, vol. 1, pp. 1–9, 2018.

[107] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "A Brief Survey of Deep Reinforcement Learning," *IEEE Signal Processing Magazine*, May 2017.

[108] B. Lake, T. Ullman, J. Tenenbaum, and S. Gershman, "Building Machines That Learn and Think Like People," *The Behavioral and Brain Sciences*, 2016.

[109] T. Park, N. Abuzainab, and W. Saad, "Learning How to Communicate in the Internet of Things: Finite Resources and Heterogeneity," *IEEE Access*, vol. 4, pp. 7063–7073, November 2016.

[110] T. E. Bogale, X. Wang, and L. B. Le, "Machine Intelligence Techniques for Next-Generation Context-Aware Wireless Networks," *Arxiv*, vol. 19, pp. 1–10, January 2018.

[111] A. LHeureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, "Machine Learning With Big Data: Challenges and Approaches," *IEEE Access*, vol. 5, pp. 7776 – 7797, April 2017.

[112] M. Taylor, D. Reilly, and B. Lempereur, "An access control management protocol for internet of things devices," *Network Security*, vol. 2017, no. 7, pp. 11 – 17, 2017.

[113] A. Ouaddah, H. Mousannif, A. A. Elkalam, and A. A. Ouahman, "Access control in the internet of things: Big challenges and new opportunities," *Computer Networks*, vol. 112, pp. 237 – 262, 2017.

[114] A. A. A. El-Aziz and A. Kannan, "A comprehensive presentation to xacml," in *Third International Conference on Computational Intelligence and Information Technology (CIIT 2013)*, pp. 155–161, Oct 2013.

[115] F. P. Diez, A. C. Vasu, D. S. Touceda, and J. M. S. Cmara, "Modeling xacml security policies using graph databases," *IT Professional*, vol. 19, pp. 52–57, November 2017.

[116] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "Iot-oas: An oauth-based authorization service architecture for secure services in iot scenarios," *IEEE Sensors Journal*, vol. 15, pp. 1224–1234, Feb 2015.

[117] D. Rivera, L. Cruz-Piris, G. Lopez-Civera, E. de la Hoz, and I. Marsa-Maestre, "Applying an unified access control for iot-based intelligent

agent systems," in *2015 IEEE 8th International Conference on Service-Oriented Computing and Applications (SOCA)*, pp. 247–251, Oct 2015.
[118] P. P. Pereira, J. Eliasson, and J. Delsing, "An authentication and access control framework for coap-based internet of things," in *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, pp. 5293–5299, Oct 2014.
[119] Y. Yang, X. Zheng, W. Guo, X. Liu, and V. Chang, "Privacy-preserving smart iot-based healthcare big data storage and self-adaptive access control system," *Information Sciences*, 2018.
[120] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, pp. 2130–2145, June 2018.
[121] L. Yeh, P. Chiang, Y. Tsai, and J. Huang, "Cloud-based fine-grained health information access control framework for lightweight iot devices with dynamic auditing and attribute revocation," *IEEE Transactions on Cloud Computing*, vol. 6, pp. 532–544, April 2018.
[122] Q. Huang, Y. Yang, and L. Wang, "Secure data access control with ciphertext update and computation outsourcing in fog computing for internet of things," *IEEE Access*, vol. 5, pp. 12941–12950, 2017.
[123] Q. Liu, H. Zhang, J. Wan, and X. Chen, "An access control model for resource sharing based on the role-based access control intended for multi-domain manufacturing internet of things," *IEEE Access*, vol. 5, pp. 7001–7011, 2017.
[124] F. Li, J. Hong, and A. A. Omala, "Efficient certificateless access control for industrial internet of things," *Future Generation Computer Systems*, vol. 76, pp. 285 – 292, 2017.
[125] F. Li, Y. Han, and C. Jin, "Practical access control for sensor networks in the context of the internet of things," *Computer Communications*, vol. 89-90, pp. 154 – 164, 2016. Internet of Things Research challenges and Solutions.
[126] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, pp. 1184–1195, April 2018.
[127] L. Xiao, Y. Li, G. Han, G. Liu, and W. Zhuang, "Phy-layer spoofing detection with reinforcement learning in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 65, pp. 10037–10047, Dec 2016.
[128] L. Xiao, X. Wan, and Z. Han, "Phy-layer authentication with multiple landmarks with reduced overhead," *IEEE Transactions on Wireless Communications*, vol. 17, pp. 1676–1687, March 2018.
[129] B. Chatterjee, D. Das, S. Maity, and S. Sen, "Rf-puf: Enhancing iot security through authentication of wireless nodes using in-situ machine learning," *IEEE Internet of Things Journal*, vol. 6, pp. 388–398, Feb 2019.
[130] C. Shi, J. Liu, H. Liu, and Y. Chen, "Smart user authentication through actuation of daily activities leveraging wifi-enabled iot," in *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Mobihoc '17, (New York, NY, USA), pp. 5:1–5:10, ACM, 2017.
[131] R. Das, A. Gadre, S. Zhang, S. Kumar, and J. M. F. Moura, "A deep learning approach to iot authentication," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2018.
[132] J. Chauhan, S. Seneviratne, Y. Hu, A. Misra, A. Seneviratne, and Y. Lee, "Breathing-based authentication on resource-constrained iot devices using recurrent neural networks," *Computer*, vol. 51, pp. 60–67, May 2018.
[133] S. Rathore and J. H. Park, "Semi-supervised learning based distributed attack detection framework for iot," *Applied Soft Computing*, vol. 72, pp. 79 – 89, 2018.
[134] I. Hafeez, A. Y. Ding, M. Antikainen, and S. Tarkoma, "Toward secure edge networks: Taming device-to-device (D2D) communication in iot," *CoRR*, vol. abs/1712.05958, 2017.
[135] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 1773–1786, Aug 2016.
[136] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for internet of things," *Future Generation Computer Systems*, vol. 82, pp. 761 – 768, 2018.
[137] A. Abeshu and N. Chilamkurti, "Deep learning: The frontier for distributed attack detection in fog-to-things computing," *IEEE Communications Magazine*, vol. 56, pp. 169–175, Feb 2018.
[138] N. Vlajic and D. Zhou, "Iot as a land of opportunity for ddos hackers," *Computer*, vol. 51, pp. 26–34, July 2018.
[139] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "Ddos in the iot: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.
[140] G. Kambourakis, C. Kolias, and A. Stavrou, "The mirai botnet and the iot zombie armies," in *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*, pp. 267–272, Oct 2017.
[141] D. Yin, L. Zhang, and K. Yang, "A ddos attack detection and mitigation with software-defined internet of things framework," *IEEE Access*, vol. 6, pp. 24694–24705, 2018.
[142] L. A. B. Pacheco, J. J. C. Gondim, P. A. S. Barreto, and E. Alchieri, "Evaluation of distributed denial of service threat in the internet of things," in *2016 IEEE 15th International Symposium on Network Computing and Applications (NCA)*, pp. 89–92, Oct 2016.
[143] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *SIGCOMM Comput. Commun. Rev.*, vol. 31, pp. 38–47, July 2001.
[144] S. Alharbi, P. Rodriguez, R. Maharaja, P. Iyer, N. Subaschandrabose, and Z. Ye, "Secure the internet of things with challenge response authentication in fog computing," in *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*, pp. 1–2, Dec 2017.
[145] Q. Yan, W. Huang, X. Luo, Q. Gong, and F. R. Yu, "A multi-level ddos mitigation framework for the industrial internet of things," *IEEE Communications Magazine*, vol. 56, pp. 30–36, Feb 2018.
[146] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning ddos detection for consumer internet of things devices," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 29–35, May 2018.
[147] J. Ye, X. Cheng, J. Zhu, L. Feng, and L. Song, "A ddos attack detection method based on svm in software defined network," *Security and Communication Networks*, vol. 2018, p. 8, 2018.
[148] S. Bera, S. Misra, and A. V. Vasilakos, "Software-defined networking for internet of things: A survey," *IEEE Internet of Things Journal*, vol. 4, pp. 1994–2008, Dec 2017.
[149] S. Tomovic, K. Yoshigoe, I. Maljevic, and I. Radusinovic, "Software-defined fog network architecture for iot," *Wireless Personal Communications*, vol. 92, pp. 181–196, Jan 2017.
[150] K. RT, S. T. Selvi, and K. Govindarajan, "Ddos detection and analysis in sdn-based environment using support vector machine classifier," in *2014 Sixth International Conference on Advanced Computing (ICoAC)*, pp. 205–210, Dec 2014.
[151] Z. Tan, A. Jamdagni, X. He, P. Nanda, and R. P. Liu, "A system for denial-of-service attack detection based on multivariate correlation analysis," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, pp. 447–456, Feb 2014.
[152] Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "Sinr-based dos attack on remote state estimation: A game-theoretic approach," *IEEE Transactions on Control of Network Systems*, vol. 4, pp. 632–642, Sept 2017.
[153] E. Hodo, X. Bellekens, A. Hamilton, P. Dubouilh, E. Iorkyase, C. Tachtatzis, and R. Atkinson, "Threat analysis of iot networks using artificial neural network intrusion detection system," in *2016 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, May 2016.
[154] R. V. Kulkarni and G. K. Venayagamoorthy, "Neural network based secure media access control protocol for wireless sensor networks," in *2009 International Joint Conference on Neural Networks*, pp. 1680–1687, June 2009.
[155] J. W. Branch, C. Giannella, B. Szymanski, R. Wolff, and H. Kargupta, "In-network outlier detection in wireless sensor networks," *Knowledge and Information Systems*, vol. 34, pp. 23–54, Jan 2013.
[156] M. A. Alsheikh, S. Lin, D. Niyato, and H. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Communications Surveys Tutorials*, vol. 16, pp. 1996–2018, Fourthquarter 2014.
[157] W. Meng, "Intrusion detection in the era of iot: Building trust via traffic filtering and sampling," *Computer*, vol. 51, pp. 36–43, July 2018.
[158] J. Li, Z. Zhao, R. Li, and H. Zhang, "Ai-based two-stage intrusion detection for software defined iot networks," *CoRR*, vol. abs/1806.02566, 2018.
[159] D. Li, Z. Cai, L. Deng, X. Yao, and H. H. Wang, "Information security model of block chain based on intrusion sensing in the iot environment," *Cluster Computing*, Mar 2018.
[160] A. A. Gendreau and M. Moorman, "Survey of intrusion detection systems towards an end to end secure internet of things," in *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*, pp. 84–90, Aug 2016.
[161] I. Butun, S. D. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," *IEEE Communications Surveys Tutorials*, vol. 16, pp. 266–282, First 2014.
[162] A. Abduvaliyev, A. K. Pathan, J. Zhou, R. Roman, and W. Wong, "On the vital areas of intrusion detection systems in wireless sensor

networks," *IEEE Communications Surveys Tutorials*, vol. 15, pp. 1223–1237, Third 2013.

[163] J. F. Colom, D. Gil, H. Mora, B. Volckaert, and A. M. Jimeno, "Scheduling framework for distributed intrusion detection systems over heterogeneous network architectures," *Journal of Network and Computer Applications*, vol. 108, pp. 76 – 86, 2018.

[164] E. Benkhelifa, T. Welsh, and W. Hamouda, "A critical review of practices and challenges in intrusion detection systems for iot: Towards universal and resilient systems," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2018.

[165] P. Shukla, "Ml-ids: A machine learning approach to detect wormhole attacks in internet of things," in *2017 Intelligent Systems Conference (IntelliSys)*, pp. 234–240, Sept 2017.

[166] J. Caedo and A. Skjellum, "Using machine learning to secure iot systems," in *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, pp. 219–222, Dec 2016.

[167] N. Nesa, T. Ghosh, and I. Banerjee, "Non-parametric sequence-based learning approach for outlier detection in iot," *Future Generation Computer Systems*, vol. 82, pp. 412 – 421, 2018.

[168] E. Viegas, A. Santin, L. Oliveira, A. Frana, R. Jasinski, and V. Pedroni, "A reliable and energy-efficient classifier combination scheme for intrusion detection in embedded systems," *Computers & Security*, vol. 78, pp. 16 – 32, 2018.

[169] H. Sedjelmaci, S. M. Senouci, and M. Al-Bahri, "A lightweight anomaly detection technique for low-resource iot devices: A game-theoretic methodology," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2016.

[170] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," in *2016 International Conference on Platform Technology and Service (PlatCon)*, pp. 1–5, Feb 2016.

[171] A. Saeed, A. Ahmadinia, A. Javed, and H. Larijani, "Intelligent intrusion detection in low-power iots," *ACM Trans. Internet Technol.*, vol. 16, pp. 27:1–27:25, Dec. 2016.

[172] J. Moos, "Iot, malware and security," *ITNOW*, vol. 59, no. 1, pp. 28–29, 2017.

[173] I. Makhdoom, M. Abolhasan, J. Lipman, R. P. Liu, and W. Ni, "Anatomy of threats to the internet of things," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2018.

[174] I. You and K. Yim, "Malware obfuscation techniques: A brief survey," in *2010 International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 297–300, Nov 2010.

[175] C. S. Veerappan, P. L. K. Keong, Z. Tang, and F. Tan, "Taxonomy on malware evasion countermeasures techniques," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pp. 558–563, Feb 2018.

[176] M. Guerar, A. Merlo, M. Migliardi, and F. Palmieri, "Invisible cappcha: A usable mechanism to distinguish between malware and humans on the mobile iot," *Computers & Security*, vol. 78, pp. 255 – 266, 2018.

[177] J. Abawajy, S. Huda, S. Sharmeen, M. M. Hassan, and A. Almogren, "Identifying cyber threats to mobile-iot applications in edge computing paradigm," *Future Generation Computer Systems*, vol. 89, pp. 525 – 538, 2018.

[178] M. S. Alam and S. T. Vuong, "Random forest classification for detecting android malware," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pp. 663–669, Aug 2013.

[179] W. Zhou and B. Yu, "A cloud-assisted malware detection and suppression framework for wireless multimedia system in iot based on dynamic differential game," *China Communications*, vol. 15, pp. 209–223, Feb 2018.

[180] H.-S. Ham, H.-H. Kim, M.-S. Kim, and M.-J. Choi, "Linear svm-based android malware detection for reliable iot services," *Journal of Applied Mathematics*, vol. 2014, p. 10, 2014.

[181] N. An, A. Duff, G. Naik, M. Faloutsos, S. Weber, and S. Mancoridis, "Behavioral anomaly detection of malware on home routers," in *2017 12th International Conference on Malicious and Unwanted Software (MALWARE)*, pp. 47–54, Oct 2017.

[182] M. Esmalifalak, N. T. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," in *2013 IEEE Global Communications Conference (GLOBECOM)*, pp. 808–813, Dec 2013.

[183] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Systems Journal*, vol. 11, pp. 1644–1652, Sept 2017.

[184] H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K.-K. R. Choo, "A deep recurrent neural network based approach for internet of things malware threat hunting," *Future Generation Computer Systems*, vol. 85, pp. 88 – 96, 2018.

[185] A. Azmoodeh, A. Dehghantanha, and K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning," *IEEE Transactions on Sustainable Computing*, pp. 1–1, 2018.

[186] E. B. Karbab, M. Debbabi, A. Derhab, and D. Mouheb, "Maldozer: Automatic framework for android malware detection using deep learning," *Digital Investigation*, vol. 24, pp. S48 – S59, 2018.

[187] J. Su, D. V. Vargas, S. Prasad, D. Sgandurra, Y. Feng, and K. Sakurai, "Lightweight classification of iot malware based on image recognition," *CoRR*, vol. abs/1802.03714, 2018.

[188] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiotnetwork-based detection of iot botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, pp. 12–22, Jul 2018.

[189] P. Mohamed Shakeel, S. Baskar, V. R. Sarma Dhulipala, S. Mishra, and M. M. Jaber, "Maintaining security and privacy in health care system using learning based deep-q-networks," *Journal of Medical Systems*, vol. 42, p. 186, Aug 2018.

[190] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, "Robust physical-world attacks on machine learning models," *CoRR*, vol. abs/1707.08945, 2017.

[191] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS P)*, pp. 372–387, March 2016.

[192] R. Rothe, "Applying deep learning to real-world problems." https://medium.com/merantix/applying-deep-learning-to-real-world-problems-ba2d86ac5837.

[193] A. Vidhya, "How to handle imbalanced classification problems in machine learning?." https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/.

[194] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 2672–2680, Curran Associates, Inc., 2014.

[195] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities and challenges," *arXiv preprint arXiv:1908.06847*, 2019.

[196] L. Georgopoulos and M. Hasler, "Distributed machine learning in networks by consensus," *Neurocomputing*, vol. 124, pp. 2 – 12, 2014.

[197] Y. Qian, L. Hu, J. Chen, X. Guan, M. M. Hassan, and A. Alelaiwi, "Privacy-aware service placement for mobile edge computing via federated learning," *Information Sciences*, vol. 505, pp. 562 – 570, 2019.

[198] J. Brownlee, "A gentle introduction to generative adversarial networks (gans)," Jun 2019.

[199] V. Belenko, V. Chernenko, M. Kalinin, and V. Krundyshev, "Evaluation of gan applicability for intrusion detection in self-organizing networks of cyber physical systems," in *2018 International Russian Automation Conference (RusAutoCon)*, pp. 1–7, Sep. 2018.

[200] A. Ferdowsi and W. Saad, "Generative adversarial networks for distributed intrusion detection in the internet of things," *CoRR*, vol. abs/1906.00567, 2019.

[201] Y. Intrator, G. Katz, and A. Shabtai, "MDGAN: boosting anomaly detection using multi-discriminator generative adversarial networks," *CoRR*, vol. abs/1810.05221, 2018.

[202] G. Caminero, M. Lopez-Martin, and B. Carro, "Adversarial environment reinforcement learning algorithm for intrusion detection," *Computer Networks*, vol. 159, pp. 96 – 109, 2019.

[203] S. Chen, M. Xue, L. Fan, S. Hao, L. Xu, H. Zhu, and B. Li, "Automated poisoning attacks and defenses in malware detection systems: An adversarial machine learning approach," *Computers & Security*, vol. 73, pp. 326 – 344, 2018.

[204] L. Chen, Y. Ye, and T. Bourlai, "Adversarial machine learning in malware detection: Arms race between evasion attack and defense," in *2017 European Intelligence and Security Informatics Conference (EISIC)*, pp. 99–106, Sep. 2017.

[205] A. Abusnaina, A. Khormali, H. Alasmary, J. Park, A. Anwar, U. Meteriz, and A. Mohaisen, "Breaking graph-based iot malware detection systems using adversarial examples: Poster," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '19, (New York, NY, USA), pp. 290–291, ACM, 2019.

[206] A. Al-Dujaili, A. Huang, E. Hemberg, and U. OReilly, "Adversarial deep learning for robust detection of binary encoded malware," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 76–82, May 2018.

[207] P. McDaniel, N. Papernot, and Z. B. Celik, "Machine learning in adversarial settings," *IEEE Security Privacy*, vol. 14, pp. 68–72, May 2016.

[208] Y. E. Sagduyu, Y. Shi, and T. Erpek, "Iot network security from the perspective of adversarial deep learning," *CoRR*, vol. abs/1906.00076, 2019.

[209] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," *CoRR*, vol. abs/1805.06605, 2018.

[210] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "https://arxiv.org/pdf/1904.12901.pdf," *https://arxiv.org/pdf/1904.12901.pdf*, 2019.

[211] L. Lei, Y. Tan, S. Liu, K. Zheng, and X. S. Shen, "Deep reinforcement learning for autonomous internet of things: Model, applications and challenges," *https://arxiv.org/pdf/1907.09059.pdf*, 2019.

[212] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *CoRR*, vol. abs/1710.08864, 2017.

[213] D. Li, D. Chen, L. Shi, B. Jin, J. Goh, and S. Ng, "MAD-GAN: multivariate anomaly detection for time series data with generative adversarial networks," *CoRR*, vol. abs/1901.04997, 2019.

[214] H. Wang, M. Li, F. Ma, S. Huang, and L. Zhang, "Poster abstract: Unsupervised anomaly detection via generative adversarial networks," in *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 313–314, April 2019.

[215] Q. Yan, M. Wang, W. Huang, X. Luo, and F. R. Yu, "Automatically synthesizing dos attack traces using generative adversarial networks," *International Journal of Machine Learning and Cybernetics*, Feb 2019.

[216] L. Muñoz-González, B. Pfitzner, M. Russo, J. Carnerero-Cano, and E. C. Lupu, "Poisoning attacks with generative adversarial nets," *CoRR*, vol. abs/1906.07773, 2019.

[217] J. Clements, Y. Yang, A. A. Sharma, H. Hu, and Y. Lao, "Rallying adversarial techniques against deep learning for network security," *CoRR*, vol. abs/1903.11688, 2019.

[218] W. Yang, D. Kong, T. Xie, and C. A. Gunter, "Malware detection in adversarial settings: Exploiting feature evolutions and confusions in android apps," in *Proceedings of the 33rd Annual Computer Security Applications Conference*, ACSAC 2017, (New York, NY, USA), pp. 288–302, ACM, 2017.

[219] R. Hussain and S. Zeadally, "Autonomous cars: Research results, issues and future challenges," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2018.

[220] R. Hussain, J. Lee, and S. Zeadally, "Autonomous cars: Social and economic implications," *IT Professional*, vol. 20, pp. 70–77, Nov 2018.

[221] C. Li and B. Palanisamy, "Privacy in internet of things: From principles to technologies," *IEEE Internet of Things Journal*, vol. 6, pp. 488–505, Feb 2019.

[222] G. Vojkovic, "Will the gdpr slow down development of smart cities?," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1295–1297, May 2018.

[223] L. Deng, D. Li, X. Yao, D. Cox, and H. Wang, "Mobile network intrusion detection for iot system based on transfer learning algorithm," *Cluster Computing*, Jan 2018.

[224] "Computanional complexity of machine learning algorithms," *https://www.thekerneltrip.com/machine/learning/computational-complexity-learning-algorithms/*, 2018.

[225] F. Hussain, A. Anpalagan, A. S. Khwaja, and M. Naeem, "Resource Allocation and Congestion Control in Clustered M2M Communication using Q-Learning," *Wiley Transactions on Emerging Telecommunications Technologies*, 2015.

[226] M. K. Pakhirag, "A Linear Time-Complexity k-Means Algorithm Using Cluster Shifting," *IEEE International Conference on Computational Intelligence and Communication Network*, pp. 1049–1053, November 2014.

[227] I. M. Johnstone and A. Y. Lu, "Sparse principal components analysis," *http://statweb.stanford.edu/ imj/WEBLIST/AsYetUnpub/sparse.pdf*, 2004.

[228] S. Koenig and R. G. Simmons, "Complexity analysis of real-time reinforcement learning," *http://idm-lab.org/bib/abstracts/papers/aaai93.pdf*, 2019.

[229] H. H. Pajouh, R. Javidan, R. Khayami, D. Ali, and K. R. Choo, "A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in iot backbone networks," *IEEE Transactions on Emerging Topics in Computing*, pp. 1–1, 2018.

TABLE IV: Machine Learning Techniques Used in IoT Security.

Supervised Learning			
Machine Learning Algorithm	**Description**	**Suitability/Application**	**Complexity**
Naive Bayes	It is the classification algorithm. It is named as "Naive", as over-simplified assumptions are made for the calculation of probabilities for specific hypothesis. All the attributes are assumed to be independent of each other [223].	• Used in binary and multi-class environment. • Effective in anomaly and intrusion detection problems. • It works best when used with discrete data, and it can fall into wrong prediction if continuous data is used.	$O(nP)$ [224]
K-Nearest Neighbour	It is supervised learning algorithm and is used for associating new data points to the existing similar points by searching through the available dataset. The model is trained and grouped according to some criteria and incoming data is checked for similarity within K neighbours [146].	• A very simple (only two parameters required to implement KNN, i.e., the value of K and the distance function (e.g., Euclidean or Manhattan, etc.)) algorithm that can be used as an initial assessment of simple classification in small networks. • The performance of the algorithm is degraded in large datasets and bigger networks. This is due to the fact that the cost of calculating distance between new points and each existing point is very big. It is also sensitive to the noisy data. • KNN does not work well with high dimensional data. This is because of the fact, high computational cost, for distance calculation in each dimensions, is associated with large number of dimensions. • KNN is a very fast algorithm and does not require any training period (e.g., SVM, Linear Regression etc.). It stores the training dataset and learns from it only at the time of making real time predictions.	$O(nP)$ [224]
Random forest and Decision Tree (DT)	It is a supervised learning method. It defines a model by implementing certain rules inferring from the data features. Afterwards, this model is used to predict the value of new targeted variable. DT is used in classification and as well as regression problems. Essentially, these trees are used to split dataset into several branches based on certain rules [178].	• Random forests work well with high dimensional data (as it can work on subsets of features in data and also can store the generated forests for future use). However, for large datasets, the size of the trees can incur huge storage cost. • It can handle unbalanced data and has methods for balancing error in class population unbalanced datasets.	If n is the number of training sample, P be the number of features, and n_{trees} be the number of trees. • $O((n^2)p)$ (RF) • $O((n^2)Pn_{trees})$ (DT) [224]

Support Vector Machines (SVM)	SVM is a supervised ML algorithm with low computational complexity, used for classification and regression. It classifies input data into n dimensional space and draws $n-1$ hyperplane to divide the entire data points into groups [179], [180].	• It has the ability to work with binary as well as with multi-class environments. • Works well with unstructured and semi structured data such as, text, images, and trees. • SVM algorithm is not suitable for large data sets.	$O(n^2P + n^3)$ [224]
Neural Network (NN)	This is a supervised learning algorithm used to develop a cascaded chain of decision units for solving the complex problems [132]. It essentially constructs network with certain number of inputs to trigger outputs. Various types of neural networks have been proposed in the literature, e.g., Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs) [184], [186], [187].	• NNs algorithms can be used to perform nonlinear statistical modeling. • NNs require less formal statistical training, and can implicitly detect complex nonlinear relationships between dependent and independent variables. Therefore, these algorithms can detect all possible interactions between predictor variables. • NNs suffer from its "black box" nature, high computational overhead, and is prone to overfitting. • NNs have ability to work with inadequate knowledge, and after training phase, output is obtained even with incomplete information.	If n is the number of training sample, P be the number of features, and nl_i be the number of neurons at layer i in a neural network: $O((Pnl_1)+(nl_1)(nl_2)+..)$ [224]
Deep Learning	It is essentially a Feed-forward Neural Network (FNN) in which each neuron is connected to another layer and no connection exists within the layer. The term deep learning refers to multiple hidden layers between the input and output layers such that each layer receives input from the previous layer and feeds the results to following layer [185].	• It has the ability to work with unstructured data as almost all the real-world problems deal with unstructured data (data exists in different types and formats such as image, text, etc.). Essentially, different data formats can be used to train DL algorithms and still obtain meaningful insights to the data (which is the main purpose of the training). • Feature engineering is not required and features are automatically deduced and optimally tuned for desired outcome. A DL algorithm scans the data to identify features that correlate, and afterwards combine them to promote faster learning without being told to do so explicitly. As a result, it also eliminates the need for labelling the data. • It can handle large amount of data and massive parallel computations can be performed using GPUs. Also, performance is improved if results are obtained from large amount of data. However, it will not work well if input data is not enough for proper fearture extraction and learning.	For the Multi-Layer Perceptron (MLP) and other neural networks, it is $O(2^n)$ to converge to optimal solution and $O(\#epochs \times \#examples \times \#features \times \#neurons)$ for approximate solution
Unsupervised Learning			

K-Means algorithm	The most commonly used technique belonging to the unsupervised category of ML family is the K-means clustering algorithm. It is used to classify or group devices based on attributes or parameters, into K number of groups, where K is a positive integer number and its value has to be known for the algorithm to work [225].	• while dealing with data classification without knowing the correlation among them, K-Means clustering is the first thing researchers do. • It does not learn the number of clusters from the data and is required to be known a priori. • K-Means assumes spherical shapes of clusters (with radius equal to the distance between the centroid and the farthest data point) and does not work well when the clusters are in different shapes such as elliptical clusters. Moreover, it does not work well in case of overlapping clusters because it does not have the intrinsic measure to determine for which cluster to assign each data point.	If n is the input data size: $O(n^2)$ [226]
Principal Component Analysis (PCA)	It is an unsupervised ML algorithm and a multivariate technique for data compression. It performs dimensionality reduction in large datasets and extracts useful information in the form of a set comprised of orthogonal variables known as "principal components". These components are organized in an increasing order of variance where first component is associated with highest variance of the data and it continues to the last. The components having the least variance can be discarded [181].	• PCA is an excellent choice in real-world situations when there are numerous features and it is difficult to visualize the relationship and to find the correlation among all the features. Also, it is very hard to reduce to the number of useful features from huge number of available features. • PCA can speed up the ML algorithm by eliminating the correlated variables (which do not contribute to any decision making) and training time of the algorithm reduces significantly with less number of features. • PCA transforms a high dimensional data to low dimensional data (2 dimensions) and helps in overcoming the overfitting issue.	If n is the number of data points and each represented by P features: $O(n^2P+n^3)$ [227]
Reinforcement Learning			

Q-Learning	It belongs to Reinforcement Learning (RL) class of ML. In RL, an agent learns by trial and error as to how its actions effect the environment. It estimates the reward after each action and moves to the new state accordingly [225].	• RL prefers to achieve long-term results and can be used to solve very complex problems that cannot be solved by conventional techniques. It is the most suitable choice when there is no training dataset available and learning is obtained through experience and interaction with the environment. • RL assumes real-world problems as Markovian model (which is not the case in reality). As probability of each event depends only on the state obtained in the previous event (as per Markovian model), in contrast to real world applications. • It suffers from curse of dimensionality and limits its applicability to the real physical systems. Also, it is data hungry and needs lots of data and computation. • To deal with various limitations of RL, it is used in combination with other ML techniques. One such popular combination is; RL with DL known as DRL.	$O(n^2)$ [228]

TABLE V: Security problems in IoT networks and applied Machine Learning techniques

Research objective/problems	**Machine Learning techniques (surveyed references)**	**Comparison of ML techniques**
Authentication and access control	• Deep learning and Long Short-Term Memory (LSTM) [131] • Artificial Neural Networks (ANNs) [129] • Recurrent Neural Networks (RNNs) [132] • Q-learning and Dyna-Q [127] • Deep Neural Network (DNN) [130]	RNN outperforms the other ML/ DL algorithms not only in terms of efficiency, but also in terms of accuracy and complexity with a system accuracy of 90%. Furthermore, LSTM outperforms SVM. Furthermore, DL algorithms can perform well in authentication than mainstream ML algorithm despite the small amount of available data.
Attack detection and mitigation	• SVM • Deep learning [136], [137], [133] • Unsupervised learning, stacked autoencoders • Extreme Learning Machine (ELM)-based semi-supervised Fuzzy C-Means (ESFCM) • K-Nearest Neighbour (NN) and SVM [135]	• ESFCM deals with labeled data, and hence increases the detection rate of the distributed attacks. However, DL-based models have better detection accuracy as compared to the traditional ML algorithms for attack detection. • K-NN performs better in small networks whereas SVM performs better in large networks in terms of attack detection accuracy.
Distributed DOS attack	• K-Nearest Neighbour [146] • Support Vector Machine [146] • Random Forest and Decision Tree [146] • Neural Network [146] • Multivariate Correlation Analysis (MCA) [151] • Q learning [152]	• Random forest algorithms have low number of control and model parameters, and are resistant to over-fitting. These algorithms do not require feature selection as it can use a large number of potential attributes. However, the variance of the model decreases as the number of trees in the forest increases, whereas the bias remains the same. • SVMs are known for their generalization capability and suitability for data consisting of a large number of feature attributes but a small number of sample points. • MCA is based on behavioral analysis of the traffic and have overall detection accuracy of 99%. • ANN techniques achieve 99.4% accuracy in experimental evaluation in a limited dataset for DDoS attacks [146].

Anomaly/Intrusion detection	• K-means clustering [165] • Artificial Neural Network ANN [166] • Novelty and Outlier Detection [167] • Decision Tree [168] • Naive Bayes [168], [229]	• K-Means clustering is used when labelled data is difficult to generate, therefore it is used for anonymizing the private data in the IoT applications where labelled data is not available. The main advantage of using clustering for intrusion detection is that it can learn from audit data without requiring the system administrator to provide explicit descriptions of various attack classes [36]. • DT is easy-to-use and has intuitive knowledge expression, high classification accuracy, and is easy to implement. • NB is known for its simplicity, ease of implementation and low training sample requirement. NB classifiers can handle continuous/categorical arbitrary number of independent features. NB reduces high-dimensional density estimation tasks to single dimension by assuming that all the features are independent. It is an optimal classifier if the features are conditionally independent given the true class, and its training can be completed in linear time. • DL models have shown an outstanding performance in large-scale data analysis [51]. Traditional learning techniques such as NNs, fuzzy model, and Hidden Markov Model (HMM) used for IDS suffer from having shallow architecture and have limitations in dealing with big network traffic data. Furthermore, traditional learning methods have specific constraints and cannot be applied properly to complex classification problems. However, sufficient data specific to the problem domain is required for training and classification of the algorithm. For instance, for IDS to learn various attack scenarios, terabytes of data should be provided to the DL classifier to train itself.
Malware analysis	• Recurrent Neural Network (RNN) [184] • Random Forest supervised classifier [178] • Deep Eigen sapce Learning and Deep Convolutional Networks [185] • SVM [179] • PCA, one-class SVM, and naive anomaly detector based on unseen n-grams [181] • CNN [187] • Artificial Neural Network [186] • Linear SVM [180] • SVM and PCA [182], [183] • Deep Q networks [189]	• PCA achieves dimensionality reduction and reduces the complexity of models developed for malware analysis. • CNN-based method achieves 94% accuracy in classification of DDoS malware. ANN achieves even better accuracy than CNN; however, if network size is large, it will incur huge processing overhead. • Deep Q networks consume less energy as compared to MLP and Learning Vector Quantization (LVQ). Furthermore, it has highest accuracy of 98.79% and mean error ratio of 0.12% among LVQ, MLP, and Back Propagation Neural Network (BPNN). • DL models and techniques (RNN, DCN etc.) are recommended for malware analysis as traditional ML algorithms such as SVM, Baysian Networks, Logistic Regression (LR), and MLP are unsuccessful in identifying obfuscated or repackaged malware. Traditional ML algorithms also suffer from having shallow architectures and do not scale well to large databases. Furthermore, their feature extraction phase is not automatic and need to utilize hand-crafted features. • RNN-based DL technique achieved 98% detection accuracy.