Journal of Network and Systems Management (2022) 30:8
https://doi.org/10.1007/s10922-021-09621-9

Deep Learning in IoT Intrusion Detection

Stefanos Tsimenidis[1] · **Thomas Lagkas**[1] · **Konstantinos Rantos**[1]

Received: 21 December 2020 / Revised: 16 May 2021 / Accepted: 9 August 2021

Abstract
The Internet of Things (IoT) is the new paradigm of our times, where smart devices and sensors from across the globe are interconnected in a global grid, and distributed applications and services impact every area of human activity. With its huge economic impact and its pervasive influence over our lives, IoT is an attractive target for criminals, and cybersecurity becomes a top priority for the IoT ecosystem. Although cybersecurity has been the subject of research for decades, the large-scale IoT architecture and the emergence of novel threats render old strategies largely inefficient. Deep learning may provide cutting edge solutions for IoT intrusion detection, with its data-driven, anomaly-based approach and ability to detect emerging, unknown attacks. This survey offers a detailed review of deep learning models that have been proposed for IoT intrusion detection. Solutions have been classified by model in a comprehensive, structured analysis of how deep learning has been applied for IoT cybersecurity and their unique contributions to the development of effective IoT intrusion detection solutions.

Keywords Deep learning · Intrusion detection · Internet of Things · Deep Neural Networks · Cybersecurity · IDS

✉ Thomas Lagkas
tlagkas@cs.ihu.gr

Stefanos Tsimenidis
sttsime@cs.ihu.gr

Konstantinos Rantos
krantos@cs.ihu.gr

[1] Department of Computer Science, International Hellenic University, Kavala Campus, Greece

Published online: 08 October 2021

1 Introduction

The emergence and development of new technologies such as sensors, broadband Internet, 5G and beyond wireless communications, radio frequency identification, smartphones, portable computers, industrial automations, semi-autonomous vehicles, satellites, cloud computing, and others, converge into what we broadly refer to as the Internet of Things (IoT), an all-encompassing network where smart devices and computational units are inter-connected, communicating and interacting with each other. At the same time, cyber-physical systems, comprising IoT devices, control critical infrastructures.

IoT ecosystems introduce new attack vectors in the expanded attack surface they introduce to their environments, and as a consequence, new forms of cyberattacks emerge that either use IoT devices as a stepping stone towards other systems or target IoT devices themselves. IoT systems are susceptible to attacks, and their complexity and novel architectures make the current cybersecurity paradigm inadequate, if not obsolete. A vicious circle occurs, where the interconnectivity of IoT devices and the novelty of IoT architectures make IoT vulnerable to threats, and where the plethora of attacks and breaches poses a serious setback in the further development of the IoT ecosystem [120]. The numerous companies that produce IoT devices use various different protocols and standards, rendering uniform encryption standards across these devices impossible. Given their limited computational and power resources, IoT devices have difficulty supporting sophisticated security solutions [3]. Vlajic et al. [115] perceived the IoT as the "Land of opportunity for DDoS attackers," due to the number of devices connected on the Internet and the security weaknesses posed by their resource-constrained nature. Generally, it is estimated that 25% of all cyberattacks target IoT devices [80], and this number is expected to increase. All in all, the pervasiveness of IoT devices, which will impact every facet of human activity, makes the subject of IoT security extremely significant and concerning [51, 102].

Intrusion detection systems (IDS) identify attacks, unauthorized intrusions, and malicious activity in networks, and constitute one of the main security measures found in contemporary networks. An IDS monitors the activity in a network, or a host, determining whether these are normal or anomalous, alerting the system administrator in the latter case.

Being a well-established approach for securing assets and resources against malicious activities, IDSs have been the subject of extensive research. A lot of progress has been made, but now, with the emergence of IoT and distributed networks, the landscape changes. As networks rise in complexity they become more prone to errors. Novel attacks are being invented constantly, while insiders take advantage of their authorization to access the network, to attack leaving limited suspicious traces behind. The traditional knowledge-based IDS must now give way to intelligent, data-driven systems. These systems must be able to learn from data, note any patterns and statistical regularities, and detect intrusions not by comparing traffic to a rigid set of signatures and patterns, but by identifying them as anomalies that stand out from the patterns of normal traffic.

Machine learning systems are flexible, robust, and scalable, and in many ways can meet the unique demands of IoT security better than any other approach currently used. They learn by examples from the available data, extract patterns on their own, and generate classifications on new data without needing hard-coded patterns to be programmed into them by humans. However, even traditional machine learning models are becoming hard pressed to meet the challenges of modern IoT architectures and the new types of threats and network weaknesses that arise. These models need hand-crafted features, they cannot take advantage of big data, and their accuracy often hits a threshold. Deep learning models, on the other hand, extract features on their own, can process vast amounts of data, and surpass traditional machine learning in performance and accuracy.

The scope of this study is the deep learning approach to IoT intrusion detection. Our motivation for conducting a comprehensive survey on the application of deep learning for IoT intrusion detection is twofold. First, the significance of the subject. Simply stated, the IoT paradigm is the next frontier in technology and engineering, cybersecurity poses a major challenge for it, and deep learning may well be the single best solution for the unique characteristics of the problem. Second, the previous reviews on the subject (see next section) discuss either deep learning in the context of general cybersecurity and conventional networks or, in the case of focusing on IoT security, they tackle general machine learning and/or other approaches. Hence, our unique contribution is a focused and detailed analysis of what deep learning methods have been proposed for intrusion detection, specifically targeting IoT environments. Within the vast fields of cybersecurity and networked systems, we focus on the topics of intrusion detection, IoT, and deep learning, and we offer an exhaustive, detailed exposition of all the research that has been conducted in this area.

In this survey we analyse the various solutions that have been proposed, classified by model. We assume the reader is versed in the basic concepts of both deep learning and IoT architecture, and we structure our survey as follows. Section 2 presents the research methodology we followed in conducting this research. In Sect. 3, relevant terms are explained and related surveys are reviewed. Section 4 presents the main contribution of this study as it elaborates on the deep learning models and variations proposed for IoT intrusion detection. In Sect. 5 we discuss some of the challenges in applying deep learning for IoT IDS and we offer some recommendations. In Sect. 6 we reach to our conclusions.

2 Research Methodology

The research methodology that has been followed in this paper included an initial search on Scopus with the following query:

```
TITLE-ABS-KEY (deep AND learning AND iot AND intrusion
AND detection)
```

This search, conducted in June 2020, yielded about 90 papers. To pursue a more exhaustive exploration of the literature we went beyond Scopus and coupled it with a search in Google Scholar, using the same terms. Then, our probing branched out in two parts. First, we conducted a series of secondary, more detailed Google Scholar

searches, using as terms specific deep learning models together with the phrase 'IoT Intrusion Detection' (i.e. 'Deep Neural Networks IoT Intrusion Detection', 'Convolutional Neural Networks IoT Intrusion Detection', 'Recurrent Neural Networks IoT Intrusion Detection', etc). Second, we sought out and gathered any relevant works referenced in the papers from our primary Scopus search. These works were examined and properly evaluated. All in all, this wide-reaching reconnaissance brought 150 papers under our scrutiny, which we screened for relevancy.

As criteria for the filtering process, we selected those works that explicitly focused on all three of the following axes: (a) deep learning, as opposed to general machine learning and pattern recognition, (b) IoT, as opposed to conventional networks, and (c) intrusion detection, as opposed to general cybersecurity and relevant issues. This process yielded around 100 papers, those that are included in this systematic review.

Early on in our exploration of the bibliography we noticed that this field is quite new. The first publications start from 2016, with the volume of research increasing steadily in the following years (Fig. 1). Another observation is that a portion of the studies we examined was more or less applicable to intrusion detection in both IoT environments and regular networks. We project that with the massive expansion and development of the IoT landscape currently taking place, (a) the number of relevant publications per year will increase, and (b) given the peculiarities of the IoT ecosystem, cybersecurity solutions designed for conventional networks are not easily adopted in IoT environments.

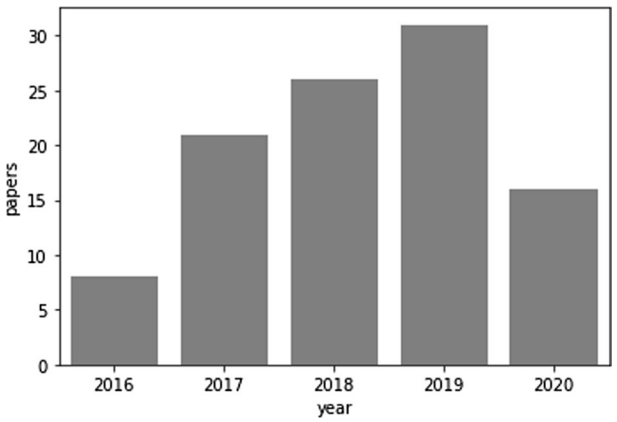

Fig. 1 Number of relevant publications per year, on the subject of IoT intrusion detection based on deep learning (statistics from June 2020)

3 Background

Although IoT can be seen as an extension and scaling-up of traditional networks and the Internet, novel features will often emerge when a system becomes more complex. The current cybersecurity paradigm will need to be reexamined and revised to account for IoT's novel features, which means these features must be defined. Deep learning models are an improvement upon traditional machine and statistical learning techniques, and a comparative analysis of these will shed light on the unique attributes of deep learning, suggesting avenues for its deployment in intrusion detection. Therefore, in this section we lay out some background information to help put the results of our survey in context.

We start with the foundations of IoT architecture, IoT intrusions, and IDS. Then, a discussion of traditional machine learning and deep learning will reveal how deep learning can contribute to IoT intrusion detection. Finally, previous reviews on the subject are examined.

3.1 IoT Architecture and Security Aspects

For the purposes of this study, we adopt the general definition of IoT as a network of physical objects [87]. Subsuming conventional networks of servers and client computers, IoT is the domain where the physical and the cyber worlds interact with each other. Sensors, actuators, smart devices, distributed applications, networks, protocols, and the Internet, all converge into a smart grid through which we both monitor the physical world and act on it. From a baby-cam connected on the Internet to industrial pipelines monitored through networked systems, IoT is a pervasive force

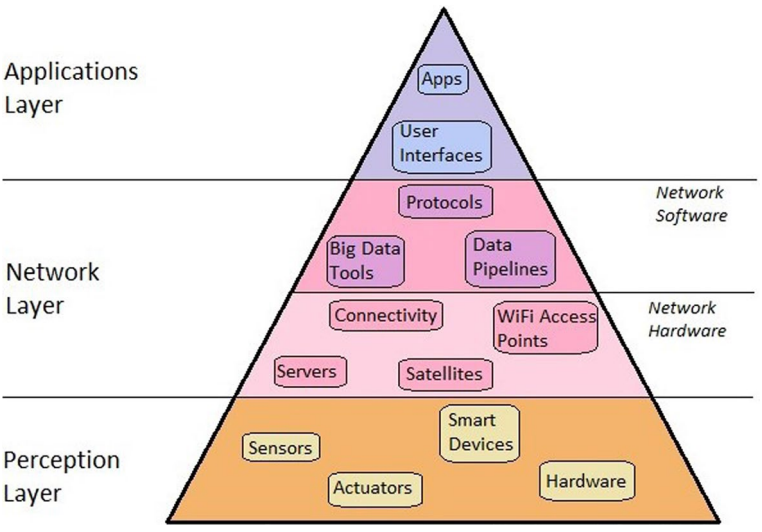

Fig. 2 IoT architecture

that touches every facet of our lives. On an abstract level, IoT can be seen as consisting of three layers: the perception layer, the network layer, and the application layer (Fig. 2).

The perception layer is mostly comprised of sensors that collect data from the physical environment. It also includes actuators and smart devices. Most of these physical objects have limited computation and power resources. This heterogeneity of devices and objects must be integrated into a common network, therefore they must adhere to certain standards and protocols in order to interconnect, ideally, in a plug-and-play fashion. A big portion of the data circulating IoT are generated at this layer.

The network layer defines protocols that smart objects use to connect to each other, using various conventional and IoT specific communication protocols. This can be a lossy and noisy environment that receives low-power communications from resource-constrained objects. The data may be transmitted and re-transmitted multiple times, to and from numerous intermediate relay nodes, before they reach their destination, or a data storage location. Some data analytics is also applied on this layer, to keep track of the traffic.

The application layer consists of the applications, user interfaces, frameworks, APIs, that users have at their disposal, either to process the data generated at the perception layer and control actuators, or for further functionalities and services. In the metaphor of IoT being the domain where the physical and cyber worlds meet, the perception layer can be seen as the physical world, the application layer as the cyber world, and the network layer as the interface between these two, as depicted in Fig. 2.

Due to the vast number of inter-connected devices, there are numerous possible vectors that attackers can exploit to reach their targets, practically from anywhere in the world. Being constrained in their computational resources, and largely heterogeneous devices cannot easily adopt support sophisticated preventative measures and are potentially exposed to cyberattacks. Many systems in IoT are distributed in nature, where devices communicate directly with each other without supervision from a centralized location, thereby a compromised device can be easily deployed to attack others.

3.2 IoT Intrusions

Fernandes et al. [26] explored the differences and similarities regarding cybersecurity, between IoT and conventional IT devices. In IoT, intrusions and threats can come for each one of the three layers and from various groups of threat agents. On the physical layer we can have physical attacks, node capture attacks, and user tracking. On the network layer we can have wireless-based attacks (e.g. DoS, wormhole, man-in-the middle, eavesdropping), Internet attacks (e.g. Hacking and intrusions, DDoS), routing attacks, and attacks on network ports. On the applications layer we are threatened by attacks like malware, spyware, ransomware, DoS, viruses, spoofing, eavesdropping, and others. Distributed attacks, particularly, can have a large

Table 1 Types of attacks and intrusions afflicting each layer of the IoT architecture

Layer	Type of attack	Example of feature bases
Perception/physical layer	Physical attacks, node capture, user tracking, social engineering to access devices, side-channels, radio frequency jamming, sleep deprivation attack	PHYL specific features, such as signal voltage, received signal strength indicator, signal to noise ratio, bitstream patterns
Network layer	DoS, DDoS, spoofing, traffic analysis, wormhole, man-in-the-middle, eavesdropping, sybil, cloning, sinkhole, hacking, viruses, identity theft	Network specific features, such as source IP address, destination IP address, transaction bytes, time to live, packets retransmitted, packets dropped, packet size
Applications layer	Malware, spyware, ransomware, buffer overflow, DoS, DDoS, viruses, botnets, phishing, SQL injection, spoofing, privilege escalation, eavesdropping	Application specific features, such as transaction protocol, number of connections, rate of connections, connection duration, PDU sequence number, login attempts, status flags
Multi-layer	Man-in-the-middle, traffic analysis, side-channels	(*all*)

impact, and DDos attacks are widely used [55]. Table 1 lists the various types of attacks that can afflict each layer, as well as multi-layered attacks. The term "attack" is used loosely, as many of these intrusions are passive and not perceived as attacks per se (e.g. traffic analysis, where intruders passively monitor the traffic, waiting for an opportunity to loot valuable data, and the administrators may never know the network was compromised). Moreover, Table 1 lists the representative features of each layer, which are adopted for intrusion detection [1, 31, 50]. Evidently, features at the perception/physical layer concern characteristics of the transmitted and received signal, at the network layer features are associated with properties of packets and dataflows identified through their inspection, whereas features at the application layer are specific to protocols which are under inspection of the IDS, such as HTTP, FTP, SSH, CANbus, Modbus and other SCADA protocols, etc.

Attacks on any of the three layers can potentially compromise the confidentiality, integrity, availability, and privacy of the data, causing harm to organizations, individuals, nations. Extensive research has been conducted on how traditional cyberattacks morph and adapt to the IoT, and what new challenges lie ahead [3, 86, 99]. Notra et al. [82] have demonstrated how easily IoT devices can be compromised, and Garcia-Morchon et al. [30] provide a taxonomy of the various attacks that have been used.

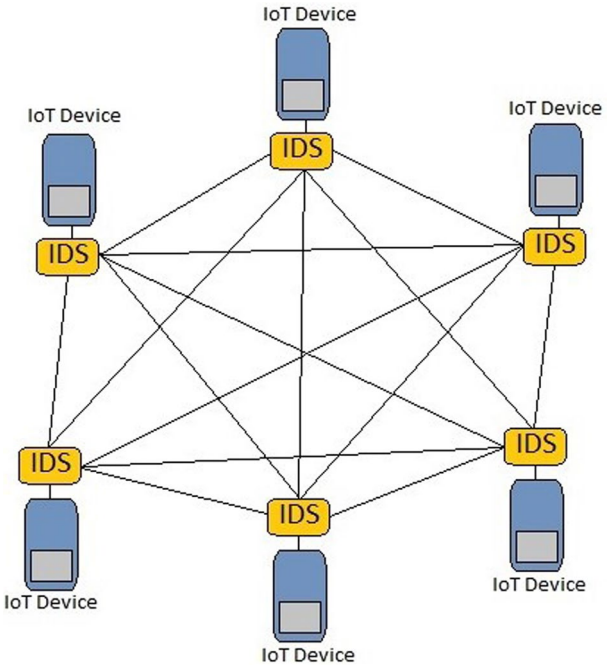

Fig. 3 Placement of IDS in distributed architectures. Devices communicate directly with each other, and each one is responsible for its own security

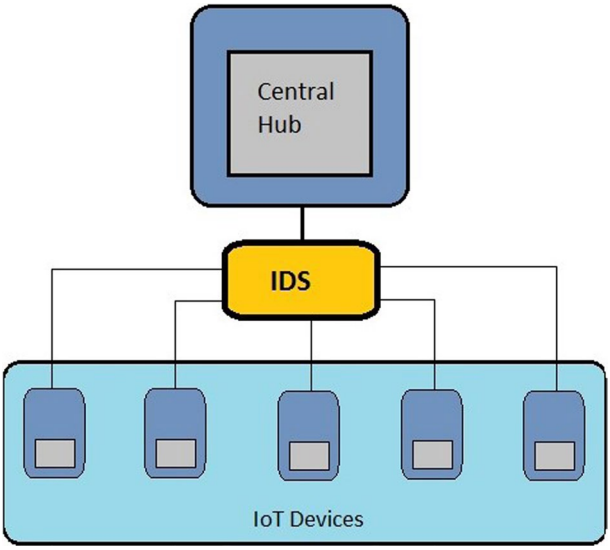

Fig. 4 Placement of IDS in centralised architectures. All traffic passes through a central hub that monitors the system and ensures only healthy packets circulate

3.3 IDS in IoT

According to their placement strategy, IDS are categorized as either distributed, centralized, or hybrid. In the distributed topology (Fig. 3) IDSs, or agents thereof, are placed in each physical device, depending on the available computational resources of the device, and nodes may monitor their neighbors as well. In the centralized topology (Fig. 4) the system is placed in a specific location, such as a network host, monitoring all traffic and scrutinizing packets for any suspicious activity. Hybrid placement combines aspects of the distributed and centralized placements, either to correspond to a certain network's architecture, or to exploit the strengths of the two types of placement and alleviate their weaknesses. Obviously, the choice of IDS placement in a given system will consider the environment's peculiarities. Where devices communicate directly with each other distributed approaches are called for, and where all traffic must pass through a central hub a supervising IDS must be placed to monitor and intercept threats.

The distributed approach has been recognized as particularly significant. Due to the large-scale nature of the IoT, over-reliance on centralized control is generally seen as non-ideal, and a distributed approach to problem solving and communications, whenever possible, is pursued. Also, as an increasing amount of data is being stored though the cloud in centralized locations, cyberattacks at cloud-based systems increase. With the emergence of cloud-IoT and fog-to-things networks, a centralized IDS could be attacked from multitudes of devices and multitudes of physical locations. Distributed DoS attacks have often compromised some of the most robust systems. Hence, research on systems for distributed intrusion detection on edge devices

is being conducted, some of which oriented towards deep learning based solutions [2, 22, 112].

According to their detection method, IDS can be classified as knowledge-based (specification-based or signature-based), anomaly-based, and hybrid.

- Knowledge-based detection systems, store patterns and signatures in databases and compare incoming traffic with them to determine whether there is malicious activity. They excel in detecting known attacks, but they're computationally inefficient and cannot detect new attacks, since there are not any patterns or indicators stored in the database for attacks that have not been invented yet. These two weaknesses make knowledge-based detection systems inappropriate for the dynamic and fast expanding IoT ecosystem, and although research on knowledge-based systems for IoT intrusion detection has been conducted in the past [24, 46], machine learning and deep learning approaches quickly gain traction and become more prevalent.
- Anomaly-based detection systems, usually relying on machine learning and statistical techniques, identify statistical regularities from normal traffic and single out any anomalous packets that deviate from the normal patterns. They can detect novel attack methods, but may suffer from high false positive rates. Also, traffic flagged by the system as anomalous may not constitute malicious activity but may occur due to a technical disturbance, such as a malfunctioning sensor.

The characteristics of knowledge-based and anomaly-based IDS are summarized in Table 2.

3.4 Traditional Machine Learning and Deep Learning Approaches

Machine learning models do not need human experts to manually define rules on what constitutes benign or malicious network traffic. In the case of supervised learning, they are trained on annotated examples, historical data with class labels for each instance, and through automated fine-tuning of the model's parameters they learn to classify new data into the assorted classes. Most of the conventional machine learning algorithms have been applied for IoT IDS, such as: Association Rules [106], Support Vector Machines [37, 64], k-Nearest Neighbors [61], k-Means Clustering [79], Logistic Regression [33, 90], Decision Tree [27, 68], Random Forest [60], Naïve Bayes [72], Neural Networks [36, 95], Multi-Layer Perception (MLP) [56], Extreme Learning Machines [89].

Deep learning is quickly replacing shallow machine learning due to its superior performance, and as expected, this trend applies to IoT intrusion detection as well. Deep learning is a subfield of machine learning that consists of more complex models, mostly deep neural networks and a number of variations, but due to its unique characteristics it is treated as a separate field.

Traditional machine learning is entirely dependent on extensive feature engineering to select which features of the data will be taken into account by the model

Table 2 Characteristics of knowledge-based and anomaly-based IDS

	Knowledge-based detection	Anomaly-based detection
Features	Hand-crafted patterns rigid/deterministic logic	Learning from data probabilistic classification
Pros	Low false positive rate, detects known attacks with high precision Interpretable	No expert knowledge required Low false negative rate Detects unknown attacks Can detect insider attacks
Cons	Expert knowledge required Efficiency depends on system complexity High false negative rate Only detects known attacks Sensitive to attack from inside the organization	High false positive rate Efficiency depends on model complexity Largely uninterpretable (black box)

and contribute to its training and classification. Depending on which features will be used the performance of the model will vary dramatically. This dependence on feature engineering and the effects it has on the model's performance has been demonstrated in the context of IoT IDS [12]. Unfortunately, in IoT intrusion, given the complexity of the subject, there are quite complex feature selection considerations [80]. Depending on which features you select, each feature has its pros and cons. Deep learning needs no hand-crafted features but automatically extracts the most significant. Representations are learned directly from raw data. Feature engineering, considered an especially important, difficult, and time-consuming task in conventional machine learning, is completely unnecessary in deep learning.

Deep learning yields better performance than traditional machine learning. This is due to the number of parameters that need to be calibrated through training in order to fit the input data. The structure of deep learning models is complex and they have thousands, even millions, of parameters. Theoretically, such a high number of parameters may lead to overfitting. In practice though, deep learning is effective, since each model is trained for a narrow domain, with limited need for generalization.

Stemming from the high number of parameters, deep learning needs huge amounts of data to be trained effectively, whereas shallow machine learning models can be trained with fewer data. In the context of IoT intrusion detection this may or may not put deep learning at a relative disadvantage. There may be no training data available for a new device or protocol, but this will be an equal shortcoming for traditional machine learning as well. Therefore, even though shortage of training data does pose a problem in deploying deep learning, this does not make it any less preferable than traditional machine learning.

Another issue with the higher complexity of deep learning models is their diminished, compared to shallow machine learning, time efficiency. After having emphasized, in the previous section, the significance of distributed IDS solutions for resource-constrained devices, we now face the high computational expense of deep learning. Thousands, or millions, of parameters are trained iteratively for numerous epochs, the computations being repeated for each sample in a dataset that may contain hundreds of thousands of samples. The drawback, though, turns out to be mostly theoretical. There is no need for exceedingly complex models in IoT intrusion detection. The infamous complexity and computational overhead of deep learning mostly applies to tasks like vision and natural language processing. Detecting anomalous patterns in network traffic is orders of magnitude simpler than performing object segmentation in 36fps video. Also, training a neural network is the most time-consuming and demanding part of the process, and this can be done on a central workstation. After the model has learned its parameters, it can be transferred to an IoT device. Classification is performed by passing the data once through a set of straightforward numerical computations, without having to go through the iterative process of back-propagation and parameter fine-tuning. Indeed, as we shall see in the main section of this paper, deep learning models have been deployed, tested, and recommended for distributed IDS run on resource-constrained devices.

3.5 Related Work

In recent literature, there have been a few endeavors to survey the latest approaches related with deep learning, IDS, and IoT.

Zarpelao et al. [124] provide a high-level view of IDS in IoT, classified according to their placement strategy (centralized, distributed, and hybrid), detection method (signature-based, specification-based, anomaly-based, hybrid), security threats being addressed (conventional attack, man-in-the-middle, routing attack, DoS), and validation strategy (hypothetical, empirical, simulation, theoretical, none), and give an overview of what studies have been conducted for each subcategory. Their work does not touch upon machine learning, but focuses on how IDS proposals approach the IoT architecture.

Amanullah et al. [11] focus on the current technologies for deep learning and big data analytics, the various frameworks and APIs, and provide a general introductory exposition of deep learning and IoT security. Yet, the authors do not discuss the various models and solutions that have been proposed for IoT intrusion detection.

Immamverdiyev and Abdullayeva [44] give an overview of the deep learning models that have been applied in intrusion detection, but their study pertains to traditional cybersecurity, and not to the IoT ecosystem.

Liu and Lang [62] also summarize machine and deep learning for IDS in traditional cybersecurity, without supplying comprehensive information on the deep learning models and their different variations that have been proposed for IDS, and without addressing the research that has been conducted specifically on IoT intrusion detection. They provide a taxonomy of IDSs based on the source of data used, which is relevant to the data-driven approach of machine learning and deep learning.

Hussain et al. [40] focus on IoT architectures and technical specifications, and on the kinds of threats, weaknesses, and challenges for security that arise from these characteristics. After a brief description of machine and deep learning techniques, they list the proposed solutions, classified according to the type of attack or threat they address. Their survey does not focus on deep learning, but on the types of attacks commonly found in IoT environments, and briefly lists some machine learning methods proposed for each category.

Ferrag et al. [28] focus on deep learning intrusion detection on conventional environments, and not on IoT. However, their survey is accompanied by the results of an experimental comparison the researchers conducted of seven deep learning models on the Bot-IoT dataset. They tested the models both for binary and multi-class classification, on IoT intrusion data, and reported the results in detail, therein the survey is useful, although the literature review only deals with general intrusion detection.

It is noted that most relevant surveys either review deep learning in the context of traditional cybersecurity or review IoT intrusion detection without focusing on deep learning. Those that do combine IoT and deep learning seem to only give a general overview of how deep learning works and what it can do for IoT intrusion detection, without providing a comprehensive survey on the specific proposals that have been made. Others combine deep learning with traditional machine learning, failing to offer a focused, thorough analysis of the former.

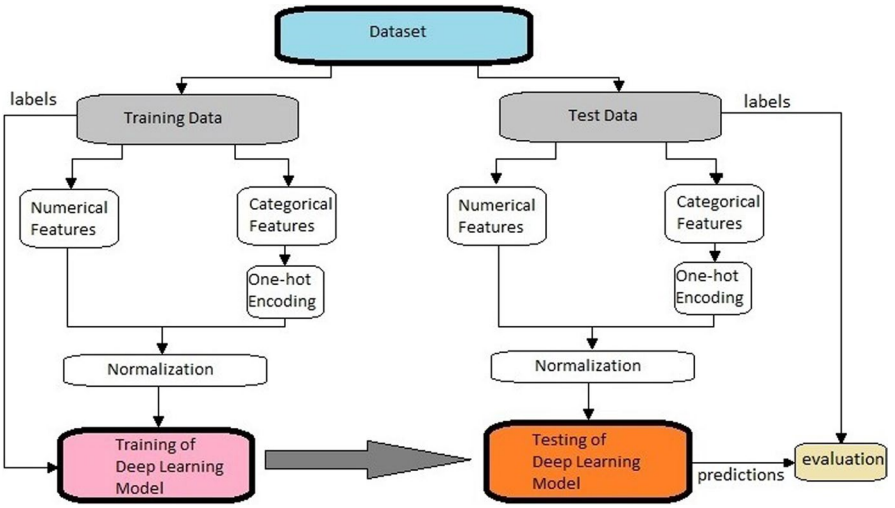

Fig. 5 Model training and testing

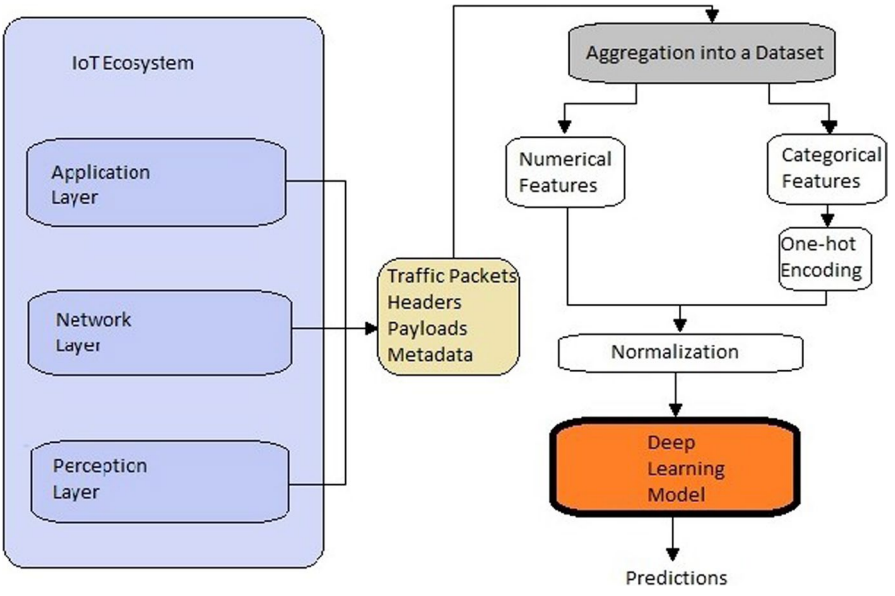

Fig. 6 Model deployment

4 Deep Learning Models for Intrusion Detection in IoT Systems

This section is the main contribution of our study. It presents the research conducted on deep learning in IoT intrusion detection. As depicted in Fig. 5, a deep learning

model is first trained with data from the network in question. It learns the distributions of the various classes (i.e. benign and malicious traffic), and during testing we determine how effectively it can discriminate among the classes, going back to training and hyperparameter tweaking if its performance was suboptimal. The training-test cycle continues, and experimentation with different models and architectures ensues, until the best model is ready for deployment (Fig. 6). In deployment, the model intercepts traffic and generates predictions as to whether a packet is normal or suspicious; in the latter case the system administrator is notified, or the packet may be automatically blocked,

What follows is a review of the specific deep learning solutions that have been proposed for IoT intrusion detection, grouped by type of model.

4.1 Deep Neural Networks for Intrusion Detection in IoT

Kang et al. [47] used a Deep Neural Network (DNN) (Fig. 7) to detect intrusions in vehicular networks in a distributed architecture. To simulate the attack scenario, they injected malicious data packets into an in-vehicular controller area network (CAN), which is the standard protocol in automotive industries for the interconnectivity of vehicular electronic and informational systems. The DNN takes the feature vectors and classifies the packets either as normal or malicious, after calculating the probability value for each class. During experimental tests, this approach was shown to achieve a detection ratio of 99%, with false positive rate of less than 1–2%.

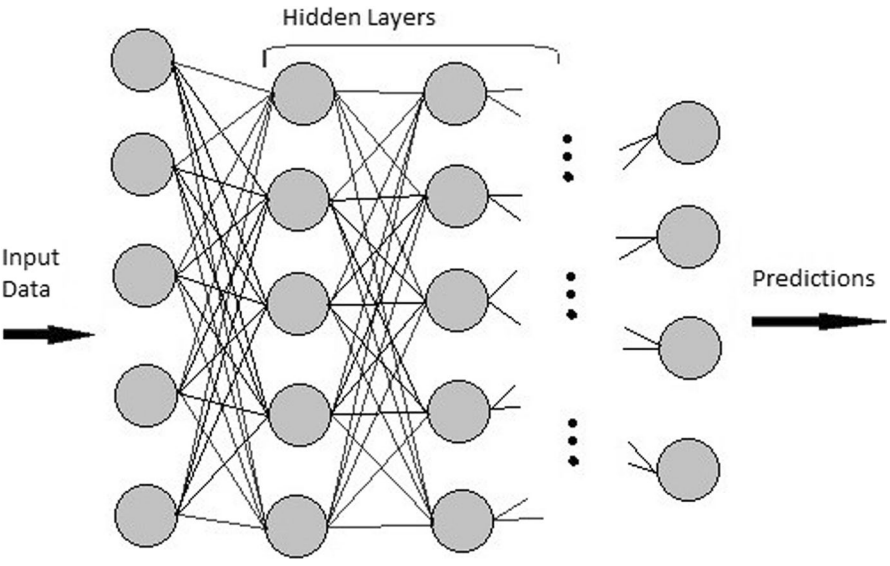

Fig. 7 A deep neural network architecture. Adding hidden layers makes the classifier able to approximate functions of increasing complexity

Ma et al. [67] combined DNNs with spectral clustering. The heterogeneity and high variance of network traffic may result in low accuracy in IDS. The researchers, through clustering traffic data based on characteristics, subdivided the original dataset into 6 highly homogeneous subsets. A separate DNN model was trained on each subset. Their method, tested on the NSL-KDD and the KDD99 datasets, reached an accuracy of 92.1%.

Some IDS implement a protocol analogous to a human-like workflow, where alerts are algorithmically ranked and human analysts evaluate the top-ranking alerts. In this way workload is reduced to a minimum, while false positives are kept low. McElwee et al. [71] developed a centralized DNN-based alert filtering method to deploy in such a human-machine interactive system. After collecting the log data, the researchers deployed a DNN model to identify high-priority security-related alerts in the logs. Security experts then analyzed the red-flagged events, the results of the human expert analysis being formulated as training data for the DNN, generating a positive reinforcement cycle. The resulting system reduced analyst workloads and rendered security analyses more time-efficient.

Tama et al. [107] used DNNs for detecting attacks on the network layer of IoT. They trained the models on three novel benchmarking datasets in wired and wireless network environments: UNSW-NB15, CIDDS-001, and GPRS. These datasets contain more modern examples of normal traffic and malware attacks than some of the older benchmark datasets such as KDD99, and have a more uniform distribution of classes. After parameter fine-tuning and experiments, the DNN model reached performance of up to 99.99% accuracy on the CIDDS-001 dataset, and 94.04% on UNSW-NB15.

Yavuz et al. [121] experimented with DNNs for centralized IDS on a range of custom-made datasets. First, they normalized the features by applying quantile transform and min-max scaling. Then, they performed feature selection through a combination of random forest, Pearson coefficient, and visual inspection through histograms. Finally, a DNN with five hidden layers was trained with the custom data and achieved accuracy ranging, depending on the dataset, from 94.9 to 99.5%.

Darbandi et al. [19] used Feed-Forward Neural Networks for real-time stability assessment in cyber-physical systems. The FNN predicts transient stability and out-of-step conditions (OOS) for the network. Cyber-attacks and other contingencies register as anomalies by the systems which, in experimental testing yielded up to 99.2% accuracy.

4.2 Convolutional Neural Networks for Intrusion Detection in IoT

Other than being the standard choice for computer vision tasks, Convolutional Neural Networks (CNN) (Fig. 8) have been used in time-series anomaly detection [78] and, consequently, for intrusion detection. In 2017, Wang et al. [118] attempted malware detection with CNNs by converting traffic data into images. It was one of the first efforts to apply CNNs to non-image data and, in particular, to malicious packet identification. Their method had difficulty detecting unknown attacks, but later, Munir et al. [77] reported encouraging results in a study with CNNs used for

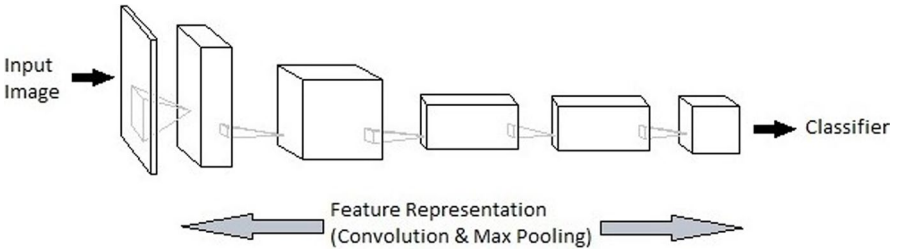

Fig. 8 Convolutional Neural Networks were originally designed for vision tasks and image recognition. By passing the image through layers of convolutions and max-pooling, higher-order features are extracted

distributed, unsupervised anomaly detection in streaming IoT-based sensors data. The researchers combined CNNs with the ARIMA statistical model for time-series forecasting and fused them into a network that learns when to rely on the CNN, and when on the ARIMA model, for the classification.

Of course, applying CNNs on non-image data entails transforming the data. A greyscale image is represented by a two-dimensional matrix with values either between 0 and 1, or between 0 and 255. Thus, a CNN-based IDS requires the dataset to be min-max scaled, and each vector reshaped into a 2D matrix, padding with zeros if needed. The computational overhead is minimal, consisting of 2-3 simple operations for each sample. For higher efficiency, the CNN could be implemented with one-dimensional convolutional layers, removing the need of reshaping vectors into matrices.

Potluri et al. [88] experimented with a CNN model on the NSL-KDD [7] and the UNSW-NB 15 datasets. The feature vectors in the datasets were converted into images for the CNN to process. To convert the vectors into 8-by-8 pixels images, categorical features were one-hot encoded and 8-byte chunks were transformed into one pixel. A three-layer CNN was trained on these images and compared with two other CNN models, GoogLeNet and Resnet-50, performed best and reached 91.14% accuracy on the NSL-KDD dataset, 94.9% on UNSW-NB2015. Teyou et al. [110] also tested a CNN on the NSL-KDD with good results, in a study for intrusion detection in cyber-physical systems. On the Bot-IoT dataset, Susilo et al. deployed a CNN that reached 91.27% accuracy, surpassing other shallow machine learning models with which it was compared [104].

CNNs can be used for intrusion detection via extracting features from logs [62]. Using a sliding window over the extracted log features takes advantage of the contextual features contained, anomalies stand out, and processing takes place in streaming or near-streaming speed. Network intrusions may leave traces of system calls and applying classifiers to analyze these system calls can identify suspicious events. Tran et al. [113], used this approach for a CNN method to analyze, in a centralized architecture, system calls for intrusion detection. Since every operation is logged in the system calls, an intrusion would be filed in its entire process. The model was trained and tested on the NGIDS-DS and the ADFA-LD datasets,

with good results, extracting features through a sliding window and then making the classifications.

Huong et al. [39], proposed a methodology where the log information of an IoT system such as address, location, service, etc., is stored into a dataset. Next, the dataset is pre-processed and converted into a matrix, like an image, and used for training a CNN that reaches an average accuracy of 98.9%.

Zhao et al. [131], used CNNs for intrusion detection and monitoring via Device-Free Localization (DFL), where the system tries, through a centrally located IDS, to pinpoint the physical location of the IoT device used wirelessly for the intrusion. The researchers formulated the DFL problem as an image classification task by converting the received-signal-strength (RSS) into a matrix. The RSS of the device is affected by the movements of its human carrier and is easily acquired through sensors. As the intruder enters within range of the DFL system's monitoring, or whenever he changes his location, the signals received from the attacking devices change. Through sampling, these changes form a matrix that is fed into a Background Elimination CNN (BE-CNN). The model outputs a location estimation for the target/intruder. Given enough sensors to capture the RSS signal of the intruder, the system was shown to yield up to 100% localization accuracy.

CNNs have also been applied for feature extraction in IoT intrusion detection research. The usual configuration has CNNs extracting features, then traditional machine learning models doing the classification. In [75], the authors used this approach to address the fact that most anomaly-based IDS focus almost exclusively on packet header information, while the omitted payload information can also prove useful. In their study, after encoding the payloads with skip-gram word embedding, the researchers used a text-CNN to learn representations of the payloads. Combining these content features with statistical features extracted from packet headers, IPs, and ports, they trained a random forest that reached 99.13% accuracy on the ISCX 2012 dataset.

Bassey et al. [13], in a similar vein, used CNNs not for the classification but for the feature extraction stage of a centralized framework designed to detect intrusions from unauthorized IoT devices. In such a situation, intrusions come from devices that did not take part in the training data. Given that the model must classify the intrusion in one of the classes it has been trained for, the classification is bound to fail. To tackle this, the researchers applied the CNN to feature extraction followed by dimension de-correlation and clustering. The unauthorized IoT devices are detected by their RF signal, like a fingerprint, in a technique called supervised bootstrapping [83]. The CNNs extract the features, followed by feature de-correlation, then t-SNE for dimensionality reduction, and finally, DBScan clustering, which is suitable for unknown number of clusters, to cluster and identify IoT devices. This method was shown to identify new devices, which were not present in the training data. In [18] the authors took the CNN-traditional ML approach to experiment with few-shot learning, as data shortage poses a challenge for IoT applications. Few-shot learning can tackle scenarios where the data is imbalanced and some classes have a limited number of instances. In this particular study, implemented on the KDD 99 and NSL-KDD datasets, the CNN is trained first, then its outputs are fed into a support vector machine and a k-nearest neighbor classifier for few-shot intrusion detection.

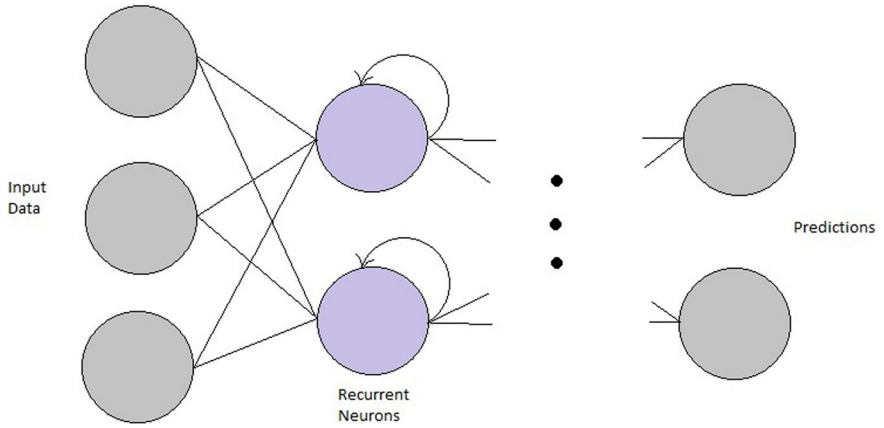

Fig. 9 In Recurrent Neural Networks, neurons take their previous states as input, giving them a type of memory. Thus, RNNs can extract features from the temporal dimension of the data

Liu at al. [63], proposed a two-level anomaly detection for networked Industrial Control Systems using CNNs and a process state algorithm. In the first state, the CNN extracts features from the data and detects anomalies based on the wider context. In the second stage, the process state algorithm, designed according to the characteristics of the industrial system's stability, takes as input the features extracted from the CNN and detects whether the data corresponds to the system's normal state. In this way the system has been shown to identify unknown and zero-day attacks successfully.

4.3 Recurrent Neural Networks for Intrusion Detection in IoT

Recurrent Neural Network (RNN) (Fig. 9) and its variations, Long Short Term Memory RNN (LSTM-RNN) and Gated Recurrent Unit RNN (GRU-RNN), with their capacity to process time-series data and take into account their context in time, are being widely used for anomaly detection and intrusion detection.

Taylor et al. [109] proposed an RNN-based anomaly detector scheme to detect attacks against network-connected vehicles. The system is anomaly-based, deploying an LSTM to predict the values of new packet data based on past instances, packets with large errors being classified as anomalies.

In [54], the authors developed a system, that can be used in hybrid architecture, for the verification of a human user and the detection of intruders, based on the keystroke dynamics on devices by the user. The keystrokes are encoded as sequential data for LSTM and GRU-type RNNs to classify as either genuine or impostor. In experiments with the Keystroke Dynamics Benchmark Dataset, the models successfully detected suspicious activity.

Pajouh et al. [32], proposed a distributed RNN-based malware detection method on IoT environments. The system focuses particularly on ARM-based IoT

applications, since IoT devices with ARM processors almost monopolize ecosystems based on Unix System V. Intrusion detection is performed by analyzing the OpCodes of IoT applications. After extracting the OpCodes, feature vectors are obtained though the TF-IDF algorithm, the dimensions of the data are reduced through PCA, and finally, the RNN classifies the samples in two classes, malicious and benign. Instead of the conventional RNN neurons, the researchers used the structure of bidirectional neural networks (BNN) [97]. In this architecture, a conventional RNN processes time-sequences both forwards and backwards. Since these two directions are entirely independent from each other, a BNN model is trained as though training a conventional RNN. One difference is the requirement for additional weight-updating computations during back-propagation. This model reached 98.18% accuracy, surpassing many shallow machine learning models that were used for comparison.

McDermot et al. [70], tested a Bidirectional LSTM (BLSTM-RNN) to classify botnet activity in consumer IoT devices, for distributed deployment. The training data are comprised of attack packets converted, through word embedding [74], into tokenised integer format, with output labels for four classes of Mirai botnet attacks. In comparison with a common LSTM, training the two models on a custom dataset, the BLSTM achieved better results, although with slight increases in overhead to each epoch.

Another Bidirectional LSTM proposal made by Radford et al. [92], was applied for session detection and was tested on the ISCX IDS dataset. The researchers clustered packets according to their IP addresses, encoded the resulting sessions with word embedding, then trained a bidirectional LSTM to detect abnormal sessions. The purpose of applying a bidirectional LSTM model was to learn the sequence features both in the forward and the backward time directions.

Hwang et al. [42], applied LSTMs to detect IoT intrusions based on the data packet level, as opposed to the flow level. Deep learning systems face challenges in real time data processing, as they need time to accumulate the packets into flows and extract features. Using the packet-level information can lead to processing time gains. In this study the researchers utilize a novel word embedding scheme to extract semantic meaning out of packets and train an LSTM to classify the packets as either normal or malicious. Examining all packets in a flow is not necessary for accurate intrusion detections. Just the first packets of the flow can be adequate for classification purposes [41]. This yields additional gains in computing time. In experiments conducted with the ISCX2012 and USTC-TFC2016 IoT datasets from Robert Gordon University, the LSTM reached accuracy up to 99.99%.

In [8], the two RNN variations, LSTM and GRU, are compared in the task of detecting attacks on the MQTT-IoT Protocol, in a centralized architecture. The GRU networks, first proposed in [16], are faster and more adaptable than LSTMs in changes of time flow. Trained on a custom dataset with traffic of an MQTT-based network [98], the LSTM achieved 93.37% accuracy and the GRU 96.08%. LSTMs have also been shown to surpass traditional machine learning models on intrusion detection in fog-to-things communication [21].

In [9], a centralized fog-computing-based IDS for IoT is presented. The system uses two cascaded RNNs, each one configured with different hyperparameters, and

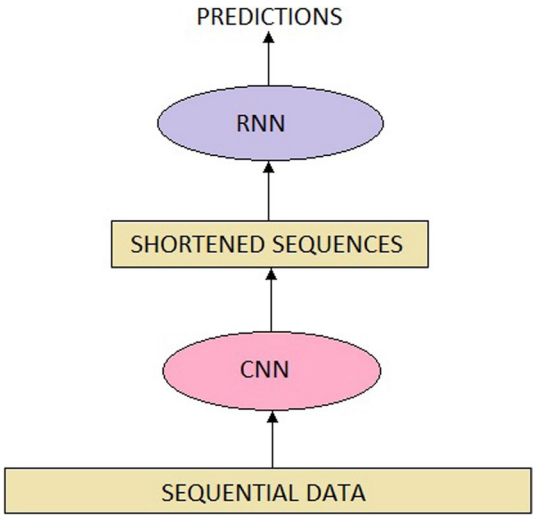

Fig. 10 CNN–RNN hybrid architecture

fine-tuned for specific types of attacks. If any of the two RNNs classifies the input instance as malign, an alarm is sent to the system administrator. The model was tested on an oversampled version of the NSL-KDD dataset, where some instances were duplicated to make for a more balanced distribution of the different classes and yielded 92.18% accuracy.

4.4 CNN–RNN Hybrids for Intrusion Detection in IoT

In the CNN–RNN hybrid model (Fig. 10), the CNN detects low-level patterns in the data, and the RNN detects mid-to-high level patterns. Hence, this approach can utilize information from multiple levels of abstraction.

In 2017, Wang et al. [117] recommended this hybrid architecture for a deep learning intrusion detection method, centrally placed, based on hierarchical feature representations. A session, in this scheme, contains the packet time sequence as well as the packet contents. The researchers applied a CNN to extract the low-level spatial features, then encoded the spatial features into sequential format and applied an LSTM to extract the high-level time features. The time features, therefore, result from a sequential arrangement of the spatial features. During experimental tests on the ISCX 2012 and the DARPA 1998 datasets, the hybrid model reached accuracy in the range of 99.92 to 99.96%.

Yuan et al. [123] experimented with a method to detect DDoS attacks, tested on the ISCX 2012 dataset. First, 20-dimensional features were extracted from the packets and encoded via the Bag of Words method. Then, after concatenating the packets in matrix form, a CNN was applied for feature extraction and an LSTM classified the sessions. Their model reached 97.606% accuracy.

In another CNN and LSTM hybrid model [49], the researchers proposed the Conv-LSTM network for intrusion detection. They, too, argue that although CNNs

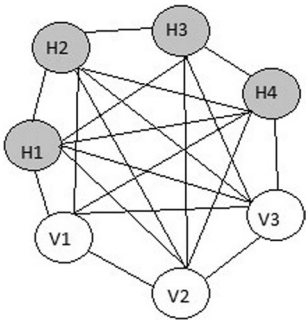

Fig. 11 In Boltzmann machines, both the visible and the hidden neurons are stochastic, interconnected bi-directionally, with the visible neurons being used as both inputs and outputs

effectively learn the local features, they miss long-range interdependencies. By adding LSTM layers after the CNN layers, global features are processed as well. Tested on the ISCX ID 2012 dataset, the model reached 97.29% accuracy.

4.5 Boltzmann Machines for Intrusion Detection in IoT

Restricted (RBN) and Deep Boltzmann Machines (DBM) (Fig. 11) can work in an unsupervised or semi-supervised fashion and were first utilized to detect intrusions by Fiore et al. [29]. Boltzmann Machines require no training labels and learn the joint probability distributions of the input data. Bengio [15] explained that a DBM (Deep Boltzmann Machine), trained on a large volume of unlabeled data and fine-tuned with a few labeled samples, yields good classification performance. Restricted BMs were used in [35] as a pre-training step for a Deep Belief Network.

In IoT intrusion detection, Boltzmann machines are mostly used as part of Deep Belief Networks (see next subsection), with a few cases of RBMs applied autonomously. When Dawoud et al. [20] proposed an SDN (Software-Defined Network)-based IoT architecture for enhanced security, they experimented with RBMs as a potential centralized IDS. On the KDD99 dataset their model had 94–95% accuracy. Otoum et al. [84] used RBM for a clustered IDS in wireless sensor networks, named RBC-IDS. The RBC-IDS system was tested with the Network Simulator-3 (NS-3) and KDD Cup 1999 dataset, and it achieved an accuracy rate of 99.91%.

Karimipour et al. [48] used RBMs in a hybrid, unsupervised approach for intrusion detection in large-scale smart grids. First, feature extraction takes place with symbolic dynamic filtering, a technique which can infer, via dynamic Bayesian networks, causal relationships between the smart grid's subsystems. Then, RBMs extract features out of the system's behavior and learn the attacks' patterns. Experimental tests with cyberattacks on the IEEE 39 bus system showed that the proposed approach yields almost 99% accuracy and 98% true positive rate.

In a study on intrusion detection for smart city networks [23], RBMs are proposed due to their unique capacity to learn from the raw, unlabeled data generated by sensors and smart meters. Used in conjunction with other classifiers, RBMs perform unsupervised feature extraction, and contribute to better classifications than the system would yield without the feature learning step.

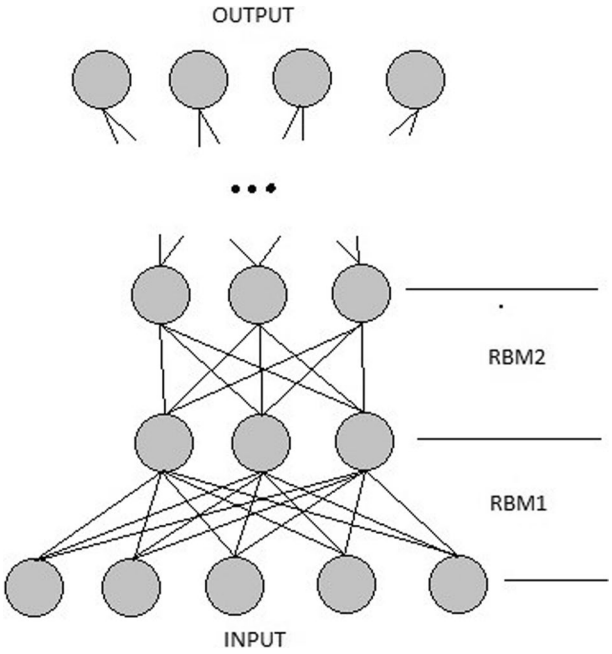

Fig. 12 By layering multiple Boltzmann Machines on top of each other results in Deep Belief Networks, which can be used for semi-supervised learning

4.6 Deep Belief Networks for Intrusion Detection in IoT

Training a stack of RBMs with a number of hidden layers, where each RBM's activation values on one RBM as the input for the next one, results in a Deep Believe Network (DBN) (Fig. 12). Just like RBMs, DBNs use both unsupervised pre-training and supervised fine-tuning. Each node is independent of other nodes in the same layer, and each layer is trained separately.

To detect false data injection attacks in smart grids, He et al. [34] designed a centralized intrusion detection model based on the extended DBN architecture. The researchers deployed conditional Gaussian-Bernoulli RBMs that extract high-dimensional temporal features. Experimental evaluations on the IEEE 118-bus power test system and the IEEE 300-bus system showed the system's accuracy to reach 98.5%.

Huda et al. [38], while researching centralized intrusion detection for industrial control systems in the cloud of things (CoT), designed a system that used DBNs for the classification of unauthorized traffic. To train their DBN model they generated log data by running malicious executables in a sandbox environment. In their experiments they concluded that 30–40 is the optimal number of hidden units that yields the best performance. Their models reached up to 99.8% accuracy, on a model with 34 hidden units.

In [38], the authors coupled DBN with support vector machines for detecting intrusions on SCADA network traffic. Rather than the traditional API-based or

signature-based malware detection techniques commonly used, the proposed architecture uses features extracted from both the network traffic and the packet payloads. The semi-supervised DBN, in this case, was used with the intention of addressing the complexity and big data challenges of the IoT ecosystem. Experimental tests with real SCADA network data showed the ensemble to be promising, and to outperform traditional machine learning techniques.

Aloqaily et al. [10] combined a DBN with a decision tree for detecting intrusions in network-connected vehicles. The DBN reduces the dimensionality of the data, while the decision tree classifies attacks. Traffic data were collected and preprocessed with a cluster-head selection mechanism and then, along the NSL-KDD dataset, used to train the hybrid system, yielding encouraging results.

Zhang et al. [130] used a genetic algorithm to train each layer of the DBN separately, then trained the last layer with back propagation. Tested on the NSL-KDD dataset, the model reached accuracy up to 99.45%. Manimurugan et al. [69] used a DBN on the CICIDS 2017 dataset, for an IDS based on hybrid placement strategy. After deploying a genetic algorithm to fine-tune the model's parameters, it achieved up to 99.37% accuracy. In most of these studies the researchers agree that, since in IoT labeled data are limited, the appeal of RBMs and DBNs come from their ability (along with Autoencoders—see next section) to be pre-trained with unlabeled data and then fine-tuned with a small volume of annotated samples.

4.7 Autoencoders for Intrusion Detection in IoT

Meidan et al. [73] used deep autoencoders (Fig. 13) on a novel network-based centralized anomaly detection method. The system responds to packets transmitted from compromised IoT devices. According to the researchers, IoT devices are easier to compromise than desktop computers, hence, the ever-increasing numbers of IoT

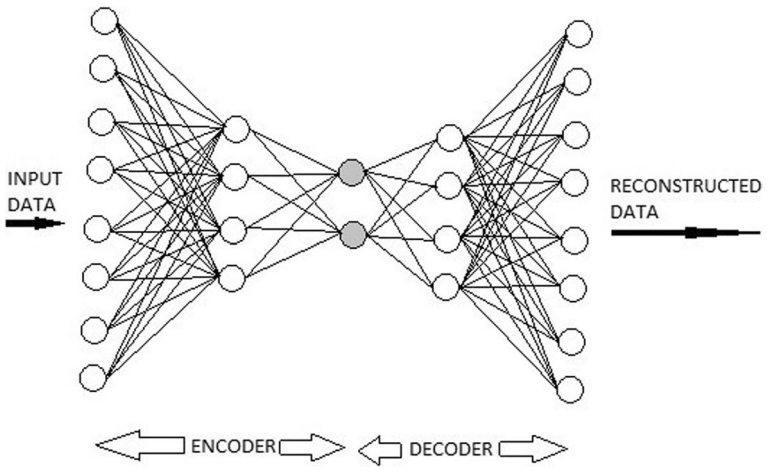

Fig. 13 Originally designed for data compression and denoising, autoencoders are widely used for anomaly detection. By passing the data through the bottleneck, high-level features are extracted

botnet attacks. The autoencoders take snapshots of the network's behavior and evaluate anomalous traffic. The method was tested on commercial IoT devices infected by Mirai and BASHLITE botnets. A separate deep autoencoder was trained for each one of nine IoT-based devices on benign traffic data. The autoencoder extracts statistical features of this benign flow, and when malignant data come through, they register as anomalies.

Lopez-Martin et al. [65] proposed a centrally placed model based on the conditional variational autoencoder (CVAE) [52, 103], named Intrusion Detection CVAE (ID-CVAE), that comes with two unique characteristics. First, it takes the labels of the classes inside its decoder layers, and second, it performs both classification and feature reconstruction. With a regular variational autoencoder we need a separate model to classify each class, each model deciding whether the input belongs to its class or not. With the ID-CVAE there's only a single model, which is trained with all the data and all the class labels in a single training step and has been shown to produce better classifications than shallow machine learning models. In terms of reconstructing incomplete features, the ID-CVAE will learn the distributions of the various features, and whenever it receives incomplete data it will recover the missing values. This ability was designed for the purpose of dealing with some of the inherent challenges in IoT environments, where faulty connections and sensor errors may distort some of the generated data, rendering them incomplete, or even invalid. Categorical features that carry an IoT device's state values are considered critical, and provided there are some related features present, the model can recover missing categorical values with accuracy of over 99%.

Autoencoders are also used for feature extraction, their outputs being fed into conventional machine learning classifiers. Yu et al. [122] proposed a system that uses a convolutional autoencoder for unsupervised feature extraction. The autoencoder extracts features from the payload of the packets, after the packets have been converted into images, to take advantage of the convolutions. The classification takes place using these learned features. On the CTU-UNB dataset the system yields precision, recall and F-measure 98.44%, 98.40%, and 98.41%, respectively.

Zhang et al. [127] used a sparse autoencoder for feature extraction and an XGBoost model for attack classification, on the NSL-KDD dataset. The NSL-KDD is a highly imbalanced dataset, and the researchers utilized the SMOTE technique to oversample the minority classes and divide the majority classes into a number of smaller categories that have similar distributions, resulting in a dataset with equally represented classes. The sparse autoencoder may surpass the original autoencoder in detecting unknown (outlier) samples, due to its sparsity constraint. This combination of sparse autoencoder feature extractor and XGBoost classifier achieved accuracy up to 99.96% on the NSL-KDD dataset.

Jin Lee et al. [58] applied the autoencoder-classifier combination in a study to develop a lightweight distributed machine learning system for resource-constraint IoT devices. They deployed a stacked autoencoder to extract features from the Aegean Wi-Fi Intrusion Dataset (AWID), and used the features to train a Support Vector Machine. The study showed that the stacked autoencoder succeeded in extracting the most significant features, enabling the classifier to reach 98.22% accuracy.

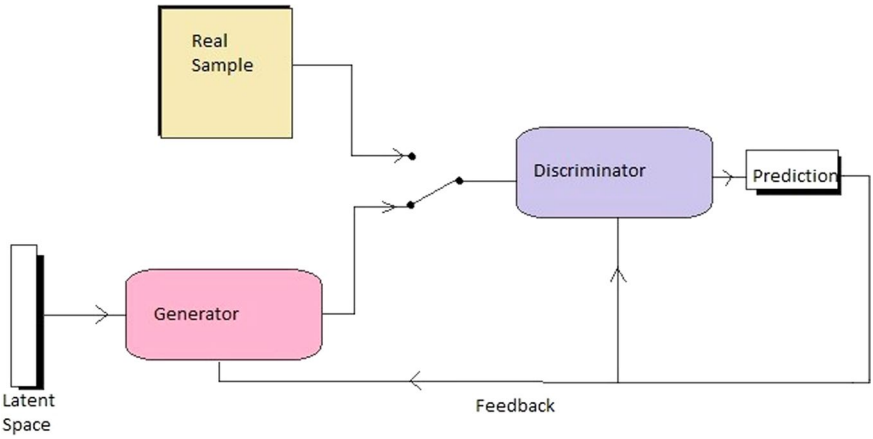

Fig. 14 Generative adversarial networks are used to test and improve the robustness of intrusion detection systems. The generator synthesizes fake but realistic data, and the discriminator becomes better at classifying them

4.8 Generative Adversarial Networks for Intrusion Detection in IoT

Belenko et al. [14] assessed the applicability of Generative Adversarial Networks (GANs) (Fig. 14) to detect intrusions in large-scale networks of cyber devices, showing how GANs may make up for the lack of comprehensive datasets. [116, 119] used adversarial attacks to test robustness of neural networks for general intrusion detection, and [43] applied adversarial attacks to evaluate the robustness of deep learning based IDS for IoT. To generate the adversarial data, they worked with IBM's Adversarial Robustness Toolbox (ART) [81], which is available for public use. During experiments they also showed that normalization of input lowers the model's effectiveness to resist adversarial attack, although the raw, un-normalized input has very low performance on adversarial-free data.

Rigaki et al. [93] applied a GAN to augment the effectiveness of IDS against attacks. Malware programs typically aim at producing packets as similar to normal ones as possible, in an effort to evade detection. The researchers used the malware FLU as an example, a malware whose packets mimic those generated by Facebook. Setting up the environment for the study, they initialized a virtual network system with servers, hosts, and an IPS. A GAN model was trained to guide the malware FLU to generate packets resembling those of Facebook, and with time more and more of these fraudulent packets begun to pass inspection, having approximated the distribution of the benign packets. These synthetically evolved malware packets where analyzed, and the resulting information was utilized to upgrade the robustness of the IPS.

Deep learning models may face challenges with small or imbalanced datasets, which are a common phenomenon in the IoT ecosystem, with its constant emergence of new technologies and devices, and the constant appearance of new threats and cyber-attacks. Zhang et al. [129] addressed this by conducting data augmentation

with a GAN, trying out the technique on the KDD99 dataset. The KDD99 dataset is imbalanced and contains outdated data, therefore using it in its original form to train machine learning models for real-world deployment would lead to poor performance. In the study a GAN model was produced data similar to those of KDD99. Inserting this synthetic data, which consisted of 8 attack classes, into the training set, resulted in a model that achieved better detection accuracy in 7 of the 8 attack classes.

In a similar vein, Lee and Park [57] noted that although there is a number of studies on imbalanced data, most methods yield sub-optimal results, often causing either data loss (due to majority class undersampling) or overfitting (due to excessive upsampling of minority classes). The researchers experimented with GANs for enriching imbalanced datasets, applying a Random Forest classifier to determine whether the GAN-enriched datasets lead to better detection performance. They concluded that not only GANs contribute to better classifications, but they also surpass other methods that were previously used widely to deal with data imbalances.

Ferdowsi et al. [25] considered GANs for fully distributed intrusion detection without relying on a centralized controller. An architecture was developed where each device is monitoring itself and its neighbors for intruders, which gives two advantages. First, each device applies the classifier to its own and to its neighbors' data, whereas with a centralized classifier we would have one massively heterogeneous and diverse dataset that would make it hard for the classifier to generalize. Second, as the data are not stored centrally, privacy of user information is feasible.

4.9 Hybrid Models for Intrusion Detection in IoT

Loukas et al. [66] proposed an IDS aimed for cyberattacks against vehicles that combines a deep multilayer perceptron and a recurrent neural network. The system detects three categories of threats to robotic vehicles: denial of service attacks, command injection attacks, and malware attacks that target the network interface. During experiments the hybrid model was shown to achieve higher accuracy than traditional machine learning techniques like k-means clustering and SVMs.

Tuor et al. [114] pursued an interpretable IDS. Lack of explainability is a major weakness of deep learning, and the researchers tackled the issue with a DNN and RNN hybrid. They applied this approach on a dataset comprised of system logs, the CERT Insider Threat dataset.First, they used a sliding window to extract features from the data. Then, a DNN and an RNN classified the logs, each model focusing on different levels of abstraction. The DNN evaluated the data from a sort-range perspective, processing the individual log's contents, and the RNN applied a long-range perspective, based on the log sequences. This approach reached a detection rate of 90% and reduced the analysis workload by 93.5%. The explainability was pursued by decomposing the malicious data samples into the contributions of each feature.

Al-Hawareh et al. [5], based on autoencoders and feed-forward neural networks developed a centrally placed intrusion detection model for the Industrial IoT. This model was designed to scrutinize TCP/IP packets and classify suspicious activity. The autoencoders smooth out the noise and extract high-level features so that the

neural network can process the data faster. Tested on the NSL-KDD and NSW-NB15 datasets, the model yields accuracy 98.6% and 92.4% respectively. Compared with other models, such as F-SVM, DMM, CVT, TANN, RNN, DBN, and DNN, the hybrid model performed better on both datasets.

Zeng et al. [125] combined three models to analyze payload features and achieve comprehensive content information, thus improving the classification. Their payload detection method included a CNN, an LSTM, and a stacked autoencoder, the three models extracting features from different viewpoints. The CNN focused locally, the RNN identified temporal patterns, and the stacked autoencoder performed feature extraction from the text of the payloads. This hybrid method yielded 99.22% accuracy on the ISCX 2012 dataset.

Thamilarasu et al. [111] focused on building a centralized system that could process IoT traffic data in real-time. They trained a DBN in unsupervised fashion, then used the nodesneurons to built a DNN that was trained with labeled data, using a binary cross-entropy loss function. In this way the overall training time of the model was faster, as the unsupervised learning of the DBN is much faster than supervised neural network training. The fast pre-train of the DNN, added to the supervised fine-tuning of the model, are faster than if the DNN was trained from scratch in a supervised fashion. The hybrid system was trained on a synthetic dataset that simulated IoT traffic with five attack classes, plus one class of normal traffic, yielding precision up to 99.5%, TPR (True Positive Rate) up to 99%, and F-1 score up to 99%.

4.10 Other Deep Learning Models for Intrusion Detection in IoT

Saeed et al. [96] used Random Neural Networks (RaNN) for anomaly-based intrusion detection in low-power IoT networks. The model consisted of two layers. The first layer was trained with the normal traffic samples, while the second was trained to detect a variety of Illegal Memory Access (IMA) bugs and data integrity attacks. This is applied on a centralized architecture where the traffic is mediated and processed by a central hub. Qureshi et al. [91] used a random neural network in a heuristic technique that yielded 95.25% accuracy on the NSL-KDD dataset.

Pamukov et al. [85] addressed the computational and power constraints of IoT devices by designing a system that is only trained with normal traffic data and red-flags any packet that diverges from the normal patterns. The result was a Negative Selection Neural Network (NSNN) that does not need attack data for training, and which performed well on the NSL-KDD dataset.

Jan et al. [45] also tackled the computational complexity of IoT intrusion detection with another experimental approach. They designed a Probabilistic Neural Network which combines AdaBoosting with locally enhanced semi-parametric base classifiers. The purpose of this architecture is to classify attacks at an affordable computational complexity, aiming for real-time IoT monitoring. In experiments with benchmark datasets, the model achieved comparable performance at a reduced computational cost.

Ibitoye et al. [43] used a Self-normalizing Neural Network (SNN), which is a variation of the Feed-Forward NN (FNN) that maintains stability during the gradient descent process [53]. They compared its performance with the normal FNN for detecting unauthorized activity in an IoT network using the BoT-IoT dataset from the Cyber Range Lab of the center of UNSW Canberra Cyber. It was shown that although the FNN outperformed the SNN on the dataset, the SNN was more robust to adversarial attacks. This may lead to fruitful future research, although in this particular study the adversarial attacks did lower the accuracy of the SNN tremendously, albeit much less than the FNN.

Li et al. [59], experimented with deep migration learning for IoT cybersecurity and smart city intrusion detection. The researchers divide deep migration learning techniques into four categories: sample migration, parameter migration, feature representation migration, and related knowledge migration. The algorithm was used both for feature extraction and classification, tested on the KDD CUP 99 dataset, and shown to reach a false alarm rate of 0.56% and a detection rate of 91.05%.

5 Discussion

5.1 Challenges with Deep-Learning-Based Intrusion Detection

Deep learning is totally dependent on training data. In the domain of IoT intrusion detection there is shortage of high-quality labeled datasets, which poses a challenge for the design of new effective methods. Curating good datasets can be costly and time consuming. They need to reflect the architecture and protocols of the network we want to design an IDS for, contain data for all the new attacks, be balanced and represent all classes adequately, and be as free as possible from redundancies, missing values, and noise. Some of the benchmark datasets are too old to reflect the current landscape, do not contain data for modern attacks, and their features may not correspond to current protocols. Expert knowledge could be exploited for curating good datasets and for creating the conditions, within an environment, for high-quality data to be collected. In Table 3 we document some information on the benchmark datasets commonly utilized in the IoT IDS literature. Naturally, the more

Table 3 The benchmark datasets used on the studies reviewed

Dataset	Year	Number of samples	Number of features	Number of classes	Data origin
KDD99 [108]	1999	494,020	42	23	Simulated
NSL-KDD [108]	1999	148,517	42	23	Simulated
ISCX 2012 [101]	2012	500,000++	8	2/6	Simulated
UNSW-NB15 [76]	2015	257,673	44	10	Hybrid (real/simulated)
CIDDS-001 [94]	2017	2,000,000++	16	5/4	Simulated
CICIDS 2017 [100]	2017	618,976	80	13	Simulated

recent datasets are more relevant to modern networks, with CICIDS2017 being created explicitly to alleviate the shortcomings of the previous datasets, as discussed [100].

Since most models and IDS solutions found in the literature are tested on benchmark datasets, their performance may not always reflect performance in the real-world. These models are fine-tuned to yield maximum results on obsolete data [100], while the current landscape could require different models. However, the theoretical foundations of deep learning are strong, and it has been demonstrated to approximate functions of very high complexity. Ironically enough, although deep models have been criticized for performing better in the controlled environment and standardized datasets of academia than in the real world, it was their real-world success that sparked the current academic enthusiasm in the first place. Most deep learning models can be deployed successfully after training with realistic data, or after hyperparameter tweaks, which is why they are widely adopted in nearly every industry.

The necessary next step is to transition deep-learning-based IDS into the real world, and this entails making them efficient enough for real time processing. Unfortunately, most attention during research is being focused on the detection performance of the models, with little emphasis on computational and time efficiency. Kang et al. [47] measured the time needed for a trained DNN to generate predictions and found that, depending on model complexity, it amounts to 10-12 ms. Meidan et al. [73] discuss model training times for centralized IDS, while Lee et al. [58], addressing distributed architectures, recommend to couple classification with feature extraction, and discuss training times for pipelines that perform both. For the IoT IDS solution to be realistic and practical, the system needs not only to detect intrusions accurately, but to do it in real-time and in resource-constrained devices. A trade-off between effectiveness and efficiency needs to be made, and research efforts should take this into consideration.

Being anomaly-based, IDSs that employ deep learning may reach high accuracy, but they also suffer from high false-positive rates. Shallow machine learning may suffer from more than 20% false-positive rates [105], and in a recent study it was concluded that deep learning alleviates the problem, though not solving it completely [6]. The problem is not model-dependent, but is inherent in the anomaly-based approach itself, which, by default, registers all novel patterns, both normal and malign, as anomalies, and flags them as suspicious. The problem could be mitigated by coupling deep learning with knowledge-based systems. Knowledge-based systems have low false-positive rate but high false-negative rates. Thus, a hybrid approach could well be the next step in IoT IDS. At any rate, expert knowledge should be taken advantage of whenever available, whether for the design of knowledge-based modules of an IDS, or for evaluating a deep learning model.

Deep learning models are black boxes, and it cannot be explained in human-understandable terms how a classifier makes its decisions. Whenever a packet is labeled as suspicious we cannot know why exactly the algorithm classified it as such. Given the false-positive problem, there will be false alarms, and it will be time consuming for human administrators to resolve them. Research on the explainability of blackbox models is being conducted, with limited success [126, 128], but eventually reproducibility makes up for the lack of explainability. If a deep learning pipeline

Table 4 Accuracy of deep learning models on benchmark datasets

Model	Dataset	Accuracy (%)	References
DNN	NSL-KDD, KDD99	92.1	[67]
	CIDDS-001	99.99	[107]
	UNSW-NB2015	94.04	[107]
CNN	NSL-KDD	91.14	[88]
	UNSW-NB2015	94.9	[88]
	ISCX2012	99.13	[75]
RNN	ISCX2012	99.99	[42]
	NSL-KDD	92.18	[9]
CNN–RNN	ISCX2012	99.92	[117]
	ISCX2012	97.60	[123]
	ISCX2012	97.29	[49]
Boltzmann	KDD99	95.50	[20]
Machine	KDD99	99.91	[84]
DBN	NSL-KDD	99.45	[130]
	CICIDS2017	99.37	[69]
Autoencoder (AE)	NSL-KDD	99.96	[127]
RaNN	NSL-KDD	95.25	[96]
AE & FFNN	NSL-KDD	98.6	[5]
	UNSW-NB2015	92.4	[5]
CNN, LSTM, & AE	ISCX2012	99.22	[125]

detects intrusions accurately over a period of time it can be deemed trustworthy, perhaps with a human administrator providing feedback in high-stake situations.

5.2 Emerging Best Practices

The application of deep learning for intrusion detection in the IoT is a new field, subject to constant change, as the IoT ecosystem keeps evolving and cybercrime keeps adapting. Nevertheless, due to the widespread research in this field some conclusions as to what would constitute 'best practices' have begun to emerge.

Table 4 presents data on the performance of deep learning models on some benchmark datasets, although in practice strategies will be customized for the topology and traffic of individual cyber-physical systems. Most studies have been geared towards specific situations, making it hard to compare their effectiveness against each other. The emerging 'best practices' we extract from our wide-range review should be considered generalized and tentative, although the success of deep learning and its wide adoption in the cybersecurity industry indicates the field is moving in the right direction.

Deep neural networks are a straightforward, easy-to-implement architecture with good all-around performance in a wide range of topologies and datasets. They form

the basis of deep learning, and in developing novel IDS solutions experimentation will start here.

CNNs, RNNs, and CNN–RNN hybrids can tackle time-sensitive and sequential data. Any security system that monitors traffic based on its temporal dimension should consider these sequence-processing models. CNNs do not surpass RNNs in performance (Table 4) but are more computationally efficient [17]. They could be the choice for distributed architectures, where the intrusion detection must take place within resource-constrained IoT devices. For centralized architectures, where a high-performance central hub can run a sophisticated model with low-latency, an RNN could be the final choice.

All deep learning models yield state-of-the-art performance, depending on the quantity and quality of the training data, but Boltzmann Machines and Deep Belief Networks excel when there are limited amounts of annotated data. Being semi-supervised learning models, they can be used in novel networks and systems where adequate volumes of training data have not been generated yet. These models can be trained with unlabeled data, learn the patterns in the distributions, and then be fine-tuned with a small number of labeled samples.

Autoencoders are best for anomaly-based intrusion detection. Each autoencoder is trained with samples of a specific class, learns the distributions in the data, and registers traffic packets that fall outside of these distributions as anomalies. The anomaly-based approach to intrusion detection manages to detect novel attacks that had not been previously observed by the system, or used as training data for a deep learning model.

A single best model for specific areas, network topologies, and IDS placement strategies does not seem to have emerged yet. The studies reviewed have used a variety of models for a variety of situations and datasets, the models' performances being comparable, as evidenced from Table 4. What is needed for definitive conclusions to arise is to conduct comparative studies, where a range of different deep learning models are tested on a single dataset or task, their performance metrics and time complexity measured, compared with each other, and evaluated.

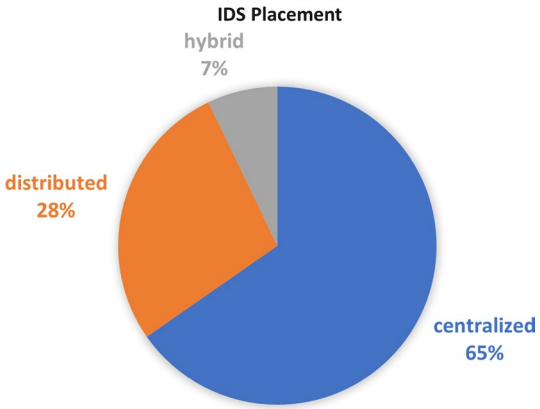

Fig. 15 The proportion of the number of studies reviewed, in relation to IDS placement

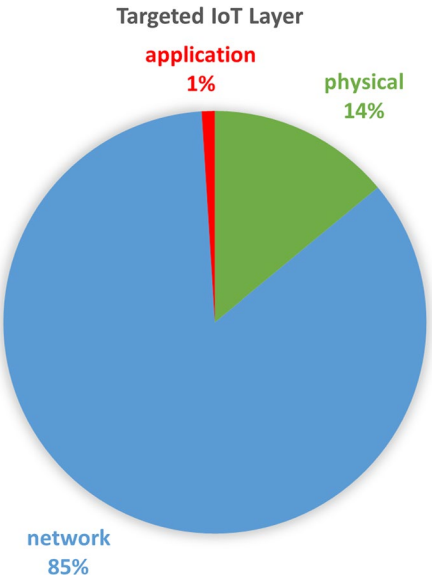

Fig. 16 The proportion of the number of studies reviewed, in relation to the targeted IoT layer

5.3 Trends and Observations

Figure 15 depicts how much attention different areas of research have received, categorized according to the IDS placement strategy (centralized, distributed, hybrid), whereas in Fig. 16 the categorization is based on the IoT layer being monitored (physical, network, application layers). The reader should be mindful that trends do not constitute recommendations. For example, centralized IDSs have received more than twice the attention of distributed, but numerous researchers highlight the importance of distributed IDS.

One trend that does seem meaningful is the low activity on deep learning-based IDS research for the application layer. In [4], the major challenges for IDSs on this layer are discussed, with those most relevant to anomaly-based systems being: (1) the IDS cannot examine packet data, since packet data is encrypted, (2) high variability and complexity of client data, rendering the system hard-pressed to extract patterns and statistical regularities, (3) parameters such as login information and user profile settings take arbitrary values, and the distinction between normal and anomalous activity is not obvious, and (4) applications change and evolve continually, and anomaly-based IDSs need constant retraining.

Another observation, not apparent from the pie charts, is the high degree of overlap between IDSs with distributed placement and IDSs on the physical layer. A major portion of the IDSs proposed for the physical layer follow the distributed topology, and vice versa [32, 78]. On the other hand, most IDSs that monitor the network layer adopt the centralized placement, especially those that use flow-based data, since flows are generated by intermediate networking devices (routers, switches, etc), rather than the IoT devices themselves.

Within the general framework of IoT, a number of areas have been the subject of deep learning-based cybersecurity studies. General networks, Industrial IoT (IIoT) and Industrial Control Systems, smart city, fog computing, vehicular networks, power grid, and smart home appliances, have all been in the scopes of researchers. Our study revealed that the majority of research has been conducted on general purpose networks, followed by IIoT and vehicular networks. The intrusion detection task itself has been formulated in a range of ways, from mere flagging of anomalous patterns to classifying different types of attacks, even to pinpointing the physical location of a human intruder. As the field is still new, trends in terms of specific areas of research have hardly begun to emerge and, again, trends are not necessarily recommendations. Researchers in touch with how the IoT field develops and evolves will focus on areas where research is most needed, as opposed to areas which simply have received the most attention.

6 Conclusions

Unlike knowledge-based IDS, and perhaps even traditional machine learning, deep learning takes advantage of big data, exploiting large quantities of data to train complex models that perform, in classification tasks, better than anything invented thus far. It also fits the dynamic nature of the IoT ecosystem, rendering it the best solution for IDS.

In this survey, we presented the models proposed for IoT intrusion detection, the specific tasks they were applied to, and the performance they achieved. We also examined some of the reasons why deep learning is a more preferable strategy for IDSs than shallow machine learning models, and what challenges this new paradigm faces.

Despite the superiority of deep learning over all other approaches for IoT intrusion detection, there's tremendous potential for future research. Distributed deep-learning-based IDS could meet the requirements of the large-scale, distributed, self-organizing nature of IoT networks. More computationally efficient models could be more easily supported by resource-constrained devices. Efficiency would also help deep learning work with real-time data streams, an absolute necessity for ensuring the safety of IoT communications. The problem of labeled data scarcity could be mitigated by breakthroughs in unsupervised learning. Decentralized, efficient, and unsupervised techniques and models would be a fruitful direction of research for deep-learning-based IoT IDS.

References

1. Abdullah, M., Alshannaq, A., Balamash, A., Almabdy, S.: Enhanced intrusion detection system using feature selection method and ensemble learning algorithms. IJCSIS **16**(2), 48–55 (2018)
2. Abeshu, A., Chilamkurti, N.: Deep learning: the frontier for distributed attack detection in fog-to-things computing. IEEE Commun. Mag. **56**(2), 169–175 (2018). https://doi.org/10.1109/MCOM.2018.1700332

3. Abomhara, M., Køien, G.M.: Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks. J. Cyber Secur. Mobil. **4**(1), 65–88 (2015)
4. Agarwal, N., Hussain, S.Z.: A closer look at intrusion detection system for web applications. Secur. Commun. Netw. (2018)
5. Al-Hawawreh, M., Moustafa, N., Sitnikova, E.: Identification of malicious activities in industrial internet of things based on deep learning models. J. Inf. Secur. Appl. **41**, 1–11 (2018). https://doi.org/10.1016/j.jisa.2018.05.002
6. Al Jallad, K., Aljnidi, M., Desouki, M.S.: Anomaly detection optimization using big data and deep learning to reduce false-positive. J. Big Data **7**(1), 1–12 (2020)
7. Al-Jarrah, O.Y., Maple, C., Dianati, M., Oxtoby, D., Mouzakitis, A.: Intrusion detection systems for intra-vehicle networks: a review. IEEE Access **7**, 21266–21289 (2019). https://doi.org/10.1109/ACCESS.2019.2894183
8. Alaiz-Moreton, H., Aveleira-Mata, J., Ondicol-Garcia, J., Muñoz-Castañeda, A.L., García, I., Benavides, C.: Multiclass classification procedure for detecting attacks on MQTT-IoT protocol. Complexity **2019**, 1–11 (2019). https://doi.org/10.1155/2019/6516253
9. Almiani, M., AbuGhazleh, A., Al-Rahayfeh, A., Atiewi, S., Razaque, A.: Deep recurrent neural network for IoT intrusion detection system. Simul. Model. Pract. Theory **101**, 102031 (2020). https://doi.org/10.1016/j.simpat.2019.102031
10. Aloqaily, M., Otoum, S., Ridhawi, I.A., Jararweh, Y.: An intrusion detection system for connected vehicles in smart cities. Ad Hoc Netw. **90**, 101842 (2019). https://doi.org/10.1016/j.adhoc.2019.02.001
11. Amanullah, M.A., Habeeb, R.A.A., Nasaruddin, F.H., Gani, A., Ahmed, E., Nainar, A.S.M., Akim, N.M., Imran, M.: Deep learning and big data technologies for IoT security. Comput. Commun. **151**, 495–517 (2020)
12. Bahşi, H., Nõmm, S., La Torre, F.B.: Dimensionality reduction for machine learning based IoT botnet detection. In: 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 1857–1862. IEEE (2018)
13. Bassey, J., Adesina, D., Li, X., Qian, L., Aved, A., Kroecker, T.: Intrusion detection for IoT devices based on RF fingerprinting using deep learning. In: 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), pp. 98–104. IEEE (2019)
14. Belenko, V., Chernenko, V., Kalinin, M., Krundyshev, V.: 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. IEEE (2018)
15. Bengio, Y.: Learning deep architectures for AI, the essence of knowledge, vol. 2, no. 1, 2009. Now, Boston and Delft (2009). http://www.nowpublishers.com/product.aspx?product=MAL&doi=220000000
16. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
17. Chollet, F., et al.: Deep Learning with Python, vol. 361. Manning, New York (2018)
18. Chowdhury, M.M.U., Hammond, F., Konowicz, G., Xin, C., Wu, H., Li, J.: A few-shot deep learning approach for improved intrusion detection. In: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), pp. 456–462. IEEE (2017)
19. Darbandi, F., Jafari, A., Karimipour, H., Dehghantanha, A., Derakhshan, F., Choo, K.K.R.: Real-time stability assessment in smart cyber-physical grids: a deep learning approach. IET Smart Grid **3**(4), 454–461 (2020)
20. Dawoud, A., Shahristani, S., Raun, C.: Deep learning and software-defined networks: towards secure IoT architecture. Internet of Things **3**, 82–89 (2018)
21. Diro, A., Chilamkurti, N.: Leveraging LSTM networks for attack detection in fog-to-things communications. IEEE Commun. Mag. **56**(9), 124–130 (2018). https://doi.org/10.1109/MCOM.2018.1701270
22. Diro, A.A., Chilamkurti, N.: Distributed attack detection scheme using deep learning approach for Internet of Things. Fut. Gener. Comput. Syst. **82**, 761–768 (2018)
23. Elsaeidy, A., Munasinghe, K.S., Sharma, D., Jamalipour, A.: Intrusion detection in smart cities using restricted Boltzmann machines. J. Netw. Comput. Appl. **135**, 76–83 (2019)
24. Eswari, T., Vanitha, V.: A novel rule based intrusion detection framework for Wireless Sensor Networks. In: 2013 International Conference on Information Communication and Embedded Systems (ICICES), pp. 1019–1022. IEEE, Chennai (2013). 10.1109/ICICES.2013.6508172

25. Ferdowsi, A., Saad, W.: Generative adversarial networks for distributed intrusion detection in the internet of things. In: 2019 IEEE Global Communications Conference (GLOBECOM), pp. 1–6. IEEE (2019)
26. Fernandes, E., Rahmati, A., Eykholt, K., Prakash, A.: Internet of things security research: a rehash of old ideas or new intellectual challenges? IEEE Secur. Priv. **15**(4), 79–84 (2017)
27. Ferrag, M.A., Maglaras, L., Ahmim, A., Derdour, M., Janicke, H.: RDTIDS: rules and decision tree-based intrusion detection system for internet-of-things networks. Fut. Internet **12**(3), 44 (2020)
28. Ferrag, M.A., Maglaras, L., Moschoyiannis, S., Janicke, H.: Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. J. Inf. Secur. Appl. **50**, 102419 (2020)
29. Fiore, U., Palmieri, F., Castiglione, A., De Santis, A.: Network anomaly detection with the restricted Boltzmann machine. Neurocomputing **122**, 13–23 (2013)
30. Garcia-Morchon, O., Kumar, S., Keoh, S., Hummen, R., Struik, R.: Security considerations in the IP-based Internet of Things draft-garciacore-security-06. Internet Engineering Task Force (2013)
31. Gül, A., Adalı, E.: A feature selection algorithm for ids. In: 2017 International Conference on Computer Science and Engineering (UBMK), pp. 816–820. IEEE (2017)
32. HaddadPajouh, H., Dehghantanha, A., Khayami, R., Choo, K.K.R.: A deep recurrent neural network based approach for Internet of Things malware threat hunting. Fut. Gener. Comput. Syst. **85**, 88–96 (2018). https://doi.org/10.1016/j.future.2018.03.007
33. Hasan, M., Islam, M.M., Zarif, M.I.I., Hashem, M.: Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet of Things **7**, 100059 (2019). https://doi.org/10.1016/j.iot.2019.100059
34. He, Y., Mendis, G.J., Wei, J.: Real-time detection of false data injection attacks in smart grid: a deep learning-based intelligent mechanism. IEEE Trans. Smart Grid **8**(5), 2505–2516 (2017). https://doi.org/10.1109/TSG.2017.2703842
35. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**(7), 1527–1554 (2006). https://doi.org/10.1162/neco.2006.18.7.1527
36. Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P.L., Iorkyase, E., Tachtatzis, C., Atkinson, R.: In: Threat analysis of IoT networks using artificial neural network intrusion detection system, pp. 1–6. IEEE (2016)
37. Hu, W., Liao, Y., Vemuri, V.R.: Robust support vector machines for anomaly detection in computer security. In: ICMLA, pp. 168–174 (2003)
38. Huda, S., Miah, S., Yearwood, J., Alyahya, S., Al-Dossari, H., Doss, R.: A malicious threat detection model for cloud assisted internet of things (CoT) based industrial control system (ICS) networks using deep belief network. J. Parallel Distrib. Comput. **120**, 23–31 (2018)
39. Huong, P.V., Thuan, L.D., Hong Van, L.T., Hung, D.V.: Intrusion detection in IoT systems based on deep learning using convolutional neural network. In: 2019 6th NAFOSTED Conference on Information and Computer Science (NICS), pp. 448–453. IEEE, Hanoi, Vietnam (2019). https://doi.org/10.1109/NICS48868.2019.9023871. https://ieeexplore.ieee.org/document/9023871/
40. Hussain, F., Hussain, R., Hassan, S.A., Hossain, E.: Machine learning in IoT security: current solutions and future challenges. IEEE Commun. Surv. Tutor. **22**(3), 1686–1721 (2020). https://doi.org/10.1109/COMST.2020.2986444
41. Hwang, R.H., Peng, M.C., Huang, C.W.: Detecting IoT malicious traffic based on autoencoder and convolutional neural network. In: 2019 IEEE Globecom Workshops (GC Wkshps), pp. 1–6. IEEE (2019)
42. Hwang, R.H., Peng, M.C., Nguyen, V.L., Chang, Y.L.: An LSTM-based deep learning approach for classifying malicious traffic at the packet level. Appl. Sci. **9**(16), 3414 (2019). https://doi.org/10.3390/app9163414
43. Ibitoye, O., Shafiq, O., Matrawy, A.: Analyzing adversarial attacks against deep learning for intrusion detection in IoT networks. In: 2019 IEEE Global Communications Conference (GLOBECOM), pp. 1–6. IEEE (2019)
44. Imamverdiyev, Y.N., Abdullayeva, F.J.: Deep learning in cybersecurity: challenges and approaches. IJCWT **10**(2), 82–105 (2020)
45. Jan, T.: Ada-boosted locally enhanced probabilistic neural network for IoT intrusion detection. In: Conference on Complex, Intelligent, and Software Intensive Systems, pp. 583–589. Springer (2018)
46. Jun, C., Chi, C.: Design of complex event-processing IDS in internet of things. In: 2014 Sixth International Conference on Measuring Technology and Mechatronics Automation, pp. 226–229. IEEE (2014)

47. Kang, M.J., Kang, J.W.: Intrusion detection system using deep neural network for in-vehicle network security. PLoS ONE **11**(6), e0155781 (2016)
48. Karimipour, H., Dehghantanha, A., Parizi, R.M., Choo, K.K.R., Leung, H.: A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids. IEEE Access **7**, 80778–80788 (2019)
49. Khan, M.A., Karim, M., Kim, Y., et al.: A scalable and hybrid intrusion detection system based on the convolutional-LSTM network. Symmetry **11**(4), 583 (2019)
50. Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J.: Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity **2**(1), 1–22 (2019)
51. King, J., Awad, A.I.: A distributed security mechanism for resource-constrained IoT devices. Informatica **40**(1), 133 (2016)
52. Kingma, D.P., Mohamed, S., Jimenez Rezende, D., Welling, M.: Semi-supervised learning with deep generative models. Adv. Neural Inf. Process. Syst. **27**, 3581–3589 (2014)
53. Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S.: Self-normalizing neural networks. In: Advances in Neural Information Processing Systems, pp. 971–980 (2017)
54. Kobojek, P., Saeed, K.: Application of recurrent neural networks for user verification based on keystroke dynamics. J. Telecommun. Inf. Technol. **2016**(3), 80–90 (2016)
55. Kolias, C., Kambourakis, G., Stavrou, A., Voas, J.: DDoS in the IoT: Mirai and other botnets. Computer **50**(7), 80–84 (2017)
56. Kulkarni, R.V., Venayagamoorthy, G.K.: Neural network based secure media access control protocol for wireless sensor networks. In: 2009 International Joint Conference on Neural Networks, pp. 1680–1687. IEEE (2009)
57. Lee, J., Park, K.: GAN-Based Imbalanced Data Intrusion Detection System. Personal and Ubiquitous Computing, pp. 1–8. Springer, Berlin (2019)
58. Lee, S.J., Yoo, P.D., Asyhari, A.T., Jhi, Y., Chermak, L., Yeun, C.Y., Taha, K.: IMPACT: impersonation attack detection via edge computing using deep autoencoder and feature abstraction. IEEE Access **8**, 65520–65529 (2020)
59. Li, D., Deng, L., Lee, M., Wang, H.: IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning. Int. J. Inf. Manage **49**, 533–545 (2019)
60. Li, J., Zhao, Z., Li, R., Zhang, H.: Ai-based two-stage intrusion detection for software defined IoT networks. IEEE Internet of Things J. **6**(2), 2093–2102 (2018)
61. Li, W., Yi, P., Wu, Y., Pan, L., Li, J.: A new intrusion detection system based on KNN classification algorithm in wireless sensor network. J. Electric. Comput. Eng. (2014) 10.1155/2014/240217
62. Liu, H., Lang, B.: Machine learning and deep learning methods for intrusion detection systems: a survey. Appl. Sci. **9**(20), 4396 (2019). https://doi.org/10.3390/app9204396
63. Liu, J., Yin, L., Hu, Y., Lv, S., Sun, L.: A novel intrusion detection algorithm for industrial control systems based on CNN and process state transition. In: 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), pp. 1–8. IEEE (2018)
64. Liu, Y., Pi, D.: A novel kernel SVM algorithm with game theory for network intrusion detection. KSII Trans. Internet Inf. Syst. **11**(8), 20 (2017)
65. Lopez-Martin, M., Carro, B., Sanchez-Esguevillas, A., Lloret, J.: Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in IoT. Sensors **17**(9), 1967 (2017)
66. Loukas, G., Vuong, T., Heartfield, R., Sakellari, G., Yoon, Y., Gan, D.: Cloud-based cyber-physical intrusion detection for vehicles using deep learning. IEEE Access **6**, 3491–3508 (2017)
67. Ma, T., Wang, F., Cheng, J., Yu, Y., Chen, X.: A hybrid spectral clustering and deep neural network ensemble algorithm for intrusion detection in sensor networks. Sensors **16**(10), 1701 (2016). https://doi.org/10.3390/s16101701
68. Madhawa, S., Balakrishnan, P., Arumugam, U.: Roll forward validation based decision tree classification for detecting data integrity attacks in industrial internet of things. J. Intell. Fuzzy Syst. **36**(3), 2355–2366 (2019). https://doi.org/10.3233/JIFS-169946
69. Manimurugan, S., Al-Mutairi, S., Aborokbah, M.M., Chilamkurti, N., Ganesan, S., Patan, R.: Effective attack detection in internet of medical things smart environment using a deep belief neural network. IEEE Access **8**, 77396–77404 (2020). https://doi.org/10.1109/ACCESS.2020.2986013
70. McDermott, C.D., Majdani, F., Petrovski, A.V.: Botnet detection in the internet of things using deep learning approaches. In: 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE, Rio de Janeiro (2018). 10.1109/IJCNN.2018.8489489. https://ieeexplore.ieee.org/document/8489489/

71. McElwee, S., Heaton, J., Fraley, J., Cannady, J.: Deep learning for prioritizing and responding to intrusion detection alerts. In: MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM), pp. 1–5. IEEE (2017)
72. Mehmood, A., Mukherjee, M., Ahmed, S.H., Song, H., Malik, K.M.: NBC-MAIDS: Naïve Bayesian classification technique in multi-agent system-enriched IDS for securing IoT against DDoS attacks. J. Supercomput. **74**(10), 5156–5170 (2018). https://doi.org/10.1007/s11227-018-2413-7
73. Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., Elovici, Y.: N-BaIoT-network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Comput. **17**(3), 12–22 (2018). https://doi.org/10.1109/MPRV.2018.03367731
74. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746–751 (2013)
75. Min, E., Long, J., Liu, Q., Cui, J., Chen, W.: TR-IDS: anomaly-based intrusion detection through text-convolutional neural network and random forest. Secur. Commun. Netw. (2018)
76. Moustafa, N., Slay, J.: Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS), pp. 1–6. IEEE (2015)
77. Munir, M., Siddiqui, S.A., Chattha, M.A., Dengel, A., Ahmed, S.: FuseAD: unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models. Sensors **19**(11), 2451 (2019)
78. Munir, M., Siddiqui, S.A., Dengel, A., Ahmed, S.: DeepAnT: a deep learning approach for unsupervised anomaly detection in time series. IEEE Access **7**, 1991–2005 (2019). https://doi.org/10.1109/ACCESS.2018.2886457
79. Münz, G., Li, S., Carle, G.: Traffic anomaly detection using k-means clustering. In: GI/ITG Workshop MMBnet, pp. 13–14 (2007)
80. Ngo, Q.D., Nguyen, H.T., Nguyen, L.C., Nguyen, D.H.: A survey of IoT malware and detection methods based on static features. ICT Express (2020)
81. Nicolae, M.I., Sinn, M., Tran, M.N., Buesser, B., Rawat, A., Wistuba, M., Zantedeschi, V., Baracaldo, N., Chen, B., Ludwig, H., et al.: Adversarial Robustness Toolbox v1. 0.0. arXiv preprint arXiv:1807.01069 (2018)
82. Notra, S., Siddiqi, M., Gharakheili, H.H., Sivaraman, V., Boreli, R.: An experimental study of security and privacy risks with emerging household appliances. In: 2014 IEEE conference on communications and network security, pp. 79–84. IEEE (2014)
83. O'Shea, T.J., West, N., Vondal, M., Clancy, T.C.: Semi-supervised radio signal identification. In: 2017 19th International Conference on Advanced Communication Technology (ICACT), pp. 33–38. IEEE, Pyeongchang, Kwangwoon Do, South Korea (2017). https://doi.org/10.23919/ICACT.2017.7890052. http://ieeexplore.ieee.org/document/7890052/
84. Otoum, S., Kantarci, B., Mouftah, H.T.: On the feasibility of deep learning in sensor network intrusion detection. IEEE Netw. Lett. **1**(2), 68–71 (2019)
85. Pamukov, M.E., Poulkov, V.K., Shterev, V.A.: Negative selection and neural network based algorithm for intrusion detection in IoT. In: 2018 41st International Conference on Telecommunications and Signal Processing (TSP), pp. 1–5. IEEE (2018)
86. Pan, J., Yang, Z.: Cybersecurity Challenges and Opportunities in the New" Edge Computing+ IoT" World. In: Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, pp. 29–32 (2018)
87. Patel, K.K., Patel, S.M., et al.: Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges. Int. J. Eng. Sci. Comput. **6**(5), (2016)
88. Potluri, S., Ahmed, S., Diedrich, C.: Convolutional neural networks for multi-class intrusion detection system. In: International Conference on Mining Intelligence and Knowledge Exploration, pp. 225–238. Springer (2018)
89. Prabavathy, S., Sundarakantham, K., Shalinie, S.M.: Design of cognitive fog computing for intrusion detection in Internet of Things. J. Commun. Netw. **20**(3), 291–298 (2018)
90. Prokofiev, A.O., Smirnova, Y.S., Surov, V.A.: 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. IEEE (2018)
91. Qureshi, A.u.H., Larijani, H., Ahmad, J., Mtetwa, N.: A Heuristic intrusion detection system for Internet-of-Things (IoT). In: K. Arai, R. Bhatia, S. Kapoor (eds.) Intelligent Computing, vol. 997, pp. 86–98. Springer International Publishing, Cham (2019). https://doi.

org/10.1007/978-3-030-22871-2_7. http://link.springer.com/10.1007/978-3-030-22871-2_7. Series Title: Advances in Intelligent Systems and Computing
92. Radford, B.J., Apolonio, L.M., Trias, A.J., Simpson, J.A.: Network traffic anomaly detection using recurrent neural networks. arXiv preprint arXiv:1803.10769 (2018)
93. Rigaki, M., Garcia, S.: Bringing a gan to a knife-fight: adapting malware communication to avoid detection. In: 2018 IEEE Security and Privacy Workshops (SPW), pp. 70–75. IEEE (2018)
94. Ring, M., Wunderlich, S., Gruedl, D., Landes, D., Hotho, A.: Technical report cidds-001 data set. Tech. rep., Tech. rep. 2017 (cit. on p. 19) (2018)
95. Roux, J., Alata, E., Auriol, G., Nicomette, V., Kaâniche, M.: Toward an intrusion detection approach for IoT based on radio communications profiling. In: 2017 13th European Dependable Computing Conference (EDCC), pp. 147–150. IEEE (2017)
96. Saeed, A., Ahmadinia, A., Javed, A., Larijani, H.: Random neural network based intelligent intrusion detection for wireless sensor networks. Proc. Comput. Sci. **80**, 2372–2376 (2016)
97. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Trans. Signal Process. **45**(11), 2673–2681 (1997)
98. Sethi, P., Sarangi, S.R.: Internet of things: architectures, protocols, and applications. J. Electr. Comput. Eng. (2017)
99. Sfar, A.R., Natalizio, E., Challal, Y., Chtourou, Z.: A roadmap for security challenges in the Internet of Things. Digital Commun. Netw. **4**(2), 118–137 (2018)
100. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSp, pp. 108–116 (2018)
101. Shiravi, A., Shiravi, H., Tavallaee, M., Ghorbani, A.A.: Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput. Secur. **31**(3), 357–374 (2012)
102. Sicari, S., Rizzardi, A., Grieco, L.A., Coen-Porisini, A.: Security, privacy and trust in Internet of Things: the road ahead. Comput. Netw. **76**, 146–164 (2015)
103. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. In: Advances in Neural Information Processing Systems, pp. 3483–3491 (2015)
104. Susilo, B., Sari, R.F.: Intrusion detection in IoT networks using deep learning algorithm. Information **11**(5), 279 (2020)
105. Syarif, I., Prugel-Bennett, A., Wills, G.: Unsupervised clustering approach for network anomaly detection. In: International Conference on Networked Digital Technologies, pp. 135–145. Springer (2012)
106. Tajbakhsh, A., Rahmati, M., Mirzaei, A.: Intrusion detection using fuzzy association rules. Appl. Soft Comput. **9**(2), 462–469 (2009)
107. Tama, B.A., Rhee, K.H.: Attack classification analysis of IoT network via deep learning approach. ReBICTE **3**, 1–9 (2017)
108. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the kdd cup 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6. IEEE (2009)
109. Taylor, A., Leblanc, S., Japkowicz, N.: Anomaly detection in automobile control network data with long short-term memory networks. In: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 130–139. IEEE (2016)
110. Teyou, D., Kamdem, G., Ziazet, J.: Convolutional neural network for intrusion detection system. In: Cyber Physical Systems. arXiv preprint arXiv:1905.03168 (2019)
111. Thamilarasu, G., Chawla, S.: Towards deep-learning-driven intrusion detection for the internet of things. Sensors **19**(9), 1977 (2019)
112. Tian, Z., Luo, C., Qiu, J., Du, X., Guizani, M.: A distributed deep learning system for web attack detection on edge devices. IEEE Trans. Ind. Inform. **16**(3), 1963–1971 (2019)
113. Tran, N.N., Sarker, R., Hu, J.: An approach for host-based intrusion detection system design using convolutional neural network. In: International Conference on Mobile Networks and Management, pp. 116–126. Springer (2017)
114. Tuor, A., Kaplan, S., Hutchinson, B., Nichols, N., Robinson, S.: Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. arXiv preprint arXiv:1710.00811 (2017)
115. Vlajic, N., Zhou, D.: IoT as a land of opportunity for DDoS hackers. Computer **51**(7), 26–34 (2018)

116. Wang, Z.: Deep learning-based intrusion detection with adversaries. IEEE Access **6**, 38367–38384 (2018)
117. Wang, W., Sheng, Y., Wang, J., Zeng, X., Ye, X., Huang, Y., Zhu, M.: HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection. IEEE Access **6**, 1792–1806 (2017)
118. Wang, W., Zhu, M., Zeng, X., Ye, X., Sheng, Y.: Malware traffic classification using convolutional neural network for representation learning. In: 2017 International Conference on Information Networking (ICOIN), pp. 712–717. IEEE (2017)
119. Warzyński, A., Kołaczek, G.: Intrusion detection systems vulnerability on adversarial examplhwanges. In: 2018 Innovations in Intelligent Systems and Applications (INISTA), pp. 1–4. IEEE (2018)
120. Wong, W.G.: Developers discuss IoT security and platforms trends (2014). https://www.electronicdesign.com/technologies/embedded-revolution/article/21800154/developers-discuss-iot-security-and-platforms-trends
121. Yavuz, F.Y., Ünal, D., Gül, E.: Deep learning for detection of routing attacks in the internet of things. Int. J. Comput. Intell. Syst. **12**(1), 39–58 (2018). https://doi.org/10.2991/ijcis.2018.25905181
122. Yu, Y., Long, J., Cai, Z.: Network intrusion detection through stacking dilated convolutional autoencoders. Secur. Commun. Netw. (2017)
123. Yuan, X., Li, C., Li, X.: DeepDefense: identifying DDoS attack via deep learning. In: 2017 IEEE International Conference on Smart Computing (SMARTCOMP), pp. 1–8. IEEE (2017)
124. Zarpelão, B.B., Miani, R.S., Kawakani, C.T., de Alvarenga, S.C.: A survey of intrusion detection in Internet of Things. J. Netw. Comput. Appl. **84**, 25–37 (2017)
125. Zeng, Y., Gu, H., Wei, W., Guo, Y.: \$ Deep-Full-Range \$: a deep learning based network encrypted traffic classification and intrusion detection framework. IEEE Access **7**, 45182–45190 (2019)
126. Zhang, A.Y., Lam, S.S.W., Ong, M.E.H., Tang, P.H., Chan, L.L.: Explainable AI: classification of MRI brain scans orders for quality improvement. In: Proceedings of the 6th IEEE/ACM International Conference on Big Data Computing, Applications and Technologies, pp. 95–102 (2019)
127. Zhang, B., Yu, Y., Li, J.: Network intrusion detection based on stacked sparse autoencoder and binary tree ensemble method. In: 2018 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6. IEEE (2018)
128. Zhang, C., Shang, B., Wei, P., Li, L., Liu, Y., Zheng, N.: Building explainable AI evaluation for autonomous perception. In: CVPR Workshops, pp. 20–23 (2019)
129. Zhang, H., Yu, X., Ren, P., Luo, C., Min, G.: Deep adversarial learning in intrusion detection: A data augmentation enhanced framework. arXiv preprint arXiv:1901.07949 (2019)
130. Zhang, Y., Li, P., Wang, X.: Intrusion detection for IoT based on improved genetic algorithm and deep belief network. IEEE Access **7**, 31711–31722 (2019)
131. Zhao, L., Su, C., Huang, H., Han, Z., Ding, S., Li, X.: Intrusion detection based on device-free localization in the era of IoT. Symmetry **11**(5), 630 (2019)