energies

MDPI

Review

Application of Deep Learning for Quality of Service Enhancement in Internet of Things: A Review

Nasser Kimbugwe [1,2], **Tingrui Pei** [1,3,*] **and Moses Ntanda Kyebambe** [2]

[1] School of Computer Science, Xiangtan University, Xiangtan 411105, China; nasser.kimbugwe@mak.ac.ug
[2] Department of Networks, College of Computing & I.S, Makerere University, Kampala 7062, Uganda; moses.ntanda@mak.ac.ug
[3] Key Laboratory of Hunan Province for Internet of Things and Information Security, Xiangtan 411105, China
[*] Correspondence: peitingrui@xtu.edu.cn

Abstract: The role of the Internet of Things (IoT) networks and systems in our daily life cannot be underestimated. IoT is among the fastest evolving innovative technologies that are digitizing and interconnecting many domains. Most life-critical and finance-critical systems are now IoT-based. It is, therefore, paramount that the Quality of Service (QoS) of IoTs is guaranteed. Traditionally, IoTs use heuristic, game theory approaches and optimization techniques for QoS guarantee. However, these methods and approaches have challenges whenever the number of users and devices increases or when multicellular situations are considered. Moreover, IoTs receive and generate huge amounts of data that cannot be effectively handled by the traditional methods for QoS assurance, especially in extracting useful features from this data. Deep Learning (DL) approaches have been suggested as a potential candidate in solving and handling the above-mentioned challenges in order to enhance and guarantee QoS in IoT. In this paper, we provide an extensive review of how DL techniques have been applied to enhance QoS in IoT. From the papers reviewed, we note that QoS in IoT-based systems is breached when the security and privacy of the systems are compromised or when the IoT resources are not properly managed. Therefore, this paper aims at finding out how Deep Learning has been applied to enhance QoS in IoT by preventing security and privacy breaches of the IoT-based systems and ensuring the proper and efficient allocation and management of IoT resources. We identify Deep Learning models and technologies described in state-of-the-art research and review papers and identify those that are most used in handling IoT QoS issues. We provide a detailed explanation of QoS in IoT and an overview of commonly used DL-based algorithms in enhancing QoS. Then, we provide a comprehensive discussion of how various DL techniques have been applied for enhancing QoS. We conclude the paper by highlighting the emerging areas of research around Deep Learning and its applicability in IoT QoS enhancement, future trends, and the associated challenges in the application of Deep Learning for QoS in IoT.

Keywords: internet of things; quality of service; machine learning; deep learning

Citation: Kimbugwe, N.; Pei, T.; Kyebambe, M.N. Application of Deep Learning for Quality of Service Enhancement in Internet of Things: A Review. *Energies* **2021**, *14*, 6384. https://doi.org/10.3390/en14196384

Academic Editors: Jaume Segura-Garcia and Santiago Felici-Castell

Received: 4 September 2021
Accepted: 3 October 2021
Published: 6 October 2021

1. Introduction

Computers, smartphones, systems, wireless sensors, actuators, and virtually every single automated device are connected together through the internet, creating the "Internet of Things (IoT)", as shown in Figure 1. The communication can be either through long-range mobile networks, such as WiMAX, GSM, GRPS, and cellular networks, such as LTE, 3G, 4G, and 5G, or through short-range technologies, such as Bluetooth, Wi-Fi, and ZigBee. Because of the massive usage of IoT networks, applications, and services in all aspects of our daily life, guaranteeing high levels of Quality of Service is very critical.

Our daily life is massively dependent on the IoT in many aspects. Almost every device currently has internet capabilities, and it is estimated that by 2040 the number of connected devices on the internet will exceed 75 billion, generating over 100 trillion

GB of data [1]. With the huge amounts of data, IoT has a great potential for the future smart world. However, the deployment of IoT on a considerably larger scale comes with many challenges, which include security and privacy challenges, resource allocation, and management challenges, all of which directly impact the Quality of Service (QoS). Because many of our critical daily life applications depend on IoT, it is important that the QoS of IoT networks and applications is guaranteed. Data-driven Machine Learning (ML) and Deep Learning (DL) methods can exploit IoT data to enhance the QoS of the automated IoT services and applications. As a way of ensuring high QoS, IoT applications sometimes require real-time responses or actions after processing data [1]. For example, object recognition and identification by security cameras require very little detection latency to capture and respond to specific events. This is becoming increasingly impossible using traditional means due to gigantic multimedia data generated. DL techniques have the capability of extracting meaningful information from this multimedia data [2]; and for this reason, Deep Learning models have been applied in different domains to revolutionize Information Technology (IT). As such, researchers in the IoT domain started exploring the application of DL to transform various aspects of IoT [3–5]. However, it is not yet clear how DL has been applied to enhance the QoS of various IoT-based systems and services. The review aims at addressing these gaps by providing researchers with; an overview of commonly used DL techniques that have been applied for enhancing QoS, future trends, and the associated challenges in the application of Deep Learning of QoS in IoT.

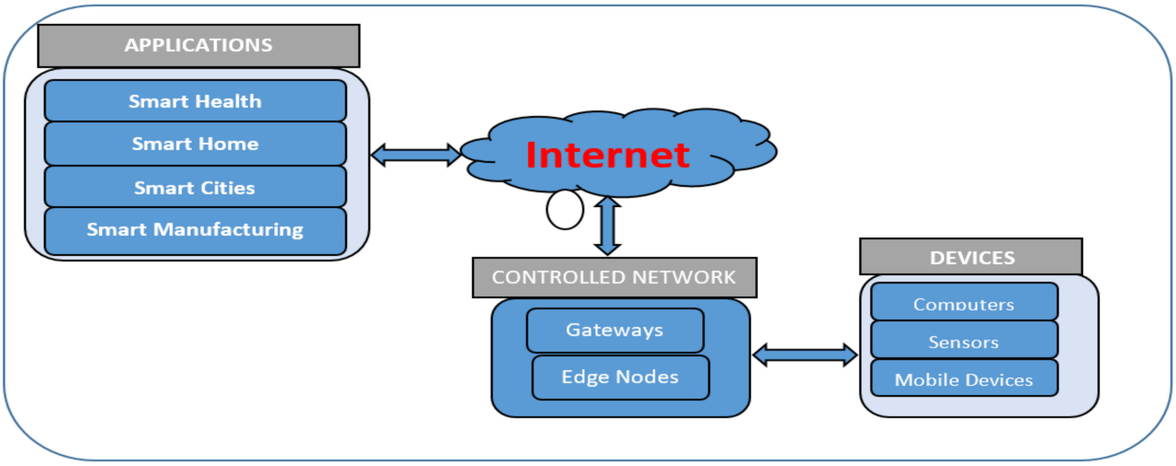

Figure 1. Internet of Things framework.

1.1. IoT Applications

Smart home: Home appliances, such as washing machines, dishwashers, lights, fridges, televisions, and radios, through a network, especially the internet, can be controlled remotely by authorized owners or users [6]. This offers better monitoring and management, hence saving resources, such as power. Control and monitoring are done using mobile phones, tablets, or computers. With the help of smart technologies, smart automatic doors, and smart human recognition sensors have also been incorporated as elements of smart homes to enhance home security [7].

Smart health: IoT has transformed health care services, from the traditional face-to-face consultations to telemedicine [8]. Wearable smart sensors and implants [9] that collect an array of health-related information, such as heartbeat rates, blood pressure, oxygen levels, blood sugar level, and body temperature [10], have been developed. Human activity recognition [11–20] technologies for health purposes have also been enhanced by the advancement of IoT and Deep Learning technologies.

Smart manufacturing: Smart manufacturing involves the application of innovative data processing and analytic techniques to improve decision making and performance within manufacturing systems [5,21]. Using IoT, Machine Learning, and Deep Learning, many manufacturing companies are applying more intelligent techniques in the manufacturing process [22]. Manufacturing enterprise systems are now able to be self-sensing, self-adaptive, and self-deciding. Self-sensing, in this case, means that the systems can obtain data from the environment by themselves and carry out processing depending on the obtained data. Self-adaptive means systems have the ability to adjust their run-time behavior in order to realize system goals. During irregular conditions, such as modifications in the settings of the system, system errors, or change of requirements, the systems are able to trigger self-adaptive functions in order for the system to continue being operational [23].

Smart Transportation and Smart cities: Smart transportation is a general term that refers to deciding the best route in terms of time and distance, optimization of parking space, streetlights, avoidance of accidents, and road anomalies [10,24]. Sensors are embedded into the vehicles, cameras, and other devices installed in the city to collect environmental information that is processed to make informed decisions. IoT and Deep Learning techniques have also been employed to ensure air quality in cities [25]. Insurance companies nowadays place cameras and sensors in insured vehicles to lower insurance compensation rates [26]. In this case, they are able to monitor and recognize driving styles of different drivers and have an informed assessment in case of an accident using GPS data.

The advance of the DL for IoT has enabled applications to become smarter. When the amount of data being collected increases, traditional Machine Learning techniques, which are mainly supervised, are not the best options. DL techniques come into play to extract meaningful features from unlabeled data for intelligent decision making, which enhances the capabilities of IoT applications, thus ensuring high QoS.

1.2. Problem Statement

Most critical systems and applications nowadays are IoT-based. This means that any failure or compromise on the Quality of Service of these systems can be detrimental. As such, ensuring the high Quality of Service of these IoT-based services should be given high priority. Deep Learning, due to its numerous advantages as a data-driven technique, has been applied to revolutionize various sectors in the IT world. However, it is not clear how DL-based approaches have been applied to ensure and guarantee high Quality of Service in IoT. It is also not clear which Deep Learning models best suit various aspects of QoS in IoT systems. In this review paper, we investigate how DL models have been applied to enhance QoS in IoT-based systems and why some DL models are applied in particular QoS aspects but not applied in others.

1.3. Related Previous Review Papers

As more and more systems become IoT-based, huge amounts of data are being generated, which requires advanced data-driven techniques for quick processing. Deep Learning, due to its capabilities in extracting relevant features from unlabeled data, is more appealing to be applied for huge IoT-generated datasets than traditional data-driven Machine Learning models that can only be applied to labeled data. We give a summary of the related literature reviews with an emphasis on papers published between 2015 and 2021 (see Table 1). We chose this period because it gives state-of-the-art research in the application of Deep Learning to IoTs. From these literature reviews, we note that QoS in IoT is mainly compromised when: (1) The security and privacy of IoT-based systems are breached, and (2) When IoT resources are not properly and efficiently managed. Most related review articles are, therefore, about Deep Learning for security and privacy in IoT (mainly intrusion detection, anomaly detections, and defect detections) and resource allocation and management (mainly resource scheduling, channel access, energy consumption, and management). Our search did not return any unequivocal review paper that had investigated the application of DL to security and privacy or resource allocation and management in IoTs for the years 2015–2017.

In 2018, D. Andročec and N. Vrček [27] carried out a review on the applicability of ML and DL models to IoT security. The note in their finding that ML and Deep Learning have not been widely applied for IoT security, and they identified Support Vector Machines (SVM) as the most used ML technique for IoT security. However, SVM works perfectly with labeled data, which is hard to find currently due to the huge amounts of data generated by the IoT. The paper does not consider the application of DL to resource allocation and management.

In 2019, P. Fraga-Lamas et al. [28] reviewed Collision Avoidance and Obstacle Detection in IoT DL-based Unmanned Aerial Vehicles (UAV) systems. By avoiding collisions and having the ability to detect obstacles, QoS is improved in IoT-based UAV systems. In this review paper, DL techniques for collision avoidance and the detection of obstacles autonomously are presented together with various datasets for IoT DL-UAV systems. Lateef et al. [29], carried out a survey of DL-based Intrusion Detection Systems (IDS) for IoT-based systems. They note that unsupervised DL models, such as AutoEncoders (AE), are more suitable for implementing IDS, but there is still a challenge of lack of training data.

In 2020, J. Asharf et al. [30] provided a survey of how ML and DL have been applied to solve intrusion detection in IoT-based systems. They also provide an overview of the available datasets for intrusion detection research. One of the challenges they note is "resource constraints issue with IoT devices limits the use of DL/ML algorithms". This suggests that there is a need to develop resource allocation and management DL techniques, which is one of the aims of this review paper. F. Hussain et al. [31] carried out an extensive review about the application of Machine Learning and Deep Learning to resource management in IoT networks. In their future work suggestion, they propose the development of DL models that are more reliable for mission-critical IoT systems. This calls for the consideration of security and privacy as major factors in achieving this. However, their review does not provide how DL models can be used to enhance security.

Table 1. Previous review papers about the use of DL-based models for QoS enhancement in IoT.

Year	Review Paper Reference	QoS Enhancement Factor
2015	No paper	
2016	No paper	
2017	No paper	
2018	D. Andročec and N. Vrček [27]	IoT security
2019	P. Fraga-Lamas et al. [28]	Obstacle detection and Collision Avoidance
	A. Lateef et al. [29]	Intrusion Detection
2020	J. Asharf et al.	Intrusion Detection
	F. Hussain et al. [31]	Resource management
2021	R. Al-amri et al. [32], M. A. Alsoufi et al. [33]	Anomaly Detection
	L. Aversano et al. [34]	IoT Security

In 2021, R. Al-amri et al. [32] provide a review of anomaly detection within IoT data and systems using ML and DL. They note that DL models are more suited to anomaly detection for IoT data streams than ML models because DL techniques have the capability of automatically extracting features from this data. They suggest future research on how to handle challenges that hinder the development of DL models for IoT anomaly detection. Some of the stated challenges include data streams and features that keep evolving, the complexity of data, which is usually noisy, visualization of data, and windowing problems. More so, Alsoufi et al. [33] also investigated the application of Deep Learning in IoT Intrusion Detection Systems based on anomaly detection. L. Aversano et al. [34] carried out a systematic review of how DL has been applied to security in IoT. Their review only concentrates on the security aspect of IoT QoS, leaving out the resource allocation and management aspects.

Based on the summary in Table 1, we conclude that all the related previous review papers focus on specific IoT QoS enhancement factors, including IoT security, obstacle detection and collision avoidance, intrusion detection, anomaly detection, and resource management. Our review is the first to explicitly cover the application of DL for QoS enhancement.

1.4. Purpose of This Review

Although the earlier literature research is helpful to review and describe the current application state of DL-based models, specifically for QoS in the Internet of Things, there are research gaps that we hope to address in this paper.

(1) Based on the previous review papers, there is a lack of papers that explicitly focus on the application of Deep Learning for QoS guarantee in IoTs. Yet, DL has been applied in many data-driven domains, including IoT. This review paper's objective is to address this gap.

(2) Various research papers recommend future research for the application of DL-based techniques for intrusion detection [29,30] and resource allocation and management [31], which are the main factors that determine the QoS of IoT networks and systems. Therefore, this review takes up this recommendation to provide researchers with the application of DL to QoS enhancement in IoTs.

(3) On top of providing the state-of-art, this research also discusses challenges hindering the application of DL techniques for QoS enhancement in IoTs. With challenges well-identified, future researchers about this topic can easily know where to focus.

In summary, the purpose of this review paper is four-fold: (1) To provide a review of the application of Deep Learning-based techniques in IoT networks and systems to enhance the QoS of such systems, (2) Identify Deep Learning models that have been applied in QoS enhancement in IoTs, (3) Elaborate on the reasons behind the use of DL techniques for QoS enhancement of IoT-based applications, and (4) Identify and discuss challenges in applying DL models for QoS enhancement in IoT-based services. This paper addresses the antecedently declared gaps in the analysis found over the assorted literature review papers revealed in Table 1.

1.5. Research Questions

The following research questions were followed in this research.

1. How are Deep Learning techniques being applied for QoS enhancement in IoTs?
2. Which Deep Learning models are being applied in various aspects of QoS enhancement in IoT-based applications, and why those models in particular?
3. Why have researchers opted for the use of Deep Learning techniques for QoS enhancement compared to the existing QoS enhancement approaches?
4. What challenges are faced by developers when applying DL models for QoS enhancement for IoTs?

1.6. Research Methodology

In this paper, we used the cataloging research method [35] to accomplish our review. We first carried out a search of the previous review papers published from 2015 to 2021. We chose this period because it represents the current state-of-the-art research carried out in the area of DL and QoS in IoTs. A summary of the previous review papers is presented in Section 1.3, and the respective QoS measurement factors addressed in the review papers are summarized in Table 1.

The second step was to search for research papers within the same period that investigated the application of Deep Learning techniques towards the enhancement of QoS in IoTs. We considered papers that fall into two categories: (1) Research papers that investigated the application of Deep Learning to improve the security and privacy in IoTs, and (2) Papers that investigated the application of Deep Learning for resource allocation and management in IoTs. We chose papers that belong to these two categories because, for the QoS of any IoT system to be compromised, it means that either the security of the IoT system has been breached or the IoT system's resources have been misallocated and or mismanaged.

Papers were searched for online from websites, including: https://ieeexplore.ieee.org (accessed on 30 July 2021), https://mdpi.com (last accessed on 20 September 2021), https://dl.acm.org/ (access on 30 July 2021), https://www.sciencedirect.com/ (accessed on 30 July 2021), https://www.springer.com/ (accessed on 30 July 2021), and https://scholar.google.com/ (accessed on 30 July 2021). The research articles were filtered according to their content. We only considered papers that investigated the application of at least one ML or DL technique towards the enhancement of IoT security and privacy or resource management for a reason already stated above.

Finally, we analyzed the selected papers to find out the DL application trends in IoT. We based our research questions in Section 1.5 on this analysis.

1.7. Contributions of This Review

The key contributions of this paper are listed below.

(a) We review Quality of Service in the Internet of Things and various metrics of QoS.
(b) We review the challenges of enhancing QoS using traditional methods (methods not related to DL) and show how DL techniques can be used to solve these challenges
(c) We review how the various DL algorithms have been applied in enhancing QoS in IoT-based systems. We identify the research gaps for the application of DL techniques for QoS in IoT. More of the observations and contributions are explained in the discussion, Section 4.

The rest of the paper is organized as follows. In Section 2, we give an overview of the Quality of Service in relation to IoT and Deep Learning algorithms in general, with a bias on those mostly applied to enhance QoS in IoT. In Section 3, we provide an extensive review of how DL-based techniques have been applied in enhancing QoS. Section 4 provides the discussion and description of the challenges of using DL for QoS enhancement in IoTs, and in Section 5, we conclude the review.

2. An Overview of Quality of Service and Deep Learning Algorithms for Internet of Things

2.1. Quality of Service in Internet of Things

QoS is the measurement of the general performance of any service, mainly the performance seen by the users of the service [36–38]. Owing to the widespread usage and application of IoT services in our daily life situations, the cost of IoT devices should be low without compromising the level of QoS. If IoT applications are to provide high-quality services to the users, latency and reliability must be guaranteed by these applications [39].

QoS assurance in IoT networks and systems requires clear support at different levels. At the network layer, for example, specific technical communication principles are needed to guarantee systematics and reliable distribution of data. For the application layer, dedicated support from application protocols and the development of innovative resource allocation procedures are needed to manage synchronized access and implementation of proper management of resources.

QoS aids in managing the system proficiencies and its resources in delivering IoT services. QoS metrics are the benchmarks upon which service providers can have perfect perceptibility of their services' performance and how best clients can use these services. QoS metrics aid customers in identifying the best IoT service for their applications and how best they can optimize the service quality. Anything that can negatively influence the performance of an IoT-based service affects its QoS [40]. According to M. Singh and G. Baranwal [37], QoS in IoT can be divided into three categories: (1) QoS of communication, (2) QoS of things, and (3) QoS of computing.

2.1.1. QoS of Communication

Transporting real-time data within IoT networks and applications is one of the fundamental determinants of the Quality of Service of the IoT network. Therefore, to meet the needs of various applications of IoT, we must consider adding value to it in order to improve the quality of the network applications and services. Anything that compromises the efficiency of an IoT service or application compromises the Quality of Service offered by that application. QoS of communication can be compromised by many factors, which may include bandwidth problems, jitter, and cyber-attacks. Bandwidth is the measure of the quantity of data that goes through a network in a particular period. Bandwidth determines the throughput and efficiency of the network [41].

To avoid cyber-attacks, which would compromise the QoS of communication, we need to pay close attention to the security and privacy of the network. Technologies, such as Virtual Private Networks (VPN), Transport Layer Security (TLS), Onion Routing, and Private Information Retrieval (PIR), have been invented to handle the privacy issue. Of all the known QoS of Communication challenges, security and privacy have the most significant influence on the adoption of IoT [42]. Denial of Service (DoS) is one of the most catastrophic attacks against IoT [43]. IoT is slightly different from traditional computers in that most IoT devices, such as sensors, are designed for deployment on a large scale, which makes them more vulnerable to security threats. IoT is increasingly becoming a target for cybercriminals. Most IoT architectures have three layers, i.e., the network layer, the application layer, and the perception layer [44,45]. QoS of things corresponds to the perception layer, QoS of computing corresponds to the application layer, and QoS of communication corresponds to the network layer. The architectural representation of these IoT layers is shown in Figure 2.

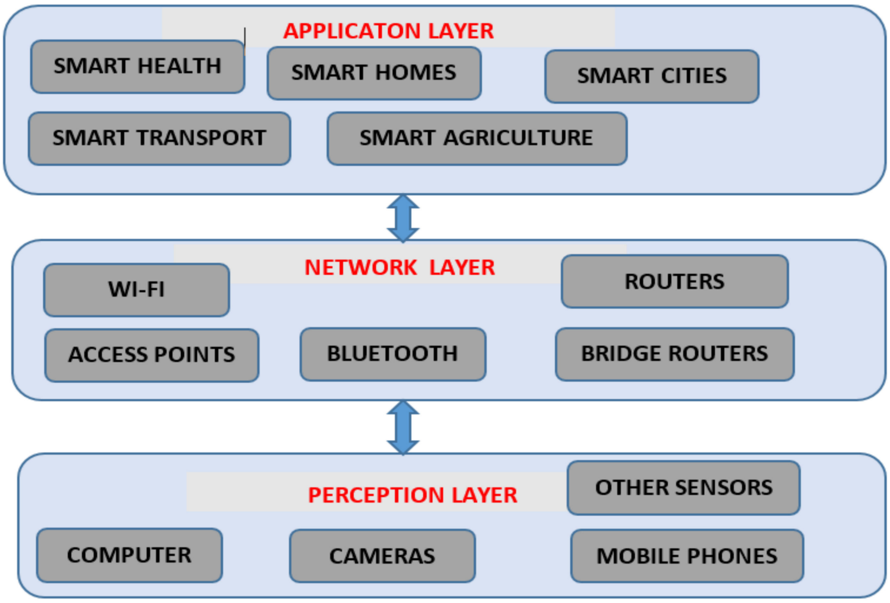

Figure 2. A Three-Layer IoT Architecture.

2.1.2. QoS of Things

Individual "things" in IoT networks must have quality parameters that can guarantee the Quality of Service of the network. Sensors, as one major element of IoT networks, for example, must be of low weight, reliable, and with low power consumption.

2.1.3. QoS of Computing

IoT networks generate huge amounts of data per unit of time. It is, therefore, important that the computing components of IoT networks be of high quality. To enhance the QoS and efficiency of computing, cloud computing was invented and is being used in IoT mainly because the applications produce an enormous volume of data that may not be easy to process within the network [46]. Computations can also be executed at fog or edge to lower the tremendous pressure put on the network [47]. Thus, edge computing shifts the data processing from the cloud to the edge nodes, which improves the QoS for IoT applications with low-latency requirements [48]. QoS of computing in IoT must ensure reliability, scalability, availability, privacy, and security, as well as appropriate management of the available resources.

The perception layer is also known as the "sensor" layer, and its main function is to detect and collect data, which is transmitted, to the network layer. The routing of data and transmission to the IoT applications and devices is then handled by the network layer over the internet.

To enhance QoS in IoTs, we must ensure that mainly two factors are well managed: (1) Ensure network security in order to guarantee privacy and security of the network resources, and (2) Ensure that network resources are well-managed, i.e., proper resource allocation and management. A QoS breach is mainly because of the mismanagement of these two factors. Either security is compromised, or resources are not well managed or allocated. This paper focuses on how Deep Learning techniques have been applied to guarantee QoS in IoT by handling security issues and resource allocation and management challenges of the network.

2.2. Deep Learning Algorithms

With the advent of big data technologies, various IoT-based services have emerged to make use of this data. Smart manufacturing [5,21,22,49,50], smart cities [24,51], smart homes [6,7], smart agriculture [52–54], and smart health [8,17,55–57], among others, have undergone tremendous development with the aid of deep-learning techniques. IoT based-services today are faced with an unprecedented surge in generated sensory data, which comes in different formats, structures, and semantics. From this massive amount of sensory data, DL has attracted wide attention as a revolution in computational automation and intelligence. By mining knowledge from different sources of data, DL technology is vital in the extraction of features from data automatically in order to identify various patterns and make informed decisions.

Deep Learning can handle huge volumes of data because DL techniques and algorithms are more scalable with increasing amounts of data compared to traditional Machine Learning algorithms, and hence are more suited to model training. On top of this, Deep Learning techniques can automatically extract hidden features and relevant correlations from unlabeled input data. Because IoT data is generated from different sources, the data tends to be of various patterns and usually in an unlabeled form. Deep Learning can exploit this unlabeled data in an unsupervised way to learn useful patterns.

IoT-based services have become an integral part of our daily lives, including very critical systems, such as airplane environmental detection systems [58] and life support systems [59]. This means that the Quality of Service (QoS) of such systems is also critical. Anything that compromises the Quality of Service must be approached with equal measures. Numerous DL algorithms have been developed, and the relevant research topics are increasing at a very rapid pace. To smoothen the study of smart services in IoT, several DL algorithms have been proposed, including Restricted Boltzmann Machine (RBM), Convolutional Neural Network (CNN), Autoencoder, and Recurrent Neural Networks (RNN) [60].

2.2.1. Convolutional Neural Network (CNN)

CNN, first proposed by G. E. Hinton et al. [61] for two-dimensional image processing, is a multilayer artificial neural network that uses a forward-feed algorithm and backpropagation [62]. Similar to other neural networks, CNN operates in the same way the brain's visual cortex recognizes and processes things and learns to classify them [63]. CNN has also been applied to speech recognition [64–66] and natural language processing (NLP) [67,68].

CNN networks contain three layers, i.e., input layer, hidden layers, and output layer. The hidden layers also consist of pooling layers, convolution layers, normalization layers, and other connected layers. When applied to images for example, the convolution layer transforms the image into convolution processes while the pooling layer combines the adjacent pixels of an image into one pixel. The convolutional layer creates the feature map, which is a list of new features, by extracting some special and unique features from the initial data. This representative value is generally the average or the largest value of the pixels being selected. To conduct operations in the pooling layer, the criterion of selecting the pixels and how to set the representation value must be decided. In Figure 3 for example, the adjacent pixels are selected from the 2 × 2 square matrix. The convolution layer is the fundamental module of CNNs much as each specific problem requires different structures of CNNs. Given the input feature map χ, and a filter matrix W, then the output of the input feature map Y is given by:

$$Y = \sum W_{ij} * \chi + b_i \tag{1}$$

where b is the bias parameter and i represents the ith row, j represents the jth column of the input matrix. The convolution layer output is in many cases run through a function known as the activation function, which is generally non-linear. An activation function can be a sigmoid function, a tanh function or reLU function as listed as follows:

$$\begin{cases} Sigmoid\ activation\ function: \ f = \frac{1}{1+e^{-x}} \\ ReLU\ activation\ function: \ f = \max(0, x) \\ \tanh activation\ function: f = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \end{cases} \tag{2}$$

where x is the input value e and an exponent constant.

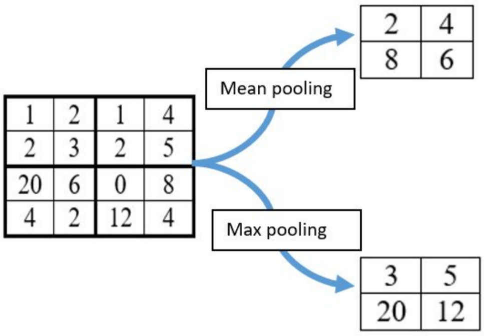

Figure 3. Illustration of the two pooling methods: Mean and Max pooling.

2.2.2. Restricted Boltzmann Machine

Restricted Boltzmann Machines (RBMs) consist of two layers; the visible and hidden layers. Unlike other neural networks, neurons within a single layer in RBM have no connections with each and every other neuron, as illustrated in Figure 4. RBMs are Artificial Neural Networks that belong to an Energy-Based Model [69] where the data is input through the visible layers, and unique features are extracted by the hidden layers. RBM models are probabilistic in nature. This means that instead of assigning a discrete value, RBM models assign probabilities. For dimensionality reduction and data encoding, hidden layers provide parameters that are considered to be features that define the input data. ML techniques, such as Naïve Bayes, logistic regression, and Support Vector Machine, are then applied for data classification. Since RBM automatically extracts the required features from data, it avoids the local minimum value, and it has received a growing number of considerations. RBM is always in a particular state. That state denotes the values attached to each neuron within the input (layer-v) plus inner layers (hidden layers, h). The possibility (P) for a given h and v to be detected is defined by the equation below.

$$P(v,h) = \frac{1}{W} e^{-E(v,h)}$$

$$W = \sum_{v,h} e^{-E(v,h)}$$

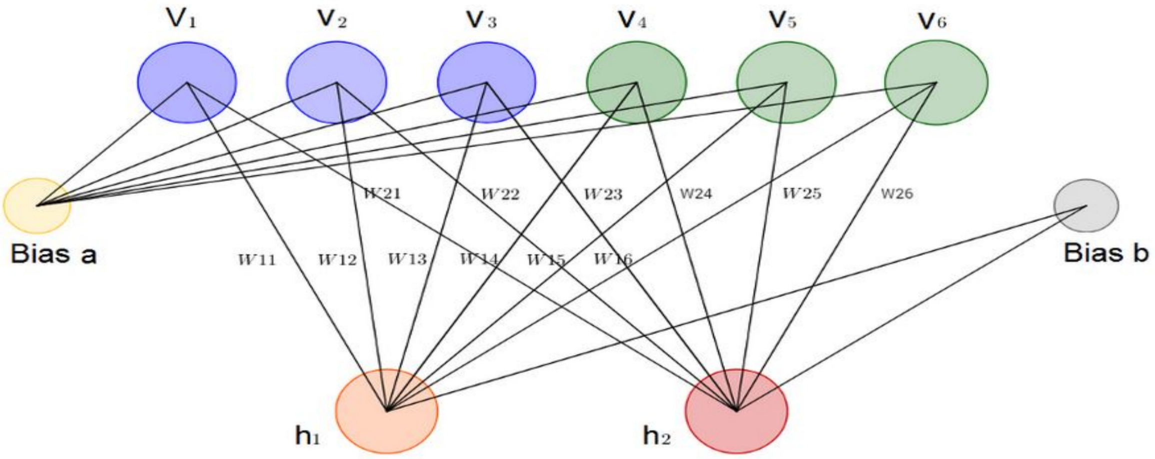

Figure 4. Restricted Boltzmann Machines Architecture.

W defines partition function for the hidden and visible neuron values and E is the RMB energy function.

The energy function for RBMs is defined as

$$E(v,h) = -\sum_i a_{iv_i} - \sum_j b_j h_j - \sum_{i,j} v_i h_i w_{ij} \tag{3}$$

where v represents input layers, h represents hidden layers, and a and b are the bias values.

2.2.3. Autoencoders (AE)

An autoencoder is a neural network mainly used in unsupervised learning to efficiently learn codings from unlabeled data. Through encoding and decoding techniques, AE can regenerate the original data input. An AE neural network uses a backpropagation algorithm [70], by equating the output values to the inputs, that is Y(i) = X(i) [71]. According to J. Jordan [72], an ideal autoencoder model should be sensitive to the original inputs enough to precisely regenerate a reconstruction and insensitive to the inputs so that the designed model does not merely overfit or simply memorize the data. The autoencoder can compress the input and then reconstruct the output according to the compressed representation [73]. Autoencoders are data-specific, meaning that they can only be applied to data similar to the training data, and their output is not always the same as the input. For example, if the model is trained using handwritten digits, it is not appropriate to apply it to landscape photos. Autoencoder techniques have been applied in various domains. For example, in civil engineering for bearing defect detections [74], health-related human activity recognition [75,76], medical imaging [77,78], recommendation systems [79–81], and many other domains. Figure 5 shows the components of an autoencoder algorithm. Autoencoder can be combined with LSTM algorithm to create LSTMAE as shown in Figure 6.

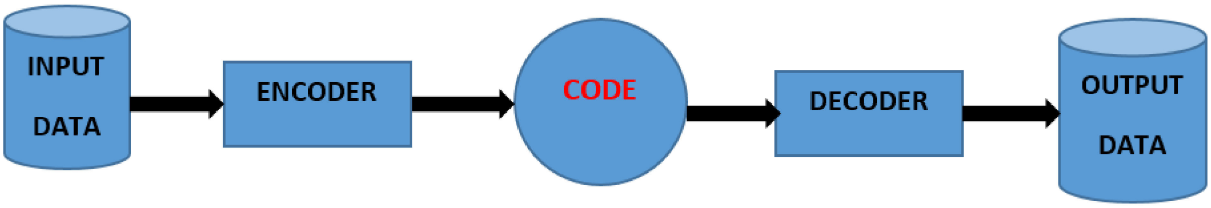

Figure 5. Illustration of components of an autoencoder.

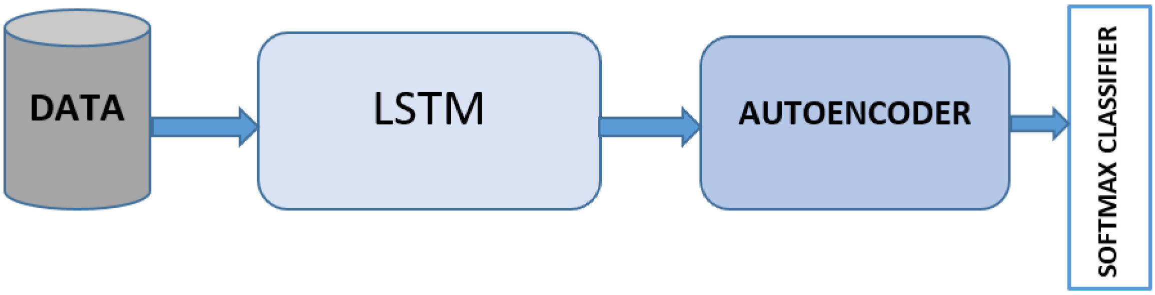

Figure 6. An architectural view of the LSMTAE algorithm.

2.2.4. Recurrent Neural Networks (RNN)

RNNs are a type of NN where inputs for the succeeding layers are generated from the preceding layers while having hidden states [82]. An RNN is very suitable for feature learning and extraction from sequential data [83] because of the connections between the preceding and the succeeding data items. RNNs recall the past, and their decisions are affected by whatever they learned from the past. Much as rudimentary feed-forward networks also recall things, they only recall things they learn while training. Even though RNNs learn in a similar way during the learning process, they can evoke states learned from previous inputs when constructing the output for the next stage. RNNs are capable of taking one or more input vectors and generating more output vectors, and unlike NN, where outputs are only determined by the weights of input vectors, they also use hidden state vectors, which show the context of the previous data [15,84]. The hidden state is calculated at various time steps using an updated rule. Consider a sequential input vector; we may calculate the current hidden state in two parts using the same sigmoid or tanh activation function. The first component is obtained using the original input, while the second is based on the preceding time step's hidden state. A softmax function can then be used to extract the desired final output from the up-to-date hidden state. Over raw input data, pooling methods, such as max pooling and mean pooling, are used to find the hidden state. The largest value of one vector in the feature map is chosen as the most significant feature by max pooling. Mean pooling takes the mean value of one vector and uses it as the vector's pooling value. In this scenario, a vector can represent a patch of pixel values on a picture being investigated. Max pooling is a great way to get sparse features.

2.2.5. Deep Reinforcement Learning (DRL)

Deep Reinforcement Learning techniques do not require huge training data sets but rather learn by interacting with the environment. It is similar to how humans learn from their actions. Deep Q-learning is one of the most prevalent Reinforcement Learning techniques. It combines Deep Neural Networks (DNN) and Reinforcement Learning (RL), with DNN serving as a learning agent for RL. In this scenario, DNN uses interactions with environmental data to gradually approximate the best policy function, obviating the requirement for extensive training data. Because RL alone cannot handle circumstances where the number of system states and data is very large, and the environment is not stationary, DNN is integrated with RL. In IoT networks, IoT devices can use Reinforcement Learning to make judgments based on inference under dynamic and uncertain network conditions. For example, RL has been utilized in cognitive radio networks during spectrum sharing for channel access between the primary users and secondary users [85–87].

2.2.6. Generative Adversarial Network

Generative Adversarial Networks (GANs) are methods of generative modeling using Deep Learning methods. Generative modeling is a form of the unsupervised learning task, which involves automatic discovery and learning the patterns in input data in a way that the model can be used to plausibly produce new examples "resembling" the original dataset. GANs are an ingenious way of training DL models by turning the problem being investigated into a supervised learning problem that has two other models: (1) A model that is trained to generate new examples (the generator model), and (2) A mode that attempts to classify data as either real or just generated fake data (discriminator model).

2.2.7. Deep Learning Frameworks

The implementation of the above Deep Learning algorithms within IoT-based services is accomplished using Deep Learning frameworks that support various programming languages. The most notable examples of Deep Learning frameworks are described in Table 2.

Table 2. Deep Learning frameworks.

DL Framework	Description	Type
Chainer [88]	Dynamic, intuitive, and highly powerful tool that is based on python. Chainer is mainly deployed in machine recognition, speech recognition, and sentiment analysis.	Open source
Caffe [89]	Supported by c, c++, python, and Matlab. It is popularly used for vision recognition. Caffe does not provide support for fine granularity network layers as compared to tensor flow or CNTK. Caffe's biggest bragging right is its speed. However, sometimes it may require usage of low-level language, which many users do not like. Caffe is also open source.	Open source
CNTK [90]	Known as the Microsoft cognitive tool. It supports C++ and python. It provides high scalability in terms of training a CNN and Generative Adversarial Networks (GAN) especially for images, speech of any text-based data. Mainly deployed in handwriting recognition and speech recognition. It is easy to train, and above all, open source.	Open source
MXNet [91]	Provides the users the ability to code in a variety of different programming languages, including python, C++, R, Scala, Julia. Designed for high efficiency, high flexibility, and high productivity. Mainly used in Natural language processing and speech recognition, as well as forecasting. Mxnet is the certified DL reference library for Amazon.	Open Source
DeepLearning4j [92]	Deep Learning for java (DL4J). Java is one of the most widely used programming languages; DL4J development was a respite for java programmers. DL4J provides parallel training though iterative modules and micro service architectures option coupled with distributed CPUs and GPUs. Binds together the whole java ecosystem to implement Deep Learning. Can be administered on top of hadoop and Apache spark. DL4J supports LSTM Networks, CNN, RNN, RBM, and DBN among other Deep Learning algorithms. Deployed for image recognition and fraud detection.	Open Source
Keras [93]	Official high-level API of TensorFlow. Supports both convolutional and Recurrent Neural Networks. Keras can run on top of Theano, Tensorflow, or CNTK. Keras is modular, and building models is as simple as stacking layers and connecting graphs. Keras is open source, actively developed by contributors across the globe, and has a good amount of documentation.	Open source
Pytorch [94]	PyTorch is an optimized tensor library for Deep Learning using GPUs and CPUs. Provides support for both python and c++. It is also an open source framework with a lot of support from the developers the world over.	Open source
Tensorflow [95]	TensorFlow is an open source machine-learning platform that features a robust ecosystem of tools, libraries, and community resources that enable researchers to advance the state-of-the-art in Machine Learning and developers to quickly build and deploy Machine Learning powered apps [96].	Open Source

3. DL Application to QoS Guarantee in IoT

DL, as a trending technological method, has been researched in various types of applications in IoT recently. In this section, we review the application of DL in ensuring that the Quality of Service of the IoT networks and applications is ensured.

3.1. Data Processing, Analytics and Transmission

Some IoT networks transfer the data to the cloud for analysis. However, this is ineffective due to high communication costs and QoS requirements. In addition, when the data is analyzed within the IoT network, there are increased bandwidth requirements and communication delays. DL-based prediction techniques play an important role in predicting the bandwidth that may be required.

Liang [97] proposed a data processing method for Deep Learning in IoT by applying Singular-Value Decomposition(SVD)-QR for the preprocessing of Deep Learning data and limited memory subspace optimization for SVD-QR algorithm to speed up data processing.

Liang outlines two possible data processing schemes for Deep Learning: (1) Data is reduced via keeping a subset, and its original features are kept through down-sampling, and (2) Data is transformed, and some of the original features are lost, e.g., through compression. The purpose of these two methods is to speed up data processing in IoT for reliable QoS.

The authors in [98] proposed a Deep Learning-based approach for IoT data transfer that is both latency and bandwidth-efficient. They suggest a solution for the missing data IoT data problem by enabling Deep Learning models on resource-restricted IoT devices. In many cases, IoT devices do not accurately collect data due to various reasons, such as malfunctioning within the devices, unreliable network communication, and external attacks. Subsequently, missing data may lead to wrong decision-making and impact the QoS, especially for time-intensive and emergency applications. To test the DL models, they used data from the Intel Berkeley Research Lab. They [98] used a Long Short Term Memory (LSTM) model for model formulation and TensorFlow plus Keras frameworks to implement the model. Their results demonstrated that Deep Learning-based techniques can greatly improve network delay and bandwidth requirements, hence an improved QoS for IoTs.

3.2. Deep Learning for IoT Security

Because IoT-based solutions are utilized for control and communication in critical infrastructure, these systems must be safeguarded from vulnerabilities in order to ensure the Quality of Service metric of availability [3].

3.2.1. Intrusion Detection in IoT

IoT networks are susceptible to attacks and detecting the adversaries' actions as early as possible and can help safeguard data from malicious damages, which guarantees Quality of Service of the network. Because of its high-level feature extraction capacity, the adoption of DL for attack and intrusion detection in cyberspace and IoT networks could be a robust mechanism against tiny mutations or innovative attacks. When malicious attacks on IoT networks are not recognized in a timely manner, the availability of important systems for end-users is harmed, which leads to an increase in data breaches and identity theft. In such a scenario, the Quality of Service is drastically compromised.

Koroniotis et al. [99] created the BoT-IoT dataset, and it was used to evaluate RNN and LSTM. They used feature normalization to scale the data within the range 0–1 and estimated the correlation coefficient within the features and joint entropy of the dataset for feature selection. They evaluated the performance of their model based on Machine and Deep Learning algorithms using the botnet-IoT dataset compared with popular datasets. The results show an improved intrusion detection using Deep Learning compared to traditional methods.

In [100], the authors employ Machine Learning classifiers; SVM, Adaboost, decision trees, and Naïve Bayes to classify data into normal and attack classes. In their work, they used Node MCU-ESP8266, DHT11-sensor, and a wireless router to simulate an IoT environment. They then built an adversary scheme with a computer, which implements poisoning and sniffing attacks on the IoT environment. The steps they followed while building their system are as follows: Develop a testbed to mimic an IoT-based environment → Develop an attack-like system to obtain attack data → Obtain the flow of data in the system and generate normal and attack scenarios features → Build Machine Learning and DL methods to identify and categorize network attacks.

Susilo, Bambang and Riri, Fitri Sari [101] discuss numerous Machine Learning and DL strategies, plus standard datasets that can be used to enhance the safety performance in IoT networks and systems. Using Deep Learning techniques, they presented a method for identifying Denial-of-Service (DoS) assaults. Tensorflow, Seaborn, and Scikit-learn were among the tools they employed using the Python programming language. According to their findings, a Deep Learning model could improve accuracy, ensuring that attacks on IoT networks are mitigated as effectively as possible, hence guaranteeing the QoS in IoT networks and applications. They used the BoT-IoT and KDD data sets to evaluate their algorithm. They used Random Forest, CNN, and multilayer perceptron (MLP) to classify the attacks.

Yingfei Xu et al. [102] proposed an autoencoder anomaly-monitoring model based on LSTMs-AE, where LTSM is used to capture time-series characteristics, and AE is used for intrusion detection. Their tests revealed that the model outperforms the standard autoencoder in terms of intrusion detection.

In [103], the authors developed a hybrid intelligent Intrusion Detection System (HIIDS) for IoT to efficiently and automatically extract important features representation from vast unlabeled raw IoT network traffic data. In their work, the authors also combined the LSTM algorithm because of its ability to capture long dependencies and the autoencoder to carry out their experiments, hence the LSTMAE algorithm. They carried out their experiments on ISCX-2012, and the results showed 97.3% accuracy.

In [104], the authors proposed RNN-CNN, an RNN and CNN hybrid. To avoid overfitting, they added layers, such as max pooling, batch normalization, and dropout. They tested their model using RedIRIS real data. RedIRIS is a Spanish research and academic backbone network that offers enhanced communication services to scientists and researchers. Results from their work show that RNN combined with CNN effectively monitored network traffic for abnormal detection with over 97% accuracy and outperformed traditional abnormality detection techniques.

Using Gated Recurrent Neural Networks, a DL model for IDS in the IoT Network was presented by Manoj Kumar Putchala, in his master's degree thesis [105]. For feature selection, the Random Forest classifier was applied. The UNB ISCX 2012 and KDD cup'99 data sets were used to validate the model.

A novel anomaly detection approach based on unsupervised DL techniques was suggested by Dawoud et al. [106]. The model compares the usage of Restricted Boltzmann machines as generative energy-based models to autoencoders as non-probabilistic algorithms to see if Deep Learning can discover abnormalities. The simulation results show ≈99% anomaly detection accuracy, which guarantees QoS in IoT.

Using bi-directional long short-term memory Recurrent Neural Networks, B. Roy and H. Cheung [107] proposed a DL approach for intrusion detection in the IoT networks. They translated categorical features to numeric values using feature normalization. Using the UNSWNB15 data set, they built a multilayer DL Neural Network. Working with the IoT network, their research focused on the binary classification of normal and attack patterns. The experimental findings demonstrate the effectiveness of the proposed model, which achieves over 95% accuracy in attack detection while ensuring QoS in intrusion detection.

In [108], on the NSL-KDD dataset, a Deep Neural Network (DNN) is used. To minimize the loss function of DNN, the authors employ stochastic gradient descent (SGD).

They employ fog nodes for training the DL model. Local parameters are provided to a fog coordinator node for updating, and the DL model is developed using fog nodes. This allows the optimum parameters to be shared and helps to avoid local overfitting.

In [109], M. Roopak et al. proposed a Deep Learning model for cyber security using various classification DL algorithms, which included multilayer perceptron, 1D Convolutional Neural Network, Long Short Term Memory (LSTM), and a combined Convolutional Neural Network +LSTM on CICIDS2017 dataset. Their model provides 97.17% accuracy in DDOS attack detections. The higher the accuracy of attack detection, the higher the Quality of Service a particular IoT network can guarantee.

The authors of [110] developed an intelligent intrusion-detection system for the IoT environment. Using an IoT simulation dataset, they proposed a feed-forward DNN using a Deep Belief Network. They allocate a cost function to each layer of the model in order to optimize DNN. For several attack scenarios, such as DDoS, wormhole attacks, sinkhole, opportunistic service, and blackhole attacks, their method achieved a recall rate of 97 percent and an average precision rate of 95%. An IoT would ensure dependable Quality of Service with such precision and recall rates because security for the Internet of Things is ensured.

Mohammadi et al. [111] developed a self-organizing map (SOM) algorithm, Radial Basis Function (RBF), and multilayer perceptron networks-based IoT Intrusion Detection System. To generate the parameters for the perceptron neural network, they employ the Imperialist Competitive Algorithm (ICA). Their tests were carried out using the KDD99, and results show tremendous improvement in intrusion detection, which enhances the QoS in IoTs.

Deepcoin, a Deep Learning and blockchain-based energy exchange concept for smart grids, was suggested by Ferrag and Maglaras [112]. On two non-IoT datasets and the BoT-IoT [113] dataset, they utilized the RNN algorithm and the truncated backpropagation through time (BPTT) [70] algorithm. Before being fed into the model, features are normalized. Their approach generates blocks with small signatures to thwart smart grid assaults.

Aldhaheri et al. [114] proposed a DeepDCA model, a hybrid between DL and Dendritic Cell Algorithm (DCA) [115] in order to handle intrusion detection. Their model implements DCA and SNN (Self Normalizing Neural Network) [116]. Their research was directed at classifying IoT intrusion and minimizing false alarm generation. Their suggested Intrusion Detection System selects the appropriate collection of features from the IoT-Bot dataset, then uses the SNN to categorize signals before using the DCA for classification. DeepDCA performs exceptionally well in detecting IoT threats, with a detection rate of over 98.73% accuracy and a low false-positive rate, according to the simulation data. The authors validated their results with other ML and DL algorithms, which showed that their model performs better classification tasks than SVM, KNN, and MLP.

Using the Bot-IoT dataset, Soe et al. [117] proposed an Artificial Neural Network to detect Distributed Denial-of-Service (DDoS) attacks in the IoT environment. They applied the Synthetic Minority Over-sampling Technique (SMOTE) to overcome data imbalances and normalized the features before feeding the input data to their proposed neural network. Their results show that the suggested model can successfully detect DDoS attacks within the IoT environment.

Ge et al. [118], using the BoT-IoT dataset applied feed-forward neural networks to detect malicious attacks in IoT. They used the Adam optimizer to optimize the model, and cross-entropy loss function, a sparse categorical in nature, was used for weights updating. Regularization techniques, such as L1, L2, and dropout, were used to avoid deal overfitting. The results obtained by evaluating the implemented model on the BoT-IoT data demonstrate a high accuracy in the classification of malicious attacks.

Muna et al. [119] proposed a framework to detect malicious activities in industrial IoT using deep autoencoder (DAE) and deep feed-forward NN. They compared their model with Computer Vision Technique (CVT) [120], Filter-based Support Vector Machine

(F-SVM) [121], Triangle Area Nearest Neighbors (TANN)[122], Dirichlet Mixture Model (DMM) [123], Deep Belief Networks (DBN) [124], Recurrent Neural Networks (RNN), Deep Neural Networks (DNN), and Ensemble-DNN. Their model outperformed all the herein mentioned techniques.

Zhong et al. [125], using Deep Learning models, proposed a sequential model-based Intrusion Detection System for Internet of Things (IoT) servers. Their model uses tcpdump packets to get information from the network layer and system procedures to gather information from the application layer. Their approach greatly improves the detection of intrusive attacks in IoT networks, hence enhancing QoS.

In [126], the authors used the Self-Normalizing Neural Network (SNN) and compared the results of their model with the feed-forward neural networks (FNN) for classifying intrusion attacks in an IoT network. They used the BoT-IoT data set, and their experimental results show that FNN outperforms SNN in terms of accuracy, precision, and recall for intrusion detection in IoT. However, the SNN shows better resilience than FNN as far as adversarial robustness is concerned.

3.2.2. Defect Detection in IoT

Ola Salman et al. [127] suggested a Machine Learning-based framework for identifying IoT devices and detecting aberrant data. By pushing intelligence to the network edge, their approach extracts features per network flow to identify the source, the type of generated traffic, and to detect network threats. They analyze different machine-learning algorithms and find that Random Forest produces the best results, with up to 94.5% accuracy for device type identification, 93.5% accuracy for traffic type classification, and 97% accuracy for abnormal traffic detection.

3.3. DL for Resource Allocation and Management in IoT

Another metric of QoS in IoT is how effective resources are allocated and managed. Poor resource management and allocation can compromise the QoS offered by a particular IoT network or application. Resource allocation is conventionally done using optimization methods, Heuristic techniques, and game theoretical approaches by considering the QoS requirements of the user [31]. Optimization method approaches have challenges whenever the number of users and devices increase or when the multicellular situations are considered. The reason is that optimization space becomes tremendously huge to satisfy the entire network; thus finding solutions becomes computationally too high. Heuristic and game theoretical approaches suffer from a lack of scalability, slow convergence, and information exchange overload. DL, on the other hand, has the ability to deduce information from data and then utilize that knowledge to alter a DL agent's behavior depending on that knowledge. Since IoT networks produce gigantic volumes of data, researchers have applied DL techniques [128,129] to extract useful features that can be used to dynamically and intelligently handle resource allocation efficiently.

Generally, each type of IoT network faces different challenges in relation to resource allocation (RA) and management. For example, RA challenges in cellular IoT are different from those in cognitive IoT networks, low-power IoT, and mobile IoT networks [31].

General IoT resource management challenges include session management and setup [130], interference management, and channel dynamic access [131]. Conventional resource allocation and management methods in IoT networks mainly make use of optimization techniques. However, as the number of users increases, the optimization computational complexity also increases tremendously, hence affecting the QoS of that network.

Cognitive IoT networks have primary users and secondary users. Primary users are the "rightful" owners of the source, but a resource can be assigned to the secondary user once the primary user is idle or absent. When the primary user in cognitive networks is stimulated, the secondary user must be removed from that channel [132]. Therefore, there is a need to consider QoS requirements for both the primary and secondary users as far as resource allocation is concerned. Static techniques are used to manage resource allocation problems, such as channel sensing, detection, and acquisition. However, these techniques have a number of drawbacks, including collisions and reduced system performance.

Mobile IoT (MIoT) networks have one distinguishing feature from traditional IoTs mobility. In MIoT, the services and applications of IoT can be transferred from one physical location to another. The communicating things move but maintain their interconnection and accessibility, for example, in the case of smart transport where cars move from one location to another but maintain connectivity. Resource allocation and management using traditional methods is more complex in MIoT than in static IoT networks because of the extra information required to maintain connectivity among mobile devices.

To address the challenges of using traditional resources allocation methods, Machine Learning and Deep Learning techniques can be an appropriate remedy where IoT networks can learn the context of users. IoT devices, through progressive learning, can autonomously be able to access the available spectrum. IoT entities can also adaptively learn and adjust the transmission power to conserve energy. Deep Reinforcement Learning techniques [133] and linear regression [134] have been used in resource allocation in IoT.

In [135], the authors investigate a combined task scheduling and resource distribution for Deep Neural Network (DNN) inference in the Industrial IoT (IIoT) networks. They formulate a resource management issue with the goal of optimizing mean inference accuracy while also meeting the QoS of DNN inference jobs in IIoT networks with limited spectrum and computational resources for huge DNN inference projects. They convert the problem to a Markov Decision Process and offer a deep deterministic policy gradient-based learning technique to quickly find a solution.

Deng et al. [136] proposed a reinforced learning method for dynamic resource allocation for edge computing-based IoT systems. In order to improve trustworthiness, IoT services declare a service-level agreement (SLA), which is used as a basis for the measurement of QoS of that particular service. The authors encode the state of the service provisioning system as well as the resource allocation scheme using the SLA as a measure and then describe the adjustment of resources allocated for that specific service as a Markov Decision Process (MDP). With the help of reinforcement learning, they obtain the trained resource allocation model, which dynamically allocated the resources according to the system states and requirements. They carried out experiments on Youtube request data, and results show that their approach has a 21.72% better performance compared to the Low Inter-reference Recency Set (LIRS) algorithm, Locality Frequency (LF) algorithm, and Long Short Term Memory (LSTM) algorithm.

In [137], the authors proposed a resource allocation approach for IoT, which uses Reinforcement Learning based on the Quality of Experience (QoE) status. They proposed two RF-based algorithms to accomplish the resource allocation task. Reinforcement Learning-based Mapping Table (RLMT) and Reinforcement Learning Resource Allocation (RLRA) algorithm. RLMT is aimed at creating an efficient cost-mapping table, which dynamically adjusts table items depending on the feedback of QoE. The RLRA algorithm then chooses the optimum path for allocating a resource based on the task-mapping table.

Shah and Zhao [138] proposed a multi-agent virtual resource allocation scheme for IoT based on Deep Reinforcement Learning. They accessed network resources using the Network Function Virtualization (NFV) approach, then handle resource allocation in IoT networks using the Deep Reinforcement Learning (DRL) algorithm. By learning the network's behavior, DRL eliminates the need for exact Channel State Information (CSI). They frame their issue as a Markovian Decision-Making Process (MDP).

In [31], the authors review various Machine Learning techniques for resource allocation in cellular and IoT networks. They also provide several resource allocation and management challenges in IoT networks and applications, which include massive channel access, power allocation and interference, cell selection, energy management, and real-time processing.

3.3.1. Massive Simultaneous Channel Access

When a large number of devices connect to the same wireless channel at the same time, the channel can become overloaded. In order to accommodate significant capacity and connection while efficiently utilizing network resources, load balancing and access control must be handled. In [139], the authors proposed an ML-based channel assignment algorithm that applies Tug-Of-War (TOW) dynamics to select channels for communication in cognitive massive IoT networks. They formulate their problem as a MAB problem. Their experimental results show great improvement in interference detection compared to conventional interference detection approaches that do not use ML techniques.

3.3.2. Power Allocation and Interference Management

Power allocation serves an important role in improving the performance of IoT networks by reducing the interference to other IoT network entities. Choosing transmission power dynamically in line with varying physical channel and network conditions is very challenging. Therefore, dynamic and intelligent power allocation and interference management techniques are needed. Machine learning techniques are best suited for this.

In [140], the authors presented a Deep Learning-based long-term Power Allocation (DL-PA) scheme for satellite-based Internet of Things non-orthogonal multiple access (NOMA) downlink system (S-IoT). They use a neural network as an approximation function to compute the Successive Interference Cancellation (SIC) decoding order according to a particular queue state and channel state. Their approach produces more accurate results than when Deep Learning is not used. In device-to-device IoT networks, the author applied Q-learning and CART Decision Tree algorithms for power control interference management. A multi-agent Q-learning algorithm is used to solve the power allocation to various users by allowing each user an optimal share of power resources. Complexity time is reduced by using binary trees, which improves the system capacity and energy efficiency.

Per Lynggaard [141] proposed a system for interference detection and dynamic power allocation based on the interference level in the radio channels. In order to minimize power wastages and interference, the author applied a linear regression algorithm on Channel State Information (CSI) to predict the transmission power levels. Linear regression can handle continuous dependent variables, and its computational complexity is lower compared to other methods, such as SVM.

3.3.3. Energy Consumption and Management

Because many of the sensors and actuators in the Internet of Things are small and have limited battery capacity and charging capabilities, having energy-efficient connectivity is critical. It is, therefore, paramount to intelligently manage and allocate this scarce resource. The authors of [142] proposed a Machine Learning-based system for managing energy efficiency in IoT-based smart cities. They used Deep Artificial Neural Networks, CART decision trees, and Random Forest learning methods to predict energy consumption for IoT-based smart cities. They used real data from the Croatian energy management information system. Their results show improved energy consumption predictions compared to non-Machine Learning techniques. Isaac at el. [143] also proposed a big-data and Machine Learning technique, which they called HEMS-IoT, for IoT-based smart home's energy saving. A Deep Learning framework for intelligent energy consumption management in IoT is proposed in [144]. They proposed a novel sequential learning-based energy prediction and estimation approach with less time complexity compared to existing approaches. A summary on the application of DL for QoS enhancements in IoT is given in Table 3.

Table 3. Deep Learning application to Security and resource allocation in IoT.

QoS Measurement Factor	Application Scenarios	Learning Model	Reference
Security and Privacy	Attack classification	SVM	[100]
		Decision Trees	[100]
		Naïve Bayes	[100]
		Random Forest	[101]
	Intrusion Detection	CNN	[99,101,104,118]
		RNN	[104,107,112]
		Autoencoders	[119]
		Restricted Boltzmann machine	[106]
		Self-normalizing Neural Network (SNN)	[114,126]
		Multilayer perceptron (MLP) neural network	[101,109,111]
		LSTMs-AE	[102,103]
		LSTM	[109]
		Gated Recurrent Neural Networks	[105]
		Deep Neural Network (DNN)	[108]
		Random Forest	[127]
		Deep Belief Network (DBN)	[110]
	Defect Detection	SDPN-stacked-deep polynomial network	[127]
Resource Allocation and management	Task scheduling and resource distribution	Deep Reinforcement Learning	[136–138,145,146]
		DNN	[135]
	Power allocation and interference detection	Deep Neural Networks-DNN	[31]
	Massive channel access	Linear Regression	[139]

4. Discussion on the Application of DL to Enhance QoS in IoTs

In this age of big data, DL provides innovative analytics and offers great potential for QoS enhancement in IoT applications and networks. Various IoT networks have different QoS requirements. However, guaranteeing QoS in IoT is a challenging task. To enforce QoS in IoTs, we must ensure that two aspects are well managed: (1) Ensure network and equipment security in order to guarantee privacy and security of the network resources. and (2) Ensure that IoT network resources are well-managed, i.e., proper resource allocation and management. This paper focuses on how Deep Learning techniques have been applied in order to guarantee QoS in IoT by handling security issues and resource allocation and management challenges of the network.

IoT has the potential to revolutionize a wide range of facets of our daily lives, including school environments, health, lifestyle, environment, business, and infrastructure. Some of these aspects are so critical in our lives, and any compromise in QoS may be detrimental. It is, therefore, important that any factor that can lead to a compromise of QoS is quickly handled. IoT QoS breaches emerge from poorly managed resources or from compromising the security of IoT networks and systems. Traditional resource management methods, such as optimization and heuristics-based methods, cannot intelligently learn from the data and make appropriate actions during run-time. Deep Learning methods guarantee automatic resource management and dynamic and intelligent decision-making for large and distributed IoT networks and applications.

In Section 3, we showed the various DL algorithms and how they have been applied in IoTs for QoS enhancement and guarantee. Table 3 shows the summary of various Deep Learning models and the respective QoS metric that they have been applied to. Table 3 assists in answering various research questions as outlined in Section 1.5.

RQ1: How are Deep Learning techniques being applied for QoS enhancement in IoTs? We note that Deep Learning has been widely applied in IoT-based systems to enhance QoS through designing security and privacy DL-based models or the development of DL-based models for resource allocation and management in IoT. Concerning the Security and privacy QoS aspect in IoT-based systems, intrusion detection has received the most attention as far as the application of Deep Learning is concerned. This is attributed to the availability of public datasets, which makes it easy for researchers to implement, test, and validate their models. The attack classification has also been massively researched, but researchers mainly apply ML models, such as Decision trees, SMV, and Naïve Bayes. Defect detection has so far received the least attention, as shown in Table 3. More future research should explore the application of DL to defect detection. As far as the resource allocation and management aspect of QoS in IoT-based systems is concerned, the use of DL for task scheduling and resource distribution has received more attention from researchers compared to power allocation and interference detection and massive channel access (see Table 3).

RQ2: Which Deep Learning models are being applied to various aspects of QoS enhancement in IoT-based applications, and why those models in particular? Still from Table 3, we note that CNN and RNN are the most widely applied Deep Learning models as far as the security and privacy aspect of QoS enhancement in IoTs is concerned. However, the two models have only been applied to the intrusion detection aspect of security and privacy. In addition, other DL models that have been widely applied to intrusion detection include MLP, autoencoders, SNN, and LSTM-AE. In all the papers we reviewed, we did not find any that applied CNN, RNN, MLP, AE, SNN, or LSTM-AE to defect detection. Only SDPN was applied to defect detection. According to [127], SDPN is suitable for the development of Deep Learning models where the size of the dataset is small. This explains why Deep Learning models, such as CNN, RNN, and other data demanding DL algorithms, have not been applied to defect detection due to the scarcity of data sets in that area. For the Resource Allocation and Management aspect of QoS, Deep Reinforcement Learning (DRL) is the most widely applied DL technique, especially for task scheduling and resource distribution. DRL is able to learn progressively from its environment and learn to take appropriate actions. This reason makes RL more qualified for task scheduling tasks than other DL models that must learn from datasets. DNN is also applied to task scheduling and resource distribution but has not been widely used by researchers compared to DRL.

RQ3: Why have researchers opted for using Deep Learning techniques for QoS enhancement compared to the existing QoS enhancement approaches? In the preamble of Section 3.3, we note that resource allocation is conventionally done, using optimization methods, heuristic techniques, and game theoretical approaches, and is based on the QoS requirements of the user [31]. Optimization method approaches have challenges whenever the number of users and devices increases or when the multicellular situations are considered. The reason is that the optimization space becomes tremendously huge to satisfy the entire network; thus finding the optimal resource allocation and management solution becomes computationally too high. Heuristic and game theoretical approaches suffer from a lack of scalability, slow convergence, and information exchange overload. For these reasons, DL approaches have been used by the researcher to overcome the problems of optimization, heuristic techniques, and game theoretical approaches for resource allocation and management. Since IoT networks produce huge amounts of data, researchers have applied DL techniques [128,129] to extract useful features that can be used to dynamically and intelligently handle resource allocation efficiently, which could not be handled using traditional non-DL techniques.

RQ4: What challenges are faced by developers when applying DL models for QoS enhancement for IoTs? Four major challenges have been identified: scarcity of datasets, heterogeneity of datasets, data storage, and privacy of IoT data. The challenges are further elaborated below:

Scarcity of datasets: Generally, DL models require huge amounts of data to train. Much as IoT generates huge amounts of data, refining that data for a particular training model is also complex. Some data is not available due to data laws and policies.

Heterogeneity of data sets: IoT networks are of diverse types and each generates data with different dimensions. As such, DL models have to be developed to extract useful and relevant features from such data. Therefore, there is a need for data preprocessing and ordering in order to be fit for the respective DL models.

Storage of data: Some IoT devices have limited storage capacities, and as such, they are unable to store huge volumes of data for analysis. Data is usually sent to servers for storage. However, this increases the communication cost involved in sending data to the respective storage servers.

Privacy of IoT data: Depending on the nature of the IoT network or application, some data may be considered private and others public. In health-based IoT networks, for example, data is usually private and may not be readily available for use in many DL models.

5. Conclusions

The aim of this paper was to provide a review of how DL-based techniques have been applied to enhance QoS in the IoTs. We first give an overview of QoS in the IoTs and the most common Deep Learning techniques. We then provide a breakdown of how various DL-based techniques have been applied in IoTs in order to enhance QoS. We finally identify challenges that hinder the application of DL-based techniques for QoS enhancement in IoTs. From our review, it was observed that DL-based techniques have been widely applied in IoTs to improve some aspects of QoS measurement factors but have not been widely applied to others. For example, DL-based techniques have been widely applied to improve IoT security through intrusion detection. More so, in regard to IoT resource allocation and management, DL-based techniques have not been widely applied for massive channel access. We note the absence of research papers that provide a performance-based comparison of various DL techniques as far as improving QoS in IoT is concerned. Thus, a lack of clarity on DL algorithms that have achieved the best results as far as improvement of QoS in IoT is concerned. What is currently clear is that DL-based models are promising, and in most cases, if well trained, perform far better than the traditional techniques. In our future research, we intend to carry out a performance-based comparison study to determine which DL techniques outperform others in various aspects of QoS in IoTs. We hope this comparison will help provide insights on DL techniques that are more suitable for application in a particular QoS enhancement situation.

As a lot of research has been done on some aspects of QoS, such as intrusion detection through Deep Learning, there are some QoS aspects that have received very little attention as far as the application of DL models is concerned. Thus, we suggest future research on the application of Deep Learning to power allocation, interference detection, massive channel access, defect detection, and other QoS areas that have not been widely researched. We hope that the discussion and findings of this review paper will help researchers and professionals in the IoTs to confidently choose DL-based techniques for various QoS situations in IoT and subsequently contribute to the growth of the field.

Author Contributions: Conceptualization N.K.; Writing—original draft, N.K.; Writing—review and editing, T.P. and M.N.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Hunan province science and technology project fund (2018TP1036).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Acronym	Description	Acronym	Description
QoS	Quality of Service	GPU	Graphics Processing Unit
DL	Deep Learning	DBN	Deep Belief Network
IoT	Internet of Things	RL	Reinforcement Learning
IDS	Intrusion Detection System	SOM	Self-Organizing Map algorithm
RBF	Radial Basis Function	DRL	Deep Reinforcement Learning
SOM	Self-organizing Map algorithm	MDP	Markov Decision Process
CNN	Convolutional Neural Networks	FIFO	First In first Out
TCNN	Temporal Convolutional Neural Networks	DQN	Deep Q Networks
RNN	Recurrent Neural Network	QoE	Quality of Experience
DOS	Denial-of-Service	AML	Adversarial Machine Learning
DDOS	Distributed Denial-of-Services	DAE	Denoising autoencoders
ICA	Imperialist Competitive Algorithm	ReLU	Rectified Linear Unit activation function
MLP	Multilayer perceptron neural network	RLRA	Reinforcement Learning Resource Allocation Algorithm
SAE	Sparse autoencoders	ANN	Artificial Neural Networks
CAE	Contractive autoencoders	CSI	Channel State Information
RLMT	Reinforcement Learning-based Mapping Table	ML	Machine Learning
		RQ	Research Question
NFV	Network Function Virtualization		
MAB	Multi-Armed Bandit		

References

1. Patel, K.K.; Patel, S.M. Internet of Things-IOT: Definition, characteristics, architecture, enabling technologies, application & future challenges. *Int. J. Eng. Sci. Comput.* **2016**, *6*, 6122–6131. [CrossRef]
2. Wang, H.; Hu, J.; Deng, W. Face Feature Extraction: A Complete Review. *IEEE Access* **2018**, *6*, 6001–6039. [CrossRef]
3. Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.K.; Du, X.; Ali, I.; Guizani, M. A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1646–1685. [CrossRef]
4. Mohammadi, M.; Al-Fuqaha, A.; Sorour, S.; Guizani, M. Deep learning for IoT big data and streaming analytics: A survey. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2923–2960. [CrossRef]
5. Wang, J.; Ma, Y.; Zhang, L.; Gao, R.X.; Wu, D. Deep learning for smart manufacturing: Methods and applications. *J. Manuf. Syst.* **2018**, *48*, 144–156. [CrossRef]
6. Suryadevara, N.K.; Mukhopadhyay, S.C. Smart Home Related Research. In *Smart Homes: Design, Implementation and Issues*; Springer International Publishing: Basel, Switzerland, 2015; Volume 14, pp. 11–51. ISBN 9783319135564.
7. Jaihar, J.; Lingayat, N.; Vijaybhai, P.S.; Venkatesh, G.; Upla, K.P. Smart home automation using machine learning algorithms. In Proceedings of the 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 5–7 June 2020; pp. 8–11. [CrossRef]
8. Islam, M.M.; Rahaman, A.; Islam, M.R. Development of Smart Healthcare Monitoring System in IoT Environment. *SN Comput. Sci.* **2020**, *1*, 1–11. [CrossRef] [PubMed]
9. Budida, D.A.M.; Mangrulkar, R.S. Design and implementation of smart HealthCare system using IoT. In Proceedings of the 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 17–18 March 2017; pp. 1–7. [CrossRef]
10. Zantalis, F.; Koulouras, G.; Karabetsos, S.; Kandris, D. A review of machine learning and IoT in smart transportation. *Future Internet* **2019**, *11*, 1–23. [CrossRef]
11. Ravi, D.; Wong, C.; Lo, B.; Yang, G.Z. A Deep Learning Approach to on-Node Sensor Data Analytics for Mobile or Wearable Devices. *IEEE J. Biomed. Health Inform.* **2017**, *21*, 56–64. [CrossRef]
12. Hossain, T.; Ahad, M.A.R.; Inoue, S. A method for sensor-based activity recognition in missing data scenario. *Sensors* **2020**, *20*, 3811. [CrossRef] [PubMed]
13. Siddiqi, M.H.; Ali, R.; Rana, M.S.; Hong, E.K.; Kim, E.S.; Lee, S. Video-based human activity recognition using multilevel wavelet decomposition and stepwise linear discriminant analysis. *Sensors* **2014**, *14*, 6370–6392. [CrossRef] [PubMed]
14. Gao, X.; Luo, H.; Wang, Q.; Zhao, F.; Ye, L.; Zhang, Y. A human activity recognition algorithm based on stacking denoising autoencoder and lightGBM. *Sensors* **2019**, *19*, 947. [CrossRef]
15. Ordóñez, F.J.; Roggen, D. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors* **2016**, *16*, 115. [CrossRef]
16. Rahim, K.N.K.A.; Elamvazuthi, I.; Izhar, L.I.; Capi, G. Classification of human daily activities using ensemble methods based on smartphone inertial sensors. *Sensors* **2018**, *18*, 4132. [CrossRef] [PubMed]
17. Thapa, K.; Abdullah Al, Z.M.; Lamichhane, B.; Yang, S.H. A deep machine learning method for concurrent and interleaved human activity recognition. *Sensors* **2020**, *20*, 5770. [CrossRef]

18. Chen, R.; Chu, T.; Liu, K.; Liu, J.; Chen, Y. Inferring human activity in mobile devices by computing multiple contexts. *Sensors* **2015**, *15*, 21219–21238. [CrossRef]
19. Chen, K.; Zhang, D.; Yao, L.; Guo, B.; Yu, Z.; Liu, Y. Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities. *arXiv* **2020**. Available online: http://arxiv.org/abs/2001.07416 (accessed on 12 May 2021).
20. Wang, L. Recognition of human activities using continuous autoencoders with wearable sensors. *Sensors* **2016**, *16*, 189. [CrossRef] [PubMed]
21. Moyne, J.; Iskandar, J. Big Data Analytics for Smart Manufacturing: Case Studies in Semiconductor Manufacturing. *Processes* **2017**, *5*, 39. [CrossRef]
22. Qu, Y.J.; Ming, X.G.; Liu, Z.W.; Zhang, X.Y.; Hou, Z.T. Smart manufacturing systems: State of the art and future trends. *Int. J. Adv. Manuf. Technol.* **2019**, *103*, 3751–3768. [CrossRef]
23. Mahdavi-Hezavehi, S.; Avgeriou, P.; Weyns, D. A Classification Framework of Uncertainty in Architecture-Based Self-Adaptive Systems With Multiple Quality Requirements. In *Managing Trade-Offs in Adaptable Software Architectures*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 45–77. ISBN 9780128028551. [CrossRef]
24. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of Things for Smart Cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [CrossRef]
25. Rahman, M.M.; Shafiullah, M.; Rahman, S.M.; Khondaker, A.N.; Amao, A.; Zahir, M.H. Soft computing applications in air quality modeling: Past, present, and future. *Sustainability* **2020**, *12*, 4045. [CrossRef]
26. Lee, C.-N.; Huang, T.-H.; Wu, C.-M.; Tsai, M.-C. The Internet of Things and Its Applications. In *Big Data Analytics for Sensor-Network Collected Intelligence*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 256–279. ISBN 9780128096253. [CrossRef]
27. Andročec, D.; Vrček, N. Machine Learning for the Internet of Things Security: A Systematic Review. In *Proceedings of the 13th International Conference on Software Technologies*; SCITEPRESS—Science and Technology Publications: Setúbal, Portugal, 2018; pp. 563–570. [CrossRef]
28. Fraga-Lamas, P.; Ramos, L.; Mondéjar-Guerra, V.; Fernández-Caramés, T.M. A review on IoT deep learning UAV systems for autonomous obstacle detection and collision avoidance. *Remote Sens.* **2019**, *11*, 2144. [CrossRef]
29. Lateef, A.A.A.; Al-Janabi, S.T.F.; Al-Khateeb, B. Survey on intrusion detection systems based on deep learning. *Period. Eng. Nat. Sci.* **2019**, *7*, 1074–1095. [CrossRef]
30. Asharf, J.; Moustafa, N.; Khurshid, H.; Debie, E.; Haider, W.; Wahab, A. A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions. *Electronics* **2020**, *9*, 1177. [CrossRef]
31. Hussain, F.; Hassan, S.A.; Hussain, R.; Hossain, E. Machine Learning for Resource Management in Cellular and IoT Networks: Potentials, Current Solutions, and Open Challenges. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1251–1275. [CrossRef]
32. Al-amri, R.; Murugesan, R.K.; Man, M.; Abdulateef, A.F.; Al-Sharafi, M.A.; Alkahtani, A.A. A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data. *Appl. Sci.* **2021**, *11*, 5320. [CrossRef]
33. Alsoufi, M.A.; Razak, S.; Siraj, M.M.; Nafea, I.; Ghaleb, F.A.; Saeed, F.; Nasser, M. Anomaly-Based Intrusion Detection Systems in IoT Using Deep Learning: A Systematic Literature Review. *Appl. Sci.* **2021**, *11*, 8383. [CrossRef]
34. Aversano, L.; Bernardi, M.L.; Cimitile, M.; Pecori, R. A systematic review on Deep Learning approaches for IoT security. *Comput. Sci. Rev.* **2021**, *40*, 100389. [CrossRef]
35. Clarke, R.I. Cataloging Research by Design: A Taxonomic Approach to Understanding Research Questions in Cataloging. *Cat. Classif. Q.* **2018**, *56*, 683–701. [CrossRef]
36. Mathy, L.; Edwards, C.; Hutchison, D. Principles of QoS in group communications. *Telecommun. Syst.* **1999**, *11*, 59–84. [CrossRef]
37. Singh, M.; Baranwal, G. Quality of Service (QoS) in Internet of Things. In Proceedings of the Proceedings—2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU 2018), Bhimtal, India, 23–24 February 2018; pp. 1–6. [CrossRef]
38. Bernard, M.S.; Pei, T.; Nasser, K. QoS Strategies for Wireless Multimedia Sensor Networks in the Context of IoT at the MAC Layer, Application Layer, and Cross-Layer Algorithms. *J. Comput. Netw. Commun.* **2019**, *2019*, 9651915. [CrossRef]
39. Tanganelli, G.; Vallati, C.; Mingozzi, E. Ensuring quality of service in the internet of things. *Stud. Comput. Intell.* **2018**, *715*, 139–163. [CrossRef]
40. Liao, Y.; Li, Y.F.; Shen, X.F.; Zhang, S.M.; Zhao, M.; Tan, X.H. QoS enhancement in space data communication: A network coding approach. *Int. J. Electron.* **2017**, *104*, 34–46. [CrossRef]
41. Verizon.com. “Bandwidth”, Verizon.com. 2010. Available online: https://www.verizon.com/info/definitions/bandwidth/ (accessed on 2 February 2021).
42. Tawalbeh, L.; Muheidat, F.; Tawalbeh, M.; Quwaider, M. IoT Privacy and Security: Challenges and Solutions. *Appl. Sci.* **2020**, *10*, 4102. [CrossRef]
43. Verma, A.; Ranga, V. Machine Learning Based Intrusion Detection Systems for IoT Applications. *Wirel. Pers. Commun.* **2020**, *111*, 2287–2310. [CrossRef]
44. Mahmoud, R.; Yousuf, T.; Aloul, F.; Zualkernan, I. Internet of things (IoT) security: Current status, challenges and prospective measures. In Proceedings of the 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), London, UK, 14–16 December 2015; pp. 336–341. [CrossRef]
45. HaddadPajouh, H.; Khayami, R.; Dehghantanha, A.; Choo, K.K.R.; Parizi, R.M. AI4SAFE-IoT: An AI-powered secure architecture for edge layer of Internet of things. *Neural Comput. Appl.* **2020**, *32*, 16119–16133. [CrossRef]

46. Chana, I.; Singh, S. Quality of Service and Service Level Agreements for Cloud Environments: Issues and Challenges. In *Cloud Computing: Challenges, Limitations and R&D Solutions*; Mahmood, Z., Ed.; Springer International Publishing: Berlin/Heidelberg, Germany, 2014; pp. 51–72. [CrossRef]
47. Anawar, M.R.; Wang, S.; Azam Zia, M.; Jadoon, A.K.; Akram, U.; Raza, S. Fog Computing: An Overview of Big IoT Data Analytics. *Wirel. Commun. Mob. Comput.* **2018**, *2018*. [CrossRef]
48. Xu, Z.; Liu, W.; Huang, J.; Yang, C.; Lu, J.; Tan, H. Artificial Intelligence for Securing IoT Services in Edge Computing: A Survey. *Secur. Commun. Netw.* **2020**, *2020*, 1–13. [CrossRef]
49. Esmaeilian, B.; Behdad, S.; Wang, B. The evolution and future of manufacturing: A review. *J. Manuf. Syst.* **2016**, *39*, 79–100. [CrossRef]
50. Vazan, P.; Janikova, D.; Tanuska, P.; Kebisek, M.; Cervenanska, Z. Using data mining methods for manufacturing process control. *IFAC-PapersOnLine* **2017**, *50*, 6178–6183. [CrossRef]
51. Rashid, M.M.; Kamruzzaman, J.; Hassan, M.M.; Imam, T.; Gordon, S. Cyberattacks detection in iot-based smart city applications using machine learning techniques. *Int. J. Environ. Res. Public Health* **2020**, *17*, 9347. [CrossRef] [PubMed]
52. Vindya, N.D.; Vedamurthy, H.K. Machine Learning Algorithm in Smart Farming for Crop Identification. In *Computational Vision and Bio-Inspired Computing*; Smys, S., Tavares, J.M.R.S., Balas, V.E., Iliyasu, A.M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 19–25. ISBN 978-3-030-37218-7. [CrossRef]
53. Balducci, F.; Impedovo, D.; Pirlo, G. Machine learning applications on agricultural datasets for smart farm enhancement. *Machines* **2018**, *6*, 38. [CrossRef]
54. Escamilla-García, A.; Soto-Zarazúa, G.M.; Toledano-Ayala, M.; Rivas-Araiza, E.; Gastélum-Barrios, A. Applications of Artificial Neural Networks in Greenhouse Technology and Overview for Smart Agriculture Development. *Appl. Sci.* **2020**, *10*, 3835. [CrossRef]
55. Attal, F.; Mohammed, S.; Dedabrishvili, M.; Chamroukhi, F.; Oukhellou, L.; Amirat, Y. Physical human activity recognition using wearable sensors. *Sensors* **2015**, *15*, 31314–31338. [CrossRef] [PubMed]
56. Islam, S.M.R.; Kwak, D.; Kabir, H.; Hossain, M.; Kwak, K.-S. The Internet of Things for Health Care : A Comprehensive Survey. *Access IEEE* **2015**, *3*, 678–708. [CrossRef]
57. Ferre, M.; Batista, E.; Solanas, A.; Martínez-Ballesté, A. Smart Health-Enhanced Early Mobilisation in Intensive Care Units. *Sensors* **2021**, *21*, 5408. [CrossRef] [PubMed]
58. Internetofbusiness.com. Athens International Airport Turns to IoT for Environmental Monitoring. *Internet of Business News*. 2020. Available online: https://internetofbusiness.com/athens-international-airport-turns-to-iot-for-environmental-monitoring/ (accessed on 3 April 2021).
59. Oskouei, R.J.; MousaviLou, Z.; Bakhtiari, Z.; Jalbani, K.B. IoT-Based Healthcare Support System for Alzheimer's Patients. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 1–15. [CrossRef]
60. Ball, J.E.; Anderson, D.T.; Chan, C.S. A comprehensive survey of deep learning in remote sensing: Theories, tools and challenges for the community. *J. Appl. Remote Sens.* **2017**, *11*. [CrossRef]
61. Hinton, G.E. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
62. Hirasawa, K.; Ohbayashi, M.; Koga, M.; Harada, M. Forward propagation universal learning network. In Proceedings of the International Conference on Neural Networks (ICNN'96), Washington, DC, USA, 3–6 June 1996; Volume 1, pp. 353–358. [CrossRef]
63. Kim, P. Convolutional Neural Network. In *MATLAB Deep Learning*; Apress: Berkeley, CA, USA, 2017; pp. 121–147.
64. Abdel-Hamid, O.; Mohamed, A.R.; Jiang, H.; Deng, L.; Penn, G.; Yu, D. Convolutional neural networks for speech recognition. *IEEE Trans. Audio Speech Lang. Process.* **2014**, *22*, 1533–1545. [CrossRef]
65. Passricha, V.; Kumar Aggarwal, R. Convolutional Neural Networks for Raw Speech Recognition. In *From Natural to Artificial Intelligence—Algorithms and Applications*; IntechOpen: London, UK, 2018. [CrossRef]
66. Haque, M.A.; Verma, A.; Alex, J.S.R.; Venkatesan, N. Experimental Evaluation of CNN Architecture for Speech Recognition. In *First International Conference on Sustainable Technologies for Computational Intelligence. Advances in Intelligent Systems and Computing*; Luhach, A., Kosa, J., Poonia, R., Gao, X.Z., Eds.; Springer: Singapore, 2020; pp. 507–514. ISBN 978-981-15-0028-2. [CrossRef]
67. Wang, W.; Gang, J. Application of Convolutional Neural Network in Natural Language Processing. In Proceedings of the 2018 International Conference on Information Systems and Computer Aided Education (ICISCAE), Changchun, China, 6–8 July 2018; pp. 64–70. [CrossRef]
68. Fesseha, A.; Xiong, S.; Emiru, E.D.; Diallo, M.; Dahou, A. Text Classification Based on Convolutional Neural Networks and Word Embedding for Low-Resource Languages: Tigrinya. *Information* **2021**, *12*, 52. [CrossRef]
69. Oppermann, A. Deep Learning Meets Physics: Restricted Boltzmann Machines. 2018. Available online: https://towardsdatascience.com/deep-learning-meets-physics-restricted-boltzmann-machines-part-i-6df5c4918c15 (accessed on 4 April 2021).
70. Kishore, D.R.; Kaur, T. Backpropagation Algorithm: An Artificial Neural Network Approach for Pattern Recognition. *Int. J. Sci. Eng. Res.* **2012**, *3*, 6–9.
71. Autoencoders. UFLDL Tutorial. Available online: http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders (accessed on 4 April 2021).

72. Jordan, J. Introduction to Autoencoders. Jordan, Jeremy Website. 2018. Available online: https://www.jeremyjordan.me/autoencoders/ (accessed on 4 April 2021).
73. Dertat, A. Applied Deep Learning: Autoencoders. 2017. Available online: https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798 (accessed on 4 April 2021).
74. Lu, M.; Mou, Y. Bearing defect classification algorithm based on autoencoder neural network. *Adv. Civ. Eng.* **2020**, *2020*, 1DUMM. [CrossRef]
75. Rosati, S.; Balestra, G.; Knaflitz, M. Comparison of different sets of features for human activity recognition by wearable sensors. *Sensors* **2018**, *18*, 4189. [CrossRef] [PubMed]
76. Wang, Q.; Ye, L.; Luo, H.; Men, A.; Zhao, F.; Huang, Y. Pedestrian stride-length estimation based on LSTM and denoising autoencoders. *Sensors* **2019**, *19*, 840. [CrossRef]
77. Eklund, A.; Dufort, P.; Forsberg, D.; LaConte, S.M. Medical image processing on the GPU—Past, present and future. *Med. Image Anal.* **2013**, *17*, 1073–1094. [CrossRef]
78. Kandel, I.; Castelli, M. Transfer learning with convolutional neural networks for diabetic retinopathy image classification. A review. *Appl. Sci.* **2020**, *10*, 2021. [CrossRef]
79. Ferreira, D.; Silva, S.; Abelha, A.; Machado, J. Recommendation system using autoencoders. *Appl. Sci.* **2020**, *10*, 5510. [CrossRef]
80. Barbieri, J.; Alvim, L.G.M.; Braida, F.; Zimbrão, G. Autoencoders and recommender systems: COFILS approach. *Expert Syst. Appl.* **2017**, *89*, 81–90. [CrossRef]
81. Liu, Y.; Wang, S.; Khan, M.S.; He, J. A novel deep hybrid recommender system based on auto-encoder with neural collaborative filtering. *Big Data Min. Anal.* **2018**, *1*, 211–221. [CrossRef]
82. Amidi, A.; Amidi, S. "Recurrent Neural Networks Cheatsheet", Stanford University. Available online: https://stanford.edu/~{}shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks (accessed on 12 May 2021).
83. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
84. Goldberg, Y. Neural Network Methods for Natural Language Processing. *Synth. Lect. Hum. Lang. Technol.* **2017**, *10*, 1–311. [CrossRef]
85. Zheng, W.; Wu, G.; Qie, W.; Zhang, Y. Deep Reinforcement Learning for Joint Channel Selection and Power Allocation in Cognitive Internet of Things. In *Human Centered Computing*; Milošević, D., Tang, Y., Zu, Q., Eds.; Springer: Cham, Switzerland, 2019; pp. 683–692. [CrossRef]
86. Di Felice, M.; Bedogni, L.; Bononi, L. Reinforcement Learning-Based Spectrum Management for Cognitive Radio Networks: A Literature Review and Case Study. In *Handbook of Cognitive Radio*; Zhang, W., Ed.; Springer: Singapore, 2018; pp. 1–38.
87. Yau, K.-L.A.; Poh, G.-S.; Chien, S.F.; Al-Rawi, H.A.A. Application of Reinforcement Learning in Cognitive Radio Networks: Models and Algorithms. *Sci. World J.* **2014**, *2014*, 209810. [CrossRef] [PubMed]
88. Preferred Networks, Inc. Chainer. Available online: https://chainer.org/ (accessed on 3 March 2021).
89. Berkeley AI Research. Caffe. Available online: https://caffe.berkeleyvision.org/ (accessed on 3 March 2021).
90. Microsoft. CNTK. Available online: https://docs.microsoft.com/en-us/cognitive-toolkit/ (accessed on 3 March 2021).
91. The Apache Software Foundation (ASF). MXNet. Available online: https://mxnet.apache.org/versions/1.8.0/ (accessed on 3 March 2021).
92. Konduit. DeepLearning4j. Available online: https://deeplearning4j.org/ (accessed on 3 March 2021).
93. Chollet, F.; Rahman, F.; Zhu, Q.C.; Lee, T.; De Marmiesse, G.; Zabluda, O.; Pumperla, M.; Santana, E.; McColgan, T.; Snelgrove, X.; et al. Keras. Available online: https://keras.io/Keras (accessed on 3 March 2021).
94. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G. Pytorch. Available online: https://pytorch.org/ (accessed on 3 March 2021).
95. Google. Tensor Flow. Available online: https://www.tensorflow.org/ (accessed on 3 March 2021).
96. Hope, T.; Resheff, Y.S.; Lieder, I. *Learning TensorFlow: A Guide to Building Deep Learning Systems*, 1st ed; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2017. ISBN 9781491978511.
97. Liang, S.D. Smart and Fast Data Processing for Deep Learning in Internet of Things: Less is More. *IEEE Internet Things J.* **2019**, *6*, 5981–5989. [CrossRef]
98. Kok, I.; Corak, B.H.; Yavanoglu, U.; Ozdemir, S. Deep Learning based Delay and Bandwidth Efficient Data Transmission in IoT. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; pp. 2327–2333. [CrossRef]
99. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Gener. Comput. Syst.* **2019**, *100*, 779–796. [CrossRef]
100. Sai Kiran, K.V.V.N.L.; Devisetty, R.N.K.; Kalyan, N.P.; Mukundini, K.; Karthi, R. Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques. In *Procedia Computer Science*; Elsevier B.V.: Amsterdam, The Netherlands, 2020; Volume 171, pp. 2372–2379. [CrossRef]
101. Susilo, B.; Riri, F.S. Intrusion Detection in IoT Networks Using Deep Learning Algorithm. *Information* **2020**, *11*, 279. [CrossRef]
102. Xu, Y.; Tang, Y.; Yang, Q. Deep Learning for IoT Intrusion Detection based on LSTMs-AE. In *Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture*; ACM: New York, NY, USA, 2020; pp. 64–68. [CrossRef]
103. Ashfaq Khan, M.; Kim, Y. Deep Learning-Based Hybrid Intelligent Intrusion Detection System. *Comput. Mater. Contin.* **2021**, *68*, 671–687. [CrossRef]

104. Lopez-Martin, M.; Carro, B.; Sanchez-Esguevillas, A.; Lloret, J. Network Traffic Classifier with Convolutional and Recurrent Neural Networks for Internet of Things. *IEEE Access* **2017**, *5*, 18042–18050. [CrossRef]
105. Putchala, M.K. Deep Learning Approach for Intrusion Detection System (Ids) in the Internet of Things (Iot) Network Using Gated Recurrent Neural Networks (Gru). Master's Thesis, Wright State University, Dayton, OH, USA, 2017; pp. 1188–1197. Available online: https://corescholar.libraries.wright.edu/etd_all/1848/ (accessed on 12 May 2021).
106. Dawoud, A.; Sianaki, O.A.; Shahristani, S.; Raun, C. Internet of Things Intrusion Detection: A Deep Learning Approach. In Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence (SSCI), Canberra, ACT, Australia, 1–4 December 2020; pp. 1516–1522. [CrossRef]
107. Roy, B.; Cheung, H. A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network. In Proceedings of the 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, Australia, 21–23 November 2018; pp. 1–6. [CrossRef]
108. Diro, A.A.; Chilamkurti, N. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Gener. Comput. Syst.* **2018**, *82*, 761–768. [CrossRef]
109. Roopak, M.; Yun Tian, G.; Chambers, J. Deep Learning Models for Cyber Security in IoT Networks. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 452–457. [CrossRef]
110. Thamilarasu, G.; Chawla, S. Towards deep-learning-driven intrusion detection for the internet of things. *Sensors* **2019**, *19*, 1977. [CrossRef] [PubMed]
111. Mohammadi, S.; Amiri, F. An Efficient Hybrid Self-Learning Intrusion Detection System Based on Neural Networks. *Int. J. Comput. Intell. Appl.* **2019**, *18*, 1950001. [CrossRef]
112. Ferrag, M.A.; Maglaras, L. DeepCoin: A Novel Deep Learning and Blockchain-Based Energy Exchange Framework for Smart Grids. *IEEE Trans. Eng. Manag.* **2020**, *67*, 1285–1297. [CrossRef]
113. Alsamiri, J.; Alsubhi, K. Internet of things cyber attacks detection using machine learning. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 627–634. [CrossRef]
114. Aldhaheri, S.; Alghazzawi, D.; Cheng, L.; Alzahrani, B.; Al-Barakati, A. DeepDCA: Novel Network-Based Detection of IoT Attacks Using Artificial Immune System. *Appl. Sci.* **2020**, *10*, 1909. [CrossRef]
115. Greensmith, J.; Aickelin, U.; Cayzer, S. Detecting Danger: The Dendritic Cell Algorithm. 2010. Available online: http://arxiv.org/abs/1006.5008 (accessed on 12 May 2021).
116. Klambauer, G.; Unterthiner, T.; Mayr, A.; Hochreiter, S. Self-Normalizing Neural Networks. 2017. Available online: http://arxiv.org/abs/1706.02515 (accessed on 12 May 2021).
117. Soe, Y.N.; Santosa, P.I.; Hartanto, R. DDoS Attack Detection Based on Simple ANN with SMOTE for IoT Environment. In Proceedings of the 2019 Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 16–17 October 2019; pp. 1–5. [CrossRef]
118. Ge, M.; Fu, X.; Syed, N.; Baig, Z.; Teo, G.; Robles-Kelly, A. Deep Learning-Based Intrusion Detection for IoT Networks. In Proceedings of the 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC), Kyoto, Japan, 1–3 December 2019; pp. 256–25609. [CrossRef]
119. 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.* **2018**, *41*, 1–11. [CrossRef]
120. Tan, Z.; Jamdagni, A.; He, X.; Nanda, P.; Liu, R.P.; Hu, J. Detection of Denial-of-Service Attacks Based on Computer Vision Techniques. *IEEE Trans. Comput.* **2015**, *64*, 2519–2533. [CrossRef]
121. Ambusaidi, M.A.; He, X.; Nanda, P.; Tan, Z. Building an intrusion detection system using a filter-based feature selection algorithm. *IEEE Trans. Comput.* **2016**, *65*, 2986–2998. [CrossRef]
122. Tsai, C.-F.; Lin, C.-Y. A triangle area based nearest neighbors approach to intrusion detection. *Pattern Recognit.* **2010**, *43*, 222–229. [CrossRef]
123. Moustafa, N.; Creech, G.; Slay, J. Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models. In *Data Analytics and Decision Support for Cybersecurity: Trends, Methodologies and Applications*; Palomares Carrascosa, I., Kalutarage, H.K., Huang, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 127–156. ISBN 978-3-319-59439-2.
124. Lopes, N.; Ribeiro, B. Deep Belief Networks (DBNs). In *Machine Learning for Adaptive Many-Core Machines—A Practical Approach*; Springer: Cham, Switzerland, 2015; pp. 155–186. ISBN 978-3-319-06938-8.
125. Zhong, M.; Zhou, Y.; Chen, G. Sequential model based intrusion detection system for iot servers using deep learning methods. *Sensors* **2021**, *21*, 1113. [CrossRef]
126. Ibitoye, O.; Shafiq, O.; Matrawy, A. Analyzing Adversarial Attacks against Deep Learning for Intrusion Detection in IoT Networks. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6. [CrossRef]
127. Salman, O.; Elhajj, I.H.; Chehab, A.; Kayssi, A. A machine learning based framework for IoT device identification and abnormal traffic detection. *Trans. Emerg. Telecommun. Technol.* **2019**. [CrossRef]
128. Zhang, C.; Cheng, X.; Liu, J.; He, J.; Liu, G. Deep Sparse Autoencoder for Feature Extraction and Diagnosis of Locomotive Adhesion Status. *J. Control Sci. Eng.* **2018**, *2018*, 8676387. [CrossRef]

129. Kim, M.; Kang, D.; Lee, N. Feature Extraction from Oriental Painting for Wellness Contents Recommendation Services. *IEEE Access* **2019**, *7*, 59263–59270. [CrossRef]
130. Wang, X.; Sheng, M.-J.; Lou, Y.-Y.; Shih, Y.-Y.; Chiang, M. Internet of Things Session Management Over LTE—Balancing Signal Load, Power, and Delay. *IEEE Internet Things J.* **2016**, *3*, 339–353. [CrossRef]
131. Na, W.; Jang, S.; Lee, Y.; Park, L.; Dao, N.-N.; Cho, S. Frequency Resource Allocation and Interference Management in Mobile Edge Computing for an Internet of Things System. *IEEE Internet Things J.* **2019**, *6*, 4910–4920. [CrossRef]
132. Muwonge, B.S.; Pei, T.; Otim, J.S.; Mayambala, F. A joint power, delay and rate optimization model for secondary users in cognitive radio sensor networks. *Sensors* **2020**, *20*, 4907. [CrossRef]
133. Munaye, Y.Y.; Juang, R.-T.; Lin, H.-P.; Tarekegn, G.B.; Lin, D.-B. Deep Reinforcement Learning Based Resource Management in UAV-Assisted IoT Networks. *Appl. Sci.* **2021**, *11*, 2163. [CrossRef]
134. Bashir, H.; Lee, S.; Kim, K.H. Resource allocation through logistic regression and multicriteria decision making method in IoT fog computing. *Trans. Emerg. Telecommun. Technol.* **2019**. [CrossRef]
135. Zhang, W.; Yang, D.; Haixia, P.; Wu, W.; Quan, W.; Zhang, H.; Shen, X. Deep Reinforcement Learning Based Resource Management for DNN Inference in Industrial IoT. *IEEE Trans. Veh. Technol.* **2021**, *70*. [CrossRef]
136. Deng, S.; Xiang, Z.; Zhao, P.; Taheri, J.; Gao, H.; Yin, J.; Zomaya, A.Y. Dynamical Resource Allocation in Edge for Trustable Internet-of-Things Systems: A Reinforcement Learning Method. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6103–6113. [CrossRef]
137. Gai, K.; Qiu, M. Optimal resource allocation using reinforcement learning for IoT content-centric services. *Appl. Soft Comput.* **2018**, *70*, 12–21. [CrossRef]
138. Shah, H.A.; Zhao, L. Multiagent Deep-Reinforcement-Learning-Based Virtual Resource Allocation through Network Function Virtualization in Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 3410–3421. [CrossRef]
139. Ma, J.; Nagatsuma, T.; Kim, S.-J.; Hasegawa, M. A Machine-Learning-Based Channel Assignment Algorithm for IoT. In Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Okinawa, Japan, 11–13 February 2019; pp. 1–6. [CrossRef]
140. Sun, Y.; Wang, Y.; Jiao, J.; Wu, S.; Zhang, Q. Deep Learning-Based Long-Term Power Allocation Scheme for NOMA Downlink System in S-IoT. *IEEE Access* **2019**, *7*, 86288–86296. [CrossRef]
141. Lynggaard, P. Using Machine Learning for Adaptive Interference Suppression in Wireless Sensor Networks. *IEEE Sens. J.* **2018**, *18*, 8820–8826. [CrossRef]
142. Zekić-Sušac, M.; Mitrović, S.; Has, A. Machine learning based system for managing energy efficiency of public sector as an approach towards smart cities. *Int. J. Inf. Manag.* **2021**, *58*, 102074. [CrossRef]
143. Machorro-Cano, I.; Alor-Hernández, G.; Paredes-Valverde, M.A.; Rodríguez-Mazahua, L.; Sánchez-Cervantes, J.L.; Olmedo-Aguirre, J.O. HEMS-IoT: A big data and machine learning-based smart home system for energy saving. *Energies* **2020**, *13*, 1097. [CrossRef]
144. Han, T.; Muhammad, K.; Hussain, T.; Lloret, J.; Baik, S.W. An Efficient Deep Learning Framework for Intelligent Energy Management in IoT Networks. *IEEE Internet Things J.* **2021**, *8*, 3170–3179. [CrossRef]
145. Xiong, X.; Zheng, K.; Lei, L.; Hou, L. Resource allocation based on deep reinforcement learning in IoT edge computing. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1133–1146. [CrossRef]
146. Elgendy, I.A.; Muthanna, A.; Hammoudeh, M.; Shaiba, H.; Unal, D.; Khayyat, M. Advanced Deep Learning for Resource Allocation and Security Aware Data Offloading in Industrial Mobile Edge Computing. *Big Data* **2021**, *9*, 265–278. [CrossRef] [PubMed]