applied sciences

MDPI

Article

# Deep Learning at the Mobile Edge: Opportunities for 5G Networks

**Miranda McClellan, Cristina Cervelló-Pastor * and Sebastià Sallent ***

Department of Network Engineering, Universitat Politècnica de Catalunya (UPC), Esteve Terradas, 7, 08860 Castelldefels, Spain; miranda.mcclellan@upc.edu

* Correspondence: cristina@entel.upc.edu (C.C.-P.); sallent@entel.upc.edu (S.S.)

Received: 2 June 2020; Accepted: 7 July 2020; Published: 9 July 2020


**Abstract:** Mobile edge computing (MEC) within 5G networks brings the power of cloud computing, storage, and analysis closer to the end user. The increased speeds and reduced delay enable novel applications such as connected vehicles, large-scale IoT, video streaming, and industry robotics. Machine Learning (ML) is leveraged within mobile edge computing to predict changes in demand based on cultural events, natural disasters, or daily commute patterns, and it prepares the network by automatically scaling up network resources as needed. Together, mobile edge computing and ML enable seamless automation of network management to reduce operational costs and enhance user experience. In this paper, we discuss the state of the art for ML within mobile edge computing and the advances needed in automating adaptive resource allocation, mobility modeling, security, and energy efficiency for 5G networks.

**Keywords:** 5G; edge network; deep learning; reinforcement learning; caching; task offloading; mobile computing; edge computing; mobile edge computing; cloud computing; network function virtualization; slicing; 5G network standardization

## 1. Introduction

By 2024, 5G mobile edge computing (MEC) is expected to be a multi-million-dollar industry with enterprise deployments reaching $73M [1]. Each year, the complexity of data continues to grow. The rise of network complexity systems stems from the increase of on-demand and customizable services. Internet service providers must accommodate traffic for web browsing, connected vehicles, video streaming, online gaming, voice over IP and always-on Internet of Things (IoT) device transmissions. New constraints introduced by on-demand services as listed above require a radical transformation of fixed and mobile access networks.

Fifth-generation (5G) mobile networks are being developed to serve the increasing levels of traffic demand and diversity. To cope with the complex traffic demanded by modern users, network operators are adopting cloud-computing techniques. 5G networks will use software-defined networks (SDN) and network function virtualization (NFV) to reduce the operational cost of growing mobile networks to provide on-demand services. Long-term, end users can expect performance enhancements because 5G is optimized to provide low-latency, high-availability, and high-bandwidth communication for multiple use cases, including delay-sensitive applications, such as autonomous vehicles and automated Industry 4.0 robotics.

The stringent functional requirements for 5G networks have forced designers to rethink the backbone and access network architectures to better support core functions and dynamic network services. The introduction of mobile edge computing disrupts the traditional separation between the access network (secure and reliable transport between end users) and the core network (information computing and storage). The combination of mobile edge computing and cloud computing blends

the control and management planes to enable an extension of virtualized network resources and the creation of end-to-end services based on SDN, NFV, and slicing techniques in 5G. mobile edge computing can then complement the goals of the access network to solve existing challenges including quality of service/experience, security, and power consumption as part of the necessary network transformation.

Alongside NFV and SDN, mobile edge computing was recognized by the European 5GPPP (Public Private Partnership) as a key enabling technology that will help satisfy the demanding requirements for throughput, latency, scalability and automation in 5G [2]. Mobile edge computing places computational processing power closer to the end user. This proximity alleviates the amount of traffic delivered across the core network to large data centers, improves response speed with latencies below ten milliseconds [3], and coordinates with data centers to offload some computational tasks, such as online inference from the main cloud. Mobile edge computing can enable real-time analysis through cloud-computing capabilities in a secure and context-aware manner with collaboration between network operators and application providers [2].

Managing thousands of heterogeneous connections under strict response constraints for applications, service creation, and network administration presents a complex challenge to 5G networks using mobile edge computing. To realize the benefits of mobile edge computing, there is a need to develop automated procedures to provide, orchestrate, and manage network services and applications under conditions that change over time and across locality. A promising solution is introducing machine learning (ML) to network operations to meet this new set of demands that are beyond the limitations of traditional optimization techniques.

The development of 5G core network and mobile edge computing division of labor depends on automated network management that is powered by efficient machine learning (ML) techniques. Traditional optimization techniques are not adaptable enough to handle the complex, real-time analysis required in 5G networks. In the past 20 years, machine learning has become widely known for pattern recognition.

A subset of ML, deep learning (DL), has been extensively researched and applied within the fields of computer vision [4] and natural language processing [5]. 5G networks can be enhanced to automatically configure, optimize, secure, and recover using the cognitive power of DL, even though this technique also introduces open issues in real-time response, energy consumption and optimization of OPEX and CAPEX. Together, cloud-based technologies and automation with DL in mobile edge computing will increase resource use and efficiency, increase resiliency, optimize power consumption, increase revenues and provide ease of operation for service providers.

Previous surveys have focused on categorizing and evaluating various aspects of edge computing machine learning algorithms applied to the edge of the network [6], on creating a taxonomy and description of challenges in distributed network computing [7], and the integration of mobile edge computing in 5G [8]. The authors in [6] present an in-depth theoretical study on the impact that the communications network has on ML algorithms and vice versa, while analyzing technical case studies. A survey of the integration of the mobile edge computing with 5G, focused on the various access mechanisms and technologies is presented in [8], with particular attention paid to aspects such as network densification, energy harvesting, or ML and its application to various verticals. The application of network defense through anomaly detection and attacks using deep learning is discussed in [9], and [10] focuses on creating intelligent and secure vehicular networks in a similarly narrow manner. The authors in [11] focus on deep reinforcement learning to create intelligent elastic optical network architectures. Finally, the incorporation of MAC protocols for heterogeneous wireless networks using deep learning is investigated in [12].

However, the previous work lacks the unique perspective provided in this paper in which we explain the novel applications for ML-deep learning models in mobile edge computing and the challenges the industry must overcome to ensure the success of new service automation in edge computing. We recognize the promise of using deep learning algorithms, though they do not resolve

all obstacles towards the full automation of mobile edge computing. We were motivated to carry out this survey to investigate the potential and challenges introduced by deploying deep learning at scale at the mobile edge.

*Contributions*

The contribution of this document is three-fold: **1)** Create a taxonomy of distributed computing and storage resources that involve connected the edge of the network with end users and devices, **2)** discuss the application of deep learning to edge computing to meet the functional requirements of 5G networks, and **3)** provide an overview of new applications, standardization efforts, and challenges that have arisen from introducing deep learning into mobile edge computing in 5G networks.

In this paper, we explain the work undertaken and the challenges in applying the power of deep learning in 5G mobile edge computing to serve low-latency, real-time applications by providing adaptive, application-specific resource allocation, and security, and accommodating high user mobility. With DL, mobile edge computing can drive 5G networks to meet the stringent requirements imposed by a wide range of applications, such as real time, security and energy efficiency in the industry environment.

The rest of the paper is organized as follows: Section 2, presents an overview of 5G and mobile edge computing enabling technologies. Section 3 offers a background to deep learning (DL) techniques commonly used in network management. Section 4 discusses the current work and addresses open issues in mobile edge computing that could be solved by further interdisciplinary ML work. Section 5 also provides an overview of protocols and architectures recently designed to automate network management with ML. Section 6 discusses state-of-the-art applications and use cases that mobile edge computing hopes to enable, including autonomous vehicles, industrial robotics, and massive IoT scale-up. The paper concludes in Section 7.

## 2. Mobile Edge Computing

This section is divided into two parts. In the first part, the three main functional requirements of the 5G network are introduced to show that its full deployment requires computing, storage, and network infrastructure close to the user and the infrastructure, whether fixed or mobile, of the end user. The second part introduces a mobile edge computing taxonomy, clarifying the functionalities and the geographic areas that edge computing covers to demonstrate the essentiality of mobile edge computing in 5G deployments.

At the same time, EC is shown to be a cornerstone of 5G deployment. Addressing the rapidly changing Internet demand requires rethinking network and information delivery designs. A combination of newly developed 5G networks and mobile edge computing (MEC) will enable Internet service providers (ISPs) to meet consumer demands.

### *2.1. 5G Network Purpose and Design*

Each generation of mobile network standards has been designed in response to the changing use of mobile communications. 4G and LTE networks enhanced capabilities beyond 3G support for simple mobile browsing and messaging systems. Similarly, 5G networks have been designed with three main goals to improve network performance for the next decade (see Figure 1):

1. Enhanced mobile broadband (eMBB) will support general consumers applications, such as video streaming, browsing, and cloud-based gaming.
2. Ultra-reliable low-latency communications (URLLC) will support latency-sensitive applications, such as AR/VR, autonomous vehicles and drones, smart city infrastructure, Industry 4.0, and tele-robotics.
3. Massive machine-type communications (mMTC) will support scalable peer-to-peer networks for IoT applications without high bandwidth.

5G networks accomplish the high bandwidth, high availability, and low-latency requirements of new Internet services and applications through the adoption of cloud-computing infrastructure. Cloud providers use software-defined networks (SDN) and network function virtualization (NFV) to boost the creation of services to facilitate multi-tenant and multi-service infrastructure.

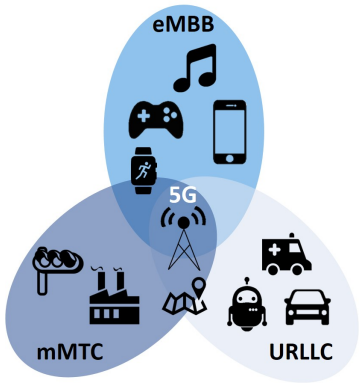


**Figure 1.** Enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (URLLC), and massive machine-type communications (mMTC) compose the network enhancements in 5G.

The move to SDN infrastructure enables the replacement of proprietary hardware and software for network functions like routers or firewalls with cheaper, standardized, and re-programmable virtual customer premises equipment (vCPE). NFVs are centrally controlled within the cloud. Virtual network functions (VNFs), as one of the key functionalities of NFV, such as load balancers, can run on any generic server to allow the network to scale up resources on demand and be migrated between different parts of the network. However, data processing in large remote cloud data centers based on SDN/NFV functionalities cannot meet the low latency required for real-time data analytics in autonomous vehicles or locally based augmented/virtual reality (AR/VR).

The increased communication taking place on smartphones, tablets, wearables, and IoT devices can congest the core network communicating the centralized cloud servers. Duplicate requests for popular videos during peak streaming times can overwhelm core networks and lead to a low quality of experience (QoE) for users and costly inefficiencies in network resource usage [2]. Mobile edge computing can help solve these challenges in 5G by creating a decentralized cloud at the network edge.

### *2.2. Mobile Edge Computing for 5G*

Various models of computing operate in the network environment, including mobile computing, cloud computing, fog computing, and edge computing. A taxonomy of the network computing paradigm is detailed in [7].

Mobile computing (MC) creates an isolated, non-centralized, network-edge, or off-network environment made up of elements (mobile devices, IoT devices, etc.) that share network, computing, and storage resources. However, cloud computing offers on-demand ubiquitous computing resources. These computing services can be public, private, or hybrid, and they use various payment-for-use mechanisms.

Edge computing (EC) is a system that offers networking, computing and storage services near the end devices, and they are generally located at the edge of the network. This system, which takes the shape of a mini data center, has high availability and can offer low-latency services, but it has computing and storage resources with lower features than cloud computing.

Mobile edge computing (MEC) combines the functions of mobile computing with edge computing. The edge computing infrastructure is complemented by the resources of mobile or IoT devices with low-consumption computing and storage hardware, and non-permanent or low-reliability communications. The mobile edge computing system has been extended and standardized by the European Telecommunication Standards Institute (ETSI), which coined the term multi-access mobile

computing (whose acronym is also MEC) [13]. ETSI proposes a platform that creates a multi-access edge system, which uses several heterogeneous access technologies, such as those proposed by 3GPP, and local or external networks, among others.

Mobile edge computing enables the implementation of new service categories such as consumer-oriented services (gaming and augmented reality), operator and third-party services (computing and storage services and V2V), and network performance and QoE improvements to enhance performance use.

Mobile edge computing creates a virtualized infrastructure that is deployed at the edge of the network and its vicinity. This architecture is closely related to NFV. Mobile edge computing can be associated with a network function virtualization (NFV), which allows applications to run efficiently and seamlessly on a multi-access network.

Mobile Edge Computing (MEC) combines Internet communication infrastructure with the cloud. Mobile edge computing brings cloud-based storage, computation, measurement, and management closer to the end user by empowering the edge network to ensure QoE, optimize resource use, and generate revenue for network operators [2]. Mobile edge computing technology has passed the proof-of-concept stage and is being deployed in networks to enable real-time applications. Instead of receiving all files from large, regional data centers, end users can receive data from local base stations to reduce latency and traffic in the backbone network. Since the first real-world deployment in 2013 [14], mobile edge computing has garnered attention as a feasible option to enable computation networks close to users by expanding the cloud-computing (CC) capabilities into a decentralized cloud using the same SDN and NFV concepts as the larger 5G. The combination of mobile edge computing and NFV allows applications to operate efficiently and seamless on a multi-access network. The mobile edge computing enabled distributed cloud will be built by deploying Network Functions in a diverse set of edge devices including LTE base stations, 5G radio network controllers (RNC), multi-Radio Access Technology (RAT) cell aggregation sites, and at aggregation points at the edge of the core network [2,15], as shown in Figure 2. A reference architecture for mobile edge computing has been proposed by the European Telecommunications Standards Institute (ETSI) [16].

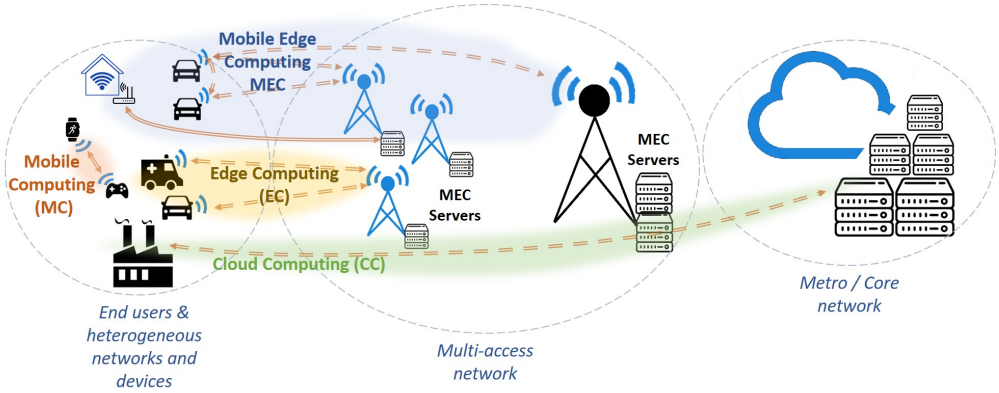


**Figure 2.** Illustration of the application area of the main network computing models focused on the edge. Mobile Edge Computing (MEC) enables cloud-based data for real-time applications and ultra-reliable services that need to be stored closer to end users with edge nodes and base stations. Mobile edge computing empowers AI-based services like navigation and connected vehicles that require large amounts of locally relevant data computation, management and data analysis.

The choice of edge device depends on the application of mobile edge computing. For example, 5G base stations can be used to assist vehicle-to-vehicle communication for autonomous driving or RAT aggregation sites can be used for delivering locally relevant, fast services to dense public locations, such as stadiums or shopping malls. Several industry cloud providers have already developed software and hardware solutions to enable mobile edge computing in edge devices (Microsoft Azure's IoT Edge [17], Google's Edge TPU [18], or Amazon's IoT Greengrass [19]), thereby establishing mobile edge

computing as a relevant technology for next-generation networks and content delivery. Layering ML on top of mobile edge computing infrastructure enables automation of efficiency-enhancing network functions at the edge. Together, ML and mobile edge computing can enable real-time applications with low-latency cloud computation.

## 3. Deep Learning Techniques

Machine learning systems use algorithms that improve their output based on experience. In the future, machine learning will replace traditional optimization methods in many fields because ML models can expand to include new restrictions and inputs without starting from scratch and they can solve mathematically complex equations. ML models are readily adapted to new situations, as we are currently witnessing with computer systems.

In the last decade, a subset of machine learning called deep learning (DL) has garnered much attention in computer vision [20,21] and has discovered new optimal strategies for games [22,23] without the costly hand-crafted feature engineering previously required. Deep learning uses neural networks to perform automated feature extraction from large data sets and then use these features in later steps to classify input, make decisions, or generate new information [24].

Research on deep learning for computer vision exploded after the release of ImageNet, a curated database of over 10 million images across 10,000 categories, which helped train ML image classification models [25]. Because 5G implementations are new and deployed in select regions, there are few representative data sets for 5G network traffic and many authors rely on simulations. 5G and slice-based networking may change the models of service demand. However, few authors have representative and detailed data from telecommunication companies [26,27] because of the risk of leaking proprietary or customer information.

Combined with the adaptation of SDN/NFV techniques within 5G networks, deep learning presents an opportunity for accurate identification and classification of mobile applications and automates the creation of adaptive network slicing [28], among other possibilities. Figure 3 shows examples of four common deep learning models, which are explained in the following subsections.

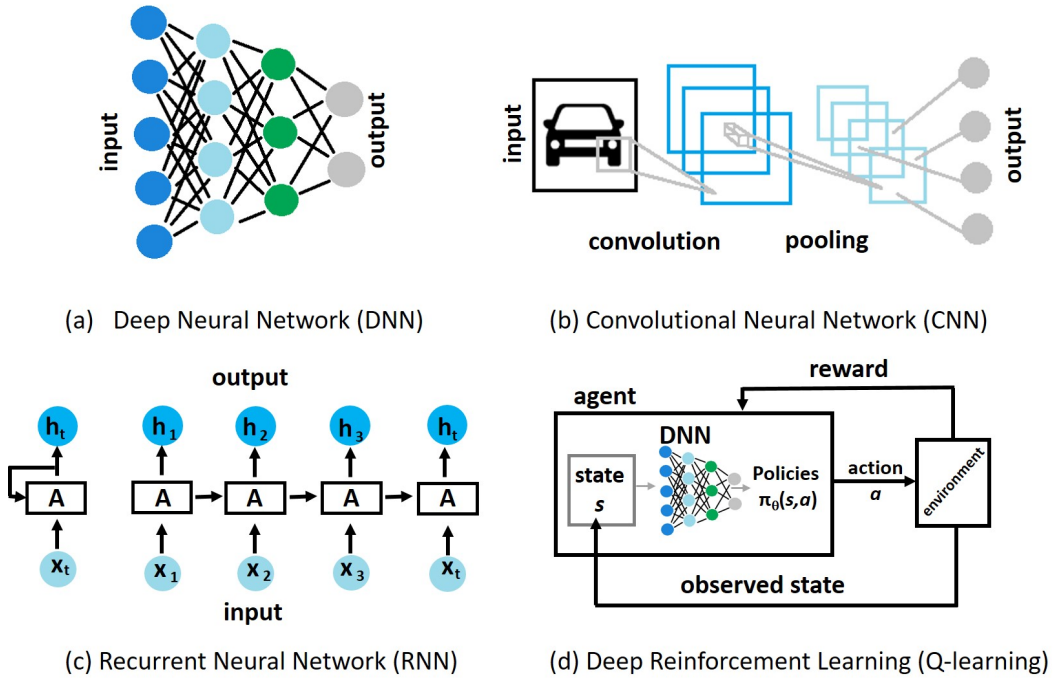


**Figure 3.** Examples of four common deep learning models.

### 3.1. Deep Neural Networks

*Deep neural networks (DNNs)* were developed to parametrically approximate functions that map a vector of input data to an output vector (e.g., an image to the percentage probability that it is in each of five classes labels) [24,29]. Expanded from original simple perceptron systems, deep neural networks are feed-forward systems that use many hidden layers and internal functions to approximate non-linear relationships between input and output. Each DNN model uses gradient descent to minimize a cost function (mean square error, maximum likelihood) in an optimization and training process called "back-propagation". Cost functions for 5G networks could also include minimizing operating costs, latency, or downtime. Within 5G networks, deep learning has already been applied to themes such as traffic classification, routing decisions, and network security [30].

*Convolutional neural networks (CNNs)* are specialized DNN models to appropriately handle large, high-resolution images as inputs [24,31]. CNNs exploit the relationship in nearby data (such as pixels in an image or location-based measurements). CNNs use mathematical "convolutions", linear operations that compute a weighted average of nearby samples, and "pooling" to summarize the data in a region typically with max-pooling or a similar operation depending on the aim [24]. CNNs have been used in 5G networks to predict mobility based on traffic flows between base stations [32] and for object classification applicable also to 5G-enabled industry robotics [33].

*Recurrent neural networks (RNNs)* scale the DNN models for function approximation to handle temporal sequences, where the output from a previous time step influences the decision the network makes for the next step [24]. RNNs require the use of a "memory" to recall information learned in previous time steps in addition to the current input to the model. Performing the gradient descent-based training on RNNs caused issues with "exploding gradients", which was corrected by new ML models called long-short term memory (LSTM) models with additional information flow structures called "gates". LSTM models have proven useful and accurate in solving traffic prediction [34] and mobility [35] problems within communication networks.

### 3.2. Reinforcement Learning

*Reinforcement Learning (RL)* has been lauded in the last decade for training ML systems to outperform humans, culminating in DeepMind's development of a winning machine even in the complex and large game space of Go [22,36]. RL is incredibly powerful because these training steps for the model do not require any prior knowledge of the rules of the games played, but rather, they optimize the model for future rewards using an "agent" who makes observations about its environment (pixels, game state, sensor inputs, etc.) and the rewards (points, coins, closeness to end goal) received in response to the actions (turning, moving, shooting, etc.) it makes. The agent determines which action to perform based on a "policy", the output of a neural network trained based on "policy gradients". In RL, discount rates can be applied to increase or decrease the importance of immediate versus projected long-term rewards when determining the optimal next action.

Q-learning is a subset of RL where models are built on Markov Decision Processes, stochastic processes where the next state is independent of previous state changes [37]. Q-values estimate the optimal state-action pairs and selects the action with the maximum reward value [36]. Q-learning is promising because even with imperfect or no information on the underlying Markov Decision Process, the transition-probabilities, the agent explores the states, rewards, and possible action pairs through greedy exploration policies that favor exploring unknown and high-reward regions. The optimal decision policy for actions to take can then be obtained with very little prior information. The agent then takes the action with the maximum Q-value calculated by the policy. Deep Q-learning networks can be used to model situations with large state spaces (e.g., Go) using DNN to avoid feature engineering to train the policy and a set of replay memories to continue using information learned in previous steps [36].

Deep Q-learning networks (DQN) are especially adaptable to open issues within the 5G sphere. Mobile networks are increasingly dynamic where the number of apps, users, and topology of the

network have become increasingly ad-hoc. The ML systems used to approximate solutions for these networks must be equally flexible. DQNs could be applied here because they discover new optimal policies after observing additional situations without requiring the model to be completely retrained. Several teams have already used deep Q-learning to address ad-hoc mobile edge computing vehicular networks for 5G [30].

### 3.3. Enabling ML in the Mobile Edge

Performing ML within edge devices can take advantage of contextual data available such as cell load, user location, connection metadata, application types, local traffic patterns, and allocated bandwidth. Latency for responses from traditional cloud-computing centers over the wide-area network hinders network and services key performance indicators (KPI). In addition, performing ML tasks at the edge can reduce the load on the core network. To take full advantage of the mobile edge computing and ML collaboration benefits, ML models must be designed to use minimal resources and still obtain useful and accurate results as they are applied to scale across expansive communication networks.

Currently, ML training and inference tasks within mobile edge computing are partially inhibited by comparatively smaller storage capabilities and limited power supplies in edge devices than those found in industrial cloud data centers. In response, ML within the mobile edge computing has been enabled by two main enhancements:

1. Efficient ML models specialized to require less energy, memory, or time to train, and
2. Distributed ML models that distribute the training and inference tasks between large data centers and smaller edge devices for parallel processing and efficiency.

#### 3.3.1. Efficient ML Models

Currently, ML models require abundant memory to store training data and computational power to train the large models. Novel ML models have been designed to operate efficiently on edge devices by employing shallow models that require low enough processing power that they can be used on IoT devices [38]. Alternatively, reducing the size of the model's inputs for classification applications can increase the speed of learning and convolutions on edge devices when less granular decisions are required [39]. The computational requirements for ML model training can be further reduced by early exiting in models designed with multiple exit points for achieved learning results [40,41] or designed in human-machine collaboration using CNNs based on existing designs by experts to explore and design new efficient ML architectures [42]. However, model redesign is only the first step to achieving efficient ML in mobile edge computing.

#### 3.3.2. Distributed ML Models

DNN is a widely adopted ML technique, but the full burden of training a DNN model is too intensive for a single resource-constrained device at the mobile edge. Distributed ML models are well-adapted to mobile edge computing because the work is distributed across many computing centers in the network (cloud, base stations, edge nodes, end devices) [43] to collectively train the DL model by giving them each a small portion of the work to perform and then combining the results [44,45]. Sub-tasks for the training can be allocated based on the edge device's resource constraints and distributed work stealing models that prioritize load balancing in inference tasks [46].

Within mobile edge computing, distributed learning aims to use multiple smaller edge devices rather than one large data center. Distributed DNNs are composed of both local and global parameter adjustments during the learning processes combined with global aggregation steps in the cloud to achieve a single well-trained model [43,47]. Optimization of the aggregation step can include methods, such as federated drop out, prioritized local updates, fast convergence, and compression [43,44] while local learning can be optimized using efficient ML models as described in the previous section.

However, distributed learning can introduce new challenges. In [48], the authors found that the latency and cost of sharing the learned gradients between devices constituted a bottleneck during the training processes. To overcome the communication bottleneck, gradients used during the back-propagation process of model training were compressed to reduce bandwidth requirements and redundancy. Additional efforts to selectively share only the important gradients in the training process have reduced communication costs with minimal impact on accuracy [49] and help reduce core network traffic and memory footprint on resource-constrained devices.

## 4. Challenges of DL for 5G Operations at the Mobile Edge

This section introduces a challenges taxonomy of applying DL at the Edge in 5G networks, which is the basis of this survey. Figure 4 shows this taxonomy, which categorizes the research articles that focus primarily on applications of deep learning techniques used in network operations discussed in this paper.

In this section, we describe how deep learning has been applied to solve operational issues at the mobile edge. Mobile edge computing (MEC) has the advantage of proximity to users, which can meet the low latency (URLLC), high bandwidth (eMBB), and high availability (mMTC) goals of 5G networks by leveraging the breakthroughs discussed in the previous section.

5G networks present interesting challenges best addressed at the mobile edge to reduce latency and incorporate locally significant information. Mobile edge computing can leverage proximity to user to address a variety of challenges in 5G networking in particular which often require automated management using DL for increasingly complex series of tasks.

Solutions combining DL for 5G promise better efficiency when conducted near the end user in mobile edge computing rather than in the core network. For instance, mixing mobile edge computing with 5G networks seamlessly connects existing cloud computing with edge computing to enable novel applications

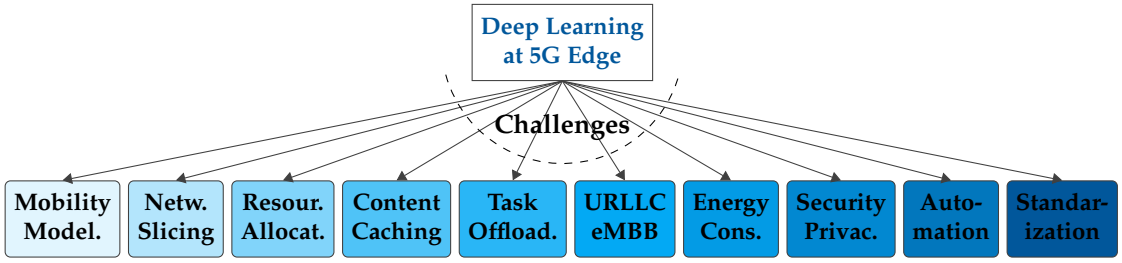


**Figure 4.** Taxonomy of challenges of deep learning used in network operations at the edge of 5G networks [26,27,32,50–105].

The potential applications of DL within the networking domain are many, but this paper focuses on a few key areas: 5G slicing using traffic prediction within the mobile edge computing, adaptive resource allocation to meet user demand in real time, predictive caching to reduce latency, task offloading from nearby end devices, meeting service quality guarantees, efficient energy usage, data security and privacy, network architectures, standards, and automation. Though these may appear to be separate tasks, they are intricately connected.

Several challenges remain before DL will be fully applicable in the 5G mobile edge, many of which are relevant in machine learning systems at large.

Table 1 provides an overview of the research discussed in this paper. Readers can view an accessible summary of the methods and key outcomes of past research.

**Table 1.** Some relevant applications of deep learning techniques used in network operations discussed in this paper. Some papers could belong in multiple categories because the themes in automated network management overlap, but for simplicity are only listed once.

| Topic | Paper | DL Model | Purpose/Methods |
|---|---|---|---|
| Slicing | [27] | DNN | Uses spatio-temporal relationships between stations to predict future traffic demand patterns |
| | [50] | RNN | Predicts base station pairing for highly mobile users to prepare for future demand |
| | [35] | RNN | Minimizes signaling overhead, latency, call dropping, and radio resource wastage using predictive handover |
| | [26] | RNN | Predicts traffic demand by exploiting space and time patterns between base stations |
| | [28] | DNN | Identifies real-time traffic and assigns to relevant network slice |
| | [54] | RL | Offers slicing strategy based on predictions for traffic and resource requirements |
| | [56] | RL | Maximizes network provider's revenue through automated admission and allocation for slices |
| | [57] | RL | Provides automated priority-based radio resource slicing and allocation |
| | [53] | RL | Constructs network services on demand based on resource use to lower costs |
| | [103] | DNN | Selects slice for resilient and efficient balancing of network load |
| Resource Allocation | [32] | DNN | Forecasts resource capacity and demands per slice using network probes |
| | [58] | DNN | Predicts traffic demand and distributes radio resources between slices |
| | [60] | RL | Minimizes cost of delay and energy consumption for multi-user wireless system |
| | [61] | RL | Minimizes end-to-end delay for caching, offloading, and radio resources for IoT |
| Caching | [62] | DNN | Reduces computational time and energy consumption for cache policy at edge |
| | [63] | RL | Jointly optimizes caching and computation for vehicular networks |
| | [64] | RNN | Forecasts user movement and service type to cache and offload tasks in advance |
| | [65] | RL | Predicts optimal cache state for radio networks |
| | [66] | RL | Optimizes cost for caching, computation, and offloading using vehicle mobility and service deadlines restraints |
| | [67] | DNN | Updates cache placement dynamically based on station and content popularity |
| | [68] | DNN | Identifies communities of mobile users and predictive device-to-device caching |
| Offloading | [70] | RL | Optimizes scheduling of offloaded tasks for vehicular networks |
| | [71] | RL | Minimizes energy, computation, and delay cost for multiple task offloading |
| QoS | [59] | RL | Dynamically meets QoS requirements for users while maximizing profits |
| | [73] | DNN | Automates management of network resources for video streaming within QoS range |
| Energy | [82] | DNN | Uses an NLP model to demonstrate DL energy requirements |
| | [83] | DNN | Creates energy model to compare consumption of cloud architectures |
| | [84] | RL | Minimizes energy usage for task offloading from mobile devices |
| Security | [86] | RL | Optimizes MEC security policies to protect against unknown attacks |
| | [87] | DNN | Protects against various types of denial-of-service attacks |
| | [88] | DNN | Develops a federated learning system with differential privacy guarantees |

### *4.1. Mobility Modeling*

User mobility prediction is necessary to achieve accurate traffic prediction. With the rise of mobile phone usage and connected vehicles, predicting mobility becomes an important step in understanding mobile network demands. Mobility models can be developed by considering different environments, such as urban [27] or highway patterns to predict the next station a user will likely connect to [50] in order to reduce costs for operational tasks such as handover [35]. Once the mobility patterns of users in a network are understood, then DL can also be used to predict traffic patterns and create more cost-efficient network operation schemes. For example, the expected demand for a base station can be predicted according to the spatial and temporal relationship it has to nearby stations [26]. Other studies apply LTSM to predict the position of the UE along time [51].

### *4.2. Slicing*

Slicing is the method by which network providers can create multiple, independent virtual networks over shared physical infrastructure for 5G networks. While traditional mobile networks treat all incoming traffic similarly, 5G network slices can provide customized network services and scale up or down as the demand for that slice changes. Slicing is made possible by SDN and NFV; the separated control plane is composed of ready-for-change software that facilitates adaptive and intelligent network management. 5G networks providers will create customized slices based on use cases (video, IoT, Industry robotics, etc.) created to meet the unique service-level agreements (SLA) [52] (see Figure 5).

5G networks can pair DL and data collected with mobile edge computing to automatically manage slice creation. Automatically spinning up resources for network slices first requires predictions of network demand and user location to assign resources correctly at edge nodes. To achieve useful traffic

prediction, it is necessary to predict user mobility and, the demand for network resources, as well as be able to classify traffic origin in real time to assign to the correct slice.

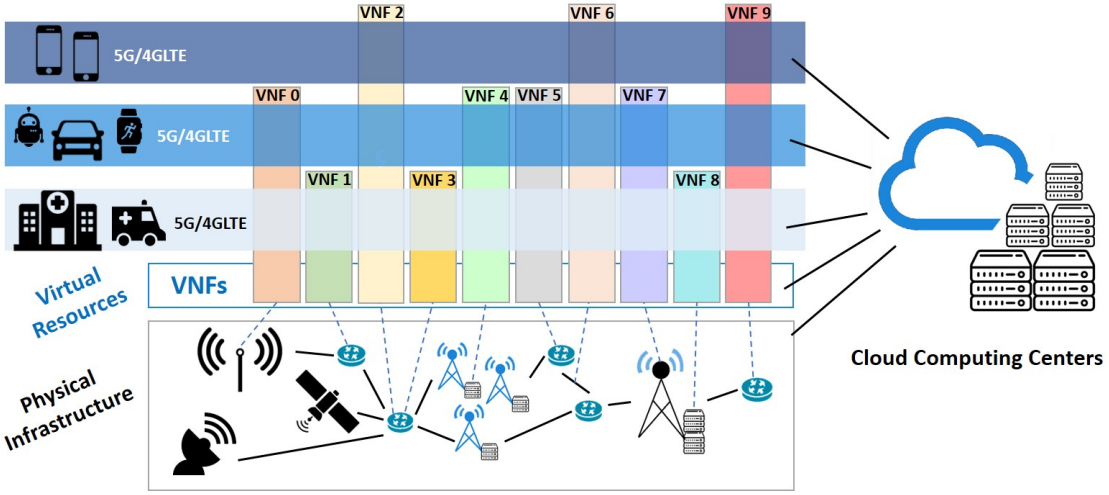


**Figure 5.** Physical infrastructure and virtual resources (network, computation, storage) in service chaining for 5G slices. Each slice has a specific purpose and type of end device, but several slices may use the same types of virtual network functions (VNFs) to deliver services to their users. The VNFs for each slice are separated for privacy and security.

The research conducted in [26] set the groundwork for this field by using the LSTM model and RNN to analyze real-world cellular data from China to understand the relationship between traffic at separate cell towers. The technique was improved by using social media and other data sources to analyze the effect of key events in a city, such as sporting games, and how this affects the network demand [27]. From here, DL can be used to classify traffic without privacy-invading techniques, such as packet inspection or strict classification based on ports or packet signatures [28]. Once the traffic type is understood, network operators can take advantage of network virtualization to create E2E slices per application and dynamically meet each SLA independently [54], while still achieving optimal resource usage.

Complete network slicing requires allocating virtual resources to a subset of traffic and isolating these resources from the rest of the network. Predicted demand influences which and how many resources are allocated per slice [55] and determines whether new users are permitted to join the network at the given station. These decisions are based on forecasts for available resources within the slice [56] in a process built on "admission control", which aims to increase revenue through efficient resource usage. DL can give some slices priority over others [57] and adapt the slicing decisions in anticipation of dynamic service demands to maximize resource use [53]. This task is complicated as unknown devices continue to join the network, but with the aid of deep learning, even these can be automatically assigned to slices to balance the network load and improve slice efficiency [103]. Maintaining efficient resource usage requires ongoing resource allocation between and within slices as discussed in the next section.

### *4.3. Resource Allocation*

Resource allocation in mobile edge computing is the task of efficiently allocating available radio and computational resources to different slices based on their requirements and priority. Historically, resource allocation was a reactive step aimed at self-healing and fault tolerance. Proactive resource allocation with DL can reduce the effects of costly mistakes for under-provisioning slices that causes SLA violations, poor user experience, and customer churn. To reduce operational costs, deep learning techniques, borrowed from image processing, are used to anticipate network capacity based on metrics gathered in the mobile edge computing, such as signal quality, occupied resource blocks, and local

computation loads [32]. Inter-slice resource allocation can be achieved by jointly optimizing according to slice priority (a slice providing remote healthcare could have priority over one for video streaming), under-stocking, fairness concerns [58], and QoE per slice [59].

Resource allocation using DL is especially useful for predicting resources needed for tasks offloaded to the mobile edge network from smaller devices [60] and proactively assigning some of the limited available resources. Because mobile edge computing can exploit local views of wireless signal and service request patterns, resource allocation models operating with this information can be used to further minimize delays [61] and respond in real time to observed changes. DL models can also be applied beyond resource optimization for a single edge node by including both spatial and temporal connections among data dependencies between traffic nearby edge nodes to predict how these dependencies and dynamic regional traffic patterns will affect resource demands [26,27].

### 4.4. Caching

Mobile edge computing can exploit proximity to the user to cache locally relevant data nearby to reduce latency and adapt to surges in popularity for certain content in a region. Employing DL to develop a proactive caching strategy has been shown to improve network efficiency and alleviate demand for radio resources by storing popular content closer to the user than regional datacenters [62,63]. Effective caching consists of two fundamental steps: predicting content requests using popularity estimates and allocating content among edge nodes.

The popularity of content influences how often and in which regions of the network users will request the content from the cache. More common content should be cached to avoid delayed retrieval from regional data centers and reduce traffic on the core network. Some studies have used user mobility and behavioral patterns to predict application choices and develop DL caching strategies to anticipate their desired content [64]. Which contents are placed in mobile edge computing caches can be optimized using DL to increase cache hit rate and decrease delays experienced by users. By using popularity models in conjunction with observed requests, an optimal cache placement strategy can be developed using DL technique such as Q-learning [65,66]; even as users move around, their desired content is more likely to be in a nearby cache.

Content popularity is necessarily dynamic (based on time of day, cultural events, or trending content) and cache content must be updated with frequency. Partial cache refreshes based on DNN provide online responses to changing popularity [67], and the content of groups of edge nodes within close proximity can be updated through joint optimization to reduce cache redundancy [68].

In [69], the authors propose an effective approach for collecting globally available resource information through a mobile network architecture based on the SDN. To minimize network latency, they designed an optimal caching strategy that consists of a small-cell cloud and a macro-cell cloud, reducing considerably the latency compared to conventional caching strategies.

Figure 6 shows the main steps for the predictive caching using ML in mobile edge computing compared with the traditional procedure.

### 4.5. Task Offloading

Due to proximity to the users and the potentially high number of stations for 5G networks, these small cell stations can be used to offload tasks that are too computationally intensive or battery-consuming for most users' mobile devices. DL systems can be trained in mobile edge computing systems to minimize the cost of offloading tasks in vehicular networks [70] and small wireless devices [60] by using both immediate and long-term rewards during the training stage. These ask offloading systems respond to changes in real-time demand for computation and supporting resources at the mobile edge computing nodes. As the scale of task offloading initiatives increases until each edge node is simultaneously receiving and running computational requests, the scheduling objective becomes almost intractable without machine learning techniques. Additional intelligent systems have been designed to simultaneously minimize costs of energy, computation, and delay by

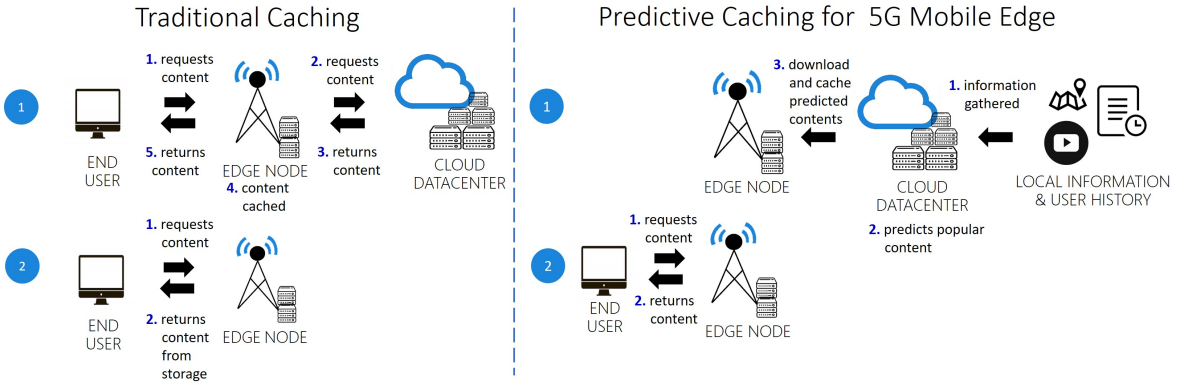


**Figure 6.** These two images show the difference between traditional caching and predictive caching using machine learning in mobile edge computing. In step 1 of predictive caching, the most popular contents that match user's predicted preferences according to their profile are downloaded from the cloud to the edge node. In the second step, when the user requests a specific content, there is a higher probability that the desired content has already been downloaded to the edge node previously, increasing QoE.

exploiting DQNs to schedule AR/VR offloading tasks [71] and then rely on additional DL techniques to reallocate resources at the mobile edge in real time. Beyond cost minimization, task offloading schemes can also be trained to minimize delay or maximize users' quality of experience /service (QoE/QoS) [70].

### *4.6. URLLC and eMBB through Quality of Service Constraints*

Quality of Service (QoS) measures the performance experienced by the end users of the network. Common QoS metrics include bandwidth, latency, and error rate, among other parameters, changes that can severely alter the network's ability to provide critical services. Every user connects to the network with a set of QoS requirements, which may be more stringent for latency-sensitive applications, such as on-demand video streaming, and voice over IP (VoIP). Meeting QoS and SLA agreements can be integrated as a goal in the DL systems for automated resource allocation in 5G mobile edge computing [59] by requiring chosen allocation schemes to maintain network operations within QoS ranges [73]. For mobile edge computing applications, such as robotic systems, resource allocation systems based on DL and QoS metrics must capture the ultra-low-latency requirements for feedback signals between end devices [74]. This ensures both safety and Quality of Experience (QoE). Furthermore, QoE, as it relates to QoS, can be optimized based on user similarity within groups or geographical regions to dynamically allocate resources according to group needs [75].

Achieving ultra-reliable and low-latency communication (URLLC) is one of the major challenges in 5G networks. This type of service offers a wide variety of challenges, such as QoS requirements [77], strict handovers for uninterrupted service [78,79], power consumption in UEs battery [80], etc. In addition, the coexistence of eMBB and URLLC with different service requirements is also a challenge [76].

The article [69] suggests a novel network architecture using a resource cognitive engine and data engine to address the problem of achieving ultra-low end-to-end delay for the ever-growing number of cognitive applications. The resource cognitive intelligence, based on network context learning, aims to attain a global view of the network's computing, caching, and communication resources.

It is quite a challenge for 5G networks to meet URLLC specifications and this will entail major changes to the system architecture of the existing telecom infrastructure. While current user requirements are initially based on high bandwidth, it is also expected that latency and reliability will play a vital role in real-time applications and mission-critical networks [81].

### 4.7. Energy Consumption

First, training DL models requires high-energy consumption. Even as training speeds have improved, the level of energy consumption for DL models remains high [82]. While some studies have attempted to estimate energy costs [83] and develop algorithms that increase the energy efficiency of the systems they manage [84,85], few efforts have combined these two critical topics. There are still many open questions about the deployability of large-scale DL models with resource-constrained mobile edge computing systems.

### 4.8. Security and Privacy

Two main concerns that have deterred the deployment of large-scale DL systems at the mobile edge computing are the rightful concerns regarding data security and privacy with collected data. Security must be guaranteed when working in 5G mobile edge computing because of the necessary sharing of physical infrastructure between slices and the potential for information leaks about data or usage patterns. As NFV is deployed, isolation between virtual machines and slices must also be guaranteed to promote privacy and reduce performance interference [89]. 5G mobile edge computing networks must also protect themselves from malicious actors, and can use DL to detect and protect against attacks [86,87], though new slicing infrastructure and virtualized networks may require deviation from industry-standard security techniques. Efforts to enhance DL performance should also be built with privacy in mind [88]. Network providers must investigate for any privacy violations in the collection or use of user data before large-scale ML systems are deployed, especially in the case of smart cities or personal electronics, such as connected cars and IoT, which can reveal intimate details about the public as a whole.

## 5. Standards towards 5G Automation

Developing end-to-end automated management of 5G architecture and services and the integration of the mobile edge computing into 5G introduce new requirements. The set of multi-access applications and services at the edge designed to meet these requirements is greatly increasing the complexity of managing the networked system.

This complexity manifests itself in different aspects of the network and services, such as the provision and operation of services, predictive analysis, real-time monitoring, analytics, or maintenance of thousands of entities, among others, and inexorably forces an end-to-end network and services automation. The application of ML in mobile edge computing and 5G, which will allow self-configuration, self-optimization and self-healing, will also require component standardization.

Several organizations, including the 3GPP, ETSI or ITU, have created working groups to address this problem of standardization and complexity, generating the first architectural standards and models for 5G.

ETSI Industry Specification Groups (ISGs), ENI (Experiential Networked Intelligence), and SAI (Security AI) are working in parallel with ITU-T's Q20/13 and FG ML5G (Focus Group on Machine Learning for Future Networks including 5G), and 3GPP TR 23.791 to incorporate ML in 5G and future networks, from the edge to the core network.

In this section, we discuss the most relevant standards, network and services architectures developed by the main standardization organizations and open forums, which enable the application of ML in mobile edge computing within the framework of 5G.

### 5.1. ETSI

ETSI is a European Standards Organization (ESO) that deals with electronic communications networks and services. It is a partner in the international Third Generation Partnership Project (3GPP) and developed thousands of standards for mobile and Internet technology since 3G networks. With 3GPPP and the guidance of specialists in ISGs for NFV, mobile edge computing, and ENI, ETSI

has created standards to develop automated and cognitive services based on real-time user needs, local environmental conditions, and business goals [92,93].

As researchers and technologists work to automate networks through DL, they must bear in mind the growing body of standards that will guide best practices for security, efficiency, and consumer experience in future networks. The main 5G management standard is ETSI's MANO (management and orchestration) architecture for NFV to simplify the roll-out of network services and reduce both deployment and operational costs. The three core functional blocks of MANO are:

1. NFV Orchestrator that controls network services onboarding, lifecycle management, resource management including capacity planning, migration and fault management,
2. VNF Manager configures and coordinates of VNF instances, and
3. Virtualized Infrastructure Manager (VIM) that controls and manages the physical and virtual infrastructure, i.e., computing, storage, and network resources.

Each functional block for MANO presents opportunities for meaningful DL implementations in highly virtualized 5G networks.

The mobile edge computing and NFV architectures proposed by ETSI are complementary. Mobile edge computing and VNF applications can be instantiated on the same virtual infrastructure, in fact NFV sees the mobile edge computing as a VNF. The mobile edge computing consists of various entities that can be grouped into the mobile edge computing system level, the mobile edge computing host level, and networks. The mobile edge computing supports different network infrastructures including those proposed by 3GPP for 5G in particular. Mobile edge computing can be one of the cornerstones of 5G at the edge. The 5G system designed by 3GPP makes it easier to deploy user plane functions on the edge of 5G network. The Network Exposure Function (NEF) in the Control Plane shows the capabilities of network functions to external entities.

One of the features supported by the mobile edge computing is 5GcoreConnect, which interchanges notifications between the 5G Network Exposure Function in the control plane or other 5G core network function. This feature allows the mobile edge computing to receive or send traffic, change the routing policy, or perform policy control. Mobile edge computing can use the shared information for application instantiation to manage the selected mobile edge computing host, select the mobile edge computing host, or perform various functionalities between both systems.

ETSI ISG mobile edge computing is focused on the management domain, some of its use cases being cognitive assistance, optimization of QoE and resource use or smart reallocation of instances, among others. The strict features of these use cases make a standardization framework essential for the application of ML in domain management.

To continue progress in connecting mobile edge computing and DL implementations for 5G, ETSI's ENI has developed models in which machine learning techniques can replace manual orchestration or traditional static policies in MANO architecture [94]. Use cases identified by ENI align with the goals in 5G research currently, and if properly realized, can even address some open issues discussed in Section 4, such as security and energy usage. ENI functional blocks include knowledge representation and management, context-aware management, situational-aware management, policy-based management, and cognition management. Using these functional blocks, 5G networks can apply the fundamentals of zero-touch network and service management (ZSM) to become self-configuring, self-optimizing, and self-healing. Future mobile edge computing architectures will take advantage of the information supplied by self-organizing networks (SONs) proposed in the 3GPP SA5 working group and the 3GPP technical report about "Study of Enablers for Network Automation for 5G (Release 16)", to study and specify how to collect data and how to feedback data analytics to the network functions [95,104].

An ETSI ISG produced a report for the organizations developing ZSM [96]. The report summarizes the main activities and architectures developed by standardization bodies, open-source organizations, and industry associations around ZSM.

Figure 7 shows the ETSI MANO framework with potential data sources and solution points for DL enhancements following ENI guidelines for automated management.

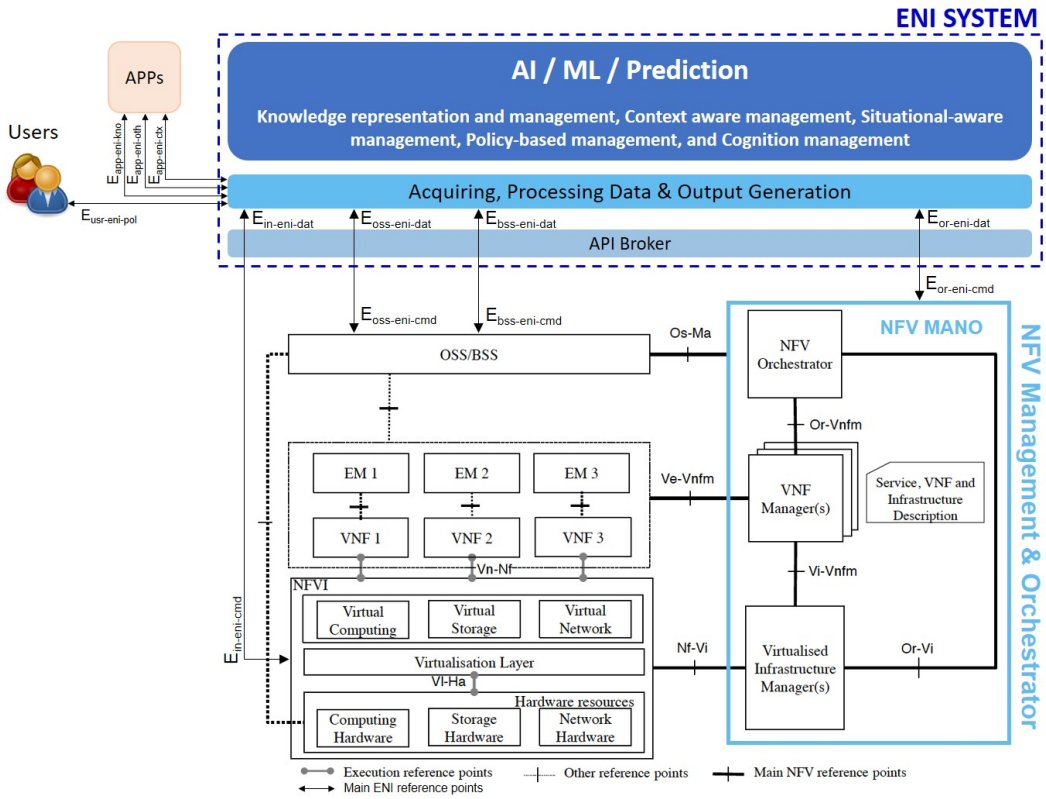


**Figure 7.** The NFV ETSI model extended with ENI ML functional blocks to automate the network operation and management [96].

## 5.2. ITU

The ITU-T Focus Group on Machine Learning for Future Networks including 5G (FG-ML5G) was established in 2017. This group drafts technical reports and specifications for ML applied to 5G and future networks, including standardization of interfaces, network architectures, protocols, algorithms and data formats [97].

The output of the FG-ML5G group includes ITU-T Y-series recommendations that provide an architectural framework for ML in future networks and use cases [98–101].

## 5.3. IETF

IETF produces open technical documentation to improve the design, use, and management of the Internet. In collaboration with the larger network engineering community, IETF produced an Internet draft to specify how ML could be introduced to distributed system pipelines with Network Telemetry and Analytics (NTA) and Network Artificial Intelligence (NAI). NTA uses telemetry and historical data to perform ML-based analytics for detection, prescription, and prediction in networks through closed loop control via SDN [102].

NTA and NAI are part of the larger effort to add intelligent management to Internet systems. Both architectures are designed to perform real-time analytics for traffic engineering and monitoring alongside existing protocols, such as BGP. Key performance indicators, such as CPU performance, memory usage, and interface bandwidth, can be used to diagnose network health even in multi-layer and virtualized environments.

IETF's Internet draft also describes intelligent service function chaining (SFC), a key task in 5G networks that allows network services to be automatically composed of several more basic network

functions. Together with application analytics and intelligent SFC, network operators could experience enhanced performance and control across locale and connection points.

### 5.4. Automation

In 5G, machine learning has been considered a useful tool to automate the network operation and management, including control and management of network slicing, service creation and orchestration, security, mobility management, etc. In [90], the authors discuss the applicability of ML to enable the 5G slicing functions to be executed autonomously. A framework is presented in [91] for the operation and control of network slices by continuously monitoring the performance, workload, and resource use, and dynamically adjusting the resources allocated to the slices.

In [105], the authors present a system of orchestration and control of E2E network slices based on service and resource modeling software that allows for custom business design and software design. They assert that applying ML in large-scale systems will yield advantages, such as better efficiency and faster integration in network management automation.

## 6. Implementations and Use Cases

This overview of the challenges and opportunities in intelligent mobile edge computing for 5G networks is timely because of the recent implementations of 5G networks in multiple countries supported by international telecommunication companies. 5G promises faster connection experiences and enhanced security through eMBB, URLLC, mMTC and design decisions for resource sharing such as network slicing and prioritized traffic. Several key world powers are competing for technology dominance in the space that will define the future of communication with new hardware, software, and data processing paradigms.

### 6.1. 5G Implementations

Following advances by the United States, Europe, and South Korea, China pledged to roll out 130,000 new 5G base stations and relay stations by the end of 2019, spread over 50 major cities [106]. These new base stations can support massive data collection to benefit network science and efficient management to provide cost-effective and efficient systems for a growing number of users. China's efforts to become a leading center for 5G were enabled by the collaboration between the country's three largest telecommunication companies (China Mobile, China Unicom, and China Telecom). Some benefits of increasing 5G small-cell base station availability are improving overall network spectral efficiency [107] and real-time insights into network capacity and performance for increasingly automated and centralized network management. However, how these new local stations and potential data processing centers will be incorporated into the 5G mobile edge computing system has not been determined. Without a doubt, ML will allow the utility of these stations and their supporting technologies to scale and empower new industries and verticals for 5G.

In 2019, Dell Technologies and the telecom Orange began working together "to jointly explore developing key technology areas for distributed cloud architectures to deliver the real-time edge use cases and new services opportunities 5G will create" [108]. Within months, Microsoft and NVIDIA announced a collaboration to advance mobile edge computing AI computing capabilities for enterprises [109] and MobiledgeX and World Wide Technology became partners to accelerate the commercialization of scalable mobile edge computing deployments [110]. Many telecommunication companies are teaming up with leaders in ML and AI to focus on intelligent mobile edge computing because of the many new applications that will benefit.

### 6.2. New Mobile Edge Applications

Corporate investment in 5G has risen rapidly because the new use cases (and revenue streams) opened by the next-generation technology will reduce expenditures and maintain flat rates for users. In their report naming mobile edge computing a key enabling technology for 5G networks,

the European 5GPPP identified multiple verticals that would be empowered by mobile edge computing: Internet of Things (IoT), caching, video streaming, augmented reality, healthcare, and connected vehicles. With the building of large-scale 5G networks, researchers are focusing on the myriad application spaces that could benefit from the low latency, proximity, high bandwidth, location-awareness, and real-time insight provided by mobile edge computing [2]. The growth of mobile edge computing will interrupt the current cloud-computing paradigm in preference to localized computing near the user.

Below we discuss a few of the new application spaces for emerging ML-enabled mobile edge computing systems.

#### 6.2.1. Internet-of-(Every)Thing

By 2022, the number of IoT devices is expected to increase to 18 billion [111], each requiring network connectivity. Mobile edge computing can benefit small-scale personal IoT devices to large-scale design situations, such as smart cities and new industrial applications. Small devices such as in-home IoT tools (Amazon Alexa, Nest Cam, Google Home), can use mobile edge computing to offload computational tasks that are too complex for their small memory capacity [112,113]. Users streaming videos from their mobile devices can enjoy cached versions of their desired content from mobile edge computing base stations [114], or videos automatically delivered in a quality/bandwidth supportable by their network based on local network conditions [115]. The growing augmented reality systems, such as Pokémon Go can store locally relevant information to overlay the user's environment in local mobile edge computing base stations such users experience reduced latency in comparison to information retrieval from regional cloud data centers [116].

#### 6.2.2. Connected Vehicles

5G mobile edge computing can enable new applications on a larger scale. Consider the coordination of increasingly ad-hoc networks from unmanned aerial vehicles (UAVs) and connected cars that must navigate new surroundings [117], offload computational tasks and download new information with the assistance of mobile edge computing stations nearby [118], all with low latency as vehicles move throughout their region. UAVs have strict memory and power-consumption restraints under which ML decision tasks must function, and, therefore, could benefit from distributed learning methods and computational offloading.

#### 6.2.3. Smart Cities

In urban settings, 5G mobile edge computing can be used to enhance smart city initiatives globally by providing points for computation and data storage relevant to local events and populations. By harnessing the power of cloud computing and Internet connectivity for large-scale IoT in cities, smart cities can provide urban services, such as electricity grids, transportation systems, and emergency response through deep learning in mobile edge computing. Cities can use mobile edge computing to manage energy consumption in growing urban areas, based on energy profiles for common activities and real-time demand [119]. In addition, placing DL at the mobile edge can also promote public safety and policing efforts in large urban areas through light-weight computer vision systems [120,121].

#### 6.2.4. Robotics and Industry

5G can be used to automate the work conducted in factories using robotic devices and real-time big data analysis at the mobile edge within the factory. Robust 5G networks provide technology advancements critical to factory automation such as high availability, low-latency, and resilience against attacks as provided by a dedicated slice of the network [122]. 5G and mobile edge computing enable the automation of critical applications, such as quality inspection of products [123]. Automated factory systems can then leverage DL to manage the offloading of computational tasks to the local edge network in energy- and resource- efficient ways [122,124]. In the medical sphere, 5G paired with

robotics can enable remote medical examination or surgeries with ultra-low-latency remote control with tactile feedback to remote surgeons.

## 7. Conclusions

Mobile edge computing plays a crucial role in helping 5G networks achieve the goals for eMBB, URRLC, mMTC as demand for network resources steadily increases with the rise of IoT and video streaming devices. Deep learning, a powerful subset of machine learning, can be adapted for use in 5G network operations to predict user behavior and automate the management of dynamic network resources. Using deep learning can both improve user experience and lower operational costs for telecommunication companies in the future.

This document provides insight into four main ideas in ML for 5G mobile edge computing. First, in Section 2, we show that the mobile edge computing is a prime candidate to implement the new verticals, features, and service categories required to deploy 5G. Second, we provide an overview of the key deep learning concepts and how they can be adapted to work best in mobile edge computing environments with computing and memory limitations. Section 3 explores the suitable deep learning techniques to automate operations required to manage services and applications over the increasingly complex 5G networks. Third, Section 5 develops a taxonomy of the challenges and trade-offs posed by the introduction of a subset of deep learning techniques in the mobile edge computing. 5G networks enable a diverse set of new applications and on-demand services with strict requirements, which substantially increase complexity for which deep learning methods are uniquely suited. Finally, Section 6 presents proofs of concept and the most relevant developments that combine the mobile edge computing with ML in the 5G environment. We also discuss mobile edge computing applications and how newly designed implementation standards for deep learning in 5G networks can enhance various verticals for 5G, including IoT, AR/VR, vehicle networks, and smart cities.

In conclusion, we hope that this survey will provide information on the use and adaptation of deep learning to improve mobile edge computing. These techniques may stimulate further research and deployment of scenarios that allow for increased automation of the network and services in the future.

**Author Contributions:** Conceptualization, S.S. and C.C.-P.; Methodology, S.S. and C.C.-P.; Investigation, M.M.; Resources, M.M., C.C.-P. and S.S.; Writing—Original Draft Preparation, M.M.; Writing—Review Editing, M.M., C.C.-P. and S.S.; Supervision, S.S. and C.C.-P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by a grant from the Fulbright Commission in Spain and by the Ministerio de Ciencia e Innovación of the Spanish Government under the project PID2019-108713RB-C51.

**Acknowledgments:** This research is supported by the Fulbright Commission in Spain and by the Ministerio de Ciencia e Innovación of the Spanish Government.

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

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| 5G | "Fifth-Generation" Mobile Networks |
| MEC | Mobile Edge Computing / Multi-Access Edge Computing |
| SDN | Software-Defined Networks |
| NFV | Network Function Virtualization |
| European 5GPPP | European 5G Public Private Partnership |
| ML | Machine Learning |
| DL | Deep Learning |
| OPEX | Operating Expenditures |
| CAPEX | Capital Expenditures |

| | |
|---|---|
| ISP | Internet Service Providers |
| eMBB | Enhanced Mobile Broadband |
| URRLC | Ultra-reliable Low-latency Communications |
| mMTC | Massive Machine-type Communications |
| vCPE | Virtual Customer Premise Equipment |
| VNF | Virtualized Network Function |
| AR/VR | Augmented Reality/Virtual Reality |
| QoE | Quality of Experience |
| MC | Mobile Computing |
| CC | Cloud Computing |
| EC | Edge Computing |
| LTE | Long-Term Evolution |
| RNC | Radio Network Controllers |
| RAT | Radio Access Technology |
| DNN | Deep Neural Network |
| CNN | Convolutional Neural Network |
| RNN | Recurrent Neural Network |
| LSTM | Long-short Term Memory model |
| RL | Reinforcement Learning |
| DQN | Deep Q-learning Network |
| KPI | Key Performance Indicators |
| ETSI | European Telecommunications Standards Institute |
| ESO | European Standards Organization |
| 3GPP | Third Generation Partnership Project |
| ITU | International Telecommunication Union |
| IETF | Internet Engineering Task Force |
| ISG | Industry Specific Groups |
| ENI | Experimental Network Intelligence |
| MANO | Management And Orchestration |
| VIM | Virtualized Infrastructure Manager |
| ZSM | Zero-touch Service Management |
| SON | Self-Organizing Network |
| NTA | Network Telemetry and Analytics |
| NAI | Network Artificial Intelligence |
| SFC | Service Function Chaining |
| AI | Artificial Intelligence |
| IoT | Internet of Things |
| UAV | Unmanned Aerial Vehicle |
| SLA | Service-Level Agreement |
| E2E | End-to-end |
| QoS | Quality of Service |
| VoIP | Voice over IP |

## References

1. Wood, L. *5G Optimization: Mobile Edge Computing, APIs, and Network Slicing 2019–2024*; Technical Report for Research and Markets: Dublin, Ireland, 22 October 2019.
2. Hu, Y.C.; Patel, M.; Sabella, D.; Sprecher, N.; Young, V. ETSI White Paper No. 11. Mobile Edge Computing: A Key Technology towards 5G. Technical Report, ETSI, 2015. Available online: https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf (accessed on 8 July 2020).
3. Porambagea, P.; Okwuibe, J.; Liyanage, M.; Ylianttila, M.; Taleb, T. Survey on Multi-Access Edge Computing for Internet of Things Realization. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2961–2991. [CrossRef]

4. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556. Available online: https://arxiv.org/abs/1409.1556 (accessed on 8 July 2020).
5. Collobert, R.; Weston, J. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In Proceedings of the 25th International Conference on Machine Learning (ICML 2008), Helsinki, Finland, 5–9 July 2008; pp. 160–167.
6. Park, J.; Samarakoon, S.; Bennis, M.; Debbah, M. Wireless Network Intelligence at the Edge. *Proc. IEEE* **2019**, *107*, 2204–2239. [CrossRef]
7. Yousefpour, A.; Fung, C.; Nguyen, T.; Kadiyala, K.; Jalali, F.; Niakanlahiji, A.; Kong, J.; Jue, J.P. All one needs to know about fog computing and related edge computing paradigms: A complete survey. *J. Syst. Archit.* **2019**, *98*, 289–330. [CrossRef]
8. Pham, Q.; Fang, F.; Ha, V.N.; Piran, M.J.; Le, M.; Le, L.B.; Hwang, W.; Ding, Z. A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art. *IEEE Access* **2020**. [CrossRef]
9. Miller, D.J.; Xiang, Z.; Kesidis, G. Adversarial Learning Targeting Deep Neural Network Classification: A Comprehensive Review of Defenses Against Attacks. *Proc. IEEE* **2020**, *108*, 402–433. [CrossRef]
10. Tang, F.; Kawamoto, Y.; Kato, N.; Liu, J. Future Intelligent and Secure Vehicular Network Toward 6G: Machine-Learning Approaches. *Proc. IEEE* **2020**, *108*, 292–307. [CrossRef]
11. Chen, X.; Proietti, R.; Yoo, S.J.B. Building Autonomic Elastic Optical Networks with Deep Reinforcement Learning. *IEEE Commun. Mag.* **2019**, *57*, 20–26. [CrossRef]
12. Yu, Y.; Wang, T.; Liew, S.C. Deep-Reinforcement Learning Multiple Access for Heterogeneous Wireless Networks. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 1277–1290. [CrossRef]
13. ETSI GS MEC 002 (V2.1.1): Multi-Access Edge Computing (MEC); Phase 2: Use Cases and Requirements, 2018. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.01.01_60/gs_MEC002v020101p.pdf (accessed on 8 July 2020).
14. Increasing Mobile Operators' Value Proposition with Edge Computing. Technical Report, Intel and Nokia Siemens Networks, 2013. Available online: https://www.intel.co.id/content/dam/www/public/us/en/documents/technology-briefs/edge-computing-tech-brief.pdf (accessed on 8 July 2020).
15. Varghese, B.; Wang, N.; Barbhuiya, S.; Kilpatrick, P.; Nikolopoulos, D. Challenges and Opportunities in Edge Computing. In Proceedings of the 1st IEEE International Conference on Smart Cloud (SmartCloud 2016), New York, NY, USA, 18–20 November 2016; pp. 20–26.
16. ETSI GS MEC 003 (V2.1.1): Multi-Access Edge Computing (MEC); Framework and Reference Architecture, 2019. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.01.01_60/gs_MEC003v020101p.pdf (accessed on 8 July 2020).
17. Enable Edge Computing with Azure IoT Edge. Available online: https://azure.microsoft.com/en-us/resources/videos/microsoft-ignite-2017-enable-edge-computing-with-azure-iot-edge/ (accessed on 8 July 2020).
18. Edge TPU. Available online: https://cloud.google.com/edge-tpu/ (accessed on 30 May 2020).
19. AWS IoT Greengrass. Available online: https://aws.amazon.com/greengrass/ (accessed on 30 May 2020).
20. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS 2012), Lake Tahoe, Nevada, 5–9 July 2008; Volume 1, pp. 160–167.
21. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. In Proceedings of the 31st International Conference on International Conference on Machine Learning (ICML 2014), Beijing, China, 21–26 June 2014; Volume 32, pp. I-647–I-655.
22. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Huberta, T.; Bakera, L.; Lai, M.; Bolton, A.; et al. Mastering the game of Go without human knowledge. *Nature* **2017**, *550*, 354–359. [CrossRef]
23. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M.A. Playing Atari with Deep Reinforcement Learning. *arXiv* **2013**, arXiv:1312.5602. Available online: https://arxiv.org/abs/1312.5602 (accessed on 8 July 2020).
24. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 8 July 2020).

25. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In Proceedings of the IEEE Computer Society Conference on Computer Vision (CVPR 2009), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
26. Wang, J.; Tang, J.; Xu, Z.; Wang, Y.; Xue, G.; Zhang, X.; Yang, D. Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach. In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM 2017), Atlanta, GA, USA, 1–4 May 2017; pp. 1–9.
27. Wang, X.; Zhou, Z.; Xiao, F.; Xing, K.; Yang, Z.; Liu, Y.; Peng, C. Spatio-Temporal Analysis and Prediction of Cellular Traffic in Metropolis. *IEEE Trans. Mobile Comput.* **2019**, *18*, 2190–2202. [CrossRef]
28. Nakao, A.; Du, P. Toward In-Network Deep Machine Learning for Identifying Mobile Applications and Enabling Application Specific Network Slicing. *IEICE Trans. Commun.* **2018**, *E101-B*, 1536–1543. [CrossRef]
29. Utgoff, P.E.; Stracuzzi, D.J. Many-Layered Learning. *Neural Comput.* **2002**, *14*, 2497–2529. [CrossRef] [PubMed]
30. Boutaba, R.; Salahuddin, M.A.; Limam, N.; Ayoubi, S.; Shahriar, N.; Estrada-Solano, F.; Caicedo, O.M. A Comprehensive Survey on Machine Learning for Networking: Evolution, Applications and Research Opportunities. *J. Internet Serv. Appl.* **2018**, *9*, 1–99. [CrossRef]
31. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of the 13th European Conference on Computer Vision (ECCV 2014), Zurich, Switzerland, 6–12 September 2014; pp. 818–833.
32. Bega, D.; Gramaglia, M.; Fiore, M.; Banchs, A.; Costa-Pérez, X. DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning. In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM 2019), Paris, France, 25–27 September 2019; pp. 280–288.
33. Shakev, N.G.; Ahmed, S.A.; Popov, V.L.; Topalov, A.V. Recognition and Following of Dynamic Targets by an Omnidirectional Mobile Robot using a Deep Convolutional Neural Network. In Proceedings of the 9th International Conference on Intelligent Systems (IS 2018), Madeira, Portugal, 25–27 September 2018; pp. 589–594.
34. Azari, A.; Papapetrou, P.; Denic, S.Z.; Peters, G. User Traffic Prediction for Proactive Resource Management: Learning-Powered Approaches. *arXiv* **2019**, arXiv:1906.00951. Available online: https://arxiv.org/abs/1906.00951 (accessed on 8 July 2020).
35. Ozturk, M.; Gogate, M.; Onireti, O.; Adeel, A.; Hussain, A.; Imran, M. A novel deep learning driven low-cost mobility prediction approach for 5G cellular networks: The case of the Control/Data Separation Architecture (CDSA). *Neurocomputing* **2019**, *358*, 479–489. [CrossRef]
36. Geron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2019.
37. Bellman, R. A Markovian Decision Process. *J. Math. Mech.* **1957**, *6*, 679–684. [CrossRef]
38. Agarwal, P.; Alam, M. A Lightweight Deep Learning Model for Human Activity Recognition on Edge Devices. *arXiv* **2019**, arXiv:1909.12917. Available online: https://arxiv.org/abs/1909.12917 (accessed on 8 July 2020).
39. Yang, S.; Gong, Z.; Ye, K.; Wei, Y.; Huang, Z.; Huang, Z. EdgeCNN: Convolutional Neural Network Classification Model with small inputs for Edge Computing. *arXiv* **2019**, arXiv:1909.13522. Available online: https://arxiv.org/abs/1909.13522 (accessed on 8 July 2020).
40. Li, E.; Zeng, L.; Zhou, Z.; Chen, X. Edge AI: On-Demand Accelerating Deep Neural Network Inference via Edge Computing. *arXiv* **2019**, arXiv:1910.05316. Available online: https://arxiv.org/abs/1910.05316 (accessed on 8 July 2020).
41. Teerapittayanon, S.; McDanel, B.; Kung, H.T. BranchyNet: Fast inference via early exiting from deep neural networks. In Proceedings of the 23rd International Conference on Pattern Recognition (ICPR 2016), Cancún, Mexico, 4–8 December 2016; pp. 2464–2469.
42. Wong, A.; Lin, Z.Q.; Chwyl, B. AttoNets: Compact and Efficient Deep Neural Networks for the Edge via Human-Machine Collaborative Design. In Proceedings of the 1st Computer Vision and Pattern Recognition Workshops (CVPR 2019), Long Beach, CA, USA, 16–29 June 2019; pp. 1–10.
43. Lin, Y.; Han, S.W.; Mao, H.; Wang, Y.; Dally, W.J. Deep Gradient Compression: Reducing the Communication Bandwidth for Distributed Training. *arXiv* **2017**, arXiv:1712.01887. Available online: https://arxiv.org/abs/1712.01887 (accessed on 8 July 2020).

44. Bonawitz, K.; Eichner, H.; Grieskamp, W.; Huba, D.; Ingerman, A.; Ivanov, V.; Kiddon, C.; Konecný, J.; Mazzocchi, S.; McMahan, H.B.; et al. Towards Federated Learning at Scale: System Design. *arXiv* **2019**, arXiv:1902.01046. Available online: https://arxiv.org/abs/1902.01046 (accessed on 8 July 2020).
45. Lim, W.Y.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated Learning in Mobile Edge Networks: A Comprehensive Survey. *arXiv* **2019**, arXiv:1909.11875. Available online: https://arxiv.org/abs/1909.11875 (accessed on 8 July 2020).
46. Teerapittayanon, S.; McDanel, B.; Kung, H.T. Distributed Deep Neural Networks Over the Cloud, the Edge and End Devices. In Proceedings of the 37th IEEE International Conference on Distributed Computing Systems (ICDCS 2017), Atlanta, GA, USA, 5–8 June 2017; pp. 328–339.
47. Wang, S.; Tuor, T.; Salonidis, T.; Leung, K.K.; Makaya, C.; He, T.; Chan, K.S. Adaptive Federated Learning in Resource Constrained Edge Computing Systems. *IEEE J. Sel. Areas Commun.* **2018**, *37*, 1205–1221. [CrossRef]
48. Liu, L.; Zhang, J.; Song, S.H.; Letaief, K.B. Client-Edge-Cloud Hierarchical Federated Learning. *arXiv* **2019**, arXiv:1905.06641. Available online: https://arxiv.org/abs/1905.06641 (accessed on 8 July 2020).
49. Tao, Z.; Li, Q. eSGD: Communication Efficient Distributed Deep Learning on the Edge. In Proceedings of the 1st Hot Topics in Edge Computing (HotEdge 2018), Boston, MA, USA, 10 July 2018; pp. 1–6.
50. Wickramasuriya, D.S.; Perumalla, C.A.; Davaslioglu, K.; Gitlin, R.D. Base station prediction and proactive mobility management in virtual cells using recurrent neural networks. In Proceedings of the 18th IEEE Wireless and Microwave Technology Conference (WAMICON 2017), Cocoa Beach, FL, USA, 24–25 April 2017; pp. 1–6.
51. Pham, Q.; Fang, F.; Ha, V.N.; Piran, M.J.; Le, M.; Le, L.B.; Hwang, W.; Ding, Z. Multiple contents offloading mechanism in AI-enabled opportunistic networks. *IEEE Access* **2020**, *155*, 93–103.
52. An Introduction to Network Slicing. Technical Report, GSM Association, 2017. Available online: https://www.gsma.com/futurenetworks/wp-content/uploads/2017/11/GSMA-An-Introduction-to-Network-Slicing.pdf (accessed on 8 July 2020).
53. Sun, G.; Gebrekidan, Z.T.; Boateng, G.O.; Ayepah-Mensah, D.; Jiang, W. Dynamic Reservation and Deep Reinforcement Learning Based Autonomous Resource Slicing for Virtualized Radio Access Networks. *IEEE Access* **2019**, *7*, 45758–45772. [CrossRef]
54. Koo, J.; Mendiratta, V.B.; Rahman, M.R.; Elwalid, A. Deep Reinforcement Learning for Network Slicing with Heterogeneous Resource Requirements and Time Varying Traffic Dynamics. *arXiv* **2019**, arXiv:1908.03242. Available online: https://arxiv.org/abs/1908.03242 (accessed on 30 May 2020).
55. Sciancalepore, V.; Samdanis, K.; Costa, X.P.; Bega, D.; Gramaglia, M.; Banchs, A. Mobile traffic forecasting for maximizing 5G network slicing resource utilization. In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM 2017), Atlanta, GA, USA, 1–4 May 2017; pp. 1–9. [CrossRef]
56. Bega, D.; Gramaglia, M.; Banchs, A.; Sciancalepore, V.; Samdanis, K.; Costa, X.P. Optimising 5G infrastructure markets: The business of network slicing. In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM 2017), Atlanta, GA, USA, 1–4 May 2017; pp. 1–9. [CrossRef]
57. Li, R.; Zhao, Z.; Sun, Q.; Chih-Lin, I.; Yang, C.; Chen, X.; Zhao, M.; Zhang, H. Deep Reinforcement Learning for Resource Management in Network Slicing. *IEEE Access* **2018**, *6*, 74429–74441. [CrossRef]
58. Khatibi, S.; Jano, A. Elastic Slice-Aware Radio Resource Management with AI-Traffic Prediction. In Proceedings of the 28th European Conference on Networks and Communications (EuCNC 2019), Valencia, Spain, 18–21 June 2019; pp. 575–579.
59. Kim, Y.; Kim, S.; Lim, H. Reinforcement Learning Based Resource Management for Network Slicing. *Appl. Sci.* **2019**, *9*, *2361*, 1–17. [CrossRef]
60. Li, J.; Gao, H.; Lv, T.; Lu, Y. Deep reinforcement learning based computation offloading and resource allocation for MEC. In Proceedings of the IEEE Wireless Communications and Networking Conference (IEEE WCNC 2018), Barcelona, Spain, 15–18 April 2018; pp. 1–6.
61. Wei, Y.; Yu, F.R.; Song, M.; Han, Z. Joint Optimization of Caching, Computing, and Radio Resources for Fog-Enabled IoT Using Natural Actor–Critic Deep Reinforcement Learning. *IEEE Internet Things J.* **2019**, *6*, 2061–2073. [CrossRef]
62. Chang, Z.; Lei, L.; Zhou, Z.; Mao, S.; Ristaniemi, T. Learn to Cache: Machine Learning for Network Edge Caching in the Big Data Era. *IEEE Wirel. Commun.* **2018**, *25*, 28–35. [CrossRef]
63. He, Y.; Zhao, N.; Yin, H. Integrated Networking, Caching, and Computing for Connected Vehicles: A Deep Reinforcement Learning Approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 44–55. [CrossRef]

64. Wang, R.; Li, M.; Peng, L.; Hu, Y.; Hassan, M.M.; Alelaiwi, A. Cognitive multi-agent empowering mobile edge computing for resource caching and collaboration. *Future Gener. Comput. Syst.* **2020**, *102*, 66–74. [CrossRef]
65. Chien, W.C.; Weng, H.Y.; Lai, C.F. Q-learning based collaborative cache allocation in mobile edge computing. *Future Gener. Comput. Syst.* **2020**, *102*, 603–610. [CrossRef]
66. Tan, L.T.; Hu, R.Q.; Hanzo, L.H. Twin-Timescale Artificial Intelligence Aided Mobility-Aware Edge Caching and Computing in Vehicular Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 3086–3099. [CrossRef]
67. Yang, J.; Zhang, J.; Ma, C.; Wang, H.; Zhang, J.; Zheng, G.H. Deep learning-based edge caching for multi-cluster heterogeneous networks. *Neural Comput. Appl.* **2019**. [CrossRef]
68. Li, Z.; Chen, J.; Zhang, Z. Socially Aware Caching in D2D Enabled Fog Radio Access Networks. *IEEE Access* **2019**, *7*, 84293–84303. [CrossRef]
69. Chen, M.; Qian, Y.; Hao, Y.; Li, Y.; Song, J. Data-Driven Computing and Caching in 5G Networks: Architecture and Delay Analysis. *IEEE Wirel. Commun.* **2018**, *25*, 70–75. [CrossRef]
70. Ning, Z.; Dong, P.; Wang, X.; Rodrigues, J.J.P.C.; Xia, F. Deep Reinforcement Learning for Vehicular Edge Computing: An Intelligent Offloading System. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 1–25. [CrossRef]
71. Huang, L.; Feng, X.; Zhang, C.; Qian, L.; Wu, Y. Deep reinforcement learning-based joint task offloading and bandwidth allocation for multi-user mobile edge computing. *Digit. Commun. Netw.* **2019**, *5*, 10–17. [CrossRef]
72. Wang, Y.; Tao, X.; Zhang, X.; Zhang, P.; Hou, Y.T. Cooperative task offloading in three-tier mobile computing networks: An ADMM framework. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2763–2776. [CrossRef]
73. Martin, A.; Egaña, J.; Flórez, J.; Montalbán, J.; Olaizola, I.G.; Quartulli, M.; Viola, R.; Zorrilla, M. Network Resource Allocation System for QoE-Aware Delivery of Media Services in 5G Networks. *IEEE Trans. Broadcast.* **2018**, *64*, 561–574. [CrossRef]
74. Aazam, M.; Harras, K.A.; Zeadally, S. Fog Computing for 5G Tactile Industrial Internet of Things: QoE-Aware Resource Allocation Model. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3085–3092. [CrossRef]
75. Wu, D.; Liu, Q.; Wang, H.; Yang, Q.; Wang, R. Cache Less for More: Exploiting Cooperative Video Caching and Delivery in D2D Communications. *IEEE Trans. Multimed.* **2019**, *21*, 1788–1798. [CrossRef]
76. Alsenwi, M.; Tran, N.H.; Bennis, M.; Shashi RajPandey, A.K.B.; Hong, C.S. Intelligent Resource Slicing for eMBB and URLLC Coexistence in 5G and Beyond: A Deep Reinforcement Learning Based Approach. *arXiv* **2020**, arXiv:2003.07651. Available online: https://arxiv.org/abs/2003.07651.pdf (accessed on 8 July 2020).
77. Maaz, D.; Galindo-Serrano, A.; Elayoubi, S.E. URLLC User Plane Latency Performance in New Radio. In Proceedings of the 25th International Conference on Telecommunications (ICT 2018), Saint Malo, France, 26–28 June 2018; pp. 225–229.
78. Bennis, M.; Debbah, M.; Poor, H.V. Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale. *Proc. IEEE* **2018**, *106*, 1834–1853. [CrossRef]
79. Park, H.; Lee, Y.; Kim, T.; Kim, B.; Lee, J. Handover Mechanism in NR for Ultra-Reliable Low-Latency Communications. *IEEE Netw.* **2018**, *32*, 41–47. [CrossRef]
80. Mukherjee, A. Energy Efficiency and Delay in 5G Ultra-Reliable Low-Latency Communications System Architectures. *IEEE Netw.* **2018**, *32*, 55–61. [CrossRef]
81. Siddiqi, M.A.; Yu, H.; Joung, J. 5G Ultra-Reliable Low-Latency Communication Implementation Challenges and Operational Issues with IoT Devices. *Electronics* **2019**, *8*, 1–18. [CrossRef]
82. Strubell, E.; Ganesh, A.; McCallum, A. Energy and Policy Considerations for Deep Learning in NLP. *arXiv* **2019**, arXiv:1906.02243. Available online: https://arxiv.org/abs/1906.02243 (accessed on 8 July 2020).
83. Ahvar, E.; Orgerie, A.; Lébre, A. Estimating Energy Consumption of Cloud, Fog and Edge Computing Infrastructures. *IEEE Trans. Sustain. Comput.* **2019**, pp. 1–12. [CrossRef]
84. Chen, W.; Wang, D.; Li, K. Multi-User Multi-Task Computation Offloading in Green Mobile Edge Cloud Computing. *IEEE Trans. Serv. Comput.* **2019**, *12*, 726–738. [CrossRef]
85. Liu, H.; Cao, L.; Pei, T.; Deng, Q.; Zhu, J. A Fast Algorithm for Energy-Saving Offloading With Reliability and Latency Requirements in Multi-Access Edge Computing. *IEEE Access* **2020**, *8*, 151–161. [CrossRef]
86. Xiao, L.; Wan, X.; Dai, C.; Du, X.; Chen, X.; Guizani, M. Security in Mobile Edge Caching with Reinforcement Learning. *IEEE Wirel. Commun.* **2018**, *25*, 116–122. [CrossRef]

87. Thantharate, A.; Paropkari, R.; Walunj, V.; Beard, C.; Kankariya, P. Secure5G: A Deep Learning Framework Towards a Secure Network Slicing in 5G and Beyond. In Proceedings of the 10th Annual Computing and Communication Workshop and Conference (CCWC 2020), Las Vegas, NV, USA, 6–8 January 2020; pp. 852–857.
88. Truex, S.; Liu, L.; Chow, K.H.; Gursoy, M.E.; Wei, W. LDP-Fed: Federated Learning with Local Differential Privacy. In Proceedings of the 3rd ACM International Workshop on Edge Systems, Analytics and Networking (EdgeSys 2020), Heraklion, Greece, 27 April 2020; pp. 61–66.
89. Akundi, S.; Prabhu, S.; BK, N.U.; Mondal, S.C. Suppressing Noisy Neighbours in 5G Networks: An End-to-End NFV-Based Framework to Detect and Suppress Noisy Neighbours. In Proceedings of the 21st International Conference on Distributed Computing and Networking (ICDCN 2020), Kolkata, India, 4–7 January 2020; pp. 1–6.
90. Kafle, V.P.; Fukushima, Y.; Martinez-Julia, P.; Miyazawa, T. Consideration on Automation of 5G Network Slicing with Machine Learning. In Proceedings of the 10th ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K 2018), Santa Fe, Argentina, 26–28 November 2018; pp. 1–8.
91. Kafle, V.P.; Martinez-Julia, P.; Miyazawa, T. Automation of 5G Network Slice Control Functions with Machine Learning. *IEEE Commun. Stand. Mag.* **2019**, *3*, 54–62. [CrossRef]
92. ETSI GS ENI 005 (V1.1.1): Experiential Networked Intelligence (ENI). System Architecture, 2020. Available online: https://www.etsi.org/deliver/etsi_gs/ENI/001_099/005/01.01.01_60/gs_ENI005v010101p.pdf (accessed on 8 July 2020).
93. ETSI GS ENI 002 (V2.1.1): Experiential Networked Intelligence (ENI). ENI Requirements, 2018. Available online: https://www.etsi.org/deliver/etsi_gs/ENI/001_099/002/01.01.01_60/gs_ENI002v010101p.pdf (accessed on 8 July 2020).
94. ETSI GR ENI 001 (V1.1.1): Experiential Networked Intelligence (ENI). ENI Use Cases, 2018. Available online: https://www.etsi.org/deliver/etsi_gr/ENI/001_099/001/01.01.01_60/gr_ENI001v010101p.pdf (accessed on 8 July 2020).
95. 3GPP TR 23.791 (V16.1.0): Study of Enablers for Network Automation for 5G, Phase 2 (Release 17), February 2020. Available online: https://www.3gpp.org/ftp/Specs/archive/23_series/23.700-91/23700-91-030.zip (accessed on 8 July 2020).
96. ETSI GR ZSM 004 V1.1.1 (2020-03) Zero-Touch Network and Service Management (ZSM), 2020. Available online: https://www.etsi.org/deliver/etsi_gr/ZSM/001_099/004/01.01.01_60/gr_ZSM004v010101p.pdf (accessed on 8 July 2020).
97. FG-ML5G-ARC5G "Unified Architecture for Machine Learning in 5G and Future Networks", 2019. Available online: https://www.itu.int/en/ITU-T/focusgroups/ml5g/Documents/ML5G-delievrables.pdf (accessed on 17 June 2020).
98. Supplement 55 to ITU-T Y.3170 "Machine Learning in Future Networks Including IMT-2020: Use Cases", 2019. Available online: https://www.itu.int/rec/T-REC-Y.Sup55-201910-I (accessed on 8 July 2020).
99. ITU-T Y.3172 "Architectural Framework for Machine Learning in Future Networks including IMT-2020", 2019. Available online: https://www.itu.int/rec/T-REC-Y.3172-201906-I/en (accessed on 8 July 2020).
100. ITU-T Y.3173 "Framework for Evaluating Intelligence Level of Future Networks including IMT-2020: Use Cases", 2020. Available online: https://www.itu.int/rec/T-REC-Y.3173-202002-I (accessed on 8 July 2020).
101. ITU-T Y.3174 "Framework for Data Handling to Enable Machine Learning in Future Networks including IMT-2020: Use Cases", 2020. Available online: https://www.itu.int/rec/T-REC-Y.3174-202002-I (accessed on 8 July 2020).
102. Zheng, Y.; Xu, S.; Dhody, D. Usecases for Network Artificial Intelligence (NAI). Internet-Draft, 2017. Available online: https://tools.ietf.org/html/draft-zheng-opsawg-network-ai-usecases-00 (accessed on 8 July 2020).
103. Thantharate, A.; Paropkari, R.; Walunj, V.; Beard, C. DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks. In Proceedings of the 10th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON 2019), New York, NY, USA, 10–12 October 2019; pp. 762–767.

104. 3GPP TR 23.791 (V16.2.0): Technical Report. 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study of Enablers for Network Automation for 5G (Release 16), June 2019. Available online: http://www.3gpp.org/ftp//Specs/archive/23_series/23.791/23791-g20.zip (accessed on 8 July 2020).
105. Boubendir, A.; Guillemin, F.; Kerboeuf, S.; Orlandi, B.; Faucheux, F.; Lafragette, J. Network Slice Life-Cycle Management Towards Automation. In Proceedings of the IFIP/IEEE Symposium on Integrated Network and Service Management (IM 2019), Arlington, VA, USA, 8–12 April 2019; pp. 709–711.
106. Si, M. Nation Ushers in 5G commercial Service Era. China Daily Press, November 2019. Available online: https://www.chinadaily.com.cn/a/201911/01/WS5dbb25a2a310cf3e35574c30.html (accessed on 8 July 2020).
107. Tran, T.X.; Hajisami, A.; Pandey, P.; Pompili, D. Collaborative Mobile Edge Computing in 5G Networks: New Paradigms, Scenarios, and Challenges. *IEEE Commun. Mag.* **2017**, *55*, 54–61. [CrossRef]
108. Dell Technologies and Orange Collaborate for Telco Multi-Access Edge Transformation. *Dell Technologies News*, May 2019. Available online: https://corporate.delltechnologies.com/en-ie/newsroom/dell-emc-and-orange-collaborate-for-telco-multi-access-edge-transformation.htm (accessed on 8 July 2020).
109. NVIDIA with Microsoft Announces Technology Collaboration for Era of Intelligent Edge: Microsoft's Intelligent Edge Solutions Extended with NVIDIA T4 GPUs to Help Accelerate AI Across Industries. Globe Newswire Press. October 2019. Available online: https://www.globenewswire.com/news-release/2019/10/21/1932901/0/en/NVIDIA-with-Microsoft-Announces-Technology-Collaboration-for-Era-of-Intelligent-Edge.html (accessed on 8 July 2020).
110. World Wide Technology and MobiledgeX Expand Partnership to Accelerate Mobile Edge Computing Deployments and Power 5G Profitability. Business Wire Press. October 2019. Available online: https://www.businesswire.com/news/home/20191021005117/en/World-Wide-Technology-MobiledgeX-Expand-Partnership-Accelerate (accessed on 8 July 2020).
111. Novo, O. Blockchain meets IoT: An architecture for scalable access management in IoT. *IEEE Internet Things J.* **2018**, *5*, 1184–1195. [CrossRef]
112. Wang, J.; Pan, J.; Esposito, F.; Calyam, P.; Yang, Z.; Mohapatra, P. Edge Cloud Offloading Algorithms: Issues, Methods, and Perspectives. *ACM Comput. Surv.* **2019**, *52*, 1–23. [CrossRef]
113. Li, S.; Tao, Y.; Qin, X.; Liu, L.; Zhang, Z.; Zhang, P. Energy-Aware Mobile Edge Computation Offloading for IoT Over Heterogenous Networks. *IEEE Access* **2019**, *7*, 13092–13105. [CrossRef]
114. Mehrabi, A.; Siekkinen, M.; Ylä-Jääski, A. Cache-Aware QoE-Traffic Optimization in Mobile Edge Assisted Adaptive Video Streaming. *arXiv* **2018**, arXiv:1805.09255. Available online: https://arxiv.org/abs/1805.09255 (accessed on 8 July 2020).
115. Sasikumar, A.; Zhao, T.; Hou, I.H.; Shakkottai, S. Cache-Version Selection and Content Placement for Adaptive Video Streaming in Wireless Edge Networks. *arXiv* **2019**, arXiv:1903.12164. Available online: https://arxiv.org/abs/1903.12164 (accessed on 8 July 2020).
116. Ren, P.; Qiao, X.; Chen, J.; Dustdar, S. Mobile Edge Computing—A Booster for the Practical Provisioning Approach of Web-Based Augmented Reality. In Proceedings of the 3rd ACM/IEEE Symposium on Edge Computing (SEC 2018), Bellevue, WA, USA, 25–27 October 2018; pp. 349–350.
117. Jeong, S.; Simeone, O.; Kang, J. Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning. *IEEE Trans. Veh. Technol.* **2018**, *67*, 2049–2063. [CrossRef]
118. Yang, C.; Liu, Y.; Chen, X.; Zhong, W.; Xie, S. Efficient Mobility-Aware Task Offloading for Vehicular Edge Computing Networks. *IEEE Access* **2019**, *7*, 26652–26664. [CrossRef]
119. Liu, Y.; Yang, C.; Jiang, L.; Xie, S.; Zhang, Y. Intelligent Edge Computing for IoT-Based Energy Management in Smart Cities. *IEEE Netw.* **2019**, *33*, 111–117. [CrossRef]
120. Nikouei, S.Y.; Chen, Y.L.; Song, S.; Xu, R.; Choi, B.Y.; Faughnan, T.R. Real-Time Human Detection as an Edge Service Enabled by a Lightweight CNN. In Proceedings of the 3rd IEEE International Conference on Edge Computing (EDGE 2018), Milan, Italy, 8–13 July 2018; pp. 125–129.
121. Chen, J.; Li, K.; Deng, Q.; Li, K.; Yu, P.S. Distributed Deep Learning Model for Intelligent Video Surveillance Systems with Edge Computing. *arXiv* **2019**, arXiv:1904.06400. Available online: https://arxiv.org/abs/1904.06400 (accessed on 8 July 2020).

122. 5G Systems Enabling the Transformation of Industry and Society. Technical Report, Ericsson, 2017. Available online: https://www.ericsson.com/en/reports-and-papers/white-papers/5g-systems--enabling-the-transformation-of-industry-and-society (accessed on 8 July 2020).
123. Li, L.; Ota, K.; Dong, M. Deep Learning for Smart Industry: Efficient Manufacture Inspection System with Fog Computing. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4665–4673. [CrossRef]
124. Li, X.; Wan, J.; Dai, H.N.; Imran, M.; Xia, M.; Celesti, A. A Hybrid Computing Solution and Resource Scheduling Strategy for Edge Computing in Smart Manufacturing. *IEEE Trans. Ind. Inform.* **2019**, *15*, 4225–4234. [CrossRef]