A Survey on Requirements of Future Intelligent Networks: Solutions and Future Research Directions

ARIF HUSEN, MUHAMMAD HASANAIN CHAUDARY, and FAROOQ AHMAD,
Department of Computer Science at COMSATS University Islamabad, Lahore Campus, Pakistan

The context of this study examines the requirements of Future Intelligent Networks (FIN), solutions, and current research directions through a survey technique. The background of this study is hinged on the applications of Machine Learning (ML) in the networking field. Through careful analysis of literature and real-world reports, we noted that ML has significantly expedited decision-making processes, enhanced intelligent automation, and helped resolve complex problems economically in different fields of life. Various researchers have also envisioned future networks incorporating intelligent functions and operations with ML. Several efforts have been made to automate individual functions and operations in the networking domain; however, most of the existing ML models proposed in the literature lack several vital requirements. Hence, this study aims to present a comprehensive summary of the requirements of FIN and propose a taxonomy of different network functionalities that needs to be equipped with ML techniques. The core objectives of this study are to provide a taxonomy of requirements envisioned for end-to-end FIN, relevant ML techniques, and their analysis to find research gaps, open issues, and future research directions. The real benefit of ML applications in any domain can only be ensured if intelligent capabilities cover all of its components. We observed that future generations of networks are heterogeneous, multi-vendor, and multidimensional, and ML can provide optimal results only if intelligent capabilities are used on a holistic scale. Realizing intelligence on a holistic scale is only possible if the ML algorithms can solve heterogeneous problems in a multi-vendor and multi-dimensional environment. ML models must be reliable and efficient, support, and possess the capability to learn and share the knowledge across the network layers and administrative domains to solve issues. First, this study ascertains the requirements of the FIN and proposes their taxonomy through reviews on envisioned ideas by various researchers and articles gathered from reputed conferences and standard developing organizations using keyword queries. Second, we have reviewed existing studies on ML applications focusing on coverage, heterogeneity, distributed architecture, and cross-domain knowledge learning and sharing. Our study observed that in the past, ML applications were focused mainly on an individual/isolated level only, and aspects of global and deep holistic learning with cross-layer/cross-domain knowledge sharing with agile ML operations are not explored at large. We recommend that the issues mentioned previously be addressed with improved ML architecture and agile operations and propose an ML pipeline based architecture for FIN. The significant contribution of this study is the impetus for researchers to seek ML models suitable for a modular, distributed, multi-domain, and multi-layer environment and provide decision making on a global or holistic rather than an individual function level.

Authors' address: A. Husen, M. H. Chaudary, and F. Ahmad, Department of Computer Science and Information Technology, Virtual University of Pakistan, 41- D Lawrence Road, Jubilee Town, Lahore, Punjab 54000, Pakistan; emails: arifhrashid@gmail.com, {sp19-pcs-006, mhchaudary, farooqahmad}@cuilahore.edu.pk.
Authors current address: Arif Husen, Department of Computer Science at COMSATS University Islamabad, Lahore Campus, Pakistan Department of Computer Science and Information Technology, Virtual University of Pakistan.

0360-0300/2022/11-ART73 $15.00
https://doi.org/10.1145/3524106

CCS Concepts: • **Networks** → **Network management**; **Mobile networks**; **Network manageability**; **Network dynamics**; *Network structure*; *Logical/virtual topologies*; • **Computing methodologies** → Distributed computing methodologies; **Machine learning**; *Planning and scheduling*; *Planning for deterministic actions*;

Additional Key Words and Phrases: Future intelligent networks, global learning, cross-administrative domain learning, knowledge sharing, cross-layer learning, feature sharing, deep holistic learning

ACM Reference format:
Arif Husen, Muhammad Hasanain Chaudary, and Farooq Ahmad. 2022. A Survey on Requirements of Future Intelligent Networks: Solutions and Future Research Directions. *ACM Comput. Surv.* 55, 4, Article 73 (November 2022), 61 pages.
https://doi.org/10.1145/3524106

1 INTRODUCTION

Traditional networks are characterized by human-assisted daily operations along with rule-based automation and decision-making matters [1–3]. However, due to the continuous proliferation of artificial intelligence applications in all fields of life, the networks must shift from the traditional approach to a new dimension [4]. The new approach requires networks to provide self-aware, customizable, flexible, and adaptable behavior with the assurance of security and privacy in its processes. These features are expected to be inducted into 6G and beyond networks, formally referred to in this study as **Future Intelligent Networks (FIN)**.

The success of FIN relies on a dynamic service level isolation enabled by **Network Slicing (NS)** and intelligent decision-making capabilities provided by **Machine Learning (ML)** techniques for networks [5]. Several researchers have envisioned the usage scenarios of ML techniques for 6G networks to realize the intelligent capabilities in terms of autonomous operations and intelligent services. However, the intelligent capabilities will extend beyond the 6G vision due to continuous networking technologies and ML techniques developments. 3GPP (the 3rd Generation Partnership Project) introduced NS in Release-15 [6] to fulfill the service level isolation requirements with the help of several recent developments in networking and computing technologies. The isolation provided by these technologies can be physical or virtual depending upon the type of resources and functions used in a network [7]. Furthermore, the configuration and optimization of resources and functions can be performed with intelligent decision-making capabilities provided by the ML algorithms in an autonomous and adoptable way [8].

The need for intelligent behavior of networks is motivated by the crucial necessity to eliminate underlying infrastructure complexity and enable service-related information exchange between multiple networks and intelligent user devices in real time [9–11]. Significant and beneficial future applications such as vehicular networks, autonomous vehicles, remote surgery, and the tactile internet will depend on the intelligent functionalities of networks. The aspects of smart services and networks have been discussed in the literature for the 6G and beyond era that will use ML capabilities intensively [12]. The intelligent capabilities to achieve self-aware automation will enhance performance in numerous essential aspects such as security, fault management, **Quality of Service (QoS), Quality of Experience (QoE)**, and energy conservation. Furthermore, the edge devices that will be used in the future will have dedicated hardware capable of running localized ML algorithms in a distributed fashion that is of a different approach in comparison to the traditional concept of **ML Operations (MLOps)**.

1.1 Motivation

Our research noted that the ML-based network functions and operations are vital enablers for FIN. It is also established that heterogeneity, multiple administrative domains, **Distributed Architecture (DA)** with a centralized control plane, the diversity of the requirements of services,

Table 1. Summary of Existing Surveys

Year	Author/s [Reference]	ML Applications	NS	Resource Optimization	GL/DHL	CAD/CL Optimization	KS	MLOps
2014	Alsheikh et al. [20]	✓						
2015	Chen et al. [21]	✓	✓					
	Zorzi et al. [22]	✓	✓					
2016	Buda et al. [23]	✓	✓					
	Keshavamurthy & Ashraf [24]	✓	✓					
	Alsheikh et al. [25]	✓	✓					
	Richart et al. [26]		✓					
2017	Klaine et al. [27]	✓		✓				
	Jiang et al. [28]	✓	✓					
	Li et al. [29]	✓		✓				
	Kasnesis et al. [30]	✓		✓				
	Wang & Jones [31]	✓		✓				
	Kato et al. [32]	✓	✓					
	Fadlullah et al. [33]	✓		✓				
	Foukas et al. [34]	✓	✓	✓				
2018	Mohammadi et al. [35]	✓		✓				
	Kaloxylos [36]		✓					
	Afolabi et al. [37]		✓					
	Condoluci & Mahmoodi [38]		✓					
2019	Luong et al. [39]	✓		✓				
	Zhang et al. [40]	✓		✓				
	Wang et al. [41]	✓		✓				
	Su et al. [42]		✓					
	Toscano et al. [43]	✓	✓					
	Zhang [7]	✓	✓					
	Laghrissi & Taleb [44]	✓		✓				
2020	Ma et al. [45]	✓		✓				
	Barakabitze et al. [46]		✓					
	This survey			✓	✓	✓	✓	✓

and automated operations in a multi-vendor environment are their main features. Thus, ML must support learning in distributed environments [12–16], extract the network-wide deep knowledge from hidden features [14], and share the knowledge across network layers and administrative domains to predict optimal actions. These actions need to represent both individual and global states of functions, operations, services, and network instances to provide holistic and pervasive intelligence [17, 18].

Furthermore, the **Knowledge Sharing (KS)** between the ML models in different layers and administrative domains requires standardized approaches to allow reliable and accurate decision making and interoperability [19]. Therefore, the ML-based network functions and operations have received significant attention in the past decade from the research community. As a result, several studies to enable intelligent and automated functions in different layers of the networks have been conducted, as shown in Table 1.

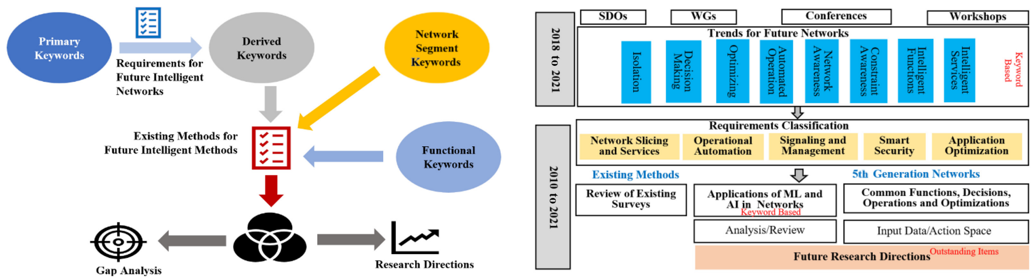
Table 2. List of Acronyms Used in the Survey

Acronym	Description	Acronym	Description	Acronym	Description
AN	Access network	IAO	Intelligent Application Optimization	QoE	Quality of Experience
ANNs	Artificial Neural Networks	IBN	Intent-Based Networking	QoS	Quality of Service
ANS	Application-Based Network Slicing	IFM	Intelligent Fault Management	RAN	Radio Access network
ANO	Automated Network Operations	ILD	Intelligent Logical Network Design and Deployment	RCA	Root Cause Analysis
BDA	Big Data Analytics	IMEI	International Mobile Equipment Identity	RF	Radio Frequency
BN	Bayesian Network (BN)	IoT	Internet of Things	RL	Reinforcement Learning
BS	Base Station	IoV	Internet of Vehicles	SA	Standalone approach
CAC	Connection Admission Control	IRA	Intelligent Resource Adaptation	SCF	Self Configuration
CAD	Cross Administrative Domain	IRM	Intelligent Radio Resource Management	SDN	Software-Defined Network
CCO	Capacity and Coverage Optimization	ISD	Intelligent Service Design	SDO	Standards Developing Organization
CL	Cross-Layer	ISO	Intelligent Security Optiizions	SDTs	Service Definition Templates
CN	Core Network	KMS	K-Means Square	SHL	Shallow Learning
CNN	Convolutional Neural Network	KPI	Key Performance Indicator	SIM	Subscriber Identity Module
CQI	Channel Quality Indicator	KS	Knowledge Sharing	SKW	Segment Keywords
CRAN	Cloud RAN	LR	Logistic Regression	SL	Supervised Learning
CRE	Cell Range Extention	LSM	Liquid State Machine	SLA	Service Level Agreement
CSI	Channel State Information	LSTM	Long Short-term memory	SMP	Smart Mobility Prediction
DA	Distributed Architecture	LTF	Long-Term Forecast	SOM	Self-Organizing Map
DDE	Dynamic Data Exchange	MBS	Macro Base Station	SONS	Self-Organized Network Slicing
DHL	Deep Holistic Learning	MC	Monte Carlo	SOP	Self Optimization
DKW	Derived Keywords	MCS	Modulation and Coding Schemes	SRCF	Self-Reconfiguration
DL	Deep Learning	MDP	Markov Decision Process	SSIM	Structural Similarity Index
DMOS	Degradation MOS	MECC	Mobile Edge Cache and Computing	SSL	Semi-Supervised Learning
DoA	Direction of Arrival	MILP	Mixed Integer Linear Programming	STC	Smart Traffic Classification
DRL	Deep RL	MIMO	Multi-Input Multi-Output	SVM	Support Vector Machine
DT	Decision Tree Based	ML	Machine Learning	TCP	Transport Control Protocol
EL	Ensemble Learning	MLOps	ML Operations	TL	Transfer Learning
ELM	Extreme Learning Machines	MLP	Multi-Layer Perceptron	TN	Transport Network
ESN	Echo State Network	MLPL	ML Pipeline	TOSCA	Topology and Orchestration Specification for Cloud Application
FIN	Future Intelligent Networks	MUE	Macrocell User Equiment	UAV	Unmanned Aerial Vehicle
FPC	Fractional Power Control	NB	Naive Bayes	UL	Unsupervised Learning
FQL	Fuzzy Q-Learning	NCR	Network Configuration Repository	UPO	User Plane Optimization
GL	Global Learning	NFV	Network Function Virtualization	VN	Virtual Networks
GRU	Gated Recurrent Unit	NS	Network Slicing	VNF	Virtual Network Function
HMM	Hidden Markov Model	PDBRs	Product Definition and Business Rules	VQM	Video Quality Metric
HNS	Holistic Network Slicing	PKWs	Primary Keywords	VUE	Vehicle User Equipment
HO	Handover	QL	Q-Learning		

Table 1 considers the ML applications for NS and resource optimization. It also evaluates the existing literature for **Global Learning (GL), Deep Holistic Learning (DHL), Cross-Administrative Domain (CAD)**, and **Cross-Layer (CL)** optimizations, KS, and MLOps. Our research has found several gaps in existing surveys, such as the absence of focus on CAD, GL/DHL, KS, and MLOps. It can be observed from Table 1 that, currently, no single study has been published in the literature covering the preceding aspects. To fulfill the gaps and help the research community align the future research, this survey presents an analysis of existing ML techniques for the networks in terms of those aspects.

1.2 Article Organization

The rest of the article is organized as follows. Section 2 discusses the survey methodology and criteria for selecting the articles and ML algorithms. It also presents a taxonomy of different ML requirements for FIN. The current vision from various researchers for 6G networks is discussed in Section 3. A brief review of the NS technologies, ML techniques, and architectures in the context of intelligent networks is covered in Section 4. After that, from Sections 5 to 10, the detailed requirements of FIN and analysis of various existing ML techniques applied to the networking domain are discussed. Section 11 discusses pipelining aspects of ML schemes with some examples. The same section also presents an ML-based architecture to fulfill the requirements of FIN. Section 12 wraps up the discussion on the requirements of the FIN and outlines various open issues, challenges, and future research directions. Finally, Section 13 concludes the study. A list of acronyms commonly used in this survey is given in Table 2 for easy reference.



(a) Literature search design (b) Requirement classification design

Fig. 1. Literature search and requirement classification designs.

2 METHODOLOGY

This section discusses the methodology used to identify the relevant articles in line with the best practices suggested by Kitchenham and Charters [47] with some modifications. The modifications are in terms of article searching strategy that uses the primary and secondary keywords instead of a forward/backward searching. This survey focuses on presenting a comprehensive summary of the requirements of FIN with a taxonomy of different network functionalities that needs to be equipped with ML techniques. The core objectives of the study are as follows:

- Develop a taxonomy for functionalities envisioned for end-to-end FIN.
- Present a summary of existing ML techniques to enable intelligent functions.
- Conduct analysis on ML techniques and in terms of FIN requirements.
- Identify open issues and future research directions.

The search method used in this study is shown in Figure 1(a). In the first step, trends and visions about FIN were searched with **Primary Keywords (PKWs)** to retrieve the articles and filtered based on the reputation of the specific event. These trends have been discussed in various articles, such as reports from working groups of various **Standards Developing Organizations (SDOs)**, research articles from reputed conferences and workshops, and a small number of journal articles from 2018 to 2020. Next, the articles found through the PKWs were used to generate new keywords called **Derived Keywords (DKWs)**. Finally, the DKWs were concatenated with network **Segment Keywords (SKWs)** and related functions to search ML-based schemes applied to the networking domain from 2010 to 2021, as shown in Figure 1(b). The distribution of the articles found with the keywords mentioned previously is shown in Table 3. The articles retrieved with DKWs were used to build the taxonomy of FIN requirements, and ML-based schemes in the articles searched through SKWs were selected for analysis in terms of key features. The article inclusion criteria for further analysis consist of the following points:

- The articles on trends, visions, and challenges for 6G and FIN and their applications are included in the study.
- The articles on using ML techniques to enable intelligent network functionality are included.
- The articles using well-established ML models, including the support for DA, **Deep Learning (DL)**, and pipeline architecture, are included.
- The latest peer-reviewed articles written in English were considered only.

Table 3. Selected Keywords

Type of Keyword	Keywords	Articles Selected					
		SDO	Conference	Journal	Website	Preprint	Book
Primary	Vision, Trends, Towards, Roadmap + Future Networks, 6G and Beyond, Intelligent networks, International Mobile Telecommunications -2000 and beyond	15	55	25	15	2	1
Derived	Intelligent, Optimization, Decision Making, Self-awareness, State-Awareness, Slicing, Autonomous	5	70	100	10	2	2
Network Segment	Access Network, Core Network, Computing, Architecture, Services, Storage						
Network Function	According to Network Segment						

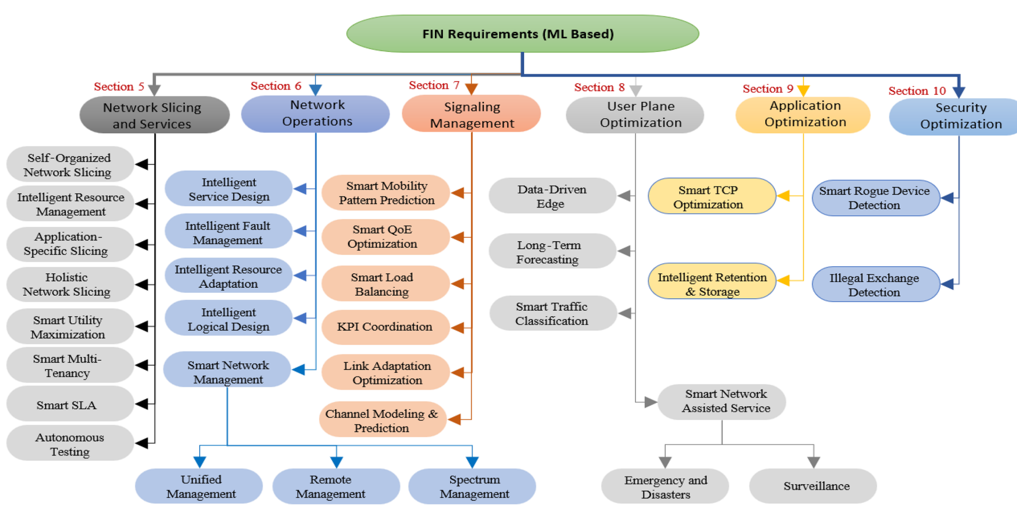


Fig. 2. The classification of ML requirements for FIN.

Studies using non-ML techniques for learning and prediction (e.g., [48–52]) were excluded from the analysis. We also excluded articles purely focused on ML techniques only and network functions only in addition to the papers that are not available as full text.

2.1 Taxonomy of FIN Requirements

The articles retrieved from different repositories using the search queries were processed with selection and exclusion criteria. The resulting articles were used to develop a taxonomy of requirements for FIN, as shown in Figure 2. The taxonomy branches are presented further from left to right for Sections 5 through 10. Each of the sections discusses the requirements of FIN in the respective branch and presents the analysis of the existing ML techniques proposed in the literature to make the networks intelligent.

The ML requirements considered in the taxonomy are also discussed in the ITU study group, SG13 [53], in the context of International Mobile Telecommunications-2020. Accordingly, requirements have been classified into six categories: intelligence requirements for NS and services, autonomous operations, signaling and management, user plane management, smart security, and application optimization. In the NS and services category, we have grouped all functions related to the creation, maintenance, and termination of network slice instances with knowledge of dynamic user requirements and network infrastructure conditions. It also covers new use cases requiring ML-level communications or coordination. The autonomous operations of FIN have diverse requirements for using ML and require a standardized reference framework that can represent the role of each network function and evaluation metrics. It also needs to cover the data generated by different network functions, and a set of potential actions that it can take. Moreover, it also needs to describe the specifications of different ML models available and their selection criteria for specific decision making or optimization.

3 CURRENT VISION FOR 6G NETWORKS

In the research literature, sixth-generation networks for mobile connectivity services are referred to as 6G networks and are still in the works [54]. Several articles have been published about future networks and connectivity requirements in 6G. We view that the 6G era services will extensively utilize artificial intelligence and ML techniques. The 6G networks will have to provide intelligent automation of functions and operations for the improved networking technologies introduced in 5G.

The exchange of network state information across the administrative domains with intelligent user devices will also be required [55]. The 6G network has been envisioned as an era of artificial intelligence and an impetus to Intelligent Networks, and thus will initiate advancements in technology such as the Internet of Intelligence [56] and Cybertwin technology [57] beyond vital automation and optimization. Essential ML requirements of 6G networks envisioned by researchers are automated management [3], intelligent control functions, programmability, and combined sensing and communications. Other characteristics of 6G networks include optimal energy conservation, dependable infrastructure, scalability, and cost-effectiveness. It is anticipated that the 6G manual design and provisioning of the services will face cost and provisioning time challenges. These challenges can potentially be addressed with ML and data-driven approaches [13]. It is observed that network management, service design, and deployment are essential to minimize human involvement and reduce the processes' time and cost [15]. Furthermore, we noted that the design, deployment, and operations support inter-user and inter-operator KS on user-centric network architecture [16].

Self-awareness, **Self-Configuration (SCF)**, and **Self-Optimization (SOP)** are also the key features of the operations of 6G networks. ML and **Big Data Analytics (BDA)** provide state-awareness and optimal decision making [10]. ML-based spectrum access [58], subnetworks, and underlay networks are also 6G era desired features that ensure guaranteed performance with predictable resource requirements and manage intra-system interference by extracting the patterns from the network traffic and related data [59]. Similarly, ML techniques will be required for vehicular networks for multi-radio access, autonomous and intelligent radio configurations, and adoptive tracking of the beamforming. Furthermore, the said networks require **Intelligent Security Optimizations (ISOs)** such as misuse detection, anomaly detection, and hybrid detection [60]. Therefore, it is expected that ML will be mandatory and will simplify the new computing architecture [11]. Furthermore, this new architecture requires intelligent radio resource allocation techniques based on an interdisciplinary approach [61]. These aspects will provide knowledge and pattern-based cognitive and self-aware optimization through the knowledge extracted with BDA techniques.

Our study noted that the super **Internet of Things (IoT)** envisioned by Zhang et al. [2] for 6G networks relies on ML. The ML techniques will address the issues in cognitive spectrum sharing, localization and sensing capabilities, and achievement of extreme performance in a subnetwork environment [13]. The 6G networks are expected to use dynamic user segmentation and resource bundling capabilities. ML will classify users, predict their needs, and determine the most efficient resource configurations. Moreover, the microservices, grafting, and streamlining procedures will enable new resource combinations and leverage the needs of resources within a novel platform-based ecosystem for 6G business models [62]. The ML is also considered as the key enabler [63] for massive machine type communications in the 6G era. It will bed for extracting the specific features from the traffic [64], predicting the node transmission time, and subscriber identification from **Radio Frequency (RF)** signatures [65]. The knowledge derived from the features can be used for autonomous resource allocation, making scheduling decisions intelligently, minimizing the resource acquisition delay, and making identity management reliable. The design [66] and provisioning processes of NS enabled by a **Software-Defined Network (SDN)** and **Mobile Edge Cache and Computing (MECC)** need to be autonomous in a distributed environment of 6G [9, 55, 58] to extend its capabilities. The reliability of automated processes depends on ML capabilities to track changes, approximating the uncertainties, making decisions, and generating the reconfiguration of heterogeneous network functions [67]. Furthermore, **Self-Organized Network Slicing (SONS)**, autonomous channel modeling, the prediction of channel conditions, and user movements are identified as requirements of ITU-T [68], for which the ML will be used to learn the state of users and networks to make optimal decisions.

4 REVIEW OF NS AND ML TECHNIQUES

Two major requirements of the FIN are end-to-end network partitioning corresponding to service or user categories and intelligently automation of the functions and operations of the underlying network. Networking slicing is the primary approach used to partition the network functions and resources, and ML techniques are used to automate and optimize configuring virtual functions and operational tasks. This section presents a brief overview of the current state-of-the-art NS, underlying technologies, services, and ML-related concepts. These topics will assist in the understanding of different aspects of slicing the network functions and resources and how the ML techniques can be used for their optimization.

4.1 Network Slicing

NS is an essential requirement for the FIN, which provides physically or virtually isolated instances of network functions, resources, and controls for different services to meet distinct requirements of users. Therefore, it is critical to provide autonomous customized subnetworks and services over the same infrastructure with independence from other users and networks. NS was incorporated in 5G network architecture to cope with diverse requirements of different types of services and multiple usage scenarios for end users and service providers. It relies on SDN, **Network Function Virtualization (NFV), Virtual Network (VN)**, MECC, and dynamic orchestration and management platforms. Additionally, NFV [69] provides the capability of instantiating various network functions in the network that includes **Access Network (AN), Core Network (CN)**, and **Transport Network (TN)** nodes. In 5G, NS provides three primary services: enhanced mobile broadband, ultra-reliable low latency, and massive machine-type communication. However, the 6G and beyond networks need to support new NS categories that can communicate with the end-user devices for optimal operations. Therefore, ML must automate the slice life cycle and perform the necessary optimization based on the knowledge learned from the user's data and network states. Moreover, ML algorithms shall evaluate the user requirements, predict the design of the end-to-end slice instances, and generate automated workflows to provision the VNs, **Virtual Network**

Function (VNF), and other computing resources. The optimizations required during the life cycle of an NS instance include the resource allocation, fault management, and migrations of VNF instances to optimal locations in the network.

4.2 ML Techniques

In the context of the FIN, ML is used to learn the state and behavior of network resources or functions and adapt them for specific objectives without explicit instructions. In addition, it is also used to automate network operations and management tasks to reduce the cost and human intervention and expedite processes. The success of ML applications depends on the performance of the individual models and their ability to share learned information with other models in the same or different related domains. It should be noted that there are no ML models currently studied that can coordinate and share learned knowledge with other models seamlessly. However, it is expected that future developments in ML algorithms and models will further help us define FIN with more precise details. ML algorithms chosen for this study are based on support for simple to complex problem representations, learning deep hidden insights, the ability to use the knowledge from similar domains to solve different but similar problems, distributed learning, and computational complexity.

4.2.1 Learning Mode and Depth. ML provides different learning modes such as **Reinforcement Learning (RL), Supervised Learning (SL), Unsupervised Learning (UL)**, and **Semi-Supervised Learning (SSL)** [70]. The RL models solve the optimization problems and generally require mathematical modeling of the given problem compared to SL, UL, and SSL. SL requires strictly labeled training data compared to UL, which does not require labeled training data. The SSL learning modes also require labeled training data but may relax the strict need to label the entire training dataset. In addition to the preceding learning modes, **Ensemble Learning (EL)** and **Transfer Learning (TL)** modes are provided by dedicated models.

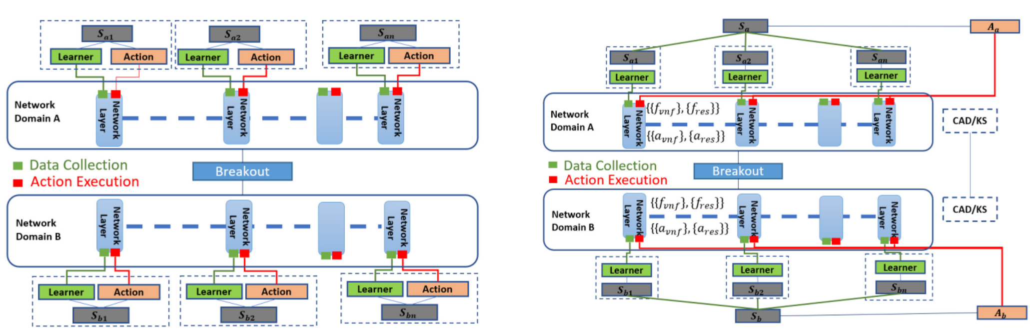
Moreover, ML models vary in terms of depth of learning. The level of knowledge learned from the given environment is represented by the depth of learning and is generally associated with **artificial neural networks (ANNs)** where fewer hidden layers provide **Shallow Learning (SHL)**. A higher number of hidden layers are used to realize DL [71]. The level of the depth requirements changes the scope of the decision or optimization or the problem nature. DL often involves computational complexity and slower response times. Some techniques support multiple learning modes and or learning depth, referred to as universal ML techniques as listed in Table 4.

The depth of universal models can be shallow or deep based on the number of hidden processing layers. Some techniques can only provide SHL or supervised or unsupervised modes, which are shown in Table 5.

4.2.2 Cross-Domain Learning and KS. CAD learning and KS allow ML algorithms to use the features extracted from one application domain to another similar domain. It expedites decision making and reduces the costs associated with redundant training. TL algorithms are often used for this purpose [81]. The future networks would require such techniques to apply the knowledge learned from one network domain or segment to another layer or segment to reduce the cost and expedite the decision making. Some studies have focused on using TL methods in the networking domain; however, the scope of their usage is quite limited. The learning methods given in Tables 4 and 5 can be adapted to realize the TL mode across multiple but closely related domains [82]. KS is generally achieved by sharing the features or weights across several ML models. Moreover, the TL itself is a mechanism to share the learned knowledge in different but similar domains.

Table 4. Universal ML Models

Model	Description	Major Features	Major Issues
ANN	Artificial Neural Network	A general term for neural networks using several distributed interconnected units arranged in layers	Blackbox, complexity [72]
ARIMA	Autoregressive Integrated Moving Average	A powerful statistical forecasting method for historical data	Requires large data size, limited extreme predictions [73]
CNN	Convolutional Neural Network	A neural network for spatial feature detections for large-scale implementations	Requires large datasets, hyperparameters [74]
ELM	Extreme Learning Machines	A feed-forward neural network model with more non-tunable hidden layers; a unifying learning platform	Infeasible generalization [75]
HMM	Hidden Markov Model	A statistical and generative model to capture unseen information from observable sequential symbols	High space and time complexity, larger seed set, no actual state transition sequence [76]
LSM	Liquid State Machine	A model for carrying out complex real-time computations on continuous input streams	Complexity [77]
LSTM	Long Short-Term Memory	A learning unit for sequential data; solves vanishing gradient and long-term dependencies	Prone to overfitting [78]
MCM	Markov Chain Model	A probabilistic graphical model for representing the dynamic processes	Not suitable for a short time interval sample [79]
VAR	Variational Autoencoder	An autoencoder that regularizes encodings distribution during training to avoid overfitting	Blurry outputs and overfitting [80]



(a) Localized learning (b) Holistic learning or GL

Fig. 3. Comparison of localized and holistic learning or GL.

4.2.3 GL and DHL. The traditional ML applications to the networking domain are localized, which involves learning from isolated domains and layers to make intelligent decisions or optimize the configurations of functions or resources of the respective domain and layer. However, for both GL and DHL, the output from several learners from heterogeneous domains and layers is used to make network-wide decisions or optimizations on a larger scale.

A comparison of localized learning and GL/DHL is shown in Figure 3(a) and (b), where S_{di} is used to represent the state of a layer in domain d and layer i, respectively. The S_{di} is the superset of the state subsets of virtual functions and resources corresponding to $\{f_{vnf}\}$ and $\{f_{res}\}$ functions, respectively. Similarly, the actions that can be performed on the function or resource of a layer are represented by subsets a_{vnf} and a_{res}. The superset of the a_{vnf} and s_{res} represents the range of an action that can be executed on a specific layer and is represented by $\boldsymbol{A_i}$. For GL, the domain of state set is the supersets $\boldsymbol{S_i}$, whose subsets are S_{di}, and for DHL, the domain of the state set is a

Table 5. Other Learning Models

Model	Description	Major Features	Major Issues
SL Techniques			
BN/NB	Bayesian Network/Naive Bayes	A conditional probability model for knowledge representation of the uncertain domain	High computation, non-automatic [83]
C4.5	A Decision Tree Classifier	Single pruning process for overfitting, handles discrete/continuous data, partial data handling	Overfitting, susceptible to noise [84]
DT/ET	Decision Tree/Ex. Decision Tree	A classification method based on trees	Overcomplexity and overfitting, prone to instability [85]
ESN	Echo State Network	An RNN model that outperforms all other non-linear dynamic models	Comparatively outdated model
			Higher space complexity
GAN	Generative Adversarial Network	A model for content generation, eliminates the need for direct data inputs	Long training time [86]
KNN	K-Nearest Neighbor	Groups data into coherent clusters and classifies the newly inputted data	Suitable for small datasets, lazy learning algorithm [87]
LDA	Linear Discriminant Analysis	A dimensionality reduction technique	Requires normal distribution, inefficient in category variables [88]
MLP	Multi-Layer Perceptron	A class of feed-forward ANN	Complexity, classify linearly separable sets
SVM	Support Vector Machine	Regression/classification, high accuracy, with low computation power	Not suitable for large datasets [89]
SW	Sliding Window	A method for framing a time series dataset for time series analysis	High computational cost [90]
RL Techniques			
ACL	Actor-Critic Learning	A learning model for simultaneous learning of policy/value	Requires homogeneous and discrete action space [91]
QL	Q-Learning	A model for measuring the action of an agent in a certain state	Applied to only discrete action and state spaces, inefficient learning in high dimensionality [92]
RBF	Radial Basis Function	A form of networks with a commonly used type of ANN for function approximation problems	Classification is slow in comparison to MLP [93]
FQL	Fuzzy Q-Learning	A learning model with prior domain knowledge in the form of a fuzzy rule set along QL	Prior domain knowledge [94]
WOLP	Wolpertinger	A DL architecture based on ACL, uses a single proto-action from the actor network and the K-closest action	Suitable for large datasets, ACL limitations [95]
XSOM	X-Self-Organizing Map	A version of the SOM algorithm that non-linearly transforms the data into a feature space	An improved version of SOM [96]

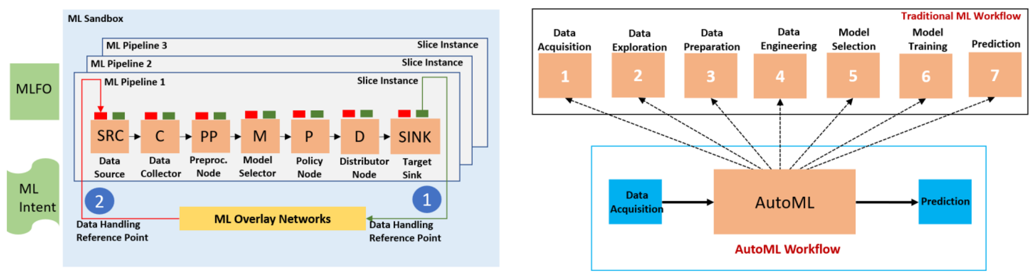
(Continued)

Table 5. Continued

Model	Description	Major Features	Major Issues
UL Techniques			
AE	Autoencoder	A model of ANN type for the task of representation learning	Not as efficient as compared to GANs [97]
KMS	K-Means Square	Well-known for clustering and simplicity	Performs poorly on non-globular clusters, highly sensitive to outliers [98]
LR	Logistic regression	Modeling a target value based on independent variables classification, cause-effect, and forecasting	Overfit the data [99]
PCA	Principal Component Analysis	A dimensionality reduction method, removes correlated features, reduces overfitting	Independent variables become less interpretable, information loss [100]
SAE	Sparse Auto Encoder	An encoder for small numbers of simultaneously active neural nodes prevents overfitting	Individual nodes of a trained model [101]
SOM	Self-Organizing Map	A model for the low-dimensional, discrete representation of the training dataset and reduced dimensionality	High computational load [102]
EL Techniques			
GAB/AB	Gradient AdaBoost/AdaBoost	A greedy boosting for high-dimensional data, also used for weak learners' identification	Needs a high-quality dataset [103]
RF	Random Forest	A learning model for low correlation features and noisy datasets	Complexity, longer training period [104]
STG	Staking	Combines multiple classifications or regression models via a meta-classifier	Complex selection of base classifiers, low accuracy, low speed and high computational cost [105]
BAG	Bagging	A technique to decrease the variance in the prediction by generating additional data for training from the dataset using combinations	Low scalability [105]
BOS	Boosting	A general ensemble method that creates a strong classifier from many weak classifiers	Low accuracy [105]
MC	Monte Carlo	MC methods provide the basis for resampling techniques like the bootstrap method for estimating a quantity related to the accuracy of a model on a limited dataset	Computationally inefficient, poor parameters and constraints, not handled

superset of states corresponding to functions $\{f_{vnf}\}$ and $\{f_{res}\}$ and action range is the superset of a_{vnf} and a_{res}. The difference between GL and DHL is that GL learners use the data consolidated at a layer or domain level only. In contrast, the DHL learners use the data generated by the learners of each of the functions, resources, and operations that a layer provides.

For FIN, there are two ways to implement GL/DHL: end-to-end service based and network instance based. The significant difference between the two is that service-based GL/DHL focuses on the user perspectives, and the network instance based approach focuses on operator or infrastructure provider objectives. Both GL and DHL also enable KS about states of the network resources



(a) ITU – MLPL (b) AutoML

Fig. 4. MLPL and AutoML.

and functions through the CAD and KS methods. From the FIN point of view, GL and DHL are required to address the issues of learning from ANs to CNs either based on service or slice instance. Besides the need for suitable methods for global representation of services and networks for the FIN, it should also be able to decide actions/optimizations from the related action space of the domain or layer. EL is a potential technique to combine features learned from different base classifiers [105]. A comparison of widely used EL algorithms in the networking domains is given in Table 5.

4.2.4 Learning Architecture. ML algorithms are generally designed with a centralized approach or **Standalone Approach (SA)** [106] in which the training data is fed to a centralized machine where training and decisions are made. However, data is generally scattered and massive for network applications, whereas the centralized approach requires massive computational resources. Furthermore, the boundary of the administrative domains may not allow the exchange of raw network data due to security and privacy concerns. Thus, SAs are least useful; instead, a DA [107] may solve the issues mentioned previously. Although several ML algorithms support the DA, common approaches adopted in the existing literature are standalone and function specific.

4.2.5 ML Pipeline and Operations. With the intense use of ML in future networks, ML algorithms need to be automated. Operations need to be deeply collaborative, designed to eliminate waste, automate as much as possible, and produce richer and more consistent insights [108]. Several automation initiatives exist in the industry, such as AutoML [109] and MLOps [110]. The ITU has outlined the **ML Pipeline (MLPL)** architecture as shown in Figure 4(a) for International Mobile Telecommunications-2020 and beyond [68]. The MLPL automates several traditional MLOps similar to AutoML, as shown in Figure 4(b); however, MLPL focuses on the network domain, and AutoML deals with the general applications. The MLPL framework consists of procedures for selecting ML models and policies managing the behavior of network nodes according to the dynamic nature of user requirements. It consists of several logical overlay nodes, such as a data source node (SRC), vendor-specific collector nodes (C), preprocessing node (PP), model selector (M), policy node (P), distributor node (D), and the target sink node (SINK). A few examples of data source nodes are user terminals, session management function, application Function, and several other virtual functions.

5 THE REQUIREMENTS FOR NS AND SERVICES

This section discusses ML-based schemes from the literature for automation tasks and optimizing parameters related to NS and services and presents their analysis in terms of FIN requirements. Each section corresponds to a requirements category in the taxonomy diagram (Figure 2) that illustrates the classification of ML requirements for FIN. We discuss the input data space, ML models,

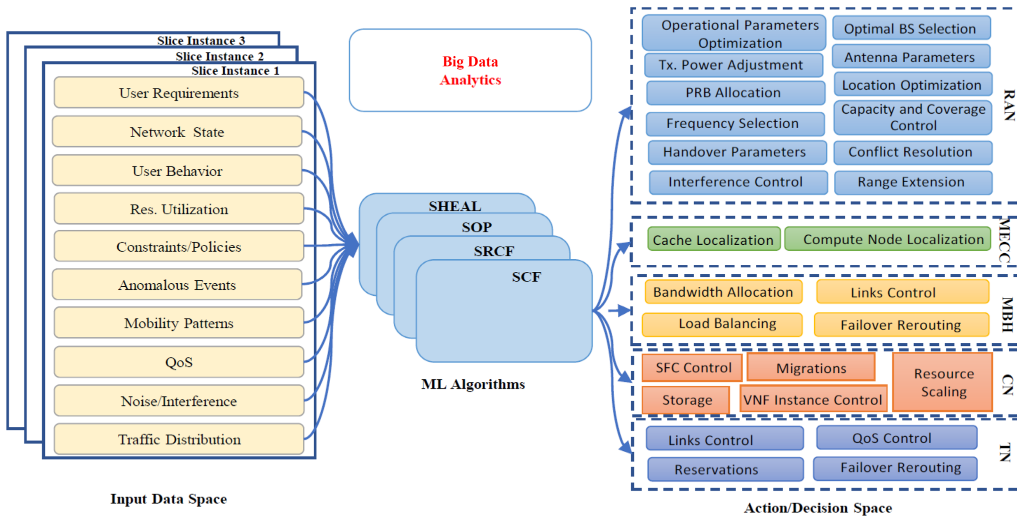


Fig. 5. Network functions, input data, and action space for SONS.

and output action space for each requirement category. Furthermore, a summary of ML models is presented for the taxonomy categories and ways to implement the MLPL for future networks. Finally, we discuss the possible ways to enable intelligent capabilities for the requirements that do not have suitable ML models.

5.1 Self-Organized Network Slicing

The automated network function slicing in the FIN is essential to fulfilling the requirements of various services, and in Figure 2, it was parked under the NS and services. SONS involves the complete intelligent operational automation, optimization, and decision during the lifecycle of the network slice instances. The primary requirements of SONS are SCF, SRCF (self-reconfiguration), SOP, and SHEAL (self-healing) [111]. The SCNF capability involves the complete automated slice services design, deployment, termination, and self-healing, discussed in detail under operational automation in Section 6. Here in the following paragraphs, we focus on SRCF and SOP. In a sliced **Radio Access Network (RAN)**, there are many options for SCNF and SOP that need to be self-organized for individual NS instances. These actions are divided into three categories, namely dynamic adjustment of parameters, capacity management, and interference coordination. The SONS techniques require learning the behavior of users, network conditions, and selecting optimal actions from the corrective action space to improve the QoS and QoE of NS instances. The data generated from input data space, as shown in Figure 5, is of large-scale, heterogeneous, and diverse formats. Therefore, the ML models require access to the data and should perform suitable analysis. The knowledge learned from this analysis represents the state of an NS instance at a particular time. Moreover, the corrective action space is also diverse and large in volume.

The global state of an instance of NS can be described by the individual state of VNFs, operations, and resources that are part of the specific instance. The global state of the individual resources state can be shared using the KS techniques with the user devices or across the administrative domains/layers for cooperative decision making.

5.1.1 Dynamic Adjustment of Parameters. Due to dynamic environmental conditions and user movements, several parameters of virtual functions and resources of NS instances require runtime adjustments. A few examples of such parameters are adjustment of operational radio parameters, antenna parameters, **Handover (HO)** optimization parameters, frequency, optimal **Base Station (BS)** selection, user association control, **Connection Admission Control (CAC)**, and location optimization. For FIN, the adjustment of the parameters mentioned previously is required to be intelligent, automated, and agile, and several ML-based schemes have been proposed in the literature in recent years. However, most of the schemes proposed for dynamic parameter adjustment exploited the RL and **Q-Learning (QL)** models and their variants [112–118].

In the work of Kong and Panaitolopol [112], QL was used to intelligently manage the dynamic resource activation and deactivation process for LTE-based RANs. The online variant of the QL was used to eliminate the training phase, and the results showed that it maximized the energy conservation by 50% without affecting the QoS constraints. In the work of Mwanje and Mitschele-Thiel [116], QL was used to automate the optimization process for values of cell individual offset parameters. The study's objective was to balance the traffic load across the cells. The results reported by the authors showed that the QL efficiently selected the optimal values of parameters and improved load balancing compared to static methods.

Similarly, Jaber et al. [117] used QL in a small cell environment to optimize the values of **Cell Range Extension (CRE)** bias based on the radio and backhaul conditions of all cells in the network. A Q table was maintained on each BS, and the QL algorithm determined an optimal CRE offset policy, improving system capacity with minimal cost and without affecting QoE. Another use of QL was demonstrated in another work by Jaber et al. [118] to intelligently adjust the bias parameters and control the cell association in a distributed environment. The general objective was to maximize the system throughput and minimize the gap between the users' achievable and required end-to-end delay. The results showed significantly improved QoE and throughput compared to the traditional cell association schemes.

A different approach from the preceding schemes was adopted by Munoz et al. [113, 114] and Dirani and Altman [115], and QL optimization was assisted with fuzzy rules representing the domain knowledge. Munoz et al. [113, 114] used **Fuzzy Q-Learning (FQL)** to self-tune the femtocell parameters to solve the localized congestion problems. Their results showed that QL with fuzzy rules provides better performance and faster response time response to the congestion events, resulting in improved performance of the femtocells. Similarly, an FQL-based scheme was also studied by Dirani and Altman [115] for a dynamic adjustment of radio resource and **Fractional Power Control (FPC)** parameters. The BS was modeled as an agent that learned from local information to optimize radio resource parameters dynamically. The FQL learned from the rapid variations in power, users' position, and interference values and adjusted the radio resource and FPC parameters. The results showed that it improved the QoS and network capacity utilization.

Some other studies in the literature used different ML models, such as RL [119, 120], ANN [121], **Self-Organizing Map (SOM)** [122], and AC [123] for dynamic adjustment of network parameters. Jaber et al. [120] were focused on multi-attribute-optimization in the distributed environment of MBH networks. RL was used on BS for optimizing the bias parameters for each **Key Performance Indicator (KPI)** to maximize the network performance and QoE. It was shown that significant improvement was achieved for QoE compared to other approaches. The schemes proposed by Galindo-Serrano et al. [119] exploited a similar ML model-based QL for automated adjustment of transmit power for femtocells in a distributed environment to optimize the capacity. ANN was studied by Adeel et al. [121] for determining the optimal radio parameters and transmit power for LTE cells. The authors used a cognition engine implementing the adaptive inertia-weighted particle-swarm-optimization, gradient descent, and differential evolution algorithms were embedded into the LTE nodes for the prediction. The adaptive inertia weight particle swarm optimization algorithm was 10.57% better than gradient descent and 8.012%

with differential evolution. However, adaptive inertia weight particle swarm optimization suffers from the disadvantages of the higher computation time.

A different ML model based on SOM was adopted in the work of Binzer and Landstorfer [122] to predict cell count, the optimal location of BS, values of transmit power, and antenna parameters to facilitate the optimal planning of code division multiple access networks. Their results indicated that the scheme had high propagation time and faster response time than other traditional techniques. Liu and Zang [123] proposed an Actor-Critic (AC) learning model to solve the CAC problem in code division multiple access cellular networks. The result showed significant performance improvement with the scheme. Santamaria and Lupia [124] also proposed a general predictor model integrated with the threshold-based statistical bandwidth multiplexing scheme for automated CAC and improved performance. However, it was implemented using the standard ML model.

5.1.2 Coverage and Capacity Management. The capacity coverage and interference management involve the physical resource block scaling, **Capacity and Coverage Optimization (CCO)**, CRE, and FPC and interference parameters control. In this regard, researchers have often used the ANN, QL, and their combination with fuzzy learning models [125–129]. For example, Debono and Buhagiar [125] analyzed cellular cluster coverage optimization with two ANNs connected in a series. The ANNs evaluated the traffic patterns discovered by statistical methods from actual network data. The scheme established a relationship between the performance of a site and clustering. The results showed improvements in optimization, primarily due to frequency reuse compared to traditional methods.

In contrast to the ANN model, a combination of fuzzy logic with QL was demonstrated in the work of Razavi et al. [126], Islam and Mitschele-Thiel [127], and Razavi et al. [128] for COO. The use of fuzzy rules representing existing knowledge of the domain allowed a jump start for the SOP problem. The existing knowledge generally represents approximate and rough estimates improved with QL as the learning iterations, an evolutionary approach. Moreover, Fan et al. [129] further extended the optimization of capacity and coverage by adding ANN to the FQL-based scheme, which focused on antenna tilt angle and transmitted power control. The cell edge and center performance indicators were jointly compared across the neighboring cells. The results showed performance improvement; however, their scheme was never tested in a real LTE environment.

Other models used in the literature for COO include **Multi-Layer Perceptron (MLP), K-Means Square (KMS)**, and regression [130–132]. Mahmood et al. [130] studied an adaptive capacity and frequency optimization method for adaptive optimization schemes based on seasonal autoregressive integrated moving averages and MLP. Both models were used to predict the traffic forecast and capacity and frequency optimization. It was shown that MLP with two layers and six hidden nodes (6/6) was adequate to achieve the desired results. Another simple study based on KMS was conducted by Savazzi and Favalli [131] in frequency division duplex cellular networks for grouping users to configure spatial beams based on **Direction of Arrival (DoA)** of uplink channels at the BS. The results showed DoA measurement improvements in comparison to heuristics-based approaches. Finally, Franco and Marca [132] used a simpler polynomial regression model for cell selection and CRE for LTE networks. Experiments showed the dynamic expansion of the small cell coverage according to traffic conditions, the balancing of traffic load, the reduction of cell congestion, and the diminishing of packet loss.

Most of the methods discussed in the preceding paragraphs are standalone and lack the learning of deep knowledge along with GL, CAD, and CL requirements.

5.1.3 Self-Coordination. A self-coordination framework for NS is an essential requirement in FIN to avoid potential conflicts in objectives or parameters values in the self-organizing process. A purposeful explicit framework was thus developed in the work of Lateef et al. [133] for cellular

Table 6. Analysis of ML Models for SONS Requirements from the Literature

Category	Ref.	Details	ML Aspects				GL/DHL	CL	CAD	KS
			Mode	Model	Arch.	Depth				
Dynamic Adjustment of Parameters	[112]	Dynamic resource activation and deactivation	RL	QL	SA	SHL	No	No	No	No
	[113, 114]	Self-tuning of femtocell parameters	RL	FQL	SA	SHL	No	No	No	No
	[115]	Dynamic adjustment of RRM and FPC	RL	FQL	SA	SHL	No	No	No	No
	[116]	Cell individual offset parameter	RL	QL	SA	SHL	No	No	No	No
	[122]	Localization, Tx power, and antenna pattern	SML	SOM	SA	SHL	No	No	No	No
	[121]	Radio parameters and Tx power prediction	SML	ANN	DA	SHL	No	No	No	No
	[117]	Backhaul-aware CRE bias	RL	QL	SA	SHL	No	No	No	No
	[118]	Cell association scheme	RL	QL	DA	SHL	No	No	No	No
	[123]	Call admission control	SML	A3C	SA	SHL	No	No	No	No
Coverage and Capacity Management	[125]	Coverage optimization	SML	ANN	SA	SHL	No	No	No	No
	[130]	Capacity and frequency optimization	SML	MLP	SA	SHL	No	No	No	No
	[131]	Utilization-based grouping classification	SML	KMS	SA	SHL	No	No	No	No
	[126, 127]	COO	RL	FQL	SA	SHL	No	No	No	No
	[132]	Cell range expansion scheme	SML	REG	SA	SHL	No	No	No	No
Self-Coordination	[134]	Self-coordination macro and femto cells	RL	FQL	DA	SHL	No	No	No	No
	[135]	ICIC and CCO coordination	RL	SVM	SA	SHL	No	No	No	No
	[136]	Coordinated cell offloading	TL	TL	DA	SHL	No	yes	No	No

networks. The authors have identified various types of conflicts in cellular networks. The following paragraph discusses the ML-based approaches to avoid such conflicts via self-coordination. The objective of inter-cell coordination like interference control, mobility robustness optimization, mobility load balancing, and resolution of inter-cell conflict are discussed. There is also a need to cover the conflicting parameter adjustments within the cell.

Bennis and Niyato [134] proposed a distributed QL algorithm for femtocells that learns from interactions with their local environment. It uses a trial-and-error approach and adapts the channel selection strategy until the desired objectives are met. It divides the problem into two subproblems: the high and the low levels. The high-level subproblem finds the channel allocation through decentralized QL, and the low-level problem targets optimizing power allocation. The analysis of the proposed scheme shows that it provides self-organizing capability through the coordination between the macrocells and femtocells. The **Support Vector Machine (SVM)**-based techniques address the issues related to self-healing. It has specific applications in the resolution of conflict between the ICIC and CCO for LTE networks [135]. Often the SONS algorithms encounter the data sparsity problem leading to inefficiency of the learning process. The TL-based technique can resolve such problems and other similar issues for cell load offloading [136]. A summary of the SONS models is given in Table 6.

5.2 Intelligent Radio Resource Management

Intelligent Radio Resource Management (IRM) is another requirement to achieve the objectives of the FIN, which is part of the SONS in the taxonomy diagram (Figure 2). Traditional radio resource management is based on a per flow or per QoS profile, which is not an efficient approach for the FIN to guarantee QoS and resource isolation. With SONS, the objective is to guarantee minimum bandwidth, reliability, and maximum utilization of scarce radio resources for each slice instance. ML-enabled radio resource management methods are needed to continuously update the prediction models based on historical data collections. It will allow the networks to improve the accuracy and reliability of predictions in real-time mode. Moreover, ML algorithms are needed to provide optimal prediction accuracy based on resource usage patterns and slice instance behavior.

Several ML models have been considered for IRM in recent literature, where a more focused study was conducted in the work of Calabrese et al. [137]. The authors discussed various approaches, frameworks, and challenges of IRM with ML. The specific requirements for the ML models for IRM are the support of reliable operation in standalone and distributed network architecture, CL learning and KS, and global or holistic optimization.

In the literature, the commonly used ML models for IRM are based on RL (e.g., [138–141]), where both SHL and DL methods have been evaluated by Feki et al. [138], who used a QL-based scheme in LTE-based vehicular networks for resource sharing and management. A similar model was adopted by Zhou et al. [139] for early resource allocation that incorporates several network status parameters and their exploitation during the learning process resulted in improved performance and accuracy. The ML model considered the future state for resource control through the learning phase. Their results showed that the proposed scheme outperformed the existing packet loss and throughput rates. Another model-based approach was proposed by Kumar et al. [140], who focused intelligently and automated management of the radio resource pool in LTE-based networks. In their experiments, a multi-operator system was modeled on the game theory based QL scheme. The results showed that it achieved higher throughput while meeting the real-time resource requirement of each player. Another RL-based approach using the DL model was studied by Du et al. [141], who discussed the architectural and algorithmic aspects of the IRM. They used a distributed **Deep RL (DRL)** model to make lightweight deep local decisions, and the processing-intensive tasks of training and updates were implemented on a cloud. It showed that the compressed **Markov Decision Process (MDP)** and spatial TL could accurately represent low-dimensional features. Spatial TL was used to learn the efficiency of DRL entities with the use of traffic correlations.

Long Short-Term Memory (LSTM)-based ML models are well known for incorporating spatial features in their optimization. In this regard, a deep LSTM model was studied by Zhou et al. [142] for localizing the traffic prediction at the BS. The proposed scheme is determined by using a suitable policy based on traffic predictions to avoid congestion intelligently. The authors have shown that this approach provides significant performance improvement in packet loss rate, throughput, and mean opinion score.

The ANN-based models for the IRM were used in the work of Sandhir and Mitchell [143] and Fazio et al. [144], where Sandhir and Mitchell studied an intelligent and dynamic resource allocation for RAN to reduce the signaling and computational overhead compared to the fixed allocation scheme. Two ANN networks were used to adopt the RAN resources by predicting the path based on the historical data [144]. The first ANN was used to predict the next cell by considering the current position and DoA, and the second ANN was used to predict the path based on the history of the first ANN transitions. The experiments have shown that this scheme reached a precision of 69% to 84% with a system utilization of 75% to 90%.

Another interesting study on IRM was conducted by Chen et al. [145], where a model for stochastic decision making as a discrete single-agent MDP for vehicular networks was used. The MDP problem was split into a series of **Vehicle User Equipment (VUE)** paired MDPs.

Table 7. Analysis of ML Models' Utility Maximization and Radio Resource Management from the Literature

	Ref.	Objective	ML				GL/DHL	CL	CAD	KS
			Model	Mode	Arch.	Depth				
Utility Maximization	[117]	CRE-based capacity utility maximization	QL	RL	SA	SHL	No	No	No	No
	[148]	Capacity utility maximization	QL	RL	DA	SHL	No	No	No	No
	[149]	Joint utility for backhaul optimization	RL-GTA	RL	SA	SHL	No	No	No	No
	[150]	Utility maximization	RL-GTA	RL	SA	SHL	No	No	No	No
Radio Resource Management	[138]	Resource sharing for LTE	QL	RL	SA	SHL	No	No	No	No
	[139]	Network state-based resource learning	QL	RL	SA	SHL	No	No	No	No
	[141]	Traffic correlations and resource sharing	DNN + TL	DRL	DA	DL	No	Yes	No	No
	[142]	Localized decisions at enhanced node B	LSTM	SML	SA	SHL	No	No	No	No
	[145]	Stochastic decisions at the BS	MDP (LSTM-SARSA)	EL	DA	SHL	No	No	No	No
	[140]	Resource pool management in a multi-operator environment	QL	RL	DA	SHL	No	No	No	No

Using a proactive algorithm based on LSTM and DRL addressed the partial observation and high dimensionality problems in VUE pairs' local network state space. With this approach, roadside units can optimally allocate frequency bands. Furthermore, the packet scheduling decisions for all slots in a decentralized fashion were achieved according to the partial observations of the global network state of VUE pairs. The authors concluded that significant performance improvements are achieved with this approach. Although researchers have recently considered various challenges for resource allocation with different ML models [146, 147], the essential requirements of FIN, such as CL and cross-domain learning, KS, and holistic IRM techniques, have still not been considered. An analysis of different ML models used in the literature for intelligent resource management is given in Table 7.

5.3 Smart Utility Maximization for Mobile Backhaul

The FIN needs to deal with diverse requirements, services, and related issues. For example, high-density video streaming requires a higher data rate, and the lowest possible latency characterizes the Internet of Vehicles. In contrast to the two scenarios, the IoT requirements are not stringent in terms of data rate, latency, and jitter, but the massive number of users is a significant challenge. Such diversity will be addressed through NS, and the ML algorithms will solve the issues of the complexity of different utility functions. The summary of various ML-based models for the smart utility maximization is given in Table 7.

We noted that the ML-based radio parameter optimization techniques often maximize the utility of one or more resources or functions. In this regard, the most related studies include optimizing CRE offset, MBH link utilization, and congestion with the QoS/QoE constraints [117, 148–150]. For example, in the work of Xu et al. [148], a distributed QL scheme was used to study the resource utilization problem of MBH links to distribute the load on backhaul links based on the system bias. The link utilization levels, the identification of the specific BS, and the weighted difference of link utilization and the probability of BS outage were used as the bias, action, and reward. The results showed an improved capability to optimize the link resource usage and QoS for users.

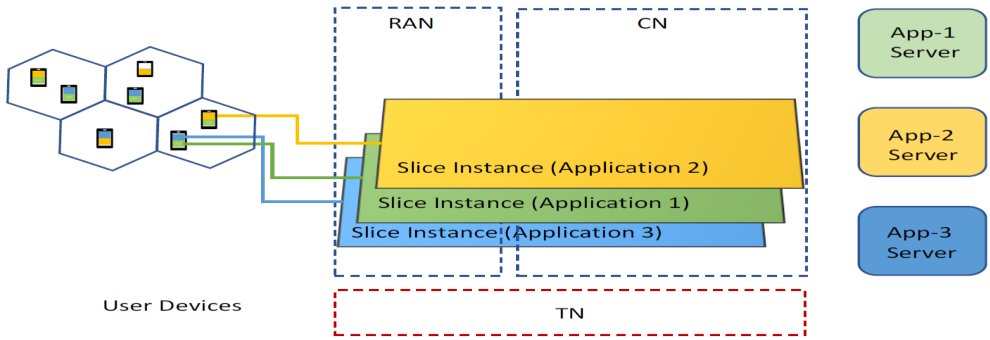


Fig. 6. Application-based network slicing.

Moreover, the RL-based game theory approach was evaluated by Hamidouche et al. [149] and Sumarakoon et al. [150] to manage backhaul links, focusing on policy estimation and joint utility for backhaul optimization. The utility maximization problem of MBH was modeled in the work of Hamidouche et al. [149] as a minimization game with the BSs as the game players. The RL was used on the BS to decide downloading content by considering high-priority requests and converging to Nash equilibrium. The RL generated the necessary information for the BSs to update their strategy with allocated utility. The results showed that the proposed scheme converges faster to reach an equilibrium. In the research work given by Sumarakoon et al. [150], the BSs acted as relays, and **Macrocell User Equipment (MUE)** can communicate with the **Macro Base Station (MBS)** via them. The heterogeneous links between the small BS and MBS are considered in their work. The non-cooperative game was modeled among the various MUEs. The action space consisted of adjustment of transmit power, selection of BS, and data rate split parameters. The authors have shown that a coarse correlation equilibrium can be obtained using the RL, and their results showed a better throughput and latency at the MUEs.

5.4 Application-Based Network Slicing

The automated deployment of applications and services with QoS constraints over a single network is one of the major objectives to be realized in the FIN. **Application-Based Network Slicing (ANS)** aims to provide NS instances based on applications instead of user groups. In Figure 6, we show that it allows the same application running on different users to be served over a single NS instance, specifically created for an application. With this approach, a user connected to a network with several applications running on his mobile device will be using different slice instances for traffic generated by various applications. The ANS-based instances are generally triggered by the application service provider having distinct objectives. It requires the knowledge of applications active on a device, its service requirements, identification and classification of traffic, and scaling up/down the slice resources dynamically. The autonomous, efficient, and reliable identification of applications is vital for the timely deployment of applications for FIN. The ML models are needed to efficiently classify applications and coordinate with other service deployment mechanisms. The critical issue is the absence of context information on the radio part of the network in contrast to CN. The traditional approaches used, such as header marking [151], deep packet inspection [152], and application signature detection [153], are inefficient due to the failure of the header marking and the existence of encrypted traffic. An application-specific NS with in-network ML is required for FIN to address the issues. This mechanism applies to RAN application or service-specific radio resource scheduling and QoS control. A similar approach is also needed for various network functions in CNs on a per application or service or per device basis.

Table 8. Analysis of ML Models for HNS from the Literature

Ref.	Objective	ML				GL/DHL	CL	CAD	KS
		Model	Mode	Arch.	Depth				
[154]	Efficient resources orchestration	DRL	RL	SA	DL	No	No	No	No
[159]	Analysis of HNS for 5GN	–	–	–	–	–	–	–	–
[155]	Open Air Interface-based slicing	–	–	–	–	–	–	–	–
[160]	delay optimization	MILP	SML	SA	SHL	No	No	No	No
	VNF placement	ANN	SML	SA	SHL	No	No	No	No
[158]	Network load efficiency: DeepSlice	ANN	SML	SA	SHL	No	No	No	No
[156]	Eliminate the security risks to the slice instances	ANN	SSML	SA	SHL	No	No	No	No

5.5 Holistic Network Slicing

The general resource allocation approaches are based on QoS provisioning techniques and only provide performance assurance on the access layer. **Holistic Network Slicing (HNS)**, in contrast, requires the resource allocations across the whole network, including the CN, RAN, TN, and interconnections. Therefore, end-to-end resource assurance is the only way to provide reliable QoS and user experience. For HNS, ML techniques are used to cluster, classify, and prioritize the resource allocations across the end-to-end network considering the network and user dynamics, thus requiring the techniques that support GL, CL, KS, and CAD features.

In the literature, a DRL-based scheme was proposed by Koo et al. [154] for efficient orchestration of several types of resources, and a derivative-based optimization framework based using DRL was studied in the work of Liu and Han [155]. The scheme proposed by Liu and Han [155] considered several heterogeneous resources, including bandwidth, computing, and links. It used the OpenAirInterface platform, LTE, SDN, and Compute Unified Device Architecture (CUDA) graphics processing unit and achieved a performance enhancement of 3.69% compared to baseline schemes. Another HNS scheme known as Secure5G used deep ANN [156] to identify and eliminate the security risks to NS instances by continuous monitoring of incoming connections. The proposed model was resilient and avoided denial of service attacks on the slice instances by quarantining potential security threats. Moreover, an end-to-end NS resource allocation algorithm based on deep QL was proposed by Li et al. [157] to jointly evaluate the RAN and CN slices to allocate resources to maximize the number of access users dynamically. The experiments showed an average access rate with the scheme higher than 97%.

In contrast to the HNS techniques mentioned previously, an RF and RNN based model, known as DeepSlice [158], focused on managing efficiency, load, and availability. The model was trained with data consisting of KPIs and analytics of received traffic, and it predicted NS instance requirements for different unknown devices. In addition to the smart selection of alternate slices, the failed instances were also considered for intelligent resource management and load-balancing across the NS instances. The challenges and issues associated with the end-to-end NS were discussed in the work of Nakao et al. [159] for 5G. The authors discussed various emerging concepts, relevant technologies, and trends for designing 5G and FIN end-to-end NS by considering different aspects such as computing and network functions. An analysis of various ML models proposed in the literature for HNS is given in Table 8.

5.6 Smart Trusted Multi-Tenancy for Crosshaul

Smart multi-tenancy provides an intelligent way of sharing the crosshaul resources in FIN. Crosshaul (Xhaul) is a TN that unifies the fronthaul and backhaul [161] to enable multi-tenancy, resource optimization, and service prediction to function [111]. To ensure cost-effectiveness for

Xhaul, the operators require sharing resources, providing high reliability, availability, and guaranteed **Service Level Agreement (SLA)** to multiple tenants; thus, the need to have GL and CAD is vital. The 3GPP has indicated one of the major challenges to be trust models for the multi-tenant Xhaul in multiple administrative domains. Furthermore, adapting to secure business for adversarial ML is crucial to managing security threats in heterogeneous environments. The multi-tenancy architecture for 5G networks was studied by Li et al. [162] using the SDN/NFV-based control plane in performance and economic efficiency perspectives using unified TN for different stakeholders. The tenants considered in their study were mobile virtual network operators, vertical industries, over-the-top, backhaul, fronthaul, and **Cloud RAN (CRAN)**. Overall, their architecture enabled the flexible and efficient allocation of resources to multiple tenants by leveraging widespread architectural frameworks for NFV and SDN; however, the aspects of GL, CAR, and KS were not considered.

5.7 Smart SLA Assurance

The FIN requires smart SLA assurance mechanisms where ML techniques shall be used to continuously learn the state of contracted performance parameters of the services. In the future, it is anticipated that the traditional networking paradigms are expected to change to a flexible network as a service, and potential applications are various industry sectors such as factories, public transportation hubs, airports, and power plants for this type of model. The SLA requirements from the sectors are significantly different, and network operators need to use the necessary methods to fulfill the requirements for each customer with a specific network slice instance [163]. The slice instance for each client must be designed to ensure the requirements of the number of users, performance, availability, and QoE. The operators will be required to constantly measure SLA KPIs for RAN, CN, and TN. The necessary actions should autonomously be triggered in case of violations of relevant KPIs in a cost-effective way by using GL, CAD, and CL techniques. We noted that SL based on a C4.5 and **Decision Tree (DT)**-based classification algorithm was proposed by Zhu et al. [164] to ensure end-to-end SLA assurance. It intelligently understands the network environment and determines the necessary action in dynamic situations by learning from the historical QoS anomalies and extracting the required information to approximate the correlations autonomously between the historical data and actual QoS anomalies. Moreover, Iannelli et al. [165] proposed a design and implementation of SLA decomposition for end-to-end slice instances consisting of multiple operator domains using the EL approach with RF, GB, and ANN-based base classifiers. The GB with ANN performance was better than the RF with ANN.

5.8 Automated Testing of Services

Before the real services are given to clients, comprehensive validation through automated testing of services is required for FIN. Autonomous road traffic monitoring, for example, is reliant mainly on intelligent activities in smart city domains, and there are specific intelligence interactions with the FIN. Therefore, it must provide tools for autonomous testing of service features and capabilities to verify network-provided services' correct functioning and dependability. The main difficulties would be the standardization of ML techniques to provide compatibility across various manufacturers or ML methodologies. Furthermore, holistically accepted procedures are necessary; however, the right solutions for these requirements are seldom addressed in the literature.

5.9 Section Summary

This section has addressed various needs for automated NS and services. It has been seen that there are several studies for ML applications in the networking domain; however, the issues of the GL,

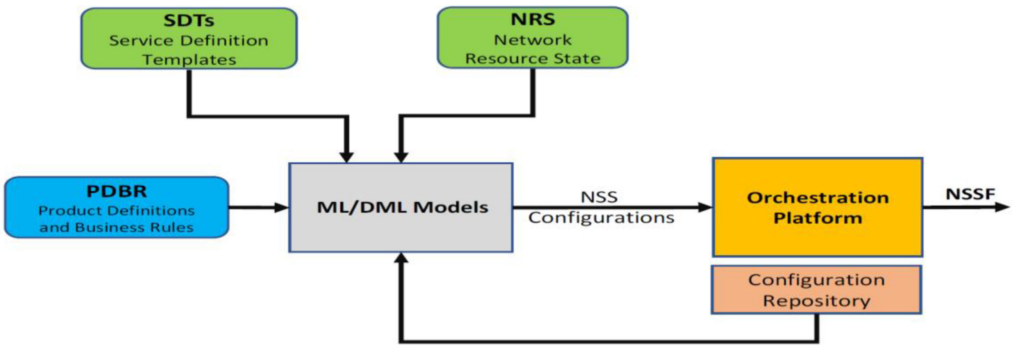


Fig. 7. Autonomous slice service design.

CAD, CL, and KS require further studies. Moreover, the requirements such as automated testing of services, smart multi-tenancy, and ANS have not been studied for localized and GL aspects.

6 AUTOMATED NETWORK OPERATIONS

The automated network operations requirement aims to manage the network routine tasks smart, intelligent, and autonomous. ML techniques for future network operations must support **Intelligent Service Design (ISD)**, resource adaptation, logical design and deployment, and fault management. In this section, we cover the specific requirements mentioned previously and discuss various existing studies that have been conducted in the past.

6.1 Intelligent Service Design

The diversity of requirements of services in terms of data rate, latency, mobility for vertical industries [166] will require distinct configurations of network functions, paths, the level of isolation, and QoS policies as part of automated ISD. The corresponding service design, network configurations, and policies will also be different for each type of service. The traditional methods of manual service design are costly in terms of deployment time, complexity, economy, and operations. The FIN needs to tackle the issues with smart, intelligent, and autonomous methods without human intervention. The ML-based methods for service design must be capable of identifying the service type and dynamically deciding the optimal network design and relevant policies. The orchestration platforms will use this design and provide the slice instance without manual intervention.

The network slice request contains specifications for the type of service, priority, isolation level, and sharing options [166] according to the product definition format published on a service provider's portal. The **Product Definition and Business Rules (PDBRs)**, network resource state, network configuration repository, and **Service Definition Templates (SDTs)** are inputted to the ML algorithm. It provides the optimal service design for network function configurations and their placement, as shown in Figure 7. The **Intent-Based Networking (IBN)** and **Topology and Orchestration Specification for Cloud Application (TOSCA)** have provided requirement specifications and formats [167]. The objective of the IBN is to provide the business objectives in abstract form to avoid the complexity of the network systems. The bottom layer uses the service contracts, which are TOSCA files used as the configuration files during the provisioning of the slice instance. This work does not consider the IBN to TOSCA translations with ML. However, the ML domain already has extensive mature solutions in natural language processing for such translations. This topic is still open for research.

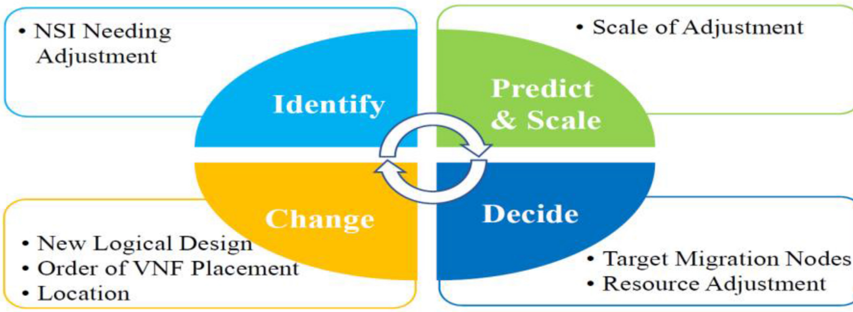


Fig. 8. Resource adaptation for the FIN.

6.2 Intelligent Resource Adaptation

In addition to the dynamic service design discussed in the previous section, the FIN would require **Intelligent Resource Adaptation (IRA)** mechanisms for heterogeneous resources such as the network functions, central processing unit, memory, links, bandwidth, and storage. The smart decisions would be required to add/remove, scale-up/down, and resource migration to fulfill the dynamics of usage and mobility of users. The network functions are chained as per the specification of service function chaining defined by IUT-T [168] that requires the adjustment of the service chains and relocation of VNFs based on the dynamic requirements to guarantee the QoS.

The IRA needs to consider the aspects of dynamic reallocation, the heterogeneity of requirements and resources such as network functions, links, and computation nodes, and SDOs have identified two basic objectives for this purpose. The most important is fulfilling the QoS requirements and the agility of the functions responsible for resource control and management. The ML requirements for IRA include the intelligent identification of slice instances for resource control, the size of resources, the decision of migration vs. scaling the resource, and the prediction of a new logical design for migration. The input data, objectives, and action space for IRA are shown in Figure 8.

In the work of Wang and Zhang [169], an RL-based framework for slice resource allocation was proposed to meet the specifications of the CRAN architecture. The framework consists of a lower layer and an upper layer. The upper layer is responsible for virtual protocol stacking functions, and the lower layer deals with power management, associations, and sub-channel allocations. It models the maximization problem as a utility function and employs an algorithm consisting of two resource allocation stages for NS. In addition, the algorithm employs QL agents to minimize the complexity of the learning process by reducing the number of entries in the Q table. The simulation results show that the proposed scheme provides better performance in terms of improved network-wise utility to understand the constraints of VN operators' baseline approaches.

In the work of Sun et al. [170], a dynamic resource reservation framework using DRL was proposed for an autonomous virtual resource slicing for the RAN. The infrastructure provider periodically reserves the free resources to instances of VN, which is based on the ratio of minimum resource requirements of all instances. The VN instances automatically control the number of resources using DRL based on the average QoS utility and resource utilization of users. Typically, mobile virtual network operators can tailor their utility and objective functions based on their specific requirements in their framework. The simulation results presented in their work show the improved performance in terms of convergence rate, utilization of resources, and satisfactory fulfillment of VN requirements.

The RF-based automated access point selection scheme was proposed by Militani et al. [171] for heterogeneous wireless networks. The experimental results show gains in the conditions of the wireless channels concerning the average throughput, and it performed better than received signal strength-based access point selection schemes. Furthermore, the issues encountered in the content caching and resource allocation in LTE-based **Unmanned Aerial Vehicle (UAV)** applications were studied in a joint caching and resource allocation scheme based on the **Liquid State Machine (LSM)** in the work of Chen et al. [172]. The LSM was used to predict the distribution of user requests for the contents with minimal information of states of the network and users and determine optimal resource allocation strategies for UAVs. The scheme results were compared with two baseline schemes, mainly QL with cache and QL without cache, and the LSM performed better than QL in terms of faster convergence and stability gains of 33.3% and 50.3%, respectively.

A deep **Convolutional Neural Network (CNN)**-based scheme was proposed by Huang et al. [173] to optimize resource allocation based on exhaustive small channel information instead of classical approaches for very dynamic wireless network environments. The results showed that the scheme performed better than the zero-forcing and is almost similar to the minimum mean squared error. However, lower computational requirements make it a promising approach. Furthermore, efficient and intelligent resource management requires historical usage information and future traffic predictions. This information is needed for RAN, CNs, and optical TNs. In this context, an ML-based scheme to predict the time-variant traffic and the blocking probability of the connections was evaluated by Singh and Jukan [174] for optical TNs for data centers. It modeled the traffic aggregation problem considering the information and requirements of applications such as latency, throughput, holding time, and traffic history.

Moreover, it aggregated and allocated the resources for the new connection requests based on mean residual time. The maximum residue limit was calculated from the mean service time and the spent time, and the mean service time was predicted with the ML algorithm. The scheme also predicted the blocking rate of the connections for the future using the future traffic forecast and historical connection blocking information. Their results showed that ML-based prediction schemes performed significantly better in reducing connection blocking and optimizing resource utilization.

6.3 Intelligent Logical Network Design and Deployment

The traditional design and deployment methods follow declarative design objectives and network specifications to avoid frequent changes to templates or scripts used in the process. A network designer generates workflows to automate the network reconfigurations in the declarative approach. The workflows are executed in different layers of the network with the help of the MLFO [68], NFV orchestrator [69], and virtual infrastructure manager [175]. However, the scripts-based automation suffers from a lack of flexibility, as only predefined scripts can be executed.

For the FIN, **Intelligent Logical Network Design and Deployment (ILD)** based on the MLPL design and deployment will be required. Furthermore, it also involves provisioning scripts, service orchestrator components, and the Open Network Automation platform. The ML techniques select a set of suitable scripts based on service requirements with objective functions to maximize the QoS/QoE. The choice of actions is limited to predefined scripts that may have been written manually. In a more flexible and smart approach, ML techniques such as natural language processing would be required to translate business requirements to a set of nodes, links, topology, policies, and their configurations. The RL techniques can optimize the different objectives like network utilization, optimal resource usage, service quality, or user experience. Such a solution will enable the complete autonomous service design and deployment without human intervention by eliminating the scripts' efforts; however, suitable ML methods have yet to be explored.

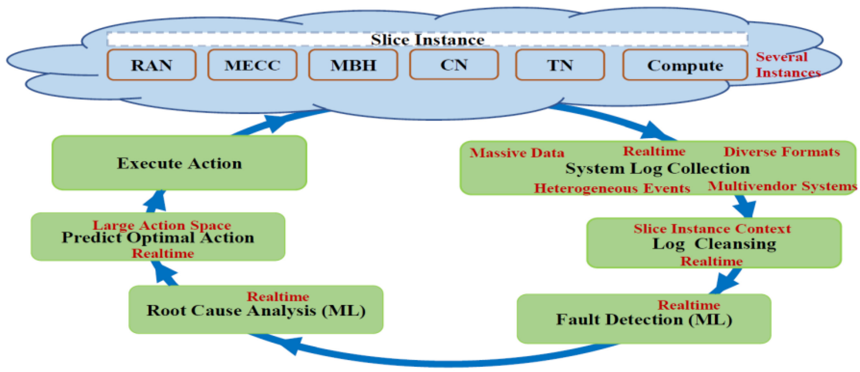


Fig. 9. Intelligent fault detection, analysis, and recovery.

6.4 Intelligent Fault Management

In traditional networks, the process of rectifying faults, errors, misconfigurations, and unexpected behaviors of VNFs is a human-based activity. In contrast, for FIN, **Intelligent Fault Management (IFM)** processes are required to automate and expedite fault detection, recovery, and reporting. IFM capability depends on the ML models that should automate the recovery process and predict the erroneous behavior based on the data analysis of heterogeneous events. We noted that, for the realization of the IFM, information about data sources, their types, and possible strategies to recover from the erroneous conditions is essential. This information allows ML developers to select a suitable ML model and train it reliably. The different kinds of input data sources for erroneous conditions and potential remedial actions are shown in Figure 9.

The input data processing required for the IFM involves massive and heterogeneous data collection, cleansing, and analysis that may hinder the timely response to the erroneous conditions. In addition, the time and space complexity of ML algorithms is essential for reliable and real-time recovery processes. Furthermore, the choice of corrective actions is vast due to the diversity of network functions and technologies. Therefore, the importance of IFM for FIN is significant, as operators will be able to quickly detect the faulty conditions and ensure autonomous recovery without human intervention. Thus, such a closed-loop operational system has been identified as the top priority for the FIN.

ML models are required to provide suitable mechanisms for the predictive detection and recovery procedures and the **Root Cause Analysis (RCA)** of abnormal conditions. It will be challenging to detect unexpected behaviors due to the heterogeneity of the software components of VNFs, software bugs, large amounts of data, and diverse formats. Furthermore, the ML algorithms used for IFM require analyzing heterogeneous log data from network functions and management data. This analysis must be performed on time to detect and classify the troublesome patterns. Detecting faults and their identification requires suitable measures incorporated in the FIN design, both in architecture and operations and maintenance processes. Such built-in characteristics will help expedite the detection of the failures and defects in the network. Moreover, ML can play an essential role in predicting faults, errors, and defects.

The future networks are expected to use the virtualized functions and resources extensively, and suitable methods and materials must be identified for the IFM in such an environment. In this regard, a dependability benchmark for NFV providers was developed by Cotroneo et al. [176, 177] to make intelligent decisions based on facts and relevant information about virtual resources and functions. The decisions are required to achieve the best dependability on virtualization, manage-

Table 9. Comparison of ML Models for IFM Requirements

Category	Ref.	Objective	ML				GL/DHL	CL	CAD	KS
			Mode	Model	Arch.	Depth				
Automated Anomaly Detection	[180]	Automated Anomaly detection	USML	SOM	SA	SHL	No	No	No	No
	[181]		EL	Several[1]	DA	SHL	No	No	No	No
	[182]		USML	KNN	SA	SHL	No	No	No	No
	[183]		SSML	BDA + LR	SA	SHL	No	No	No	No
Outage Detection and Management	[184]	Cell outage management	USML	AC + KNN	SA	SHL	No	No	No	No
	[185]	Cell outage detection	RL	AC + TD	DA	SHL	No	No	No	No
Automated Troubleshooting	[186]	Automated diagnostics and troubleshooting	USML	SOM	SA	SHL	No	No	No	No
	[187]		SML	BN	SA	SHL	No	No	No	No
	[188]		SML	Several[2]	DA	SHL	No	No	No	No
Link Failure Detection and Localization	[189, 190]	Link failure detection and localization	USML	Several[3]	SA	SHL	No	No	No	No
	[191]		SML	RNN + KNN	SA	SHL	No	No	No	No
	[192]		SML	Several[4]	SA	SHL	No	No	No	No

[1]SVM, ARIMA, and VAR.
[2]PCA + SVM, LDA + SVM, and MLP + BN + LSTM.
[3]RF, NB, LR, SVM, MLP, and DT.
[4]SVM, RF, ANN, LR, and DT.

ment, and application level solutions. The ML techniques are required to use different measures to determine the effect of faults injected in violation of the SLA, which later are used to determine latency and coverage of the management of faults. Similarly, for Industry 4.0 applications, significant work was conducted for fault detection and recovery, and a comprehensive review of various techniques was discussed in the work of Angelopoulos et al. [178].

Moreover, a detailed study on cell fault management using ML was presented in the work of Mulvey et al. [179] for 5G networks that highlighted various challenges and issues concerning cell management. A comparison of different ML approaches for IFM is given in Table 9.

Most ML-based techniques require the availability of massive logs or large amounts of data, as discussed in the literature. Various types of radio link failures have been addressed in the work of Puttonen et al. [193], for which suitable ML models are required to detect, classify, and report the events to make automated decisions. In the literature, the four major types of schemes proposed for the IFM are automated anomaly detection, outage management, automated troubleshooting, and link failure detection and localization. The details of the schemes are discussed in the following sections.

6.4.1 Automated Anomaly Detection. Network anomaly detection is a process where data analysis determines whether a specific event or condition correlates to a normal operation or abnormal. ML models in this respect can automate anomaly detection and intelligently correlate the apparent implicit events. A SOM-based scheme was evaluated by Sukkhawatchani and Usaha [180] to monitor the network traffic and detect anomalies to correlate abnormal performance indicators from the RAN traffic and facilitate the network operations by making the network troubleshooting simpler and faster. Another scheme was studied by modeling the cell behavior with EL, considering partial and complete cell performance degradation [181]. It used several base classifiers such as the sliding window, SVM, ARIMA (autoregressive integrated moving average), and VAR (variational autoencoder), and the results showed the scheme automates the processes and significantly improves anomaly detection. Finally, a KNN-based classifier was used by Xue et al. [182] for automated anomaly detection in LTE-A systems with better efficiency; however, since it used an SL approach, the labeled data availability is challenging for its broader applications. An SSL approach for anomaly detection was proposed by Hussain et al. [183] using the data generated by mobile wireless networks to avoid the dependency on the highly labeled data. It identified the low activity and high activity areas; the low activity areas are generally referred to as sleeping cells or special cases of cell outage, whereas the high activity area indicates the need for additional resources and fault avoidance. The experiments showed improved results in terms of accuracy of anomaly detection and better response time.

6.4.2 Outage Detection and Management. Cell outage detection and management is when the service downtimes are identified, and suitable actions are executed to recover from the condition. ML-based outage detection and management techniques can automate the detection and management process and determine the correct actions. An AC-based scheme for cell outage management in heterogeneous networks was proposed by Onireti et al. [184], which used a KNN and local-outlier-factor-based anomaly detection for the control plane and a heuristic grey prediction approach for anomaly detection for the data plane. The experiments have shown that it can reliably detect both control and data plane outages. Another distributed approach for cell outage compensation and frequency reuse schemes as self-healing methods was proposed by Moysen and Giupponi [185] for enhanced node Bs that control the detection of faults. It used a temporal difference learning approach with an actor-critic learning scheme that continuously interacted with the environment and learned from past actions. The experiments show significant advantages over the state-of-the-art resource allocation techniques.

6.4.3 Automated Troubleshooting. Troubleshooting is a process that is followed when a fault has occurred and usually involves tracing several logs of events generated by diverse network functions and operations. The FIN will involve many instances of the heterogeneous and virtualized functions and resources, making the troubleshooting process with manual methods expensive in terms of cost and time. Thus, suitable ML models are required to automate the troubleshooting process and expedite the recovery from faults. An LSTM-based scheme was used by Zhao et al. [188] for automated fault diagnosis, and different variants were evaluated, such as PCA with SVM, linear discriminant analysis with SVM, MLP, and **Bayesian Network (BN)** with LSTM. The classifiers were used for RCA in an SL way. The experiments have shown that BN with LSTM provides the best results compared to other schemes considered in the study. Khanafer et al. [187] proposed a BN model for automated diagnosis and troubleshooting in UMTS networks. The scheme used a BN model to automate the diagnosis and minimize entropy to improve performance by selecting optimal discretized segments of input symptoms. The scheme was tested and provided reliable results in simulated and real environments. A SOM-based automatic diagnostics system using an RCA scheme was proposed by Gomez-Andrades et al. [186] for LTE networks. A self-healing scheme refined the diagnostic process with silhouette index and accuracy improvements with a novel adjustment process. The scheme was tested in a simulated environment, and the results showed improved performance compared to the reference approaches. However, the scheme was not tested in a real LTE environment.

6.4.4 Link Failure Detection and Localization. In FIN, NS instances corresponding to distinct requirements shall cater to the services. The NS may consist of several virtual and physical links with different properties providing connectivity between different functions and resources geographically located in other locations. Traditional methods of monitoring the health and connectivity of the link with given properties are a tedious task; thus, ML techniques are required to analyze the events generated by several links and predict their status according to certain KPIs. In recent literature, different ML models are used for this purpose. For example, Srinivasan et al. [189, 190] proposed an ML-based passive scheme for link failure detection and localization using traffic engineering information and specific measurements. Since it was a passive scheme, it did not incur any overhead, as it did not inject packets. Instead, it learned from propagation delay, the number of flows, and average packet loss at every node in the network under normal working conditions and failure scenarios. Several ML models were evaluated in this scheme, such as **Naive Bayes (NB), Logistic Regression (LR)**, SVM, MLP, DT, and RF. The experimental results showed that the RF-based scheme provided 90% accuracy and took less time to localize the faulty links.

Meanwhile, Khunteta and Chavva [191] proposed an RNN-based scheme to reduce the link failure during the HO for 5G networks. The RNN continuously monitored the signaling conditions, and its output was fed to a KNN classifier to predict the failure. The proposed scheme was tested in a simulated environment showing the significance. Moreover, Truong-Huu et al. [192] proposed a centralized approach for SDN networks for fault detection and localization. Several ML models, such as SVM, RF, ANN, LR, and DT, were evaluated in their study, and the experimental results were promising. However, there was an increase in node density and a slowing down of the scheme response time.

As shown in Table 9, the analysis of ML models proposed in the recent literature for IFM shows that further studies are required to incorporate the inter-domain, inter-layer fault management techniques and share the knowledge of errors and network logs. Nevertheless, the IFM techniques with the preceding features can accurately predict the corrective actions across the network in a cost-effective and expedited way.

6.5 Smart Management

Holistic management in the 6G era covers management aspects for Industry 4.0 applications, end-to-end management of networks, spectrum management, and remote management for smart cities. For I4A, 6G networks will rely on ML and BDA techniques to enable dynamic and continuous management of the Industrial IoT network on environmental observations and manufacturing patterns. Industrial IoT networks are unique in the high standards and requirements such as reliability, availability, and security. Traditionally, there are separate management systems for access, core, transmission, and cloud computing platforms [195].

6.5.1 Cross-Domain Unified Network Management. The FIN will provide application-specific end-to-end network slice instances to fulfill the distinct service requirements. For this purpose, unified management processes are crucial to monitor performance and manage systems, supporting heterogeneous technologies and multi-vendor systems throughout the network. Human operators are used to administering the system, which contributes to delays, mistakes, and a high total cost of ownership. A unified network management system that is automated and intelligently coordinated across administrative domains and network levels is required to deal with the issues. Such a system depends on ML methods and BDA to reduce human interaction with network status and policy data. For example, the EL-based RCA proposed by Turk et al. [196] must automatically discover problems and initiate suitable treatments using ML techniques without human participation. Since the RL and DRL approaches do not rely on previous information and instead gain the needed knowledge via trial and error, they are often employed to manage large and distributed networks.

6.5.2 Remote Management for Smart Cities. Smart city services are anticipated to connect with networks to provide reliable and effective transmission and transfer of sensor data. Since smart cities will use a variety of ML approaches for automation, the operators' ML algorithms will need to collaborate and integrate with them. In this regard, the FIN needs to employ standardized ML

Table 10. Analysis of Spectrum Management Models from the Literature

Ref.	Objective	ML Aspects				GL/DHL	CL	CAD	KS
		Mode	Model	Arch.	Depth				
[198]	Throughput max. using DSS	RL	QL	SA	SHL	No	No	No	No
[199]	Spectrum efficiency	RL	LR	SA	SHL	No	No	No	No
[200]	Spectrum efficiency	RL	Custom	DA	SHL	No	No	No	No
[201]	Spectrum sharing	RL	QL	SA	SHL	No	No	No	No
[202]	Spectrum allocation	RL	ESN	SA	SHL	No	No	No	No
[203]	Spectrum prediction	RL	GAN	SA	SHL	No	No	No	No

techniques that can provide standardized interfaces for users to interact and facilitate the intelligent applications of smart cities. The ML algorithms will formulate the relationship between the specified KPIs in both domains to improve their thresholds. The performance of the TN is assessed against various KPIs such as packet loss ratio, latency, and jitter, with varying thresholds for each KPI, depending on the radio services in use.

6.5.3 Intelligent Spectrum Management. Massive data traffic rise in FIN requires intelligent spectrum management techniques for improving spectrum utilization. Intelligent spectrum management has been extensively addressed using SL and UL approaches in the literature. Spectrum prediction has also become a vital research area due to its wide applications in cognitive radios. It includes spectrum sensing, mobility, dynamic spectrum access, and intelligent topology control [197]. As can be seen in Table 10, in the literature there are several RL-based techniques for dynamic spectrum [198], spectrum efficiency [199, 200], spectrum sharing [201], spectrum allocation [202], and spectrum prediction [203].

6.6 Section Summary

This section has discussed various aspects of FIN operations and analysis of the existing ML schemes. It has been seen that areas such as ISD and ILD have not been addressed in the current literature, which is crucial for automating FIN operations intelligently. Moreover, the existing methods do not focus on the GL, CAD, and CL aspects required to enable holistic, automated operations in a heterogeneous and multi-vendor network environment.

7 INTELLIGENT SIGNALING MANAGEMENT

This section focuses on the ML requirements for the FIN in the domain of Intelligent Signaling Management. Mobility pattern prediction, load balancing, QoE optimization, the correlation of heterogeneous KPIs, holistic network management, smart and autonomous channel modeling and prediction, and smart link adaptation will be covered in this section.

7.1 Smart Mobility Pattern Prediction

Smart Mobility Pattern Prediction (SMP) is an intelligent way of determining the user trajectory, usage pattern, and context information in advance. The FIN requires suitable ML techniques for determining optimal configurations, proactive resource allocation, improved HO, intelligent caching, and energy conservation to optimize the network utilization dynamically. The SMP will enable several new applications such as adaptive public transportation solutions, adaptive streetlights, smart home heating systems, and location-based advertisements. In LTE networks, HO management is quite complicated, and there are more stringent requirements such as the zero-latency HO and consistent user experience. The model-free or hybrid model based data-driven ML

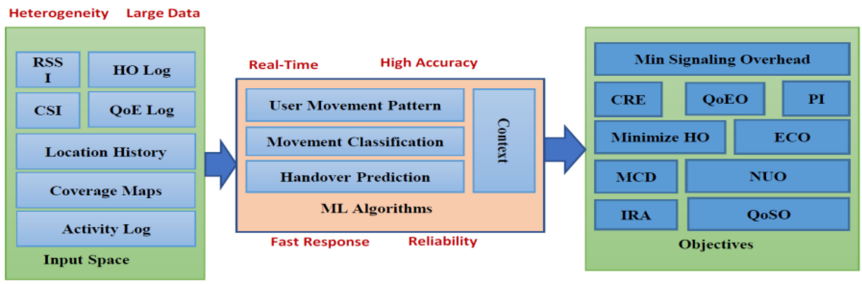


Fig. 10. Input space, analysis, and objectives for SMP.

Table 11. Analysis of ML Models for SMP from the Literature

Ref.	Description	ML				GL/DHL	CL	CAD	KS
		Model	Mode	Arch.	Depth				
[204]	User Movement Prediction	Several[1]	SML	SA	SHL	No	No	No	No
[205]	User Movement Prediction	SVM	SML	SA	SHL	No	No	No	No
[206]	User Movement Prediction	ANN	SML	SA	SHL	No	No	No	No
[207]	HO Procedure Optimization	MCH	SML	SA	SHL	No	No	No	No
[208]	Detection of Spatiotemporal Features of Mobility	MCH	SML	SA	SHL	No	No	No	No
[210]	Detection of Spatiotemporal Features of Mobility	MCH	SML	SA	SHL	No	No	No	No
[211]	Optimal Localization of BS	Several[2]	SML	SA	SHL	No	No	No	No
[212]	Optimal Localization of BS	NB	SML	SA	SHL	No	No	No	No
[213]	Online HO Prediction	MCH	SML	SA	SHL	No	No	No	No
[214]	HO Decision Optimization	ANN	SML	SA	SHL	No	No	No	No
[215]	HO Decision Optimization	ANN	SML	SA	SHL	No	No	No	No
[96]	Minimization of HO Frequency	XSOM	SML	SA	SHL	No	No	No	No
[216]	Minimization of HO Frequency	SOM	SML	SA	SHL	No	No	No	No
[217]	Mobility Robustness Optimization	QL	RL	DA	SHL	No	No	No	No
[218]	Optimal Cell Selection	QL	RL	SA	SHL	No	No	No	No
[219]	HO Management	ANN	SML	SA	SHL	No	No	No	No
[220]	HO Management	ANN	SML	SA	SHL	No	No	No	No

[1] AB, DT, NB, and KNN.
[2] DT, NB, and ANN.

techniques need to have the capability of being location-aware and predict the movement direction and speed. The SMP involves three phases: real-time data collection, strategy, and execution. The ML algorithms will be used in the CN to learn the user trajectory and formulate suitable strategies in an automated way. The execution functionality generates man-machine language [172] and communicates with the base transceiver station and other elements of the CN. The traditional mobility management methods consist of updated databases with user movements; however, such approaches cannot fulfill the intelligent SOP of mobility management. The input data space, their properties, the type of predictive analysis, and objectives for the SMP are shown in Figure 10, and analysis of different existing ML models is presented in Table 11.

Several ML applications on mobility-related aspects were studied in the literature. The focus of researchers in these studies was on user movement prediction, the effects of the spatio-temporal characteristics of user movements, optimal localization of BS, HO prediction, and minimizing the HO frequency. In the following paragraphs, we discuss different approaches adopted in the literature.

7.1.1 User Location Prediction. The availability of the user locations and their movement patterns in advance helps to make various resource allocation and optimization decisions. Several studies were conducted to predict the user location and movement patterns based on the analysis of historical data. First, the merits and demerits of different ML models were evaluated by Yu et al. [204] for user location or movement prediction using real-life trajectory data's individual and common activity patterns. The results showed that the AB-based scheme improves the prediction performance robustly and achieves an accuracy of 98% compared to DT, NB, and KNN. Finally, Chen et al. [205] studied a multi-class SVM-based classification scheme for user location prediction. The authors used the **Channel State Information (CSI)** and HO log data, and the experimental results showed that SVM provides predictions at a high rate and high accuracy with only 60% of CSI data.

Moreover, Akoush and Sameh [206] researched a hybrid Bayesian neural network scheme for user location prediction for cellular, Wi-Fi, and WIMAX networks. The data used is the usage data of user devices such as call logs, application logs, and charging status. The authors focused on reducing the location management cost and paging delay and comparing the scheme with five non-hybrid neural networks. In addition, Mohamed et al. [207] conducted a study on an online mobility prediction scheme based on the MCH to optimize the HO procedure and reduce the interruption time and HO-associated signaling overhead in LTE networks. The scheme was tested on the isolated control plane and data plane, and the scheme specifically focused on the data plane-related optimization.

7.1.2 Spatial and Temporal Pattern Analysis. Spatio-temporal analysis of the mobility data provides important features in terms of time and space. It is used to predict the number of users and their movement at a given time and position in the wireless networks. In the literature, the mobility prediction in terms of spatial and temporal features of user mobility was studied by Si et al. [208] and Lv et al. [209] with the **Hidden Markov Model (HMM)**. The scheme was implemented on base station controllers and with a lesser number of states and movement history data and provided better accuracy as compared to **Monte Carlo (MC)** and had lower time complexity bounds. Similarly, Farooq and Imran [210] used a semi-MC model for determining the user position at a given time and position in the network by using spatio-temporal features extracted from historical data. The results showed that MC provided steady state and gain with approximately 90% accuracy on real network traces.

7.1.3 Fingerprinting Cellular Devices. As with traditional networks, cell splitting techniques such as microcell, picocell, and femtocells improve wireless networks' coverage and performance. In FIN, cell splitting techniques' size shall be further reduced, and the number of smaller cells shall rise to other levels. Thus, ML-based automated and intelligent methods shall be required for fingerprinting the cellular devices by predicting their locations and determining the optimal association with BSs. Several optimizations for network resources and functions were proposed by Premchaisawatt and Ruangchaijatupon [211] with a partitioning ML classifier. It Improved the accuracy of fingerprinting indoor positions by using partial data of signal strengths and solving the clustering and classification problems. It was compared with DT, NB, and ANN. It provides improved performance with DT classifiers. Chakraborty et al. [212] also studied an alternative approach for localizing cellular devices without extensive geo-tagged data using NB and Gaussian NB. This scheme was implemented on the network side and provided higher density as the BS density was increased.

7.1.4 HO Prediction and Decision. HO prediction is another area where several ML models have been proposed in the recent literature. Mohamed et al. [213] proposed an intelligent online HO prediction scheme for preparing the resources in advance to minimize the HO latency and signaling overhead. It uses the MCM to predict the probability of the HO using the log of HO events. Bhattacharya et al. [214] studied a simple ANN-based scheme to improve the handoff decision in cellular networks and showed HO is initiated at the most suitable place, and the related overhead is reduced.

Furthermore, Ekpenyong et al. [215] proposed an ANN-based scheme for HO decision optimization. Their results showed that network performance was improved with big data and the number of neurons for signal to interference and noise ratio data. However, they have shown that increased layers resulted in degraded performance. Sinclair et al. [96] proposed an XSOM (X-Self-Organizing Map)-based scheme to detect unnecessary HOs using the history of the events in LTE femtocells. The proposed scheme was shown to reduce the number of HOs up to 70%, and, consequently, the overhead was reduced. In addition, Stoyanova and Mahonen [216] carried out a study on fuzzy logic and SOM to decide the vertical HOs. Their SOM-based scheme was not appropriate for deciding HO compared to a multi-parametric approach based on fuzzy logic. Moreover, the results revealed that SOM-based schemes were computationally costly.

7.1.5 HO Performance Optimization. ML models have also been proposed in the literature to optimize the performance during and after HO. Mwanje and Mitschele-Thiel [217] used a distributed QL scheme for mobility robustness optimization that determines the optimal values for HO parameters such as hysteresis and time-to-trigger that depend on user velocities in the network. The performance of the proposed scheme is similar to the reference models for SON. Dhahri and Ohtsuki [218] studied a QL-based scheme for the best cell selection in the HO context in dense femtocell environments. The decision was made without any necessity of prior knowledge; rather, the target cell's behavior is learned online and was used to update the parameters of a fuzzy inference system. It minimized the frequency of HOs affecting the QoE. In addition to this, Narasimhan and Cox [219] studied a HO scheme using the pattern recognition from the received signal strength by using a probabilistic neural network as a pattern classifier. The results showed that the frequency of the handoffs and related overhead signaling was reduced. Ali et al. [220] studied an ANN-based HO management scheme to improve the QoE for users in LTE networks that learned from QoE changes during HO. Experiments showed that the scheme offers improved download time and data volume performance.

An analysis of the various ML models for the SMP is given in Table 11, which shows that most of the models used for SMP focused on standalone learning models, and KS, CL/CAD, and GL are still open for research.

7.2 Smart Load Balancing and Cell Splitting/Merging

The rise in mobile traffic in the future will necessitate the extensive deployment of micro and femto level BSs in FIN. ML-based techniques would be required to efficiently, smart, and intelligently handle the coordination across them to realize smart load balancing. The traditional approach to load balancing can only transfer a load of a few users for comparably short periods and does not deliver adequate results due to the manual interventions necessary to configure numerous parameters for load balancing. Large cells are divided into smaller cells in the cell splitting approach to cope with a rise in traffic volume without deploying new BSs to increase capacity. The cell merging is alternated to cell splitting to deal with situations where the traffic volumes are reduced, and it is an effective tool for cost optimization. The manual intervention for load balancing, cell splitting, and merging is ineffective for FIN. ML-based tools are required to learn the traffic demands in a real-time mode and decide what technique must be used in a predictive way. A regression-based dynamic optimization scheme for CRE offset in small cells has been presented for roadside units [117]. It distributes the load among femtocells and macrocells. The proposed method may learn its parameters, which are used to optimize the CRE offset. Xu et al. [148] proposed a distributed QL scheme for the load balancing for MBH links to improve resource utilization. The link utilization was classified into different groups used as the state variables.

The BS used the state variables to decide how the bias value needed to be changed with a reward defined as the weighted difference between link resource utilization and the outage probability. The authors of the study presented in [148] had shown through several experiments that the scheme could optimize MBH utilization within the QoS constraints

7.3 Smart QoE Optimization

The variety of multimedia services and new heterogeneous content types is expected for 6G and beyond networks. The traditional KPI-based approach to ensure the QoS will be insufficient. Furthermore, the FIN needs to deal with the diversity of requirements and performance expectations of users. SQO (smart QoE optimization) is required to measure the user's evaluation of the performance of communication services. It reflects the actual level of user satisfaction and subjective feelings about different network services. The QoE depends on QoS and is one of the main optimization objectives of mobile networks providing complex services. ML-based QoE techniques need to optimize the network configurations to maximize the QoE score for all network users while meeting the network capacity and service requirements. In this regard, Aroussi and Mellouk [221] proposed a framework to quantify the correlation between user OoE and network QoS and analyzed ML-based correlation models such as offline batch models and incremental online models. The offline models presented in the literature are either regression based or classification based. The online QoE prediction models provide better results as compared to offline models. Generally, there are different QoE parameters for different types of services. For example, the mean opinion score, **Degradation MOS (DMOS)**, peak signal to noise ratio, structural similarity index, and video quality metric are used for video streaming services. Meanwhile, the IPTV services and the underlying QoS measurements are delays, jitter, loss ratio, bandwidth, congestion time, and out-of-order delivery ratio.

Moreover, Du et al. [222] studied ANN to correlate the DMOS with QoS parameters such as delay, the jitter, loss ratio, bandwidth, congestion time, and out-of-order delivery ratio. Their results showed that QoS parameters could predict the DMOS score without human intervention. Similar experiments have been conducted in the work of Machado et al. [223] and Calyam et al. [224]. A DRL-based study was conducted by Chen et al. [225] for QoE optimization for 5G networks. It focused on QoE-aware service function chaining optimization in SDN- and NFV-enabled NS. The scheme consisted of a QoS collector. It used a link layer discovery protocol on the southbound interface of the SDN controller and a deep QL agent to support surface function chaining in NFV contexts. The scheme approximated the reward function to optimize the QoS under the constraints of the QoS. Their outcomes showed promising results in terms of the optimization of QoS in dynamic environments. Moreover, Zheng et al. [226] proposed a BDA-based framework for the QoE optimization technique for 5G networks. It addressed the issues related to data collection, storage, analytics, and UE, RAN, and CN optimizations.

7.4 Smart Channel Modeling and Channel Prediction

The FIN will have different deployment scenarios, frequency bands, and a new physical layer, mmWave, and massive **Multi-Input Multi-Output (MIMO)** to fulfill the diversity of service requirements. Wireless systems will need channel modeling and signal propagation measurement to optimize various options. The accurate prediction of channel conditions is essential to utilize the optimal transmission capacity of a physical medium. Due to the sheer wireless network's size and diversity, both BDA and ML algorithms would be necessary for the autonomous channel modeling and prediction of channel conditions. The automated smart channel modeling will deal with large-scale data from various propagation media. Channel prediction would be challenging to fulfill strict accuracy and delay requirements. ML techniques will be used to build channel models according to medium characteristics. Historical data analysis will predict the future channel response on a real-time basis. Moreover, it would aid physical layer processes by determining the appropriate channel coding and modulation strategy to prevent needless overhead transmission.

Different ML models have been evaluated in the literature for automating channel modeling and channel state prediction. Stojanovic et al. [227] conducted a performance analysis of wireless channel state prediction with the **Echo State Network (ESN)** and **Extreme Learning Machines (ELM)** by predicting the signal to noise ration in a single-input single-output system for the picocell and microcell environments. The normalized mean squared error and response time were considered accuracy and efficiency improvement indicators. The results showed that the ESN has a shorter test time and higher accuracy in picocell environments, whereas the ELM performs better in microcell environments. In the work of Aldossari and Chen [228], a regression-based ML approach for channel modeling was investigated utilizing measurements from a specific situation. The primary goal of this research was to enhance path loss models while reducing the complexity and time required to simulate the channels. In the work of Thrane et al. [229], a deep CNN-based technique for modeling wireless channels is investigated, and the resulting channel model is compared to standard models. The technique employed the satellite images along with a simple path loss model.

The scheme consisted of two ANNs and a deep CNN where the CNN processed the satellite images, and one of ANNs was used to extract the features and positional locators. The second ANN was used to combine the output of the first ANN and CNN to get the final model outcome. Their results showed that deep CNN-based approaches improved the path loss predictions for the unseen areas by 1db and 4.7 db for 811 MHz and 2,630 MHz frequency bands, respectively. The satellite image-based path loss prediction impact is also seen with improved results. Two HMM-based protocols have been proposed by Tarek et al. [230]. For the protocols, the radio data is classified into two groups, and each group uses a different HMM model to predict the availability of channels in cognitive radios. Both datasets were evaluated with an ML model based on Bayes theorem in the first protocol. In the second proposed protocol, the authors used SVM to model the availability of the channels. Their results show that both protocols provide better performance than classical HMM techniques. SVM-based protocols provide improved performance due to the division of radio data into groups.

7.5 Smart Link Adaptation Optimization

The interference and noise level in serving cells affect the received signal quality at UEs. Using ML techniques, smart link adaptation optimization will be used to maximize the throughput without compromising the reliability of communication channels and control the transmission parameters and **Modulation and Coding Schemes (MCS)** [231, 232]. The MCS determines the transmit-block-size per stream, MIMO transmission rank, and precoding used to cope with the dynamics of the channel state. The traditional link adaptation optimization approach uses the **Channel Quality Indicator (CQI)** feedback from the UEs in which the CQI is matched against the link quality metrics to determine a suitable MCS. In addition, OLLA (the out-loop link adaptation) further improves the performance based on the link feedback. However, the CQI measure is considered outdated, as it does not capture the inter-sub carrier, multi-user/multi-layer interference in the actual downlink transmission. The preceding limitation results in a mismatch between the CQI feedback and the actual CQI for the downlink data.

However, measuring the link quality with reliable accuracy is not easy. Finally, we noted that for the FIN, the LA would be more complex due to a higher number of antennas and the number of channels that result in higher CSI dimensions. Moreover, it will be complicated to determine the suitable mapping tables for link qualities and LA parameters. OLLA is perfect for full buffer services, but it is hard to converge with a small burst and fast variation channel conditions. All issues lead to performance degradation. Therefore, the ML-based autonomous link adaptation optimization schemes are required to improve the performance more simply. It can use the historical channel condition data and corresponding KPIs to find optimized MCS and rank.

7.6 Section Summary

This section discussed various requirements of FIN related to signaling management and analyzed various existing ML schemes from the literature. We also identified input data space and action space that ML models use for optimization and decision making. Several areas related to signaling management, such as SLO, remote management, and cross-domain unified signaling management, require further studies. Furthermore, GL, cross-domain learning, and KS must be explored in multi-operator applications. Moreover, most existing solutions follow SAs and provide only shallow learning, whereas the FIN requires continuous learning from the real-time signaling data. The KS support for intelligent signaling is vital for enabling FIN to share the network state with user devices and other network operators.

8 USER PLANE OPTIMIZATION

The **User Plane Optimization (UPO)** requirement is related to ML applications specific for network functions that forward actual user data and is also referred to as the data plane. The user plane carries high-speed data or voice/video traffic to/from an external data network via the user plane function to the user terminal. This section discusses the FIN requirements in ML, the input data space, and action space to automate or optimize the user plane.

8.1 Data-Driven Architecture for ML on Edge (Dynamic Data Exchange)

Due to the higher level of complexity of the FIN infrastructure, it would be necessary to simplify the network management and orchestration with ML-based sophisticated data-driven paradigms. Furthermore, the ML techniques need to extract valuable insights from the operational data to automate provisioning workflows and enable data-driven SON capability to reduce cost and expedite service deployments. The insights extracted with ML will also be used for optimal load balancing, smart cluster formation, simplified bearer configuration, efficient and reliable sleep periods, and automated scaling of radio resources. In addition, the correctly predicted forecast for cell users will be used to optimize the network's performance.

In the literature, most of the ML applications on edge are related to content caching, and RL, ELM, and ESN learning models have been used with different learning depths. ML techniques with SHL depth were evaluated [233, 234]. Tamoor-ul-Hassan et al. [233] assessed an RL-based joint utility and strategy estimate for updating cache contents based on spatio-temporal features in a decentralized architecture of edge caching. The BS optimized the probability distribution of caching of different content classes with immediate utility. The optimal location of the cache was selected based on the weighted probability of BS and cloud and stability of the local and global popularity. He et al. [234] conducted a study on an alternative scheme based on QL in which BSs can communicate with each other to retrieve the missing contents from other BSs to minimize the

Table 12. Analysis of ML Models for Data-Driven Edge from the Literature

Ref.	Objective	ML				GL/DHL	CL	CAD	KS
		Model	Mode	Arch.	Depth				
[233]	Joint utility and strategy estimation	QL	RL	SA	S	No	Yes	No	No
[234]	Edge traffic offloading	QL	RL	DA	S	No	No	No	No
[235]	Edge resource management	DQL	RL	SA	S	No	No	No	No
[236]	Edge cache performance optimization	Wolpertinger	RL	SA	D	No	No	No	No
[237]	Dynamic caching scheme	ELM	SML	SA	S	No	No	No	No
[238]	Joint optimization of content location	ESN	SML	SA	S	No	Yes	No	No
[239]	Proactive remote radio head optimization	ESN	SML	SA	S	No	No	No	No

cost. The content location represents the state, and the action space changes the locations across the BSs. The results showed better performance as compared to other approaches.

DL ML models were evaluated as well [235–237]. The DRL-based scheme proposed by Zhong et al. [235] for vehicular networks addressed the issues in computing and network orchestration and resource management for edge caching. In the scheme, vehicle-to-BS assignment is decided based on the availability of the requested content in the respective BS. The results showed significant improvement in performance as compared to other solutions. In the work of Wang et al. [236], the Wolpertinger DRL approach was adopted to study the aspects of the cache update performance for the individual BS. The input state space consists of the concurrent request frequency and user request for content, and the action space includes the decision of whether to cache the content or not. The authors compared the scheme with several others and found improvements in the hit rates of the short- and long-term cache. A different approach adopted by Tanzil et al. [237] was based on ELM, where the focus was on evaluating a dynamic caching scheme based on content popularity. **Mixed Integer Linear Programming (MILP)** was used to determine the location of the cached content. The results of their research showed improvements in the QoE and the performance of the edge network.

Chen et al. [238] used the ESN model for joint optimization of the content location and user association with user locations to minimize the transmit power within the constraints of the QoE. In addition, the ESN was used to learn the content request distribution with minimal input data for training. The input to the ESN is user context, and the output is the request probability distribution. Another scheme using the ESN was studied by Chen et al. [239] for the optimization of remote radio head and baseband unit caches in CRAN. Moreover, a 3D-CNN technique was used by Doan et al. [240] for proactive caching, and traffic offloading at the edge was also investigated. Finally, the 3D-CNN was used for feature extraction, and SVM has been employed to generate the content vectors. Although the models mentioned previously have reported various improvements in different aspects of data-driven edge, the analysis given in Table 12 shows that holistic learning and KS with other domains and layers of the networks are still to be explored.

8.2 Network-Assisted Smart Services

The FIN aims to address several problems from real-life domains, such as smart road traffic monitoring, smart parking, automated driving, and smart factories [241] through network-assisted smart services. The ML techniques, along with network-based surveillance, will be used to provide vehicle, pedestrian, and blind spot detection, traffic signal violation, and RCA of accidents on highways. The major issues to be addressed are to ensure a faster response time, reliability, accuracy, and localization of the objects by using ML techniques.

Table 13. Analysis of ML Models for Network-Assisted Services from the Literature

Ref.	Objective	ML				GL/DHL	CL	CAD	KS
		Model	Mode	Arch.	Depth				
[242]	Object detection and counting at the edge: VIGIL	CNN	SL	SA	S	No	No	No	No
[243]	Distributed object detection and counting at the edge: VideoEdge	CNN	SL	SA	S	No	No	No	No
[244]	Amazon DeepLens integration with edge	CNN	SL	SA	S	No	No	No	No
[194]	Object detection with ensuring privacy of citizens	CNN	SL	SA	S	No	No	No	No
[245]	Urban road traffic detections bi-LSTM	CNN	SL	SA	S	No	No	No	No
[246]	Crises and disaster warning system	EL+MLP	TL	SA	S	No	Yes	Yes	No
[247]	Emergency situation management using UAVs	CNN	SL	SA	S	No	No	No	No
[248]	Smoke and fire detection	ML	SL	SA	S	No	No	No	No

A computer vision system known as Vigil [242] was proposed for MECC networks that used several network-enabled video surveillance cameras to offload computing at edge nodes. The edge nodes selected the video frames intelligently to detect objects and their enumeration, such as vehicles, humans, or various products. Like Vigil, VideoEdge [243] is also an MECC-based video system that scales well to large applications scenarios such as streets and roads. In contrast to Vigil, it uses hierarchical computing architecture, where the partial computations are offloaded to edge nodes and results are aggregated and stored on the cloud nodes. VideoEdge is compatible with Amazon DeepLens [244] devices that perform image detection locally and store results on the cloud storage systems to resolve latency issues and reduce high-density videos' traffic load over CN TN links. Another interesting project was proposed by Barthelemy et al. [194] to study edge computing for video processing with deep ANN to ensure user privacy while allowing real-time multimodal transmission. The results showed significant benefits of the proposed scheme in real work applications.

A combination of messages from the social media and weather forecast datasets has been used to predict urban road traffic in the work of Essien et al. [245], who employed a deep bidirectional LSTM based sparse autoencoder model for multi-step road traffic predictions. The model was trained data of Twitter messages and the actual traffic weather forecast dataset. The experimental evaluation and results showed improvements in effectiveness and accuracy of predictions, computational cost efficiency, and reduced environmental harm. In the work of Saeed et al. [246], an ML strategy based on EL and MLP classifiers was presented to understand the warning system better to address crises and disasters. Such a system is suitable for implementation in current configurations such as surveillance systems or closed-circuit television systems and has been tested in several trials, and the findings suggest that it detects accidents, disasters, and other crises more quickly. In the literature, vision-based CNN has been extensively utilized for surveillance, particularly for human-based emergencies such as early detection of fire and smoke. Another lightweight CNN system, presented in the work of Vinayakumar et al. [247], employed UAVs to monitor and categorize emergency and catastrophe circumstances. The emergency response primarily used an aerial picture database to improve the performance three times with only a 2% loss in accuracy compared to other standard techniques. A similar technique for video surveillance was proposed by Kyrkou and Theocharides [248] to detect smoke and fire events. It made use of two neural networks where one of the ANN used AdaBoost and numerous MLP layers to collaborate to build a hybrid model that predicted fire events more effectively using information from different gas, heat, and smoke sensors. A second CNN model was employed with the preceding model's output to identify the fire incident promptly. An analysis of various ML models for network-assisted services is shown in Table 13.

Table 14. Traffic Forecasting and Classifications: Analysis of ML Models from the Literature

Ref.	Description	ML				GL/DHL	CL	CAD	KS
		Model	Mode	Arch.	Depth				
[249]	Traffic forecasting using cross-domain big data	TL	TL	DA	S	No	Yes	No	No
[250]	STCNN: Traffic forecasting	CNN	SML	SA	S	No	No	No	No
[251]	DeepCog: Traffic forecasting	AE	SML	SA	D	No	No	No	No
[247]	IRNN: Traffic forecasting	RNN	SML	SA	S	No	No	No	No
[252]	Traffic forecasting for WiMAX	ANN	SML	SA	S	No	No	No	No
[253]	Traffic forecasting for LTE networks	Several[1]	SML	SA	S	No	No	No	No
[254]	Peer-to-peer traffic classification	VOT	EL	DA	S	Yes	No	No	No
		STK	EL	DA	S	Yes	No	No	No
[255]	Traffic classification using statistical data	SVM	SML	SA	S	No	No	No	No
[256]	IP traffic classification	Several[2]	SML	SA	S	No	No	No	No
[257]	Continuous traffic classification	Several[3]	SML	SA	S	No	No	No	No
[258]	Resource-aware traffic classification	CNN	SML	SA	S	No	No	No	No
[259]	Flow level traffic classification	BN	SML	SA	S	No	No	No	No
[260]	Optimization of traffic classification	DT	SML	SA	S	No	No	No	No
[261]	vTC: Traffic classification for VNFs	Several[4]	SML	SA	S	No	No	No	No
[262]	Traffic classification	DAB	TL	DA	S	No	Yes	No	No

[1] ANN, LSTM, SVR, and RF.
[2] MLP, RBF, C4.5, BN, and NB.
[3] NB, C4.5, and DT.
[4] KNN, SVM, DT/ET, RF, AB/GAB, NB, and MLP.

8.3 Long-Term Traffic Forecasting

The FIN is expected to deal with massive traffic volumes, and stringent performance requirements of services and ML techniques need to provide intelligent traffic engineering and demand-aware resource management based on reliable and accurate predictions of traffic forecast. Traditional traffic forecasting using various probe equipment installed network-wide generates a massive amount of data, and statistical techniques for traffic forecasting such as ARIMA and exponential smoothing cannot capture the spatial user-movement relationships from the data. The prediction of future traffic is generally garnered from time series data representing the network's usage. Several ML models for short-term forecast and **Long-Term Forecast (LTF)** proposed in recent years exploit the spatio-temporal traffic classifications from historical traffic data. An analysis of schemes in the context of this article is shown in Table 14, and the following paragraphs discuss their brief details.

The RNN models have the specific capability to detect temporal patterns from the time series data. In this regard, an LSTM-based traffic model is proposed in the work of Zhang et al. [249] and He et al. [250] that predicts traffic forecasts daily to long-term periods. In contrast, CNN-based models can learn the spatial features while considering the oversubscription policies for the resources. In addition, they can detect SLA violations and spatio-temporal dependencies for the LTF [251].

DL has exceptional feature extraction capabilities. The EC-based DL scheme called *DeepCog* proposed by Bega et al. [251] extracted the spatial and temporal features from the usage data of the network slices. Another scheme based on TL was proposed by Zhang et al. [249] for traffic forecast prediction for weekdays using CAD data. The base classifiers used were CNN and RNN for spatial and temporal features detection. Moreover, Vinayakumar et al. [247] evaluated the performance of classic RNN using the rectified linear unit and recurrent updates to an identity matrix, and a comparison of the results was made with **Gated Recurrent Unit (GRU)** and LSTM based models. The experimental evaluations showed that the LSTM-based approach performed better on real-time network traffic than classical RNN. The GRU and RNN with a rectified linear unit and identity matrix provided similar performance results with LSTM; however, the GRU incurred lesser computation cost. Using ANN and genetic algorithms, Railean et al. [252] proposed a traffic forecasting scheme based on stationary wavelet transfer for WiMAX traffic. Several ANN configurations, such as forecasts for similar days and all days, have been evaluated in this work to achieve better results as compared to traditional ANN-based schemes. Gijón et al. [253] investigated several ML approaches, and the findings for LTE data suggested that SL methods yield more accurate LTFs. They considered the RF, ANN-MLP, ANN-LSTM, and SVM models. Typically, the MLP provides the best accuracy results for 3 to 12 months of traffic forecasts. Cui et al. [263] used a convolution LSTM model to evaluate traffic prediction, which captured the spatio-temporal features of the traffic and predicted traffic of complex slice services in the vehicular networks. Based on this prediction, resources are allocated dynamically to the slice instances.

8.4 Smart Traffic Classification

Traffic categorization will be challenging because of the rising tendency of encrypted traffic on networks, and **Smart Traffic Classification (STC)** methods are required in the FIN [264]. To enable efficient management of network resources suited to different types of traffic, efficient and accurate traffic categorization algorithms have significant importance. ML-based traffic classification aims at classifying large amounts of network traffic in a real-time manner to overcome the limitations of classical approaches, such as port-based methods and deep packet inspection. Moreover, ML techniques must treat various services following QoS or QoE criteria and offer important profiling data to operators for purposes like customized advertising.

Accordingly, several ML-based traffic classification techniques from different perspectives have been studied in the literature. For example, Reddy and Hota [254] investigated stacking and voting-based EL with several base classifiers such as NB, BN, and DT to categorize peer-to-peer traffic. In this study, the authors concluded that stacking-based EL provided better results in terms of accuracy with the selected base classifiers. The advantage of EL-based techniques is that they are suitable for use in a DA and ML pipelining. Thus, they provide an opportunity to fulfill GL, CAD, CL, and KS requirements. In the context of EL models, stacking models provide better classification accuracy than the voting-based models; however, their ability to share the classification information across other layers and domains (via KS methods) is yet to be explored.

Generally, traffic classification information is required as soon as there is a need to apply specific policies. Therefore, the earlier the traffic classification information is available, the better the performance. In this regard, Sena and Belzarena [255] presented an early traffic classification approach based on SVM that does not require the complete packet to be received but can use the packet header information during the transmission of packets. In the scheme, weighted voting was used to improve the performance compared to the centroid-based classification. Another general requirement for traffic classification algorithms is that the complete packet inspection should be avoided whether online or offline classification methods are used. The partial analysis of the packets allows for early traffic classification information and reduces the computation overhead. Considering these aspects, Singh and Agrawal [256] studied a C4.5 classifier that provided accuracy up to 88%. The experiments also evaluated the MLP, RBF, BN, and NB. However, the performance was much lower as compared to the C4.5 classifier. The AB models can classify the traffic based on flows rather than per-packet classification. Typically, it detects the flows or transactions in the captured traffic and associates the related packets to the flow as per flow definition [262]. With

AB-based traffic flow classification, the accuracy was achieved between 93% and 98.1%, better than the C4.5 classifier.

Real-time traffic classification is often challenging in high-speed networks. It is further complicated by the perseverance of privacy of the users generating traffic. However, ML models have an advantage over classical classification approaches, as they only require the meta-information of the traffic flows instead of looking at the entire packet contents. In this regard, the vTC scheme proposed by He et al. [261] allowed the VNF to select a suitable classification scheme at runtime and using partial packet inspection. The experimental results show that the proposed NFV for flow classification can improve classification accuracy by up to 13%. The proposed scheme consisted of a controller and a pool of virtual machines providing classification capability based on different ML approaches. With the help of vTC, every VNF was allowed to select the specific classification model based on the desired characteristics.

An analysis of various ML-based models used for traffic classification proposed in the recent literature for the traffic classification is shown in Table 14. It can be observed that there are very few studies that can fulfill the FIN requirements, such as GL and CL; however, the aspects of CAD and KS require further studies.

8.5 Section Summary

This section has presented several requirements for optimizing the user or data plane. The data-driven edge can significantly improve data plane performance by offering reduced latencies and increased throughput. In addition, ML techniques can provide necessary insights to make various decisions such as content and computer locations and their usage. Moreover, data plane optimization requires sophisticated traffic forecasting and classification methods. Finally, the information must be shared across the administrative domains and network layers to make efficient decisions; thus, CAD, CL, GL, and KS are vital. Although several schemes in the literature use TL and EL, the aspects of GL and KS have not been studied.

9 INTELLIGENT APPLICATION OPTIMIZATION

Intelligent Application Optimization (IAO) allows sharing knowledge between applications and networks [5]. The NS instances are required to intelligently self-optimize various network-related parameters and share certain information with applications such as the effects of **Transport Control Protocol (TCP)** behavior, data retention, and storage. However, due to significant differences in the timescale of radio network fluctuations and TCP data rate adjustments, there is a misalignment between RAN and applications. It results in false network overloading, undesired congestion control actions, non-effective utilization of radio resources, and degraded user experience. In the literature, although TCP window optimization, congestion control, and throughput prediction have been studied with different ML models, correlating the underlying TCP performance to various resources or functions of the network is yet to be explored.

The commonly used ML models for IAO in the literature are QL and RL [265–267], AB [268], and SVM [269]. In the work of Li et al. [265], the QL model was used for TCP window optimization by assessing the TCP window size fluctuations and sharing information between the network and applications on a real-time basis. The shared information considered in their scheme was the BS buffer size, the load, link throughput, the packet error rate, and window size. ML-assisted TCP window optimization is used to inform the applications about the status of radio-air interface in real time, and applications adjust their transmission data rate accordingly. Specifically, the TCP window size can be optimized to match better the radio channel variations based on information provided by the access nodes. Access nodes' information is the buffer size, the load on a BS, the link throughput, and packet error rate. In the scheme proposed by Kong et al. [266], RL and loss predictor models were used for congestion control on the TCP layer. Both of the schemes were evaluated with extensive simulations, and results showed that both manage the trade-off between latency and throughput in simulated network conditions as compared to the New-Reno and QL-based approaches. Another prominent scheme called *TCP-Drinc* was proposed by Xiao et al. [267], where a DRL model was used for congestion control on the TCP layer. It used model-free learning and managed the congestion control decisions smartly by learning suitable parameters for optimizing the congestion control parameters, such as the window size, based on the previous observations. The authors reported improved performance as compared to five benchmark schemes.

Similarly, Li et al. [268] investigated AB-based TCP congestion management that used an adaptive model to effectively identify the boost level and classify the type of packet loss in TCP sessions. A combination of explicit congestion notification and loss type detected at the receiver is sent back to the sender, where the congestion control parameters are adjusted accordingly. Extensive simulations were carried out to show higher classification speed with lower response time and use fewer network resources with AdaBoost. In comparison to Hybla and Westwood and the cubic congestion control schemes, AB gained 10% higher throughput. Similar results were reported in comparison to the New-Reno congestion control scheme. Another interesting study was conducted by Mirza et al. [269] on TCP throughput prediction. The SVM regression model was used to predict throughput based on analysis of historical TCP sessions and link state data. The results of extensive simulations with the preceding scheme showed significant improvements in the throughput prediction.

Another requirement for FIN under the category of IAO is dynamic data retention and storage. Several use cases in the FIN will require massive data collection in real time and processing on the network nodes. A large amount of data may overwhelm the limited storage capacity on the real-time processing nodes, and frequent memory clearing will be required. The traditional data cleanup and retention approaches are unsuitable for FIN, as most use cases require historical data. Future decisions are made on relationships in the dataset elements. Due to the huge number of fixed rules, intelligence must be developed based on qualities connected with the data. Dynamic retention rules must be developed based on past learning, storage, and context. Data across different network nodes, particularly at the edge, require dynamic retention and storage solutions.

10 INTELLIGENT SECURITY OPTIMIZATIONS

ISO is another requirement for FIN [270], and it is concerned with managing the security and privacy of users and network operators in an automated and intelligent way. The primary objective of combating counterfeit information and devices is to identify cloned **International Mobile Equipment Identity (IMEI)**. The ML techniques are needed for both RAN and the CN and their layers. ML models can optimize security by using several device characteristics like antennas and MIMO in RANs. In the CNs, ML models can use attributes such as the IMEI and device model, operating system, and manufacturer. For example, ML models can identify the cloned IMEI utilizing this information and deploy it in the edge network nodes to prevent such mobiles from accessing the network.

The use of **Subscriber Identity Module (SIM)** boxes as an illegal exchange is a serious security issue identified for FIN. It causes heavy revenue losses to the government, service providers, and problems concerning law enforcement. Therefore, it is necessary to have an efficient and accurate method to identify such traffic to identify and stop such communications effectively. Traffic data is collected due to the ML-based detection of unlawful SIM box exchanges. First, it categorizes

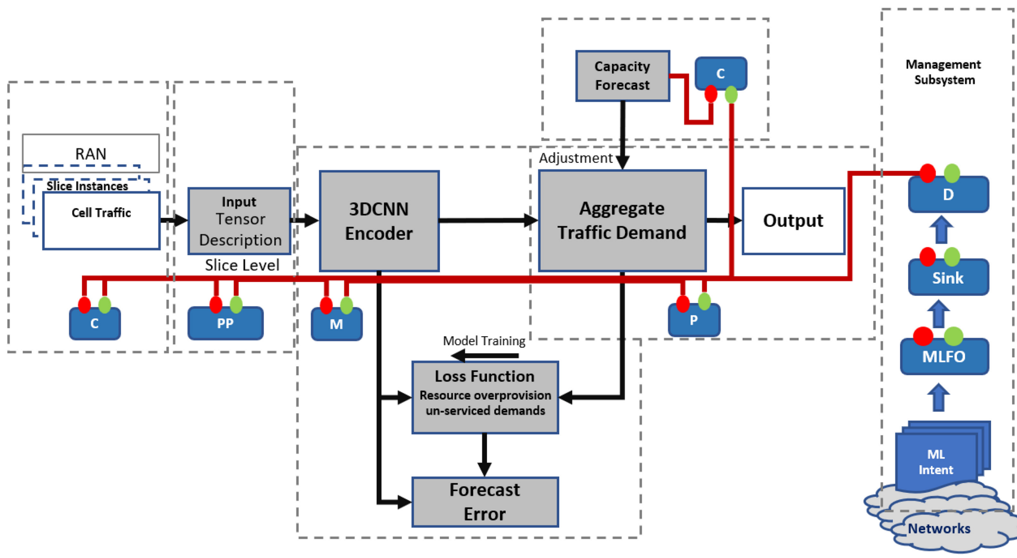


Fig. 11. 3DCNN model for LTF.

the data as an unlawful SIM exchange. Then it sends the data to the relevant node for further processing. It assures the prevention of such traffic from passing across the network.

11 PIPELINING THE ML SCHEMES

This section discusses a few examples of pipelining the existing ML models proposed in the literature and their issues. Then we present the overall view of the MLPL for the FIN that can fulfill the various requirements. The ITU-T introduced the MLPL for managing the MLOps for the network, and we discussed this briefly in Section 4. Several frameworks are used in the industry to automate initiatives, like AutoML [109] MLOps [108]. The automation of MLOps offers several advantages in terms of flexibility, scalability, and extensibility of learning life cycles and provides timely analysis and assessment, real-time predictions, and eases the transformation to ML-enabled processes. According to Microsoft, the expectations from MLPL are automation of heterogeneous resource operations, reusability, tracking, versioning, modularity, and collaboration [271]. Therefore, ML models for the FIN must include pipeline architecture to provide accurate and efficient decision making and optimization.

We will consider the cases of traffic forecasting and traffic classification schemes as discussed in Section 8. The DL-based traffic system [251] was discussed previously. It uses CNN to learn the spatial features, and MLP is used to learn temporal features. This scheme is shown in Figure 11 with possible MLPL integration. The input data is collected by collector nodes, where the source nodes can be placed on different network parts depending on the objective function. M nodes implement the 3DCNN that provides valuable traffic insights used by policy nodes to determine the appropriate policies. The traffic classification technique proposed by He et al. [261] is shown with MLPL architecture in Figure 12. First, the user and signaling traffic collected by node C from source nodes are processed at the preprocessing node for filtering and adding slice context. Then, the set of models can be executed on M nodes that provide the feature set used by policy nodes to determine suitable policies and actions to respond to the traffic forecast changes. Finally, the policies are executed on the sink nodes identified by the distributor node. Our critical observation is that the MLPL is activated with an ML intent without explicit continuous learning specification

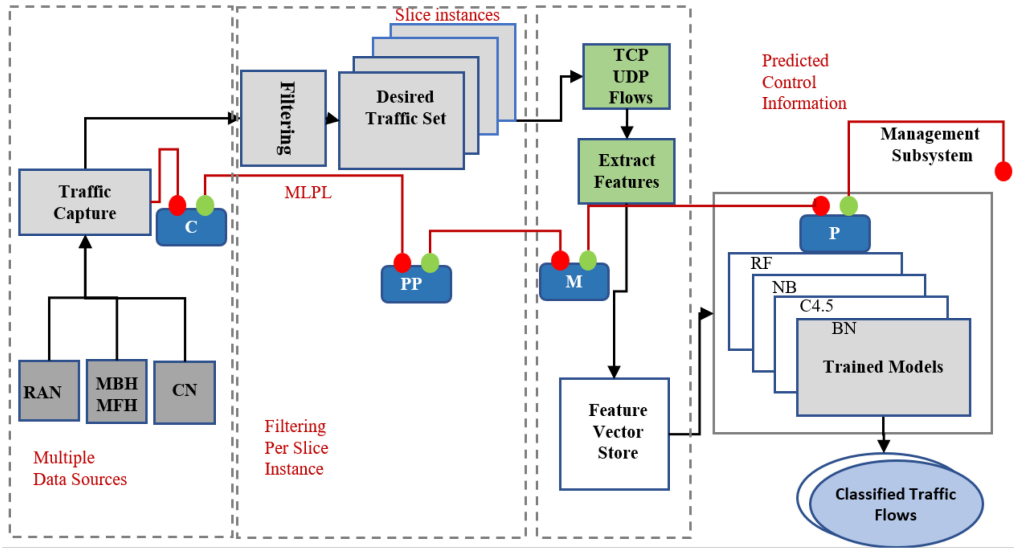


Fig. 12. Traffic classification model.

[13]. The MLPL also specifies the lifecycle management for ML models, procedures to treat different characteristics of ML models, monitoring model performance, triggering retraining, and transferring models. Another issue is that although ML model lifecycle management does not specify how to correlate the knowledge learned from different layers, CAD and GL/DHL are not specified. Furthermore, it does not specify how the agility of ML models can be ensured in the wake of the rapid advancements in ML techniques.

Considering the dynamic conditions and rapid state changes in heterogeneous environments, we view the pipeline architecture most supporting the CAD, GL/DHL, CL, and KS, as shown in Figure 13. It will facilitate the support for multi-vendor systems where the learning approaches could be different, but some standardizable feature sets represent the state of network functions and operation states. The bottom layer consists of network infrastructure. Most of the network's functions and resources are divided into virtual instances using state-of-the-art HNS to cater to different requirements. The second layer from the bottom to top implements a deep distributed learning paradigm where resources and functions are learned and localized actions are executed on the nodes.

The localized actions do not affect or are not related to other domains or layers of the networks. The global action space consists of those actions that cannot be executed in a standalone mode and requires end-to-end level coordination and KS. These actions are selected using the CAD, GL, and CL layers and populate shareable network state information. The green lines in Figure 13 represent the output of the local learners, which is input to the GL and DHL learners. Both GL and DHL make optimal decisions by considering the feature sets from other layers and domains, as discussed in Sections 5 through 10. The potential challenges with such an approach are the absence of standardized techniques for KS with GL and DHL learners and security concerns from the exposure of the information. The red color represents the action lines, which is the output from the GL and DHL learners. It specifies an action to be executed along with the identifier of the network-specific layer and target function or resource in it. This approach helps execute the primary optimization actions on a specific function or resource and the secondary actions corresponding to the primary actions. Since the GL and DHL also interface with other domains,

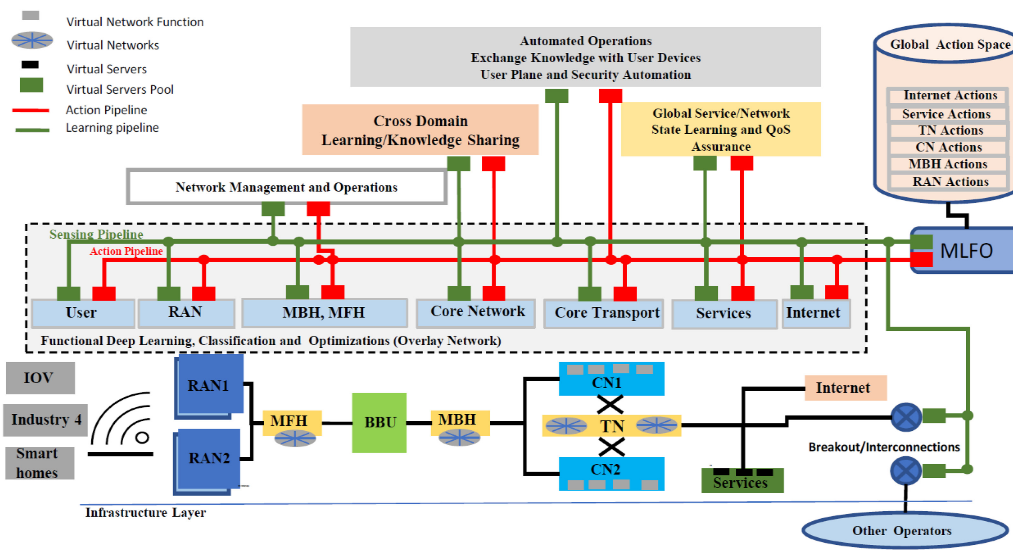


Fig. 13. ML architecture with GL/DHL, CAD, CL, and KS for FIN.

uncontrolled information exposure over the green lines could expose confidential information. Therefore, the operators must decide what information can be shared across the network layers and other domains based on their security risk analysis, policies, and potential benefits.

The potential benefits of the architecture mentioned previously are an opportunity for expedited network optimizations and reduced computational costs. In addition, several related feature sets learned from one layer or domain can also be used to make decisions in other domains. It eliminates the redundant learning and related computation costs, which are common drawbacks of the localized learning approaches.

12 OPEN ISSUES, CHALLENGES, AND FUTURE RESEARCH DIRECTIONS

In Sections 5 through 10, we discussed various requirements for the FIN and analyzed several existing ML techniques from recent literature. Section 11 proposed a high-level architecture to implement ML for the FIN. Finally, this section discusses some challenges in fulfilling the requirements mentioned previously.

The first most crucial issue is the availability of reliable and suitable ML models for all of the network functions and operations. An analysis of the existing ML models for different requirements of the FIN is shown in Figure 14. The color rank of the cells in the figure has been determined based on the number of schemes available in the literature and their evaluation of the support of GL, CL, CAD, KS, the level of learning, and DA. The green cells represent specific FIN requirements in different network segments, and the depth of the color shows the rank of the respective area. The blue cells represent the current state of the taxonomy brach, and the yellow color represents the state of the network segment. The color depth is representative of the rank of the respective FIN requirement category and network segment. Accordingly, in recent literature, the issues of intelligent networks have been widely studied with some level of focus on GL, CAD, CL, and KS. However, not all required categories have been studied for end-to-end networks, such as data-driven edge, traffic classification, forecasting, application optimization, and fault management. Similarly, from the point of view of requirement categories, UPO has been widely studied but lacks the aspects of the end-to-end networks and RAN.

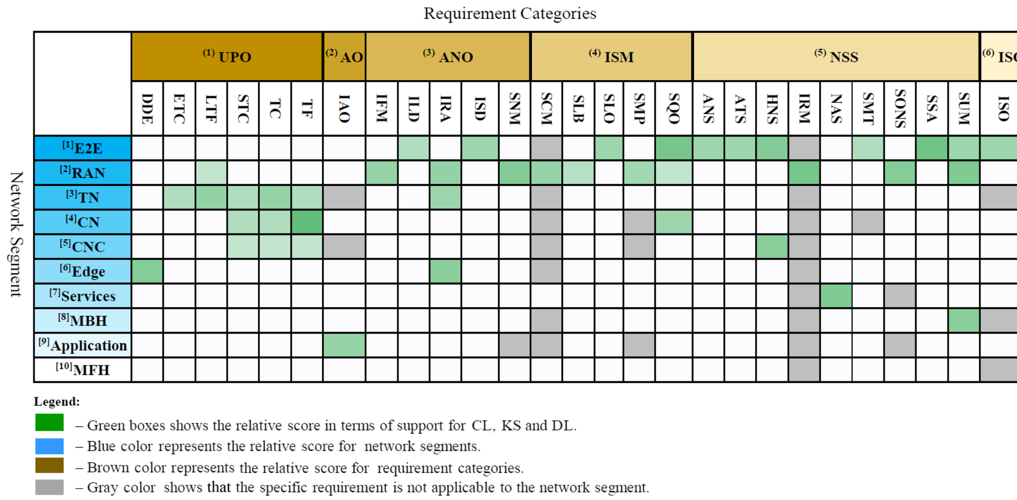


Fig. 14. Overall summary of ML techniques for 6G and beyond.

MLP-based models are the simplest and can be implemented for SL, UL, and reinforcement scenarios from the ML point of view. Still, their performance is acceptable and has a very slow convergence during learning [272]. Several MLP-based models are used for frequency selection, operational radio parameters, and configuration optimization [130], VNF classification [256], and IP traffic classification [257], which we discussed in the context of MLPL architecture.

The restricted Boltzmann machine models can robustly represent the data in an unsupervised mode, but its training is complex [273]. We also discussed traffic classification models [257] and VNF classification techniques [256]. However, the AE provides UL and can represent sparse and compact problems. Although it is challenging to pre-train with large data, it is one of the most powerful and successful UL methods [274]. Furthermore, sparse autoencoder based models have been used for video processing on the edge [245].

CNN-based models can work in an SL, UL, or RL mode. Moreover, they are well suited for spatial data modeling since they support weight sharing and affine invariance. However, the computation cost is high, and it is challenging to find the optimal values of hyperparameters and requires deep structures for complex tasks [275]. Nevertheless, CNN has been extensively used for UPOs such as LTTF and STC [249–251, 254, 260], **Dynamic Data Exchange (DDE)** optimization [276], and smart network services [194, 242–245].

Furthermore, the RNN can also provide SL, UL, and RL and address the issues related to sequential data or data streams. The RNN is very efficient in learning the temporal dependencies but faces high complexity problems, gradient vanishing, and explosion problems [277]. The GAN models provide UL to generate samples and represent actual life scenarios, but the training process is unstable and has complex convergence [278].

DRL is a model for high dimensionality but suffers from slow convergence [279]. It has been used for resource orchestration optimization [154, 155] and QoE optimization [200], DDE optimization [233], and congestion control [267]. EL methods are inherently suitable for distributed learning models, as they allow a combination of different ML models' results. These methods improve the robustness and performance [280] and have been used in RCA [189]. TL methods are intrinsically suitable for solving heterogeneous problems. They enable the training process to be completed on certain specific scenarios and the information to be applied to address the difference of related problems [281], content popularity estimation [233], and traffic correlations [282]. Except for the

EL- and TL-based models with the intrinsic capability for DA, other models requiring a centralized implementation need certain suitable techniques for MLPL support. HNS, IFM, LTTF, and STC are requirements that have been heavily focused on by researchers in the recent decade, whereas ANS, ISD, ILD, IRA, SQO, SHNM, CoKPI, SRDD, and SIED are the requirements that require further studies for different network segments.

Smart and intelligent behavior in service design, deployment, operation, and proactively responding to diverse situations are critical aspects for 6G and beyond. Network resources must be managed intelligently in functionality, links, computing, and storage. These approaches must consider user characteristics, resource status, and circumstances, and capacity and knowledge learned must include both normal and abnormal events in which a partial network failure may occur.

All intelligent capabilities in various network sections such as access, the core, transport, and data center must operate together to offer holistic provisioning of end-to-end network slice instances. Most ML-based techniques give a localized or a standalone solution to a particular network function or operation. The primary step toward smart automation starts with the product definitions and their translations to slice specific network configurations. Then, orchestration platforms use these configurations to instantiate the slice instance. Finally, the configurations need to be optimized in different steps based on the network state, operator, regulatory policies, optimal localization of the functions, and connectivity between them. The physical designs of networks are often established during network planning and development, and the logical designs are often directly associated with the service design. Hence, the NS design and deployment will be mostly focused on logical design. We also noted that CAD, CL, GL, and KS are critical to realizing the FIN to enable end-to-end optimal decisions and optimizations across the network segments and multiple operators and user devices.

12.1 Challenges and Future Research Directions

We have observed through the review of literature in this article that the successful implementation of the FIN requires that all network functions, resources, management, and operations have intelligent capabilities to learn, predict, and decide actions in an automated way on a timely basis. A significant challenge requiring further studies for the FIN is the availability of standardized frameworks for MLOps that can support GL/DHL, CL, CAD, and KS. According to datatonic, a Google AI partner, MLOps must support continuous integration, delivery, training, learning, and monitoring [283]. Furthermore, the large-scale, heterogeneous networking products and multi-vendor deployments require machine intelligence to interoperate and exhibit well-proven behavior. Therefore, besides the standardization of learning methods, the standardization of evaluation metrics for ML is also necessary [284]. Furthermore, traditional evaluation matrixes based on accuracy, precision, and F1 score, reliability, computational complexity, and agility of MLOps must be considered. Rule-based approaches may be applied in such cases, resulting in less optimal decisions.

The standardization efforts must address the heterogeneity of network functions and SLA. The network comprises many functions, and the virtualization technologies for various levels that allow their instantiation are likewise diverse. The autonomous instantiation and scaling of their capacities based on network conditions and user requirements are also disparate. Similarly, different SLAs will be required due to the different business models supported by NS. GL using the learned features from the distributed DL ML algorithms requires formal methods to represent the global state, network function, operation, and resource. Suitable ensemble techniques require further studies, and the impact of the selection of particular action needs to be identified. Moreover, the suitable methods for exchanging the learned features in different formats across the network layers, multiple operators, and user devices need to be investigated. Finally, the exchange of features must be without losing the security and privacy of users and network operators.

Specifically focused studies are required to investigate suitable methods of utilizing the information learned from one domain or network layer to other domains or layers to reduce computational costs and expedite automated operations. The QoS and QoE assurance algorithms must collaborate with various ML and network technologies to offer an end-to-end performance perspective [284]. Every single network slice needs an individual SLA. The ITU defined that the multi-service provider SLA framework in E.860 [285] shall apply to the FIN.

Another challenge is handling massive and heterogeneous data formats generated from network functions such as user mobility patterns, service usage, and logs. The ML algorithms must support heterogeneous data formats to ensure end-to-end decisions and optimizations for decision making out of insights from data. Furthermore, the information learned by the ML algorithms needs to be exchanged with operators involved in the end-to-end service delivery to ensure user-centric performance rather than network-centric performance. Operators are generally responsible for the network under their administrative domain; however, end-to-end services delivery is often not confined to a single operator. Overall service performance depends on the entire network involved. Therefore, ML techniques must be investigated to maintain and communicate service characteristics across various domains.

Energy efficiency [286] is another challenge for the FIN, as large-scale computations will affect the energy consumption at the CNs and reduce the life of user devices. Intelligence comes at the cost of computations and storage resources. The load balancing techniques are required to distribute the load across different network components. It will avoid sparsely loaded network components. Hence, it will be possible to free up less loaded components. The load from such nodes can be migrated to other places where optimal loads can be maintained. The freed-up resources like BSs can be turned off to save energy. ML will automate the network operations based on user requirements, network state, and operator policies such as oversubscription rules, security policies, and regulatory policies, which influence the way forecasted optimizations or actions are implemented. Operators require these policies to be enforced while the ML automated operations are executed to ensure compliance of services with economic business models. The intelligent charging of services and network usage is an important factor that needs to be investigated for FIN [287]. Business organizations invest in buying licensed spectrum, which requires them to charge the direct users of the services. In the FIN, indirect users will also be charged due to different business models, such as an ANS scenario.

The extensibility, modularity, evaluation metrics, and support for the DA of ML models are critical requirements for enabling agile MLOps and holistic and pervasive intelligence in FIN. However, it will not be easy to achieve automated and agile MLOps without these capabilities.

13 CONCLUSION

Future networks require intelligent and autonomous design, deployment, operations, and troubleshooting capabilities to meet diverse requirements and open new revenue streams for network operators. ML can efficiently cater to the requirements with continuous learning from user behaviors, network conditions, and operational data. This survey article has identified requirements for FIN, including SONS, intelligent operations, signaling and management, user plane management, applications coordination, and security. These requirements have been analyzed concerning the existing ML techniques and pipeline architecture to determine research gaps and future research directions. In addition, this survey has explored several existing schemes proposed in the literature for SONS. It has been concluded that existing techniques did not focus on the requirements of the FIN in terms of KS and CL and cross-domain learning. Moreover, there are significant areas such as service design, logical network design, and the autonomous translation of product definitions and business requirements that still need to be investigated for FIN. This survey has also analyzed existing ML-based schemes for network operations and management. It has been seen that most of the existing schemes are layer or domain specific and hence cannot fulfill the requirements for FIN. This survey also looked at the green applications for future networks where intelligent networking and coordination with the user devices are required; however, very few studies have been conducted in the literature. Finally, this study has also looked at optimization techniques for user plan and network security requirements of FIN. These areas require extensive efforts to correlate user applications and network KPIs. Furthermore, the heterogeneous backhaul resources must be shared and managed intelligently to ensure multi-tenancy, security, privacy, and resource guarantees. The ML models require further interoperability, reliability, and scalability investigations to extract holistic insights to realize agile processes in the slice service life cycle. The lack of ML algorithms standardization poses a critical challenge as various algorithms operate in diverse domains and network layers.

REFERENCES

[1] Changyang She, Peng Cheng, Ang Li, and Yonghui Li. 2021. Grand challenges in signal processing for communications. *Frontiers in Signal Processing* 1 (2021), Article 664331, 4 pages.

[2] Lin Zhang, Ying-Chang Liang, and Dusit Niyato. 2019. 6G visions: Mobile ultra-broadband, super Internet-of-Things, and artificial intelligence. *China Communications* 16 (2019), 8, 1–14.

[3] Zhengquan Zhang, Yue Xiao, Zheng Ma, Ming Xiao, Zhiguo Ding, Xianfu Lei, George K. Karagiannidis, and Pingzhi Fan. 2019. 6G wireless networks: Vision, requirements, architecture, and key technologies. *IEEE Vehicular Technology Magazine* 14 (2019), 28–41.

[4] Xiaohu You, Cheng-Xiang Wang, Jie Huang, Xiqi Gao, Zaichen Zhang, Mao Wang, Yongming Huang et al. 2021. Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts. *Science China Information Sciences* 64 (2021), Article 110301.

[5] Yiqing Zhou, Ling Liu, Lu Wang, Ning Hui, Xinyu Cui, Jie Wu, Yan Peng, Yanli Qi, and Chengwen Xing. 2020. Service-aware 6G: An intelligent and open network based on the convergence of communication, computing and caching. *Digital Communications and Networks* 6 (2020), 253–260.

[6] 3GPP. 2021. Release 15. Retrieved March 2, 2021 from https://www.3gpp.org/release-15.

[7] Shunliang Zhang. 2019. An overview of network slicing for 5G. *IEEE Wireless Communications* 26 (2019), 111–117.

[8] Ioannis Tomkos, Dimitrios Klonidis, Evangelos Pikasis, and Sergios Theodoridis. 2020. Toward the 6G network era: Opportunities and challenges. *IT Professional* 22 (2020), 34–38.

[9] José Marcos C. Brito, Luciano Leonel Mendes, and José Gustavo Sampaio Gontijo. 2020. Brazil 6G project—An approach to build a national-wise framework for 6G networks. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[10] Shuo Wang, Tao Sun, Hongwei Yang, Xiaodong Duan, and Lu Lu. 2020. 6G network: Towards a distributed and autonomous system. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[11] Jinkang Zhu, Ming Zhao, Sihai Zhang, and Wuyang Zhou. 2020. Exploring the road to 6G: ABC—Foundation for intelligent mobile networks. *China Communications* 17 (2020), 51–67.

[12] Benjamin Sliwa, Robert Falkenberg, and Christian Wietfeld. 2020. Towards cooperative data rate prediction for future mobile and vehicular 6G networks. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[13] Harish Viswanathan and Preben E. Mogensen. 2020. Communications in the 6G era. *IEEE Access* 8 (2020), 57063–57074.

[14] Volker Ziegler and Seppo Yrjola. 2020. 6G indicators of value and performance. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[15] Gustav Wikström, Janne Peisa, Patrik Rugeland, Nicklas Johansson, Stefan Parkvall, Maksym Girnyk, Gunnar Mildh, and Icaro Leonardo Da Silva. 2020. Challenges and technologies for 6G. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[16] Marco Giordani, Michele Polese, Marco Mezzavilla, Sundeep Rangan, and Michele Zorzi. 2020. Toward 6G networks: Use cases and technologies. *IEEE Communications Magazine* 58 (2020), 55–61.

[17] Khaled B. Letaief, Yuanming Shi, Jianmin Lu, and Jianhua Lu. 2022. Edge artificial intelligence for 6G: Vision, enabling technologies, and applications. *IEEE Journal on Selected Areas in Communications* 40, 1 (2022), 5–36.

[18] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang. 2021. Holistic network virtualization and pervasive network intelligence for 6G. *IEEE Communications Surveys & Tutorials* 24, 1 (2022), 1–30.

[19] Muhammad Waseem Akhtar, Syed Ali Hassan, Rizwan Ghaffar, Haejoon Jung, Sahil Garg, and M. Shamim Hossain. 2020. The shift to 6G communications: vision and requirements. *Human-centric Computing and Information Sciences* 10 (2020), Article 53.

[20] Mohammad Abu Alsheikh, Shaowei Lin, Dusit Niyato, and Hwee-Pink Tan. 2014. Machine learning in wireless sensor networks: Algorithms, strategies, and applications. *IEEE Communications Surveys & Tutorials* 16 (2014), 1996–2018.

[21] Xianfu Chen, Jinsong Wu, Yueming Cai, Honggang Zhang, and Tao Chen. 2015. Energy-efficiency oriented traffic offloading in wireless networks: A brief survey and a learning approach for heterogeneous cellular networks. *IEEE Journal on Selected Areas in Communications* 33 (2015), 627–640.

[22] Michele Zorzi, Andrea Zanella, Alberto Testolin, Michele De Filippo De Grazia, and Marco Zorzi. 2015. Cognition-based networks: A new perspective on network optimization using learning and distributed intelligence. *IEEE Access* 3 (2015), 1512–1530.

[23] Teodora Sandra Buda, Haytham Assem, Lei Xu, Danny Raz, Udi Margolin, Elisha Rosensweig, Diego R. Lopez, et al. 2016. Can machine learning aid in delivering new use cases and scenarios in 5G? In *Proceedings of the 2016 IEEE/IFIP Network Operations and Management Symposium (NOMS'16)*.

[24] Bharath Keshavamurthy and Mohammad Ashraf. 2016. Conceptual design of proactive SONs based on the big data framework for 5G cellular networks—A novel machine learning perspective facilitating a shift in the SON paradigm. In *Proceedings of the 2016 International Conference System Modeling Advancement in Research Trends (SMART'16)*.

[25] Mohammad Abu Alsheikh, Dusit Niyato, Shaowei Lin, Hwee-Pink Tan, and Zhu Han. 2016. Mobile big data analytics using deep learning and apache spark. *IEEE Network* 30 (2016), 22–29.

[26] Matias Richart, Javier Baliosian, Joan Serrat, and Juan-Luis Gorricho. 2016. Resource slicing in virtual wireless networks: A survey. *IEEE Transactions on Network and Service Management* 13 (2016), 462–476.

[27] Paulo Valente Klaine, Muhammad Ali Imran, Oluwakayode Onireti, and Richard Demo Souza. 2017. A survey of machine learning techniques applied to self-organizing cellular networks. *IEEE Communications Surveys & Tutorials* 19 (2017), 2392–2431.

[28] Chunxiao Jiang, Haijun Zhang, Yong Ren, Zhu Han, Kwang-Cheng Chen, and Lajos Hanzo. 2017. Machine learning paradigms for next-generation wireless networks. *IEEE Wireless Communications* 24 (2017), 98–105.

[29] Rongpeng Li, Zhifeng Zhao, Xuan Zhou, Guoru Ding, Yan Chen, Zhongyao Wang, and Honggang Zhang. 2017. Intelligent 5G: When cellular networks meet artificial intelligence. *IEEE Wireless Communications* 24 (2017), 175–183.

[30] Panagiotis Kasnesis, Charalampos Z. Patrikakis, and Iakovos S. Venieris. 2017. Changing mobile data analysis through deep learning. *IT Professional* 19 (2017), 17–23.

[31] Lidong Wang and Randy Jones. 2017. Big data analytics for network intrusion detection—A survey. *International Journal of Networks and Communications* 7, 1 (2017), 24–31.

[32] Nei Kato, Zubair Md Fadlullah, Bomin Mao, Fengxiao Tang, Osamu Akashi, Takeru Inoue, and Kimihiro Mizutani. 2017. The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective. *IEEE Wireless Communications* 24 (2017), 146–153.

[33] Zubair Md Fadlullah, Fengxiao Tang, Bomin Mao, Nei Kato, Osamu Akashi, Takeru Inoue, and Kimihiro Mizutani. 2017. State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems. *IEEE Communications Surveys & Tutorials* 19 (2017), 2432–2455.

[34] Xenofon Foukas, Georgios Patounas, Ahmed Elmokashfi, and Mahesh K. Marina. 2017. Network slicing in 5G: Survey and challenges. *IEEE Communications Magazine* 55 (2017), 94–100.

[35] Mehdi Mohammadi, Ala Al-Fuqaha, Sameh Sorour, and Mohsen Guizani. 2018. Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials* 20 (2018), 2923–2960.

[36] Alexandros Kaloxylos. 2018. A survey and an analysis of network slicing in 5G networks. *IEEE Communications Standards Magazine* 2, 60–65.

[37] Ibrahim Afolabi, Tarik Taleb, Konstantinos Samdanis, Adlen Ksentini, and Hannu Flinck. 2018. Network slicing and softwarization: A survey on principles, enabling technologies, and solutions. *IEEE Communications Surveys & Tutorials* 20 (2018), 2429–2453.

[38] Massimo Condoluci and Toktam Mahmoodi. 2018. Softwarization and virtualization in 5G mobile networks: Benefits, trends and challenges. *Computer Networks* 146 (2018), 65–84.

[39] Nguyen Cong Luong, Dinh Thai Hoang, Shimin Gong, Dusit Niyato, Ping Wang, Ying-Chang Liang, and Dong In Kim. 2019. Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys & Tutorials* 21 (2019), 3133–3174.

[40] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. 2019. Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials* 21 (2019), 2224–2287.

[41] Hongbo Wang, Jiaying Hou, and Na Chen. 2019. A survey of vehicle re-identification based on deep learning. *IEEE Access* 7 (2019), 172443–172469.

[42] Ruoyu Su, Dengyin Zhang, R. Venkatesan, Zijun Gong, Cheng Li, Fei Ding, Fan Jiang, and Ziyang Zhu. 2019. Resource allocation for network slicing in 5G telecommunication networks: A survey of principles and models. *IEEE Network* 33 (2019), 172–179.

[43] Marcos Toscano, Federico Grunwald, Matías Richart, Javier Baliosian, Eduardo Grampín, and Alberto Castro. 2019. Machine learning aided network slicing. In *Proceedings of the 2019 21st International Conference on Transparent Optical Networks (ICTON'19)*.

[44] Abdelquoddouss Laghrissi and Tarik Taleb. 2019. A survey on the placement of virtual resources and virtual network functions. *IEEE Communications Surveys & Tutorials* 21 (2019), 1409–1434.

[45] Bo Ma, Weisi Guo, and Jie Zhang. 2020. A survey of online data-driven proactive 5G network optimisation using machine learning. *IEEE Access* 8 (2020), 35606–35637.

[46] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines. 2020. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Computer Networks* 167 (2020), Article 106984.

[47] Barbara Kitchenham and Stuart Charters. 2007. Guidelines for Performing Systematic Literature Reviews in Software Engineering. Technical Report EBSE-2007-01. ESBE.

[48] Yi Zhang, Wee Peng Tay, Kwok Hung Li, Moez Esseghir, and Dominique Gaiti. 2016. Learning temporal–spatial spectrum reuse. *IEEE Transactions on Communications* 64 (2016), 3092–3103.

[49] Sergei Dotcenko, Andrei Vladyko, and Ivan Letenko. 2014. A fuzzy logic-based information security management for software-defined networks. In *Proceedings of the 16th International Conference on Advanced Communication Technology*.

[50] Leonard Barolli, Arjan Durresi, Fatos Xhafa, and Akio Koyama. 2008. A fuzzy-based handover system for wireless cellular networks: A case study for handover enforcement. In *Proceedings of the International Conference on Network-Based Information Systems (NBiS'08)*.

[51] Fozia Hanif Khan, Rehan Shams, Muhammad Aamir, Muhammad Waseem, and Mujtaba Memon. 2015. Intrusion detection in wireless networks using genetic algorithm. In *Proceedings of the 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom'15)*.

[52] Mojtaba Romoozi, Mehdi Vahidipour, Morteza Romoozi, and Sudeh Maghsoodi. 2010. Genetic algorithm for energy efficient and coverage-preserved positioning in wireless sensor networks. In *Proceedings of the 2010 International Conference on Intelligent Computing and Cognitive Informatics*.

[53] ITU-T. 2020. SG13—Machine Learning Requirements for IMT-2020. Retrieved March 16, 2022 from https://www.itu.int/ITU-T/workprog/.

[54] Chamitha De Alwis, Anshuman Kalla, Quoc-Viet Pham, Pardeep Kumar, Kapal Dev, Won-Joo Hwang, and Madhusanka Liyanage. 2021. Survey on 6G frontiers: Trends, applications, requirements, technologies and future research. *IEEE Open Journal of the Communications Society* 2 (2021), 836–886.

[55] Khaled B. Letaief, Wei Chen, Yuanming Shi, Jun Zhang, and Ying-Jun Angela Zhang. 2019. The roadmap to 6G: AI empowered wireless networks. *IEEE Communications Magazine* 57 (2019), 84–90.

[56] Yan Chen, Peiying Zhu, Gaoning He, Xueqiang Yan, Hadi Baligh, and Jianjun Wu. 2020. From connected people, connected things, to connected intelligence. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[57] Quan Yu, Jing Ren, Haibo Zhou, and Wei Zhang. 2020. A cybertwin based network architecture for 6G. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[58] Vilho Räisänen. 2020. A framework for capability provisioning in B5G. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[59] Gilberto Berardinelli, Preben Mogensen, and Ramoni O. Adeogun. 2020. 6G subnetworks for life-critical communication. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[60] Fengxiao Tang, Yuichi Kawamoto, Nei Kato, and Jiajia Liu. 2020. Future intelligent and secure vehicular network toward 6G: Machine-learning approaches. *Proceedings of the IEEE* 108 (2020), 292–307.

[61] Mohammed S. Hadi, Ahmed Q. Lawey, Taisir E. H. El-Gorashi, and Jaafar M. H. Elmirghani. 2020. Patient-centric hetnets powered by machine learning and big data analytics for 6G networks. *IEEE Access* 8 (2020), 85639–85655.

[62] Seppo Yrjölä, Marja Matinmikko-Blue, and Petri Ahokangas. 2020. How could 6G transform engineering platforms towards ecosystemic business models? In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[63] Nurul Huda Mahmood, Hirley Alves, Onel Alcaraz López, Mohammad Shehab, Diana P. Moya Osorio, and Matti Latva-Aho. 2020. Six key features of machine type communication in 6G. In *Proceedings of the 2020 2nd 6G Wireless Summit (6G SUMMIT'20)*.

[64] Fahad Qasmi, Mohammad Shehab, Hirley Alves, and Matti Latva-Aho. 2018. Optimum transmission rate in fading channels with Markovian sources and QoS constraints. In *Proceedings of the 2018 15th International Symposium on Wireless Communication Systems (ISWCS'18)*.

[65] Shamnaz Riyaz, Kunal Sankhe, Stratis Ioannidis, and Kaushik Chowdhury. 2018. Deep learning convolutional neural networks for radio identification. *IEEE Communications Magazine* 56 (2018), 146–152.

[66] Emilio Calvanese Strinati, Sergio Barbarossa, Jose Luis Gonzalez-Jimenez, Dimitri Ktenas, Nicolas Cassiau, Luc Maret, and Cedric Dehos. 2019. 6G: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication. *IEEE Vehicular Technology Magazine* 14 (2019), 42–50.

[67] Xiaofei Wang, Xiuhua Li, and Victor C. M. Leung. 2015. Artificial intelligence-based techniques for emerging heterogeneous network: State of the arts, opportunities, and challenges. *IEEE Access* 3 (2015), 1379–1391.

[68] ITU-T. 2019. Y.3172: Architectural framework for machine learning in future networks including IMT-2020. Retrieved March 16, 2022 from https://www.itu.int/rec/T-REC-Y.3172/en.

[69] ETSI. 2018. Network Functions Virtualisation (NFV). Retrieved March 2, 2021 from https://www.etsi.org/technologies/nfv.

[70] I. H. Sarker. 2021. Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science* 2 (2021), Article 160.

[71] Kitsuchart Pasupa and Wisuwat Sunhem. 2016. A comparison between shallow and deep architecture classifiers on small dataset. In *Proceedings of the 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE'16)*.

[72] Manish Mishra and Monika Srivastava. 2014. A view of artificial neural network. In *Proceedings of the 2014 International Conference on Advances in Engineering Technology Research (ICAETR'14)*.

[73] Michael Grogan. 2020. Limitations of ARIMA: Dealing with Outliers. Retrieved March 16, 2022 from https://towardsdatascience.com/limitations-of-arima-dealing-with-outliers-30cc0c6ddf33

[74] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaria, M. A. Fadhel, M. Al-Amidie, and L. Farhan. 2021. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data* 8, 1 (2021), 53.

[75] Cen Chen, Kenli Li, Mingxing Duan, and Keqin Li. 2017. Extreme learning machine and its applications in big data processing. In *Big Data Analytics for Sensor-Network Collected Intelligence*, Hui-Huang Hsu, Chuan-Yu Chang, and Ching-Hsien Hsu (Eds.). Academic Press, London, UK, 117–150.

[76] Jean-Paul Van Oosten and Lambert Schomaker. 2014. A reevaluation and benchmark of hidden Markov models. In *Proceedings of the 2014 14th International Conference on Frontiers in Handwriting Recognition*.

[77] Emmanouil Hourdakis and Panos Trahanias. 2011. Improving the classification performance of liquid state machines based on the separation property. *IFIP Advances in Information and Communication Technology* 363 (2011), 52–62.

[78] Luay Alawneh, Belal Mohsen, Mohammad Al-Zinati, Ahmed Shatnawi, and Mahmoud Al-Ayyoub. 2020. A comparison of unidirectional and bidirectional LSTM networks for human activity recognition. In *Proceedings of the 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops'20)*.

[79] Timothy DelSole. 2000. A fundamental limitation of Markov models. *Journal of the Atmospheric Sciences* 57 (2000), 2158–2168.

[80] Zhiyuan Chen, Waleed Mahmoud Soliman, Amril Nazir, and Mohammad Shorfuzzaman. 2021. Variational autoencoders and Wasserstein generative adversarial networks for improving the anti-money laundering process. *IEEE Access* 9 (2021), 83762–83785.

[81] Lei Yang, Yan Chu, Jianpei Zhang, Linlin Xia, Zhengkui Wang, and Kian-Lee Tan. 2015. Transfer learning over big data. In *Proceedings of the 2015 10th International Conference on Digital Information Management (ICDIM'15)*.

[82] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big Data* 3 (2016), Article 9.

[83] Aurélien Géron. 2019. *Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). O'Reilly Media.

[84] J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

[85] J. R. Quinlan. 1986. Induction of decision trees. *Machine Learning* 1 (1986), 81–106.

[86] Navin K. Manaswi. 2020. *Generative Adversarial Networks with Industrial Use Cases: Learning How to Build GAN Applications for Retail*, Healthcare, Telecom, Media, Education, and HRTech. BPB Publications.

[87] Kashvi Taunk, Sanjukta De, Srishti Verma, and Aleena Swetapadma. 2019. A brief review of nearest neighbor algorithm for learning and classification. In *Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS'19)*.

[88] Alan Julian Izenman 2008. Linear discriminant analysis. In *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. Springer Texts in Statistics. Springer, 237–277.

[89] Sasan Karamizadeh, Shahidan M. Abdullah, Mehran Halimi, Jafar Shayan, and Mohammad Javad Rajabi. 2014. Advantage and drawback of support vector machine functionality. In *Proceedings of the 2014 International Conference on Computer, Communications, and Control Technology (I4CT'14)*.

[90] Klevis Ramo. 2019. *Hands-On Java Deep Learning for Computer Vision*. Packt Publishing.

[91] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi. 2020. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Transactions on Cybernetics* 50 (2020), 3826–3839.

[92] Chris Gaskett, David Wettergreen, and Alexander Zelinsky. 1999. Q-learning in continuous state and action spaces. In *Advanced Topics in Artificial Intelligence. Lecture Notes in Computer Science*, Vol. 1747. Springer, 417–428.

[93] Martin D. Buhmann and Slawomir Dinew. 2007. Limits of radial basis function interpolants. *Communications on Pure & Applied Analysis* 6 (2007), 569–585.

[94] H. R. Berenji. 1994. Fuzzy Q-learning: A new approach for fuzzy dynamic programming. In *Proceedings of the 1994 IEEE 3rd International Fuzzy Systems Conference.*

[95] Hoang Viet Nguyen, Hai Ngoc Nguyen, and Tetsutaro Uehara. 2020. Multiple level action embedding for penetration testing. In *Proceedings of the 4th International Conference on Future Networks and Distributed Systems (ICFNDS'20).*

[96] Neil Sinclair, David Harle, Ian A. Glover, James Irvine, and Robert C. Atkinson. 2013. An advanced SOM algorithm applied to handover management within LTE. *IEEE Transactions on Vehicular Technology* 62 (2013), 1883–1894.

[97] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2018. Neural discrete representation learning. *arXiv:1711.00937* [cs] (2018).

[98] Junjie Wu. 2012. Advances in k-means clustering: A data mining thinking.

[99] Henrik Brink, Joseph W. Richards, and Mark Fetherolf. 2016. *Real-World Machine Learning.* Manning.

[100] Saurabh Prasad and Lori Mann Bruce. 2008. Limitations of principal components analysis for hyperspectral target recognition. *IEEE Geoscience and Remote Sensing Letters* 5 (2008), 625–629.

[101] Andrew Ng. 2011. Sparse Autoencoder. Retrieved March 16, 2022 from https://web.stanford.edu/class/cs294a/sparseAutoencoder_2011new.pdf.

[102] Junfeng Xie, F. Richard Yu, Tao Huang, Renchao Xie, Jiang Liu, Chenmeng Wang, and Yunjie Liu. 2019. A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges. *IEEE Communications Surveys & Tutorials* 21 (2019), 393–430.

[103] Peng Wu and Hui Zhao. 2011. Some analysis and research of the AdaBoost algorithm. In *Proceedings of the International Conference on Intelligent Computing and Information Science (ICICIS'11).*

[104] K. Torizuka, H. Oi, F. Saitoh, and S. Ishizu. 2018. Benefit segmentation of online customer reviews using random forest. In *Proceedings of the 2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM'18).*

[105] Shahid Ali, Sreenivas Sremath Tirumala, and Abdolhossein Sarrafzadeh. 2015. Ensemble learning methods for decision making: Status and future prospects. In *Proceedings of the 2015 International Conference on Machine Learning and Cybernetics (ICMLC'15).*

[106] Georgios Drainakis, Konstantinos V. Katsaros, Panagiotis Pantazopoulos, Vasilis Sourlas, and Angelos Amditis. 2020. Federated vs. centralized machine learning under privacy-elastic users: A comparative analysis. In *Proceedings of the 2020 IEEE 19th International Symposium on Network Computing and Applications (NCA'20).*

[107] Alex Galakatos, Andrew Crotty, and Tim Kraska. 2018. Distributed machine learning. In *Encyclopedia of Database Systems*, Ling Liu and M. Tamer Ozsu (Eds.). Springer, 1196–1201.

[108] ODSC: Open Data Science. 2019. *What Are MLOps and Why Does It Matter?* Medium.

[109] Isabelle Guyon, Lisheng Sun-Hosoya, Marc Boullé, Hugo Jair Escalante, Sergio Escalera, Zhengying Liu, Damir Jajetic et al. 2019. Analysis of the AutoML challenge series 2015–2018. In *Automated Machine Learning: Methods, Systems, Challenges.* Springer Series on Challenges in Machine Learning. Springer, 177–219.

[110] David Sweenor, Steven Hillion, Dan Rope, Dev Kannabiran, Thomas Hill, and Michael O'Connell. 2020. *ML Ops: Operationalizing Data Science.* O'Reilly Media.

[111] ITU-T. 2019. Y.3170: Requirements for Machine Learning-Based Quality of Service Assurance for the IMT-2020 Network. Retrieved March 2, 2021 from https://www.itu.int/rec/T-REC-Y.3170/en.

[112] Peng-Yong Kong and Dorin Panaitopol. 2013. Reinforcement learning approach to dynamic activation of base station resources in wireless networks. In *Proceedings of the 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'13).*

[113] Pablo Munoz, Raquel Barco, Jose Maria Ruiz-Aviles, Isabel de la Bandera, and Alejandro Aguilar 2013. Fuzzy rule-based reinforcement learning for load balancing techniques in enterprise LTE femtocells. *IEEE Transactions on Vehicular Technology* 62 (2013), 1962–1973.

[114] P. Munoz, R. Barco, I. de la Bandera, M. Toril, and S. Luna-Ramirez. 2011. Optimization of a fuzzy logic controller for handover-based load balancing. In *Proceedings of the 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring'11).*

[115] Mariana Dirani and Zwi Altman. 2011. Self-organizing networks in next generation radio access networks: Application to fractional power control. *Computer Networks* 55 (2011), 431–438.

[116] Stephen S. Mwanje and Andreas Mitschele-Thiel. 2013. A Q-learning strategy for LTE mobility load balancing. In *Proceedings of the 2013 IEEE 24th Annual International Symposium on Personal*, Indoor, and Mobile Radio Communications *(PIMRC'13).*

[117] Mona Jaber, Muhammad Imran, Rahim Tafazolli, and Anvar Tukmanov. 2015. An adaptive backhaul-aware cell range extension approach. In *Proceedings of the 2015 IEEE International Conference on Communication Workshop (ICCW'15)*.

[118] Mona Jaber, Muhammad Ali Imran, Rahim Tafazolli, and Anvar Tukmanov. 2016. A distributed SON-based user-centric backhaul provisioning scheme. *IEEE Access* 4 (2016), 2314–2330.

[119] Ana Galindo-Serrano, Lorenza Giupponi, and Gunther Auer. 2011. Distributed learning in multiuser OFDMA femtocell networks. In *Proceedings of the 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring'11)*.

[120] Mona Jaber, Muhammad Ali Imran, Rahim Tafazolli, and Anvar Tukmanov. 2016. A multiple attribute user-centric backhaul provisioning scheme using distributed SON. In *Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM'16)*.

[121] Ahsan Adeel, Hadi Larijani, Abbas Javed, and Ali Ahmadinia. 2015. Critical analysis of learning algorithms in random neural network based cognitive engine for LTE systems. In *Proceedings of the 2015 IEEE 81st Vehicular Technology Conference (VTC Spring'15)*.

[122] T. Binzer and F. M. Landstorfer. 2000. Radio network planning with neural networks. In *Proceedings of the Fall 2000 Vehicular Technology Conference (VTC'00 Fall)*.

[123] Derong Liu and Yi Zhang. 2003. A self-learning adaptive critic approach for call admission control in wireless cellular networks. In *Proceedings of the IEEE International Conference on Communications (ICC'03)*.

[124] Amilcare Francesco Santamaria and Andrea Lupia. 2015. A new call admission control scheme based on pattern prediction for mobile wireless cellular networks. In *Proceedings of the 2015 Wireless Telecommunications Symposium (WTS'15)*.

[125] C. J. Debono and J. K. Buhagiar. 2005. Cellular network coverage optimization through the application of self-organizing neural networks. In *Proceedings of the IEEE 62nd Vehicular Technology Conference (VTC'05 Fall)*.

[126] R. Razavi, S. Klein, and H. Claussen. 2010. Self-optimization of capacity and coverage in LTE networks using a fuzzy reinforcement learning approach. In *Proceedings of the 21st Annual IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications*.

[127] Muhammad Naseer ul Islam and Andreas Mitschele-Thiel. 2012. Reinforcement learning strategies for self-organized coverage and capacity optimization. In *Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC'12)*.

[128] Rouzbeh Razavi, Siegfried Klein, and Holger Claussen. 2010. A fuzzy reinforcement learning approach for self-optimization of coverage in LTE networks. *Bell Labs Technical Journal* 15 (2010), 153–175.

[129] Shaoshuai Fan, Hui Tian, and Cigdem Sengul. 2014. Self-optimization of coverage and capacity based on a fuzzy neural network with cooperative reinforcement learning. *EURASIP Journal on Wireless Communications and Networking* 2014, 1 (2014), 57.

[130] Atif Mahmood, Miss Laiha Mat Kiah, Muhammad Reza Z'Aba, Adnan N. Qureshi, Muhammad Shahreeza Safiruz Kassim, Zati Hakim Azizul Hasan, Jagadeesh Kakarla, Iraj Sadegh Amiri, and Saaidal Razalli Azzuhri. 2020. Capacity and frequency optimization of wireless backhaul network using traffic forecasting. *IEEE Access* 8 (2020), 23264–23276.

[131] Pietro Savazzi and Lorenzo Favalli. 2007. Dynamic cell sectorization using clustering algorithms. In *Proceedings of the 2007 IEEE 65th Vehicular Technology Conference (VTC'07 Spring)*.

[132] Cesar A. Sierra Franco and Jose Roberto B. de Marca. 2015. Load balancing in self-organized heterogeneous LTE networks—A statistical learning approach. In *Proceedings of the 2015 7th IEEE Latin-American Conference on Communications (LATINCOM'15)*.

[133] Hafiz Yasar Lateef, Ali Imran, and Adnan Abu-dayya. 2013. A framework for classification of self-organising network conflicts and coordination algorithms. In *Proceedings of the 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'13)*.

[134] Mehdi Bennis and Dusit Niyato. 2010. A Q-learning based approach to interference avoidance in self-organized femtocell networks. In *Proceedings of the 2010 IEEE GLOBCOM Workshops*.

[135] C. Vaishnavi, A. R.. Ashok Kumar, G. Selvakumar, and Shashikant Y. Chaudhari. 2019. Self organizing networks coordination function between intercell interference coordination and coverage and capacity optimisation using support vector machine. In *Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS'19)*.

[136] Ejder Baştuğ, Mehdi Bennis, and Mérouane Debbah. 2015. A transfer learning approach for cache-enabled wireless networks. In *Proceedings of the 2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'15)*.

[137] Francesco Davide Calabrese, Li Wang, Euhanna Ghadimi, Gunnar Peters, Lajos Hanzo, and Pablo Soldati. 2018. Learning radio resource management in RANs: Framework, opportunities, and challenges. *IEEE Communications Magazine* 56 (2018), 138–145.

[138] Souhir Feki, Aymen Belghith, and Faouzi Zarai. 2019. A reinforcement learning-based radio resource management algorithm for D2D-based V2V communication. In *Proceedings of the 2019 15th International Wireless Communications Mobile Computing Conference (IWCMC'19)*.

[139] Yibo Zhou, Fengxiao Tang, Yuichi Kawamoto, and Nei Kato. 2020. Reinforcement learning-based radio resource control in 5G vehicular network. *IEEE Wireless Communications Letters* 9 (2020), 611–614.

[140] Dhananjay Kumar, N. N. Kanagaraj, and R. Srilakshmi. 2013. Harmonized Q-learning for radio resource management in LTE based networks. In *Proceedings of 2013 ITU Kaleidoscope: Building Sustainable Communities*.

[141] Zhiyong Du, Yansha Deng, Weisi Guo, Arumugam Nallanathan, and Qihui Wu. 2021. Green deep reinforcement learning for radio resource management: Architecture, algorithm compression, and challenges. *IEEE Vehicular Technology Magazine* 16 (2021), 29–39.

[142] Yibo Zhou, Zubair Md Fadlullah, Bomin Mao, and Nei Kato. 2018. A deep-learning-based radio resource assignment technique for 5G ultra dense networks. *IEEE Network* 32 (2018), 28–34.

[143] P. Sandhir and K. Mitchell. 2008. A neural network demand prediction scheme for resource allocation in cellular wireless systems. In *Proceedings of the 2008 IEEE Region 5 Conference*.

[144] Peppino Fazio, Floriano De Rango, and Ivano Selvaggi. 2010. A novel passive bandwidth reservation algorithm based on neural networks path prediction in wireless environments. In *Proceedings of the 2010 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'10)*.

[145] Xianfu Chen, Celimuge Wu, Tao Chen, Honggang Zhang, Zhi Liu, Yan Zhang, and Mehdi Bennis. 2020. Age of information aware radio resource management in vehicular networks: A proactive deep reinforcement learning perspective. *IEEE Transactions on Wireless Communications* 19 (2020), 2268–2281.

[146] Rongpeng Li, Zhifeng Zhao, Qi Sun, Chih-Lin I, Chenyang Yang, Xianfu Chen, Minjian Zhao, and Honggang Zhang. 2018. Deep reinforcement learning for resource management in network slicing. *IEEE Access* 6 (2018), 74429–74441.

[147] Anuar Othman and Nazrul Anuar Nayan. 2019. Efficient admission control and resource allocation mechanisms for public safety communications over 5G network slice. *Telecommunication Systems* 72 (2019), 595–607.

[148] Yang Xu, Rui Yin, and Guanding Yu. 2015. Adaptive biasing scheme for load balancing in backhaul constrained small cell networks. *IET Communications* 9 (2015), 999–1005.

[149] Kenza Hamidouche, Walid Saad, Merouane Debbah, Ju Bin Song, and Choong Seon Hong. 2017. The 5G cellular backhaul management dilemma: To cache or to serve. *IEEE Transactions on Wireless Communications* 16 (2017), 4866–4879.

[150] Sumudu Samarakoon, Mehdi Bennis, Walid Saad, and Matti Latva-aho. 2013. Backhaul-aware interference management in the uplink of wireless small cell networks. *IEEE Transactions on Wireless Communications* 12 (2013), 5813–5825.

[151] Caiyun Zhao, Lizhi Peng, Bo Yang, and Zhenxiang Chen. 2012. Labeling the network traffic with accurate application information. In *Proceedings of the 2012 8th International Conference on Wireless Communications, Networking, and Mobile Computing*.

[152] S. Jayashree and N. Shivashankarappa. 2014. Deep packet inspection using ternary content addressable memory. In *Proceedings of the International Conference on Circuits, Communication, Control, and Computing*.

[153] Hyun-Min An, Myung-Sup Kim, and Jae-Hyun Ham. 2013. Application traffic classification using statistic signature. In *Proceedings of the 2013 15th Asia-Pacific Network Operations and Management Symposium (APNOMS'13)*.

[154] Jaehoon Koo, Veena B. Mendiratta, Muntasir Raihan Rahman, and Anwar Walid. 2019. Deep reinforcement learning for network slicing with heterogeneous resource requirements and time varying traffic dynamics. In *Proceedings of the 2019 15th International Conference on Network and Service Management (CNSM'19)*.

[155] Qiang Liu and Tao Han. 2019. When network slicing meets deep reinforcement learning. In *Proceedings of the 15th International Conference on Emerging Networking Experiments and Technologies (CoNEXT'19 Companion)*. 29–30.

[156] Anurag Thantharate, Rahul Paropkari, Vijay Walunj, Cory Beard, and Poonam Kankariya. 2020. Secure5G: A deep learning framework towards a secure network slicing in 5G and beyond. In *Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference (CCWC'20)*.

[157] Taihui Li, Xiaorong Zhu, and Xu Liu. 2020. An end-to-end network slicing algorithm based on deep Q-learning for 5G network. *IEEE Access* 8 (2020), 122229–122240.

[158] Anurag Thantharate, Rahul Paropkari, Vijay Walunj, and Cory Beard. 2019. DeepSlice: A deep learning approach towards an efficient and reliable network slicing in 5G networks. In *Proceedings of the 2019 IEEE 10th Annual Ubiquitous Computing, Electronics, and Mobile Communication Conference (UEMCON'19)*.

[159] Akihiro Nakao, Ping Du, Yoshiaki Kiriha, Fabrizio Granelli, Anteneh Atumo Gebremariam, Tarik Taleb, and Miloud Bagaa. 2017. End-to-end network slicing for 5G mobile networks. *Journal of Information Processing* 25 (2017), 153–163.

[160] Davit Harutyunyan, Riccardo Fedrizzi, Nashid Shahriar, Raouf Boutaba, and Roberto Riggio. 2019. Orchestrating end-to-end slices in 5G networks. In *Proceedings of the 2019 15th International Conference on Network and Service Management (CNSM'19)*.

[161] Xi Li, Giada Landi, Jose Núñez-Martínez, Ramon Casellas, Sergio González, Carla Fabiana Chiasserini, Jorge Rivas Sanchez, et al. 2016. Innovations through 5G-Crosshaul applications. In *Proceedings of the 2016 European Conference on Networks and Communications (EuCNC'16)*.

[162] Xi Li, Ramon Casellas, Giada Landi, Antonio de la Oliva, Xavier Costa-Perez, Andres Garcia-Saavedra, Thomas Deiss, Luca Cominardi, and Ricard Vilalta. 2017. 5G-Crosshaul network slicing: Enabling multi-tenancy in mobile transport networks. *IEEE Communications Magazine* 55 (2017), 128–137.

[163] Haneya Naeem Qureshi, Marvin Manalastas, Syed Muhammad Asad Zaidi, Ali Imran, and Mohamad Omar Al Kalaa. 2021. Service level agreements for 5G and beyond: Overview, challenges and enablers of 5G-healthcare systems. *IEEE Access* 9 (2021), 1044–1061.

[164] Guosheng Zhu, Jun Zan, Yang Yang, and Xiaoyun Qi. 2019. A supervised learning based QoS assurance architecture for 5G networks. *IEEE Access* 7 (2019), 43598–43606.

[165] Michael Iannelli, Muntasir Raihan Rahman, Nakjung Choi, and Le Wang. 2020. Applying machine learning to end-to-end slice SLA decomposition. In *Proceedings of the 2020 6th IEEE Conference on Network Softwarization (NetSoft'20)*.

[166] GSMA. 2018. From Vertical Industry Requirements to Network Slice Characteristics. Retrieved March 2, 2021 from https://www.gsma.com/futurenetworks/resources/from-vertical-industry-requirements-to-network-slice-characteristics/.

[167] OASIS. 2020. Topology and Orchestration Specification for Cloud Applications (TOSCA) TC. Retrieved January 6, 2020 from https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca.

[168] ITU-T. 2018. Service Function Chaining in Mobile Networks. Retrieved March 2, 2021 from https://www.itu.int/rec/T-REC-Y.2242/en.

[169] Xiaofei Wang and Tiankui Zhang. 2019. Reinforcement learning based resource allocation for network slicing in 5G C-RAN. In *Proceedings of the 2019 Computing, Communications, and IoT Applications (ComComAp'19)*.

[170] Guolin Sun, Zemuy Tesfay Gebrekidan, Gordon Owusu Boateng, Daniel Ayepah-Mensah, and Wei Jiang. 2019. Dynamic reservation and deep reinforcement learning based autonomous resource slicing for virtualized radio access networks. *IEEE Access* 7 (2019), 45758–45772.

[171] Davi Militani, Samuel Vieira, Everthon Valadão, Kátia Neles, Renata Rosa, and Demóstenes Z. Rodríguez. 2019. A machine learning model to resource allocation service for access point on wireless network. In *Proceedings of the 2019 International Conference on Software, Telecommunications, and Computer Networks (SoftCOM'19)*.

[172] Mingzhe Chen, Walid Saad, and Changchuan Yin. 2017. Liquid state machine learning for resource allocation in a network of cache-enabled LTE-U UAVs. In *Proceedings of the 2017 IEEE Global Communications Conference (GLOBECOM'17)*.

[173] Dan Huang, Yuan Gao, Yi Li, Mengshu Hou, Wanbin Tang, Shaochi Cheng, Xiangyang Li, and Yunchuan Sun. 2018. Deep learning based cooperative resource allocation in 5G wireless networks. *Mobile Networks and Applications*. Online ahead of print, December 5, 2018.

[174] Sandeep Kumar Singh and Admela Jukan. 2018. Machine-learning-based prediction for resource (re)allocation in optical data center networks. *Journal of Optical Communications and Networking* 10, 10 (2018), D12–D28.

[175] Charalampos Rotsos, Daniel King, Arsham Farshad, Jamie Bird, Lyndon Fawcett, Nektarios Georgalas, Matthias Gunkel, et al. 2017. Network service orchestration standardization: A technology survey. *Computer Standards & Interfaces* 54 (2017), 203–215.

[176] Domenico Cotroneo, Luigi De Simone, and Roberto Natella. 2017. NFV-Bench: A dependability benchmark for network function virtualization systems. *IEEE Transactions on Network and Service Management* 14 (2017), 934–948.

[177] Domenico Cotroneo, Luigi De Simone, Antonio Ken Iannillo, Anna Lanzaro, and Roberto Natella. 2015. Dependability evaluation and benchmarking of network function virtualization infrastructures. In *Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft'15)*.

[178] A. Angelopoulos, E. T. Michailidis, N. Nomikos, P. Trakadas, A. Hatziefremidis, S. Voliotis, and T. Zahariadis. 2019. Tackling faults in the Industry 4.0 era—A survey of machine-learning solutions and key aspects. *Sensors (Basel)* 20, 1 (2019), 109.

[179] David Mulvey, Chuan Heng Foh, Muhammad Ali Imran, and Rahim Tafazolli. 2019. Cell fault management using machine learning techniques. *IEEE Access* 7 (2019), 124514–124539.

[180] Pichanun Sukkhawatchani and Wipawee Usaha. 2008. Performance evaluation of anomaly detection in cellular core networks using self-organizing map. In *Proceedings of the 2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications, and Information Technology*.

[181] Gabriela F. Ciocarlie, Ulf Lindqvist, Szabolcs Nováczki, and Henning Sanneck. 2013. Detecting anomalies in cellular networks using an ensemble method. In *Proceedings of the 9th International Conference on Network and Service Management (CNSM'13)*.

[182] Wenqian Xue, Mugen Peng, Yu Ma, and Hengzhi Zhang. 2014. Classification-based approach for cell outage detection in self-healing heterogeneous networks. In *Proceedings of the 2014 IEEE Wireless Communications and Networking Conference (WCNC'14)*.

[183] Bilal Hussain, Qinghe Du, and Pinyi Ren. 2018. Semi-supervised learning based big data-driven anomaly detection in mobile wireless networks. *China Communications* 15 (2018), 41–57.

[184] Oluwakayode Onireti, Ahmed Zoha, Jessica Moysen, Ali Imran, Lorenza Giupponi, Muhammad Ali Imran, and Adnan Abu-Dayya. 2016. A cell outage management framework for dense heterogeneous networks. *IEEE Transactions on Vehicular Technology* 65 (2016), 2097–2113.

[185] Jessica Moysen and Lorenza Giupponi. 2014. A reinforcement learning based solution for self-healing in LTE networks. In *Proceedings of the 2014 IEEE 80th Vehicular Technology Conference (VTC'14 Fall)*.

[186] Ana Gomez-Andrades, Pablo Munoz, Inmaculada Serrano, and Raquel Barco. 2016. Automatic root cause analysis for LTE networks based on unsupervised techniques. *IEEE Transactions on Vehicular Technology* 65 (2016), 2369–2386.

[187] R. M. Khanafer, B. Solana, J. Triola, R. Barco, L. Moltsen, Z. Altman, and P. Lazaro. 2008. Automated diagnosis for UMTS networks using Bayesian network approach. *IEEE Transactions on Vehicular Technology* 57 (2008), 2451–2461.

[188] Haitao Zhao, Shaoyuan Sun, and Bo Jin 2018. Sequential fault diagnosis based on LSTM neural network. *IEEE Access* 6 (2018), 12929–12939.

[189] Srinikethan Madapuzi Srinivasan, Tram Truong-Huu, and Mohan Gurusamy. 2018. TE-based machine learning techniques for link fault localization in complex networks. In *Proceedings of the 2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud'18)*.

[190] Srinikethan Madapuzi Srinivasan, Tram Truong-Huu, and Mohan Gurusamy. 2019. Machine learning-based link fault identification and localization in complex networks. *IEEE Internet of Things Journal* 6 (2019), 6556–6566.

[191] Shubham Khunteta and Ashok Kumar Reddy Chavva. 2017. Deep learning based link failure mitigation. In *Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA'17)*.

[192] Tram Truong-Huu, Prarthana Prathap, Purnima Murali Mohan, and Mohan Gurusamy. 2019. Fast and adaptive failure recovery using machine learning in software defined networks. In *Proceedings of the 2019 IEEE International Conference on Communications Workshops (ICC Workshops'19)*.

[193] Jani Puttonen, Jussi Turkka, Olli Alanen, and Janne Kurjenniemi. 2010. Coverage optimization for minimization of drive tests in LTE with extended RLF reporting. In *Proceedings of the 21st Annual IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications*.

[194] J. Barthelemy, N. Verstaevel, H. Forehead, and P. Perez. 2019. Edge-computing video analytics for real-time traffic monitoring in a smart city. *Sensors (Basel)* 19, 9 (2019), 2048.

[195] 3GPP. 2016. Specs. 23.501: System Architecture for the 5G System (5GS). Retrieved March 2, 2021 from https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/.

[196] Yekta Turk, Engin Zeydan, and Zeki Bilgin. 2019. A machine learning based management system for network services. In *Proceedings of the 2019 International Conference on Wireless and Mobile Computing, Networking, and Communications (WiMob'19)*.

[197] Guoru Ding, Yutao Jiao, Jinlong Wang, Yulong Zou, Qihui Wu, Yu-Dong Yao, and Lajos Hanzo. 2018. Spectrum inference in cognitive radio networks: Algorithms and applications. *IEEE Communications Surveys & Tutorials* 20 (2018), 150–182.

[198] Chaoqiong Fan, Bin Li, Chenglin Zhao, Weisi Guo, and Ying-Chang Liang. 2018. Learning-based spectrum sharing and spatial reuse in mm-wave ultradense networks. *IEEE Transactions on Vehicular Technology* 67 (2018), 4954–4968.

[199] Ghassan Alnwaimi, Seiamak Vahid, and Klaus Moessner. 2015. Dynamic heterogeneous learning games for opportunistic access in LTE-based macro/femtocell deployments. *IEEE Transactions on Wireless Communications* 14 (2015), 2294–2308.

[200] Yaohua Sun, Mugen Peng, and H. Vincent Poor. 2018. A distributed approach to improving spectral efficiency in uplink device-to-device-enabled cloud radio access networks. *IEEE Transactions on Communications* 66 (2018), 6511–6526.

[201] Manikantan Srinivasan, Vijeth J. Kotagi, and C. Siva Ram Murthy. 2016. A Q-learning framework for user QoE enhanced self-organizing spectrally efficient network using a novel inter-operator proximal spectrum sharing. *IEEE Journal on Selected Areas in Communications* 34 (2016), 2887–2901.

[202] Mingzhe Chen, Walid Saad, and Changchuan Yin. 2017. Echo state networks for self-organizing resource allocation in LTE-U with uplink–downlink decoupling. *IEEE Transactions on Wireless Communications* 16 (2017), 3–16.

[203] Fandi Lin, Jin Chen, Guoru Ding, Yutao Jiao, Jiachen Sun, and Haichao Wang. 2021. Spectrum prediction based on GAN and deep transfer learning: A cross-band data augmentation framework. *China Communications* 18 (2021), 18–32.

[204] Chen Yu, Yang Liu, Dezhong Yao, Laurence T. Yang, Hai Jin, Hanhua Chen, and Qiang Ding. 2017. Modeling user activity patterns for next-place prediction. *IEEE Systems Journal* 11 (2017), 1060–1071.

[205] Xu Chen, François Mériaux, and Stefan Valentin. 2013. Predicting a user's next cell with supervised learning based on channel states. In *Proceedings of the 2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC'13)*.

[206] Sherif Akoush and Ahmed Sameh. 2007. The use of Bayesian learning of neural networks for mobile user position prediction. In *Proceedings of the 7th International Conference on Intelligent Systems Design and Applications (ISDA'07).*

[207] Abdelrahim Mohamed, Oluwakayode Onireti, Seyed Amir Hoseinitabatabaei, Muhammad Imran, Ali Imran, and Rahim Tafazolli. 2015. Mobility prediction for handover management in cellular networks with control/data separation. In *Proceedings of the 2015 IEEE International Conference on Communications (ICC'15).*

[208] Hongbo Si, Yue Wang, Jian Yuan, and Xiuming Shan. 2010. Mobility prediction in cellular network using hidden Markov model. In *Proceedings of the 2010 7th IEEE Consumer Communications and Networking Conference.*

[209] Qiujian Lv, Zongshan Mei, Yuanyuan Qiao, Yufei Zhong, and Zhenming Lei. 2014. Hidden Markov model based user mobility analysis in LTE network. In *Proceedings of the 2014 International Symposium on Wireless Personal Multimedia Communications (WPMC'14).*

[210] Hasan Farooq and Ali Imran. 2017. Spatiotemporal mobility prediction in proactive self-organizing cellular networks. *IEEE Communications Letters* 21 (2017), 370–373.

[211] Shutchon Premchaisawatt and Nararat Ruangchaijatupon. 2014. Enhancing indoor positioning based on partitioning cascade machine learning models. In *Proceedings of the 2014 11th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications, and Information Technology (ECTI-CON'14).*

[212] Ayon Chakraborty, Luis E. Ortiz, and Samir R. Das. 2015. Network-side positioning of cellular-band devices with minimal effort. In *Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM'15).*

[213] Abdelrahim Mohamed, Oluwakayode Onireti, Muhammad Ali Imran, Ali Imran, and Rahim Tafazolli. 2017. Predictive and core-network efficient RRC signalling for active state handover in RANs with control/data separation. *IEEE Transactions on Wireless Communications* 16 (2017), 1423–1436.

[214] P. P. Bhattacharya, Ananya Sarkar, Indranil Sarkar, and Subhajit Chatterjee. 2013. An ANN based call handoff management scheme for mobile cellular network. *International Journal of Wireless & Mobile Networks* 5 (2013), 125–135.

[215] Moses Ekpenyong, Joseph Isabona, and Etebong Isong. 2015. Handoffs decision optimization of mobile celular networks. In *Proceedings of the 2015 International Conference on Computational Science and Computational Intelligence (CSCI'15).*

[216] Milena Stoyanova, and Petri Mahonen. 2007. Algorithmic approaches for vertical handoff in heterogeneous wireless environment. In *Proceedings of the 2007 IEEE Wireless Communications and Networking Conference.*

[217] Stephen S. Mwanje and Andreas Mitschele-Thiel. 2014. Distributed cooperative Q-learning for mobility-sensitive handover optimization in LTE SON. In *Proceedings of the 2014 IEEE Symposium on Computers and Communications (ISCC'14).*

[218] Chaima Dhahri and Tomoaki Ohtsuki. 2014. Adaptive Q-learning cell selection method for open-access femtocell networks: Multi-user case. *IEICE Transactions on Communications E97.B* (2014), 1679–1688.

[219] R. Narasimhan and D. C. Cox. 1998. A handoff algorithm for wireless systems using pattern recognition. In *Proceedings of the 9th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications.*

[220] Zoraze Ali, Nicola Baldo, Josep Mangues-Bafalluy, and Lorenza Giupponi. 2016. Machine learning based handover management for improved QoE in LTE. In *Proceedings of the 2016 IEEE/IFIP Network Operations and Management Symposium (NOMS'16).*

[221] Sana Aroussi and Abdelhamid Mellouk. 2014. Survey on machine learning-based QoE-QoS correlation models. In *Proceedings of the 2014 International Conference on Computing, Management, and Telecommunications (ComManTel'14).*

[222] Haiqing Du, Chang Guo, Yixi Liu, and Yong Liu. 2009. Research on relationship between QoE and QoS based on BP neural network. In *Proceedings of the 2009 IEEE International Conference on Network Infrastructure and Digital Content.*

[223] Victor A. Machado, Carlos N. Silva, Rosinei S. Oliveira, Alexandre M. Melo, Marcelino Silva, Carlos R. L. Francês, João C. W. A. Costa, Nandamudi L. Vijaykumar, and Celso M. Hirata. 2011. A new proposal to provide estimation of QoS and QoE over WiMAX networks: An approach based on computational intelligence and discrete-event simulation. In *Proceedings of the 2011 IEEE 3rd Latin-American Conference on Communications.*

[224] Prasad Calyam, Prashanth Chandrasekaran, Gregg Trueb, Nathan Howes, Rajiv Ramnath, Delei Yu, Ying Liu, Lixia Xiong, and Daoyan Yang. 2012. Multi-resolution multimedia QoE models for IPTV applications. *International Journal of Digital Multimedia Broadcasting* 2012 (2012), 1–13.

[225] Xi Chen, Zonghang Li, Yupeng Zhang, Ruiming Long, Hongfang Yu, Xiaojiang Du, and Mohsen Guizani. 2018. Reinforcement learning based QoS/QoE-aware service function chaining in software-driven 5G slices. *arXiv-1804.02099* [cs] (2018).

[226] Kan Zheng, Zhe Yang, Kuan Zhang, Periklis Chatzimisios, Kan Yang, and Wei Xiang. 2016. Big data-driven optimization for mobile networks toward 5G. *IEEE Network* 30 (2016), 44–51.

[227] Miloš B. Stojanovic, Nikola M. Sekulovic, and Aleksandra S. Panajotovic. 2019. A comparative performance analysis of extreme learning machine and echo state network for wireless channel prediction. In *Proceedings of the 2019 14th International Conference on Advanced Technologies, Systems, and Services in Telecommunications (TELSIKS'19).*

[228] Saud Aldossari and Kwang-Cheng Chen. 2019. Predicting the path loss of wireless channel models using machine learning techniques in mmwave urban communications. In *Proceedings of the 2019 22nd International Symposium on Wireless Personal Multimedia Communications (WPMC'19)*.

[229] Jakob Thrane, Darko Zibar, and Henrik Lehrmann Christiansen. 2020. Model-aided deep learning method for path loss prediction in mobile communication systems at 2.6 GHz. *IEEE Access* 8 (2020), 7925–7936.

[230] Dina Tarek, Abderrahim Benslimane, M. Darwish, and Amira M. Kotb. 2019. Cognitive radio networks channel state estimation using machine learning techniques. In *Proceedings of the 2019 15th International Wireless Communications Mobile Computing Conference (IWCMC'19)*.

[231] ITU-T. 2018. Y.3104: Architecture of the IMT-2020 Network. Retrieved March 16, 2022 from https://www.itu.int/rec/T-REC-Y.3104/en.

[232] 3GPP. 2017. Specs. 38.214: NR-Physical Layer Procedures for Data. Retrieved March 2, 2021 from https://www.3gpp.org/ftp/Specs/archive/38_series/38.214/.

[233] Syed Tamoor-ul-Hassan, Sumudu Samarakoon, Mehdi Bennis, Matti Latva-aho, and Choong Seon Hong. 2018. Learning-based caching in cloud-aided wireless networks. *IEEE Communications Letters* 22 (2018), 137–140.

[234] Ying He, Nan Zhao, and Hongxi Yin. 2018. Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach. *IEEE Transactions on Vehicular Technology* 67 (2018), 44–55.

[235] Chen Zhong, M. Cenk Gursoy, and Senem Velipasalar. 2018. A deep reinforcement learning-based framework for content caching. In *Proceedings of the 2018 52nd Annual Conference on Information Sciences and Systems (CISS'18)*.

[236] Wei Wang, Ruining Lan, Jingxiong Gu, Aiping Huang, Hangguan Shan, and Zhaoyang Zhang. 2017. Edge caching at base stations with device-to-device offloading. *IEEE Access* 5 (2017), 6399–6410.

[237] S. M. Shahrear Tanzil, William Hoiles, and Vikram Krishnamurthy. 2017. Adaptive scheme for caching YouTube content in a cellular network: Machine learning approach. *IEEE Access* 5 (2017), 5870–5881.

[238] Mingzhe Chen, Mohammad Mozaffari, Walid Saad, Changchuan Yin, Merouane Debbah, and Choong Seon Hong. 2017. Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience. *IEEE Journal on Selected Areas in Communications* 35 (2017), 1046–1061.

[239] Mingzhe Chen, Walid Saad, Changchuan Yin, and Merouane Debbah. 2017. Echo state networks for proactive caching in cloud-based radio access networks with mobile users. *IEEE Transactions on Wireless Communications* 16 (2017), 3520–3535.

[240] Khai Nguyen Doan, Thang Van Nguyen, Tony Q. S. Quek, and Hyundong Shin. 2018. Content-aware proactive caching for backhaul offloading in cellular network. *IEEE Transactions on Wireless Communications* 17 (2018), 3128–3140.

[241] Ambar Bajpai and Arun Balodi 2020. Role of 6G networks: Use cases and research directions. In *Proceedings of the 2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC'20)*.

[242] Tan Zhang, Aakanksha Chowdhery, Paramvir (Victor) Bahl, Kyle Jamieson, and Suman Banerjee. 2015. The design and implementation of a wireless video surveillance system. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom'15)*. 426–438.

[243] Chien-Chun Hung, Ganesh Ananthanarayanan, Peter Bodik, Leana Golubchik, Minlan Yu, Paramvir Bahl, and Matthai Philipose. 2018. VideoEdge: Processing camera streams using hierarchical clusters. In *Proceedings of the 2018 IEEE/ACM Symposium on Edge Computing (SEC'18)*.

[244] AWS. 2019. Deep Learning Enabled Video Camera for Developers. Retrieved January 6, 2020 from https://aws.amazon.com/deeplens.

[245] Aniekan Essien, Ilias Petrounias, Pedro Sampaio, and Sandra Sampaio. 2020. A deep-learning model for urban traffic flow prediction with traffic events mined from Twitter. *World Wide Web* 24 (2020), 1345–1368.

[246] Faisal Saeed, Anand Paul, Won Hwa Hong, and Hyuncheol Seo. 2019. Machine learning based approach for multimedia surveillance during fire emergencies. *Multimedia Tools and Applications* 79 (2019), 16201–16217.

[247] R. Vinayakumar, K. P. Soman, and Prabaharan Poornachandran. 2017. Applying deep learning approaches for network traffic prediction. In *Proceedings of the 2017 International Conference on Advances in Computing, Communications, and Informatics (ICACCI'17)*.

[248] Christos Kyrkou and Theocharis Theocharides. 2019. Deep-learning-based aerial image classification for emergency response applications using unmanned aerial vehicles. In *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'19)*.

[249] Chuanting Zhang, Haixia Zhang, Jingping Qiao, Dongfeng Yuan, and Minggao Zhang. 2019. Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data. *IEEE Journal on Selected Areas in Communications* 37 (2019), 1389–1401.

[250] Zhixiang He, Chi-Yin Chow, and Jia-Dong Zhang. 2019. STCNN: A spatio-temporal convolutional neural network for long-term traffic prediction. In *Proceedings of the 2019 20th IEEE International Conference on Mobile Data Management (MDM'19)*.

[251] Dario Bega, Marco Gramaglia, Marco Fiore, Albert Banchs, and Xavier Costa-Perez. 2019. DeepCog: Cognitive network management in sliced 5G networks with deep learning. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'19)*.

[252] Ion Railean, Cristina Stolojescu, Sorin Moga, and Philippe Lenca. 2010. WIMAX traffic forecasting based on neural networks in wavelet domain. In *Proceedings of the 2010 4th International Conference on Research Challenges in Information Science (RCIS'10)*.

[253] Carolina Gijón, Matías Toril, Salvador Luna-Ramírez, María Luisa Marí-Altozano, and José María Ruiz-Avilés. 2021. Long-term data traffic forecasting for network dimensioning in LTE with short time series. *Electronics* 10, 10 (2021), 1151.

[254] Jagan Mohan Reddy and Chittaranjan Hota. 2013. P2P traffic classification using ensemble learning. In *Proceedings of the 5th IBM Collaborative Academia Research Exchange Workshop (I-CARE'13)*.

[255] Gabriel Gómez Sena and Pablo Belzarena. 2012. Statistical traffic classification by boosting support vector machines. In *Proceedings of the 7th Latin American Networking Conference (LANC'12)*. 9–18.

[256] Kuldeep Singh and S. Agrawal. 2011. Feature extraction based IP traffic classification using machine learning. In *Proceedings of the International Conference on Advances in Computing and Artificial Intelligence (ACAI'11)*. 208–212.

[257] Thuy T. T. Nguyen, Grenville Armitage, Philip Branch, and Sebastian Zander. 2012. Timely and continuous machine-learning-based classification for interactive IP Traffic. *IEEE/ACM Transactions on Networking* 20 (2012), 1880–1894.

[258] Wanqian Zhang, Junxiao Wang, Sheng Chen, Heng Qi, and Keqiu Li. 2019. A framework for resource-aware online traffic classification using CNN. In *Proceedings of the 14th International Conference on Future Internet Technologies (CFI'19)*. Article 5, 6 pages.

[259] Yu Jin, Nick Duffield, Jeffrey Erman, Patrick Haffner, Subhabrata Sen, and Zhi-Li Zhang 2012. A modular machine learning system for flow-level traffic classification in large networks. *ACM Transactions on Knowledge Discovery from Data* 6 (2012), 1–34.

[260] Yan Luo, Ke Xiang, and Sanping Li. 2008. Acceleration of decision tree searching for IP traffic classification. In *Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS'08)*. 40–49

[261] Lu He, Chen Xu, and Yan Luo. 2016. vTC: Machine learning based traffic classification as a virtual network function. In *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks and Network Function Virtualization (SDN-NFV Security'16)*. 53–56.

[262] Erico N. de Souza, Stan Matwin, and Stenio Fernandes. 2013. Network traffic classification using AdaBoost dynamic. In *Proceedings of the 2013 IEEE International Conference on Communications Workshops (ICC'13)*.

[263] Yaping Cui, Xinyun Huang, Dapeng Wu, Hao Zheng, and Changqing Luo. 2020. Machine learning-based resource allocation strategy for network slicing in vehicular networks. *Wireless Communications and Mobile Computing* 2020 (2020), 1–10.

[264] Thiago R. Raddo, Simon Rommel, Bruno Cimoli, Chris Vagionas, Diego Perez-Galacho, Evangelos Pikasis, Evangelos Grivas et al. 2021. Transition technologies towards 6G networks. *EURASIP Journal on Wireless Communications and Networking* 2021 (2021), Article 100.

[265] Wei Li, Fan Zhou, Kaushik Roy Chowdhury, and Waleed Meleis. 2019. QTCP: Adaptive congestion control with reinforcement learning. *IEEE Transactions on Network Science and Engineering* 6 (2016), 445–458.

[266] Yiming Kong, Hui Zang, and Xiaoli Ma. 2018. Improving TCP congestion control with machine intelligence. In *Proceedings of the 2018 Workshop on Network Meets AI and ML*.

[267] Kefan Xiao, Shiwen Mao, and Jitendra K. Tugnait. 2019. TCP-Drinc: Smart congestion control based on deep reinforcement learning. *IEEE Access* 7 (2019), 11892–11904.

[268] Ning Li, Zhongliang Deng, Qiaodi Zhu, and Qin Du. 2019. AdaBoost-TCP: A machine learning-based congestion control method for satellite networks. In *Proceedings of the 2019 IEEE 19th International Conference on Communication Technology (ICCT'19)*.

[269] Mariyam Mirza, Joel Sommers, Paul Barford, and Xiaojin Zhu. 2010. A machine learning approach to TCP throughput prediction. *IEEE/ACM Transactions on Networking* 18 (2010), 1026–1039.

[270] Minghao Wang, Tianqing Zhu, Tao Zhang, Jun Zhang, Shui Yu, and Wanlei Zhou. 2020. Security and privacy in 6G networks: New areas and new challenges. *Digital Communications and Networks* 6 (2020), 281–291.

[271] Microsoft. 2020. What are Azure Machine Learning Pipelines? Retrieved March 2, 2021 from https://docs.microsoft.com/en-us/azure/machine-learning/concept-ml-pipelines.

[272] Corinna Cortes, Xavier Gonzalvo, Vitaly Kuznetsov, Mehryar Mohri, and Scott Yang. 2017. AdaNet: Adaptive structural learning of artificial neural networks. In *Proceedings of the 34th International Conference on Machine Learning (PMLR'17)*.

[273] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y. Ng. 2009. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*.

[274] Souad BelMannoubi, Haifa Touati, and Hichem Snoussi. 2019. Stacked auto-encoder for scalable indoor localization in wireless sensor networks. In *Proceedings of the 2019 15th International Wireless Communications Mobile Computing Conference (IWCMC'19)*.

[275] Yu Xu, Dezhi Li, Zhenyong Wang, Qing Guo, and Wei Xiang. 2018. A deep learning method based on convolutional neural network for automatic modulation classification of wireless signals. *Wireless Networks* 25 (2018), 3735–3746.

[276] Byungseok Kang and Hyunseung Choo. 2016. A deep-learning-based emergency alert system. *ICT Express* 2 (2016), 67–70.

[277] Mohamed Abdel-Nasser, Karar Mahmoud, Osama A. Omer, Matti Lehtonen, and Domenec Puig. 2020. Link quality prediction in wireless community networks using deep recurrent neural networks. *Alexandria Engineering Journal* 59 (2020), 3531–3543.

[278] Hao Ye, Le Liang, Geoffrey Ye Li, and Biing-Hwang Juang. 2020. Deep learning-based end-to-end wireless communication systems with conditional GANs as unknown channels. *IEEE Transactions on Wireless Communications* 19 (2020), 3133–3143.

[279] Yiding Yu, Taotao Wang, and Soung Chang Liew. 2018. Deep-reinforcement learning multiple access for heterogeneous wireless networks. In *Proceedings of the 2018 IEEE International Conference on Communications (ICC'18)*.

[280] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*.

[281] Xin Du, Katayoun Farrahi, and Mahesan Niranjan. 2019. Transfer learning across human activities using a cascade neural network architecture. In *Proceedings of the 23rd International Symposium on Wearable Computers (ISWC'19)*.

[282] Zhiyong Du, Yansha Deng, Weisi Guo, Arumugam Nallanathan, and Qihui Wu. 2019. Green deep reinforcement learning for radio resource management: Architecture, algorithm compression and challenge. *arXiv:1910.05054* [cs, eess] (2019).

[283] Datatonic. 2021. The 2021 Guide to MLOps. Retrieved March 16, 2022 from https://datatonic.com/insights/mlops-guide-2021/.

[284] Nei Kato, Bomin Mao, Fengxiao Tang, Yuichi Kawamoto, and Jiajia Liu. 2020. Ten challenges in advancing machine learning technologies toward 6G. *IEEE Wireless Communications* 27 (2020), 96–103.

[285] ITU-T. 2002. E860: Framework of a Service Level Agreement. Retrieved March 16, 2022 from https://www.itu.int/rec/T-REC-E.860-200206-I/en.

[286] Ekram Hossain and Firas Fredj. 2021. Editorial energy efficiency of machine-learning-based designs for future wireless systems and networks. *IEEE Transactions on Green Communications and Networking* 5 (2021), 1005–1010.

[287] Ray Le Maistre. 2021. SoftBank Outlines 12 Challenges for 6G. Retrieved March 16, 2022 from https://www.telecomtv.com/content/6g/softbank-outlines-12-challenges-for-6g-42096/.

Received 18 February 2021; revised 25 January 2022; accepted 28 February 2022