ADAPTIVE DATA-DRIVEN OPTIMIZATION USING TRANSFER LEARNING FOR RESILIENT, ENERGY-EFFICIENT, RESOURCE-AWARE, AND SECURE NETWORK SLICING IN 5G-ADVANCED AND 6G WIRELESS SYSTEMS

A Dissertation
IN
Computer Networking and Communication System
and
Electrical and Computer Engineering

Presented to the Faculty of the University of Missouri – Kansas City
in partial fulfillment of
the requirements for the degree

DOCTOR OF PHILOSOPHY

by
ANURAG THANTHARATE

B.E., Rashtrasant Tukadoji Maharaj Nagpur University, Maharashtra, India, 2009
M.S., University of Missouri - Kansas City, MO, USA, 2012

Kansas City, Missouri
2022

ADAPTIVE DATA-DRIVEN OPTIMIZATION USING TRANSFER LEARNING FOR RESILIENT, ENERGY-EFFICIENT, RESOURCE-AWARE, AND SECURE NETWORK SLICING IN 5G-ADVANCED AND 6G WIRELESS SYSTEMS

Anurag Thantharate, Candidate for the Doctor of Philosophy Degree
University of Missouri – Kansas City, 2022

ABSTRACT

5G–Advanced is the next step in the evolution of the fifth–generation (5G) technology. It will introduce a new level of expanded capabilities beyond connections and enables a broader range of advanced applications and use cases. 5G–Advanced will support modern applications with greater mobility and high dependability. Artificial intelligence and Machine Learning will enhance network performance with spectral efficiency and energy savings enhancements.

This research established a framework to optimally control and manage an appropriate selection of network slices for incoming requests from diverse applications and services in Beyond 5G networks. The developed *DeepSlice* model is used to optimize the network and individual slice load efficiency across isolated slices and manage slice lifecycle in case of failure. The *DeepSlice* framework can predict the unknown connections by utilizing the learning from a developed deep-learning neural network model.

The research also addresses threats to the performance, availability, and robustness of B5G networks by proactively preventing and resolving threats. The study proposed a *Secure5G* framework for authentication, authorization, trust, and control for a network slicing architecture in 5G systems. The developed model prevents the 5G infrastructure from Distributed Denial of Service by analyzing incoming connections and learning from the developed model. The research demonstrates the preventive measure against volume attacks, flooding attacks, and masking (spoofing) attacks. This research builds the framework towards the zero trust objective (never trust, always verify, and verify continuously) that improves resilience.

Another fundamental difficulty for wireless network systems is providing a desirable user experience in various network conditions, such as those with varying network loads and bandwidth fluctuations. Mobile Network Operators have long battled unforeseen network traffic events. This research proposed *ADAPTIVE6G* to tackle the network load estimation problem using knowledge-inspired Transfer Learning by utilizing radio network Key Performance Indicators from network slices to understand and learn network load estimation problems. These algorithms enable Mobile Network Operators to optimally coordinate their computational tasks in stochastic and time-varying network states.

Energy efficiency is another significant KPI in tracking the sustainability of network slicing. Increasing traffic demands in 5G dramatically increase the energy consumption of mobile networks. This increase is unsustainable in terms of dollar cost and environmental impact. This research proposed an innovative *ECO6G* model to attain sustainability and energy efficiency. Research findings suggested that the developed model can reduce network energy costs without negatively impacting performance or end customer experience against the classical Machine Learning and Statistical driven models. The proposed model is validated against the industry-standardized energy efficiency definition, and operational expenditure savings are derived, showing significant cost savings to MNOs.

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Graduate Studies, have examined a dissertation titled “Adaptive Data-driven Optimization using Transfer Learning for Resilient, Energy-Efficient, Resource-Aware, and Secure Network Slicing in 5G-Advanced and 6G Wireless Systems,” presented by Anurag Thantharate, candidate for the Doctor of Philosophy degree, and hereby certify that in their opinion it is worthy of acceptance.

Supervisory Committee

Cory Beard, Ph.D., Committee Chair
Department of Computer Science Electrical Engineering

Deepankar Medhi, Ph.D.
Department of Computer Science Electrical Engineering

Masud Chowdhury, Ph.D.
Department of Computer Science Electrical Engineering

Sejun Song, Ph.D.
Department of Computer Science Electrical Engineering

Zhu Li, Ph.D.
Department of Computer Science Electrical Engineering

CONTENTS

ILLUSTRATIONS

TABLES

Tables Page

Acknowledgments

Dream is not that which you see while sleeping; it is something that does not let you sleep

- Dr. A.P.J Abdul Kalam

I want to dedicate this dissertation to my family for their selfless love, and blessing and for always believing in me throughout my journey in life – my mom Jyoti Thantharate, dad Deepak Thantharate, and passed grandma, who supported me spiritually throughout writing this dissertation.

Words cannot express my gratitude and profound appreciation to my committee chair and my advisor, Dr. Cory Beard, for his continuous support and trust in me. His patience and immense research knowledge have encouraged me to be a focused researcher and critical thinker. I will be forever grateful for his mentorship and assistance.

I extend my sincere appreciation to my dissertation committee members – Dr. Masud Chowdhury, Dr. Deep Medhi, Dr. Zhu Li, and Dr. Sejun Song, for their insightful and constructive feedback, which encouraged me to widen my research perspective.

I could not have undertaken this journey without the moral and emotional support of my wife, Poonam Kankariya, and my brother Pratik Thantharate. Both held me up through thick and thin these last four years. I am also grateful to my extended family Moksha Sahu and Divya Todurkar, for their unwavering belief in me and support.

I am fortunate to have friends like family and my fellow researchers - Dr. Rohit Abhishek, Dr. Vijay Walunj, Dr. Rahul Paropkari, and Priyanka Gaikwad, without whom this endeavor and a decade of life in Kansas City would not have been possible.

CHAPTER 1

INTRODUCTION

The fifth-generation (5G) wireless technology promises to be the critical enabler of use cases far beyond smartphones and other connected devices. The next-generation 5G-Advanced wireless standard confronts the changing face of connectivity by enabling elevated levels of automation through continuous optimization of several Key Performance Indicators (KPIs) such as latency, reliability, connection density, and energy efficiency. 5G and Beyond networks are boosted by integrating software-defined networking (SDN) and network function virtualization (NFV) technologies. The distributed, granular, cloud-based, event-driven architecture of 5G allows for agile resource allocation, ultra-low latency edge-based services, and more. Rapid software modifications and functional requirements make it difficult for network operators to implement continuous integration and delivery (CI/CD). By decoupling software from hardware, operators can deploy code updates more rapidly than ever to address business challenges.

Emerging use cases and applications, such as machine-to-machine communications, multi-access edge computing, autonomous driving, and data-driven network designs, have stringent reliability, latency, throughput, and security requirements. Such requirements pose new challenges to architecture design, network management, and resource orchestration in next-generation wireless networks while allowing resource sharing among multiple tenants. Zero-touch, Artificial Intelligence (AI) /Machine Learning (ML) empowered cognitive network, and service automation becomes crucial for continuously ensuring highly diverse B5G networks.

Existing cellular communications and the B5G mobile network requires meeting high-reliability standards, very low latency, higher capacity, more security, and high-speed user connectivity. MNOs are looking for a programmable solution that will allow them to accommodate multiple independent tenants on the same physical infrastructure and 5G networks allow for end-to-end network resource allocation using the concept of Network Slicing. Due to the traffic explosion, data-driven decision-making will be vital in future communication networks, and AI will accelerate the 5G network performance. A mobile radio air interface dynamically defined by AI/ML will be essential for future networks. These interfaces could allow radios, devices, and network elements to learn from one another and their surroundings.

1.1 Research Objectives

This disruptive deployment of 5G has triggered the need for transformation and a radical change in how networks and services are managed and orchestrated. The focus of this dissertation is to study the fundamental issues in network slice management, load prediction, resource management, network slice security, and energy efficiency in Beyond 5G Networks (B5G). More specifically, the major problem that this research study has raised and undertaken is providing a better understanding of the question of how the 5G network and data-driven ML models may fit together to make an efficient and optimized solution for managing network slicing resource management, slice load-balancing, and slice selection in B5G networks and further extend it towards energy-efficient networks.

A cornerstone of wireless connectivity involves trust and privacy in the data shared between users and network elements as wireless connectivity becomes an integrated, fundamental element of society. With a large influx of data in B5G systems from end-users and network elements, it is imperative to understand how data is collected and used for real-time data processing operations. The current wireless network learning involves centralizing the training data, which is inefficient as it continuously requires end devices to send their collected data to a central server.

1.2 Network Slicing in Beyond 5G Networks

Network slicing is an essential technology for 5G and Beyond mobile communication. Slicing the physical mobile communication network into multiple virtual networks maximizes the benefits of high-speed communication, ultra-low latency, and ultra-connected communication. With each segmented network, as shown in Fig. 1, a variety of specialized services can be provided. 5G network slicing is becoming increasingly crucial as 5G services with varying requirements cannot be fully utilized with a uniform network service policy. Specifically, network slicing technology is vital because 5G is focused on providing various services efficiently, such as autonomous driving, smart cities, robotic

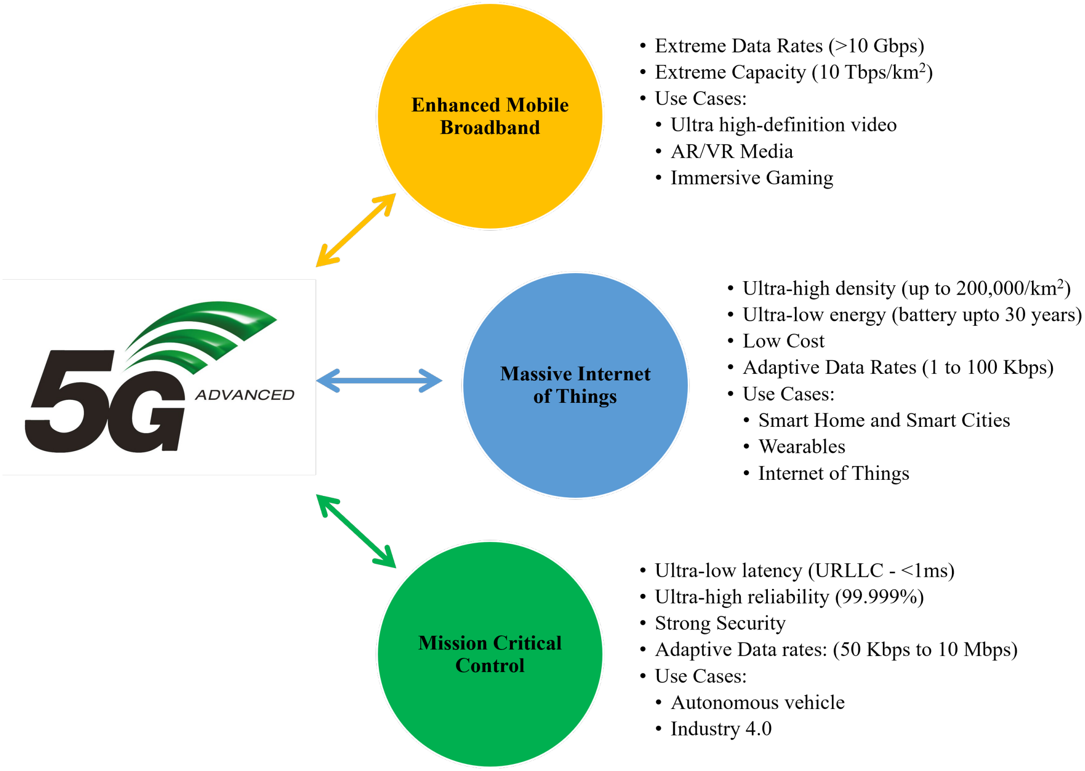

Enhanced Mobile Broadband
• Extreme Data Rates (>10 Gbps)
• Extreme Capacity (10 Tbps/km^2)
• Use Cases:
• Ultra high-definition video
• AR/VR Media
• Immersive Gaming
5G ADVANCED
Massive Internet of Things
• Ultra-high density (up to 200,000/km^2)
• Ultra-low energy (battery upto 30 years)
• Low Cost
• Adaptive Data Rates (1 to 100 Kbps)
• Use Cases:
• Smart Home and Smart Cities
• Wearables
• Internet of Things
Mission Critical Control
• Ultra-low latency (URLLC - <1ms)
• Ultra-high reliability (99.999%)
• Strong Security
• Adaptive Data rates: (50 Kbps to 10 Mbps)
• Use Cases:
• Autonomous vehicle
• Industry 4.0

Figure 1: 5G-Advanced Technology Overview

factories, augmented reality (AR), and virtual reality (VR). It provides a network for services like AR/VR streaming and 4K video streaming to provide a virtual network that guarantees communication speeds of hundreds of megabits per second (Mbps) to several gigabits per second (Gbps).

Network Slicing will play a vital role in enabling a multitude of 5G applications, use cases, and services. It will provide end-to-end isolation between slices with the ability to customize each slice based on the service demands (bandwidth, coverage, security, latency, reliability, etc.). Maintaining the isolation of resources, traffic flow, and network functions between the slices is critical in protecting the network infrastructure system from Distributed Denial of Service (DDoS) attacks. The 5G network demands and new feature sets to support ever-growing, complex business requirements have made existing approaches to network security inadequate.

The evolution of the 5G network has opened an arena of possibilities and capabilities unavailable in the 4G/LTE network. The critical aspect of 5G wireless digital transformation will enable the network operators to move their network functions into virtualized core and cloud, leading to new vertical use cases for businesses, enterprises, and consumers. The growing opportunities have resulted in a race among the network operators and service providers to deploy network-slicing functions. Network Slicing provides ease of operation and flexibility to create multiple logical networks on top of a commonly shared physical network infrastructure. Network Slicing will also allow orchestrating a dedicated end-to-end network for specific applications at scale while maintaining their respective service demands and needs.

Existing automation features like Network Function Virtualization (NFV), Software Defined Networking (SDN), and Network Slicing in 5G are the solutions to generate new sources of revenues and reduce the operation cost incurred by a single core network for various services. These features also aim to increase the elasticity and efficiency for scaling new business demands from IoT, Public Safety, Automotive, and Healthcare applications. For example, Massive Internet of Things (IoT) devices require high reliability with limited data rates and low latency for connectivity of smart electric meters and smart city sensors. Meanwhile, augmented and virtual reality (AR-VR) applications require high throughput and low latency. In contrast, mission-critical services require ultra-reliable low latency and high bandwidth in case of emergencies.

1.3 Research Significance and Accomplishments

Effective slicing of the Radio Access Network (RAN) remains extremely difficult due to network dynamics, isolation of network slices, and diverse Quality of Service (QoS) requirements of various services introduced in 5G. Moreover, radio resource management poses technical challenges for network slicing, given the scarcity of radio resources and the limited spectrum. Consequently, practical and dynamic RAN slicing will introduce unprecedented network complexity, rendering the conventional mode-based approach intractable and ineffective. The stochastic nature of wireless networks, multi-dimensional QoS requirements of services, highly dynamic service traffic, and inevitable limitation of resources available in networks contribute to not only impact our ability to study or view conceptual problems but also impede our ability to obtain better solutions or even feasible solutions in some practical situations. With a large influx of data in B5G systems from end-users and network elements, it is imperative to understand how data is collected and used for real-time data processing operations.

Resource sharing can be implemented using either partition-based sharing or elastic sharing. Due to the changing nature of the network load, dynamic resource sharing among slice tenants increases the efficiency of network resource consumption. There are specific problems that need to be resolved in resource sharing. Radio resources, for instance, can be shared amongst RAN slices. Allocating radio resources among these slices requires an efficient radio scheduling algorithm. Additionally, pooling computational resources and other resources must be considered.

Future 5G network operators are concerned with how to manage network resources to maximize their benefits. In this scenario, network slice life-cycle management is a crucial issue that must be resolved. To satisfy the most significant number of requests for various services, 5G network operators must develop virtual network functions and rapidly assign network resources to form network slices. In addition, they should be able to scale slices based on the fluctuating service traffic dynamically. On the other hand, although a network operator has the most control over his network slices, a slice may need to exercise control over itself to improve service quality. Therefore, the network slicing technique must examine how to grant partial permissions to each slice for configuration and management without causing security concerns. In addition, network slice management must be implemented automatically to eliminate manual effort and errors.

This research proposed a novel resource management framework for network slicing architecture in B5G systems, realized through the Classical ML, Statistical, DLNN and Transfer Learning-based data-driven methods. The developed framework *DeepSlice*, *Secure5G*, *ADAPTIVE6G*, and *ECO6G* considered load from network slices to forecast the total traffic demand and enabled network operators to configure slice resource automation more precisely, resulting in better management of network resources by avoiding excessively over-provisioned or under-provisioned resources in B5G systems. The simulated results demonstrate a considerable performance improvement and reduced error using transfer learning compared to a traditional neural network and classical ML algorithms.

In a larger scientific and socioeconomic context, the proposed research and developed models would accelerate the deployment of new services and applications over 5G and Beyond networks, which have not yet been conceptualized and will have huge societal benefits. The proposed research will aid value in 5G-Advanced and 6G development, and a few significant contributions are as follows.

- network slice selection while maintaining slice isolation and service requirements.
- mapping slices to a physical infrastructure while providing availability guarantees despite infrastructure failures.
- strong recovery of virtual and physical infrastructure components following catastrophic failure events.
- preventive measure against volume attacks, flooding attacks, and masking (spoofing) attacks.
- secure framework for authentication, authorization, availability, trust, and control for network slicing architecture in 5G systems.
- adaptive modeling for network load estimation using transfer learning
- predictions to manage the heterogeneous traffic and resource usage patterns of the different slices and develop energy-aware resource management for network load.
- evaluate benefit-cost-analysis to operate network in B5G systems

1.4 Organization

In chapter 2, I present the proposed *DeepSlice* and *Secure5G* framework for network slicing, which discusses the adoption of DLNN for network slicing management in B5G systems. The *ADAPTIVE6G* framework is presented in chapter 3, which introduces the concept of Transfer-learning and the findings of my research on the adaptiveness of network slicing for load prediction. Chapter 4 presents the optimization formulation for energy-efficient network slicing in B5G networks and our proposed framework *ECO6G*. The evaluation is compared with the classical ML and statistical model. Finally, the conclusion and future work is discussed in chapter 5.

CHAPTER 2

A DEEP NEURAL NETWORK FRAMEWORK TOWARDS A RESILIENT, EFFICIENT, AND SECURE NETWORK SLICING IN BEYOND 5G NETWORKS

2.1 Introduction

Mobile communication has become an essential part of human lives. The number of mobile devices has been exponential over the past two decades, where newer services and applications play the role of a catalyst. This change has led to a need for higher capacity and throughput in the network and requires close integration of multiple different technologies. The previous two generations of mobile networking have focused on mobile broadband communications, bringing faster speeds to more devices at lower prices. As the 5G nears its midpoint, the network is expanding in multiple new directions to bring 5G capabilities to new vertical industries and markets and support new types of devices with modest data-rate requirements. However, seamless operations and management have always been a challenge for heterogeneous wireless networks, but many service providers have worked their way through to meet customer demands.

5G networks are seen to be multi-service networks with a wide range of operations embedding diverse performance and services, which calls for a broader device ecosystem. B5G will enable a richer mobile experience, whether it is mobilizing media and entertainment, highspeed mobility, immersive experiences, augmented reality, or connected vehicles in the congested network environment. Our work integrates Deep Learning (DL) methods to understand traffic requirements and make accurate decisions in 5G networks.

Networks have evolved with the introduction of programmable systems like SDN and NFV and have benefited since their implementation. Some critical services that 5G networks would encapsulate are autonomous driving, enterprise business models, AR-VR solutions, industrial automation, remote monitoring, smart health, smart cities, and many more. The Third Generation Partnership Project (3GPP) considers network slicing a critical enabling technology for 5G. Slicing would allow operators to efficiently run multiple instances of the network over a single infrastructure for serving various applications, use cases, and business services with superior Quality of Service (QoS).

5G New Radio (NR), the global standard for 5G networking, is the first generation of wireless communication systems to use high spectrum and the vast bandwidth of frequencies above 24 GHz, known as millimeter Wave (mmWave), to transfer data faster over mobile connections. 5G mobile communication is also designed for spectrum bands below 3 GHz and mid-band between 3 GHz to 6 GHz. 5G NR aims to address various cellular-driven applications and systems, driving different frontiers to produce considerably higher efficiency and unprecedented cost, energy, and usage efficiency rates. Enhanced Mobile Broadband (eMBB) will allow a user to experience not only higher throughput but will also create use cases and content for augmented and virtual reality (AR/VR) to deliver immersive entertainment and experiences. 5G will create a path for Mission-Critical communications and industries requiring ultra-reliable and low-latency links like autonomous vehicles, medical applications, and Industry 4.0 infrastructures. 5G will enable Massive Internet of Things, a virtual world of billions of connected devices, by offering low cost, scale-down data rates but with full mobility functionality and future proof for services that are unknown today.

The 5G wireless industry is creating new approaches to solve optimization problems such as capacity forecasting, traffic estimation (over-provisioned, under-provisioned), scheduling of network resources, and planned maintenance by utilizing the available data points from the production network and users. These Key Performance Indicators (KPIs) are generated and collected by billions of connected UEs and RAN elements through traditional ML and DL approaches.

Data analysis is not new in wireless networks. In general, the industry has adopted a centralized ML approach in the current implementation, where the data points and ML training have been conducted in a central entity, i.e., a central server. Suppose the network can make reliable, dynamic, and faster decisions through the trained model (without having to train on the larger dataset every time). In that case, this will result in a reliable and better quality of experience (QoE) and improve low latency communications with faster response times. Additionally, the importance and growth of on-device intelligence are transformational and essential if we fully realize the benefits of our 6G future. Improved device experiences, such as smarter beamforming, better power efficiency, reduced interference, and better spectrum utilization, are a few examples of improved system performance through which the next-generation networks can be optimized.

5G supports services with vastly different requirements to optimally serve various verticals, such as enhanced mobile broadband, massive machine-type communications, and ultra-low latency applications. Network Slicing is essential in this expanding envi-

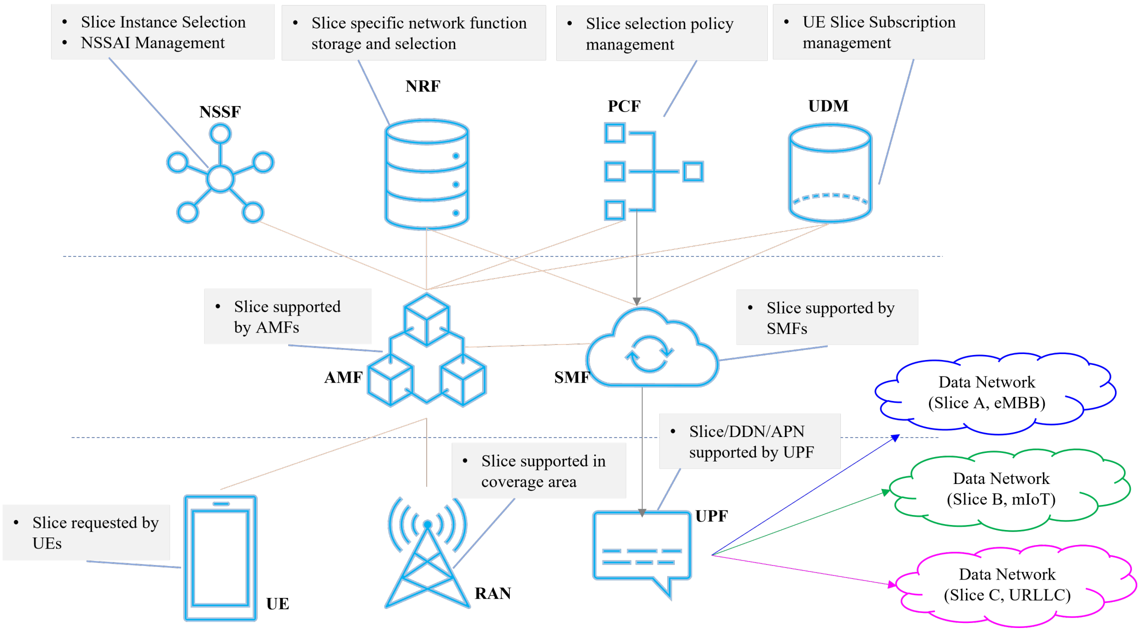

• Slice Instance Selection
• NSSAI Management
• Slice specific network function storage and selection
• Slice selection policy management
• UE Slice Subscription management
NSSF
NRF
PCF
UDM
• Slice supported by AMFs
AMF
SMF
• Slice supported by SMFs
Data Network (Slice A, eMBB)
• Slice/DDN/APN supported by UPF
• Slice supported in coverage area
Data Network (Slice B, mIoT)
• Slice requested by UEs
UPF
UE
RAN
Data Network (Slice C, URLLC)

Figure 2: 3GPP based 5G Network Architecture Systems

ronment, as it provides a flexible, secure, and scalable solution for optimizing network configurations for virtually any service capacity. Network Slicing offers the concepts and tools essential to deploy multiple virtual networks on the same infrastructure. They are utilizing software-defined networks and network function virtualization as foundational concepts. Network slices enable highly flexible, efficient, customized network deployments for any 5G service.

A network slice in the B5G network is a logical virtual network on the core physical infrastructure that can be dynamically configured to have complete network functionality and resources. Slicing allows Mobile Network Operators (MNOs) to serve various vertical applications based on customer service level agreements and requirements. Network slices contain dedicated and shared Control Plane (CP), and User Plane (UP) network functions along with Core Network (CN) and Radio Access Network (RAN) resources intending to provide end-to-end isolation (or at least protected and dependable performance) for both data traffic and network function resources. Network slicing enables an MNO to manage network resource allocation and resource utilization (e.g., bandwidth, physical resource blocks, load, and capacity). It enables a flexible UE-based subscription model to serve customers. At the time of writing this paper, mobile operators worldwide have yet to deploy network slicing in the commercial network; many of them are in trial and proof of concept phases with goals to commercialize in 2023.

When providing E2E network slicing, the CN and RAN perform slicing-related functions following the 3GPP standard architecture, where the UE during the initial registration or mobility registration updates will trigger the request for a network slice instance. The RAN routes the request to the default or appropriate Access and Mobility Management Function (AMF) in the core network, as shown in Fig. 2. AMF is unique to each UE and is shared by all network slice instances that serve a UE. The UE can request up to eight Single Network Slice Selection Assistance (S-NSSAI) in the registration request based upon its UE-type subscription and application usage. That means the UE can have eight network slices simultaneously (though only three are defined in standards, others can be operator-defined slices). Furthermore, depending upon the core network support and subscription parameters, the network may accept registration for all or some or none of the requested S-NSSAI. The AMF assigns the slice allowed by the user subscription and interacts with the Network Slice Selection Function (NSSF) for the appropriate slice assignment for that UE.

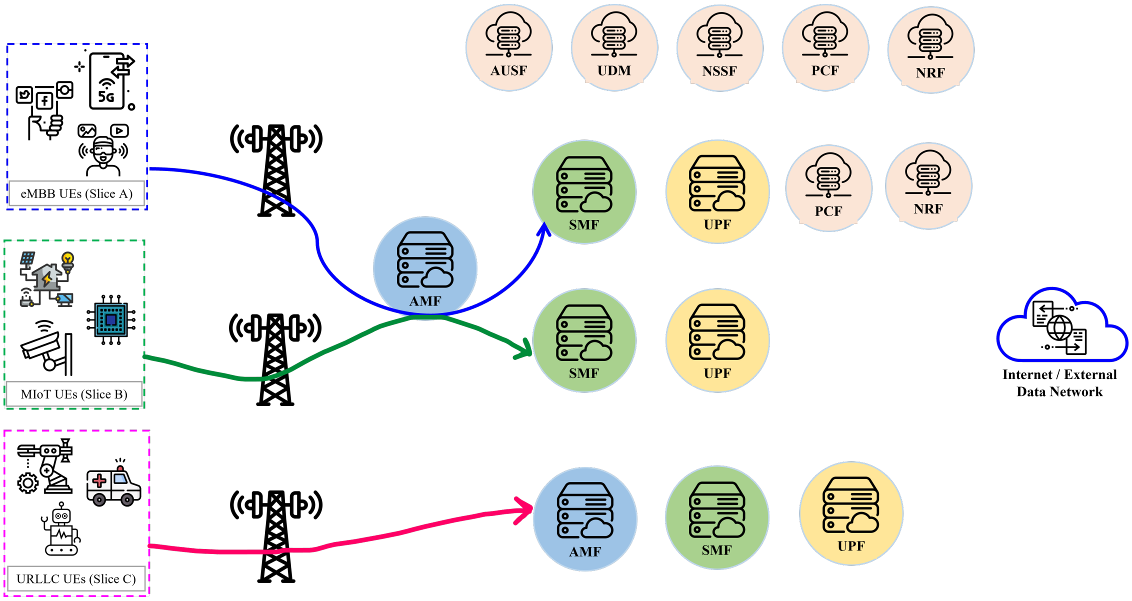


Figure 3: Network Slicing Architecture in B5G Networks

The Network Slicing function, S-NSSAI, comprises the Slice/Service Type (SST) and the Slice Differentiator (SD). The SST identifies the slice type, while the SD differentiates the SST among the slices. The 3GPP has standardized three SST values for global deployment and interoperability: eMBB (SST =1), URLLC (SST = 2), and mIoT (SST = 3). We have considered these SST values as use cases for this paper to classify slices. The other terminologies used in Fig. 3 are Session Management Function (SMF), Unified Data Management (UDM), and User plane Function (UPF), each of them serving different aspects of network slicing call flow which is used to establish end-to-end user plane connectivity between the UE and a particular Data Network (DN) via the UPF.

5G-Advanced will significantly enhance network slices' configuration, management, and control, enabling network operators to offer their customers the most granular service levels. In 3GPP Release 18, standards will define new features that provide service continuity across network slices, enable operators to configure slices for specific geographic areas or zones, extend slice functions across roaming partners' networks, and provide much finer control over how individual devices use slicing services. It is expected that MNOs will implement Network Slicing in phases. Each slice service will be offered to a diverse set of customers (public, private, hybrid) and will follow a decentralized approach. For example, a Private Network Slice can have a different network configuration model where resources can be decentralized and not managed by a single entity. These types of scenarios are more decentralized, even more so with the 5G Service-based Architecture, as MNOs are undoubtedly capable of establishing an isolated network of slices as required. Our decentralized approach for traffic forecasting from each slice could greatly benefit MNOs in managing and maintaining these customized and isolated networks.

The telecom industry is going through a massive digital transformation with the adoption of ML, AI, feedback-based automation, and advanced analytics to handle next-generation applications and services. AI concepts are not new; the algorithms used by ML and DL are currently being implemented in various industries and technology verticals. With growing data and an immense volume of information over 5G, the ability to predict data proactively, swiftly, and accurately is critically important. AI will enable network functions to deliver ultralow latency, higher throughput, and reliability by optimizing network performance and improving QoE.

2.2 Related Work

Authors in [1] explore the multi-tenancy nature of the 5G network slicing by demonstrating how the capacity of a Mobile virtual network operator (MVNO) is affected by the number of users and transmit power. SDN and NFV-based 5G core network architecture is defined in [2]. Du and Nakao propose an application-specific mobile network DL architecture to apply application-specific radio spectrum scheduling in the RAN [3]. Authors in [4] propose a framework to prioritize network traffic for smart cities using a priority management SDN approach. Authors in [5] started work early on network slicingSINR standardization of network slicing, network slice selection, identifying slice-independent functions, and then proposed architecture for slicing and the RRC frame.

No other work to our knowledge considers the easily overlooked but the complex problem of deciding which devices and connections should be assigned to which network slices. And our work here is the first to use DL to address this problem, which will provide the benefits of fast, flexible, accurate, and informative decision-making. The authors in [6] contrast Fade Duration Outage Probability (FDOP) based handover requirements with the traditional SINR-based handover methods in cellular systems. Another SDN and NFV-based work on slicing demonstrate dynamic data rate allocation and the ability to provide hard service guarantees on 5G new radio air interfaces [7]. Many industry white papers and network surveys have been published. An Ericsson mobility report predicts the growth of mobile devices, 5G network connections, and overall data usage in coming years. As for network intelligence, the authors in [8] represented handovers using B matrix distributions for public safety and emergency communications, which helps make handover decisions more accurate considering all the different parameters involved in the decision process.

Authors in [9] present network survivability framework in 5G networks demonstrating network virtualization with multiple providers, which necessitates network slicing in 5G. Virtualized networks or slices of virtualized networks are selected and assigned based on QCI and security requirements associated with a requested service in [10]. Campolo et al. share their vision about V2X network slicing by pinpointing essential needs and providing design guidelines aligned with ongoing 3GPP standard specifications and network softwarization directions in [11]. The proposed model in [12] enables a cost-optimal deployment of network slices, allowing a mobile network operator to efficiently allocate the underlying layer resources according to its user's requirements. However, their work needs to consider the possibility of multiple service requirements requested by the same device, especially those requested by an unknown device. Also, network slice load balancing and future prediction of traffic is unique in our work, especially with the use of ML and DL neural networks.

A mathematical model that can provide on-demand slice isolation and guarantee end-to-end delay for 5G core network slices is proposed in [13] to proactively mitigate Distributed Denial-of-Service attacks in 5G core using slice isolation. The network slices relying solely on common infrastructure cannot meet the highest isolation requirements. Therefore, authors in [14] introduce different novel provisioning models for 3rd-party slices and discuss their isolation properties. Authors in [14] propose an efficient and secure service-oriented authentication framework supporting network slicing and fog computing for 5G-enabled IoT services. They also introduced a privacy-preserving slice selection mechanism to preserve both configured slice types and accessing service types of users. [8] proposes a 5G network architecture framework with network virtualization among multiple providers, and a self-organizing ad hoc network among the eNBs that may use another provider for network resilience when the aggregation network and the backhaul network fail.

Three different models are demonstrated in [15] using the CoAP and MQTT application protocol, which aims at providing efficient mechanisms and methods for over-the-air (OTA) delivery of software updates and security patches to IoT devices. The authors also evaluate which protocol suits proposed models and applications better. [16] applies a deep auto-encoded dense neural network algorithm for detecting intrusion or attacks in 5G and IoT network for flooding, impersonation, and injection type of attacks. UE capability, RRC messages, and measurement reports [17] and [18] can be very helpful in identifying irregular behavior by identifying changes in device power and thermal consumption. With Secure5G, the proposed work is to utilize the UE Capability Enquiry and UE Capability Information messages along with the measurement of neighboring cells to identify the non-3GPP access or unauthorized base stations. Upon detection, the network can flag the false base station as a threat and direct another UE's to not connect to the identified false base station.

2.3 Research Contribution

In this research work, we have developed a ***DeepSlice*** and ***Secure5G*** model by implementing Deep Learning Neural Network (DLNN) to manage network load efficiency, network slice selection, network availability, and security of network slices utilizing in-network deep learning and prediction. We have used available network Key Performance Indicators (KPIs) to train our model to analyze incoming traffic and predict the network slice for an unknown device type, as shown in Fig. 4. Intelligent resource allocation allows us to efficiently use the available resources on existing network slices and offer load balancing. Our proposed *DeepSlice* model will be able to make smart decisions and select the most appropriate network slice, even in case of a network failure. Furthermore, it proposed a Neural Network-based *Secure5G* Network Slicing model to proactively detect and eliminate threats based on incoming connections before they infest the 5G core network.

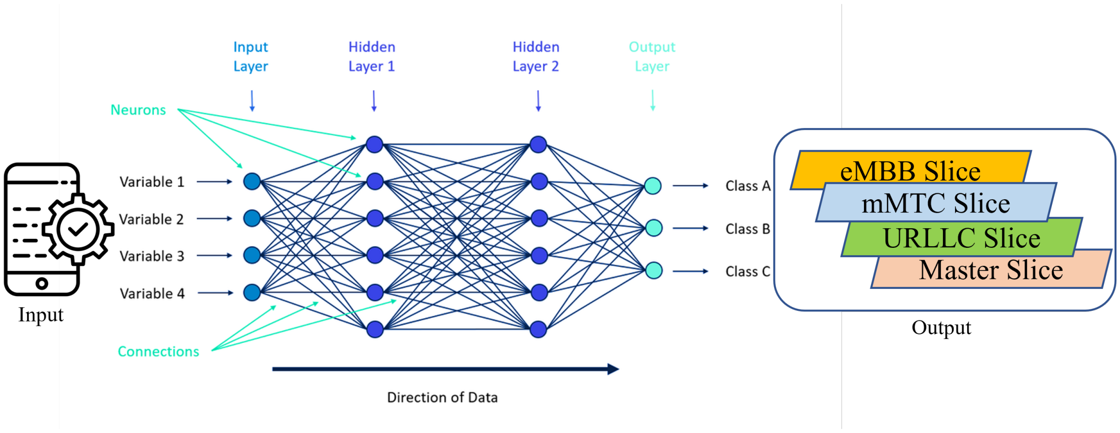


Figure 4: DLNN Model Overview

The following specific research issues are addressed through *DeepSlice* and *Secure5G*:

- optimal control and appropriate management of network slice selection for incoming requests.
- optimizes the network and individual slice load efficiency across isolated slices.
- strong recovery of network slicing infrastructure following catastrophic failure events.
- authentication, authorization, availability, trust, and control for network slicing architecture in 5G systems
- prevention measure against volume attacks, flooding attacks, and masking (spoofing) attacks strategies towards the zero trust objective (never trust, always verify, verify continuously) for securely operating on 5G networks that provide security and improve availability and resilience.

2.4 Network Slice Resource Management - *DeepSlice* Model

The legacy 4G LTE architecture has a rigid framework that could be more flexible and scalable to adapt to various use cases. It often needs more customization when offering tailored business requirements or meeting specific business demands. With growing mobile data and consumer demands, business needs for faster connectivity and higher throughput cannot be fulfilled by today's 4G LTE network. Network slicing in 5G can deliver multiple logical networks cost-effectively over the same physical infrastructure. SDN and NFV would allow us to manipulate these slices as and when needed without touching multiple physical equipments in the network. Almost 'no-disruption' to any existing services is possible. Currently, service providers must configure and stitch together several components and equipment to achieve network slicing in 4G. Use of Access Point Name (APN) or Public Land Mobile Network (PLMN) are examples that service providers implement today for Mobile Virtual Network Operators (MVNOs), enterprise customers, etc. Much work is done on optimizing and efficiently scheduling radio and network resources; however, application or service-based resource allocation is necessary and a must-have feature in 5G networks.

Operators have a huge amount of data traffic coming through their network, which will increase with a growing number of devices and additional services of 5G networks. This traffic can be segmented and dealt with individually and independently. It will benefit any service provider as they can now bill differently for each sliced segment and adjust the cost for each slice, leading to a balance between business profitability and customer satisfaction. In addition, 5G network slicing allows service providers to build for all current use cases that have been r a while and some emerging applications and services. It will provide a ‘one size fits allpieces of equipment each. Each network slice can be isolated and have individual control and policy management systems.

Including DL and ML here will allow us to analyze any unknowns and take necessary corrective actions. ML will provide network analysis of the huge data, which can be studied further to efficiently and cost-effectively modify any given slice as needed. DL will provide, process, and make intelligent decisions for network resource adaptation without human intervention. It will also combine various factors to make the best decisions, possibly too many factors for a human to consider at once or even be able to process quickly.

DL will perform real-time analysis for any given slice to determine the network performance, create a potential baseline for performance, be proactive in anticipating problems, inspect different network elements, and find out if anything is abnormal. A simple example could be on a slice for a fixed wireless enterprise network. If the network sees a sudden demand increase, automation can add more capacity in real-time to provide efficient communication. This will help to create any newly required services or slices in the network. Automation will facilitate all this quickly without causing performance issues in an ongoing session. Current hurdles in the implementation of network slicing are organizational, as one will have to touch several pieces of hardware and groups in a service provider network to make a single change. The programmability capabilities of 5G will provide flexibility to seamlessly stitch together an end-to-end service for any application. A typical consumer would request parameters like data rate, latency, mobility, isolation, power constraints, etc. Accordingly, a specific network slice type is provisioned if the existing network slice instance does not have enough capacity and associated network functions are initiated on demand.

With Network Slicing, each use case receives an optimized set of resources in the network topology covering several SLA-specified factors like connectivity, latency, priority, service availability, speed, capacity, etc., that suit the need of an application. The key parameters for network slicing are the slice type, bandwidth, throughput, latency, equipment type, mobility, reliability, isolation, power, etc. 5G enables enormous amounts of data collection, leading to the need for ML for big data analytics. Some of the most relevant and useful ML-based applications in the wireless industry are identifying and restarting sleeping cellular cells, optimizing mobile tower operations, faster wireless channel adoption, facilitating targeted marketing, autonomous decision-making in IoT networks, real-time data analysis, predictive maintenance, customer churn, sentiment analysis by social networking, fraud detection, e-commerce, etc. ML implementation in Uber-like applications will have many advantages since Uber follows differential pricing in real-time based on the demand, cars available, weather conditions, rush hour, etc. The ML-based platform will allow for better accuracy and future prediction based on enormous past and present data.

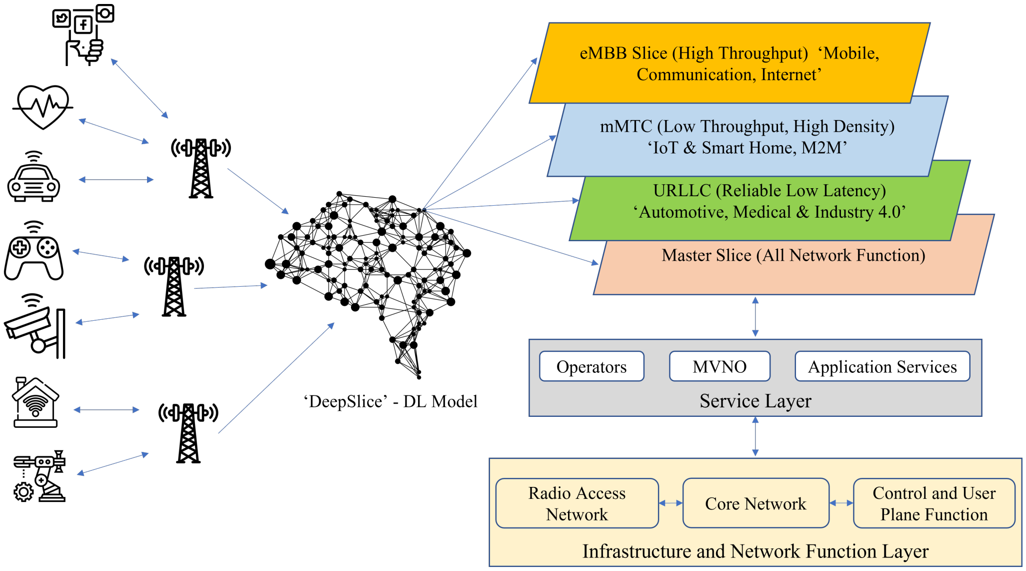

eMBB Slice (High Throughput) 'Mobile, Communication, Internet'
mMTC (Low Throughput, High Density) 'IoT & Smart Home, M2M'
URLLC (Reliable Low Latency) 'Automotive, Medical & Industry 4.0'
Master Slice (All Network Function)
'DeepSlice' - DL Model
Operators
MVNO
Application Services
Service Layer
Radio Access Network
Core Network
Control and User Plane Function
Infrastructure and Network Function Layer

Figure 5: Proposed DeepSlice Model

Fig. 5 shows the high-level overview of our developed *DeepSlice*. In the proposed model, we predict the network load of each network slice based on the incoming connection and keep track of which output 'network slice' is utilized most. We then allocate incoming devices to slices by efficiently distributing between the eMBB, URLLC, mMTC, or the master slice, depending on the load and the output predicted by our model. We have used Keras, a Python deep-learning library, for our model simulations. A DLNN is required as there are no clear rules for how each incoming device type should be treated. Cellular handovers, for example, are based on several network factors. With every new scenario, an intelligent network can learn and adapt quickly to changes or new requirements compared to traditional algorithms. DLNN can help identify and accommodate the unknowns in the network.

The DLNN works best when the data is unstructured and huge. We use the same dataset to train multiple neurons of our DLNN, and it predicts the correct network slice based on any input from the UE information. Our DLNN can predict very accurately, and we utilize this functionality to select the correct slice for unknown device types. It helps redirect traffic to the Master slice if load balancing is required in the network slices and in case of any slice failure. Neural networks are widely used in the industry today, and their usage will only grow as the ever-growing devices on 5G networks generate massive data. Accurate analysis and decision-making will be overwhelming for any human being, and faster processing times are required. We first create an ML model and later build a DLNN to help decide which network slice to use for given input information.

The developed *DeepSlice* is then used to manage network load, slice failure conditions, and detect the most appropriate slice for any new unknown device type connecting to the network. A statistical ML model is based on the Random Forest (RF) algorithm, and the DeepSlice uses a convolutional neural network (CNN) classifier. Both RF and CNN are widely used models in their respective domains. We use the same dataset for our ML and DLNN models consisting of over 65,000 unique input combinations. Our dataset includes the most relevant KPIs from both the network and the devices, including the type of device used to connect (Smartphone, IoT device, URLLC device, etc.), User Equipment (UE) category, QoS Class Identifier (QCI), packet delay budget, maximum packet loss, time and day of the week, etc. These KPIs can be captured from control packets between the UE and the network. Since our model will run internally on the network, all this information is readily available.

Table 1: Dataset inputs for *DeepSlice* and *Secure5G*

Input Type	Duration	Packet Loss Rate	Packet Delay Budget (ms)	**Predicted Slice**
Smartphone	300	10^{-2}/10^{-3}/10^{-6}	50	eMBB/mMTC
IoT Device	60	10^{-2}	50	mMTC
Smart Transportation	60	10^{-6}	10	URLLC
Industry 4.0	180	10^{-2}	300	mMTC
AR/VR/Gaming5	600	10^{-2}	100	eMBB
Healthcare	180	10^{-6}	100	eMBB
Public Safety / E911	300	10^{-6}	100	eMBB
Smart City / Home	120	10^{-6}	100	eMBB
Unknown Device Type / Home	60/120/180/300	10^{-6}	100	eMBB

We have multiple different types of input devices requesting access to our system. As shown in Fig. 5, these include smartphones, available IoT devices, AR-VR devices, Industry 4.0 traffic, e911 or public safety communication, healthcare, smart city or smart homes traffic, etc., or even an unknown device requesting access to one or multiple services. These have UE category values defined for them, and the network also allocates a pre-defined QCI value to each service request. In 5G, the packet delay budget and the packet loss rate are an integral part of the 5QI (5G QoS Identifier), and we have them included in our model. DeepSlice will also observe what time and day of the week the request is received in the system. All this information will be recorded and used by our DLNN to make smart decisions and efficiently predict future network resource reservations.

In Table 1, we have shown highlights of the features of our simulation model. The second column shows the average time spent in the system by each incoming request. All these incoming requests are directed to one or more of the network slices as predicted. We have also considered some variations in the traffic types; mMTC devices can be further categorized as ones requiring a continuous connection link and others needing only a momentary connection to send data periodically. Every day, users can use smartphone devices to make phone calls, browse the web, and at the same time, first responders in an emergency (lower packet loss and packet delay). Our pre-defined slice categories include enhanced Mobile Broad Band (eMBB), Ultra Reliable Low Latency Communication (URLLC), massive Machine Type Communication (mMTC), and the Master slice. The Master slice is the slice that will have network functions belonging to each of the other slices. It can always act as a backup slice in a hot standby and will be used depending on the load on other slices.

2.5 *DeepSlice* Use-cases

2.5.1 Objective 1: Identifying Unknown Incoming Connection

The DeepSlice model is trained using multiple unique inputs based on network and device KPIs. Our cross-validation accuracy was close to 95% as shown in Fig. 6, which included the entire test dataset of new input scenarios, those not used while training. At the same time, the classical ML approach using Random Forest shows 93.70% accuracy. Our training dataset included 6 to 8 parameters in every input. However, our model requires a minimum of two or three input KPIs to determine the services requested and allocate the correct slice.

After training, we evaluated *DeepSlice* to predict network slice selection for *unknown device* types. The unknown device types are new types or devices that are not yet known or that the 5G networks have not seen. As 5G will bring plenty of new use cases, new hardware, sensors, and devices will connect to the network and almost certainly re-

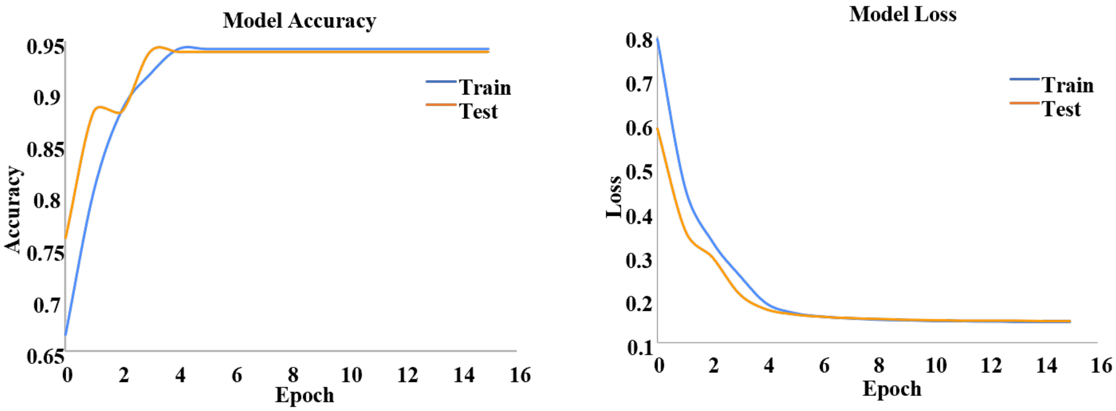


Figure 6: Unknown Slice Prediction Accuracy

quire one of these slices. Using *DeepSlice*, we can predict and co-relate the device using learned behavior and device metadata and assign appropriate slices, inline ML model. This is essential since many devices with various capabilities request different services at different times, and it is hard to rely on signature-based modeling for devices not built yet. An industry 4.0 IoT application requires very low latency (URLLC). In contrast, the same type could also be used to monitor production lines, requiring periodic connection and very low throughput (mMTC). We also included certain unknown device types with randomly selected parameters.

Table 2: Unknown Inputs for Predicting Network Slice

Input Type	Technology	Packet Loss Rate	Packet Delay Budget (ms)	**Predicted Slice**
Unknown - 1	LTE/5G or IoT	10^{-3}	50	eMBB/mMTC
Unknown - 2	IoT	10^{-2}	50	mMTC
Unknown - 3	IoT	10^{-6}	10	URLLC
Unknown - 4	IoT	10^{-2}	300	mMTC
Unknown - 5	LTE/5G	10^{-2}	100	eMBB
Unknown - 6	LTE/5G	10^{-6}	100	eMBB

We demonstrated an accurate Network slice prediction for unknown devices. Table 2 shows a few unknowns and how only a portion of input information was used to determine the network slice to be used correctly. Thus, this evaluation established a framework to optimally control and manage appropriate network slice selection for incoming requests from diverse applications and services.

2.5.2 Objective 2: Network Slice Load Balancing

We evaluated *DeepSlice* DLNN model for a load balance mechanism to distribute slice loads among base stations or within network slices. The B5G network needs to be swifter and smarter in using its resources, it should balance the load and prioritize its customer's needs in near real-time, and it should all happen without human intervention. Fig. 7 shows the simulated traffic using *DeepSlice* model. We generated approximately a quarter million user connection requests in a twenty-four simulation, of which 40% was eMBB, 25% mMTC, and 35% URLLC. The simulated DLNN model gives the number of users served in twenty-four hours. The plot began when our model reached a steady state at the one-hour mark. Based on the Table 1 information, all incoming traffic has a pre-defined time-to-live (TTL), so only a fraction remains alive every second. For example, the eMBB active user average count was 275 at any given instance. URLLC and mMTC users were allotted short TTL compared to the eMBB, which is why we have more users alive for broadband services. This can help analyze the user pattern and will allow for automated decisions based on the retrieved input information from the connected device.

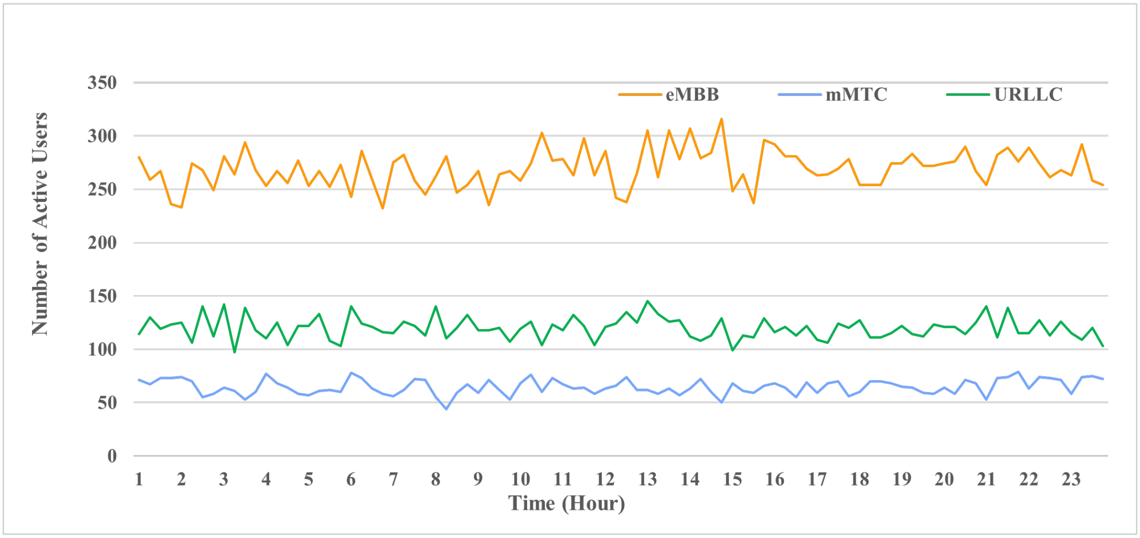


Figure 7: Simulated Network Traffic using DeepSlice

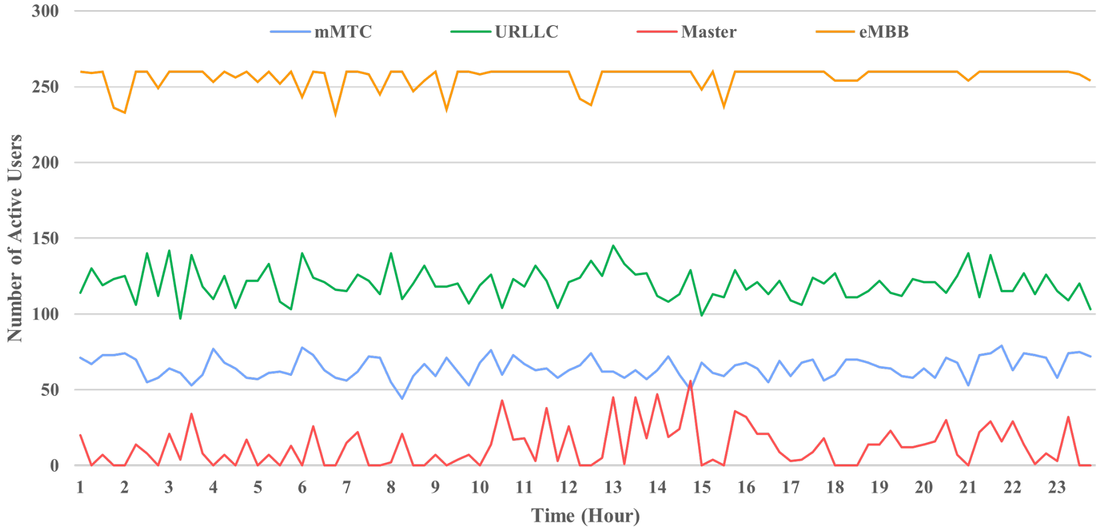


Figure 8: Load Balancing across Network Slices

As shown in 8, as afternoon and evening peak hours hit, the demand increases, and users demand more bandwidth, creating a congestion-like environment in the network. Now the network must decide whether to lower the QoS profile of users within the same slice, move users to a new slice with the same QoS, or create a new network slice for latency-sensitive and bandwidth-hungry applications. In short, the network needs to be swifter and smarter in using its resources, it should balance the load and prioritize its customer's needs in near real-time, and it should all happen without human intervention.

For evaluation, we simulated the over-utilization of one slice, i.e., the number of connections exceeded a threshold, 90% usage in our case. Fig. 9 shows an eMBB slice is detected to have over 90% utilization with its traffic to go over the set threshold, so the master slice acts as a backup for any new eMBB connections. Our DeepSlice can

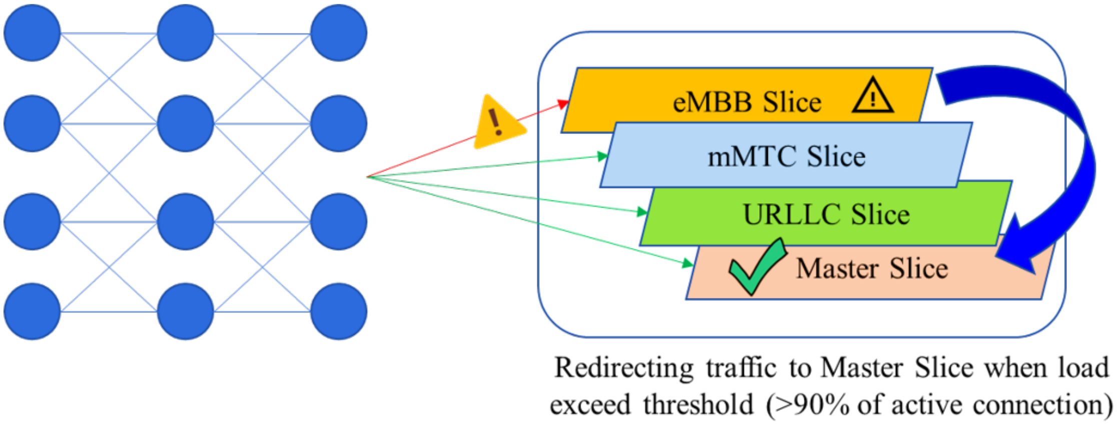

eMBB Slice
mMTC Slice
URLLC Slice
Master Slice
Redirecting traffic to Master Slice when load exceed threshold (>90% of active connection)

Figure 9: Network Slice Utilization exceeding a pre-defined threshold

realize this overload and can be prepared next time to redirect traffic without causing one specific slice to be overloaded. In this case, the master slice takes over the excess traffic, represented by a flat line on eMBB.

2.5.3 Objective 3: Network Slice Failure Scenario

In this case, we assume a complete failure of a specific slice, specifically eMBB, as shown in Fig. 11. Here, the DeepSlice directs all new eMBB related traffic all new eMBB-related traffic to the master slice and avoids any traffic transmission loss in the network. We made the network slice unavailable for a set period and rerouted incoming traffic to the Master slice. The graphs show that mMTC slices failed between 3 and 5 a.m., and eMBB slices failed between 15 and 18. However, any ongoing communication on that slice would be hampered, and all existing connections would be lost due to an unexpected slice failure. The master slice was identified as a backup and used to redirect this traffic during those slice failures. We had substantial resources reserved in the master

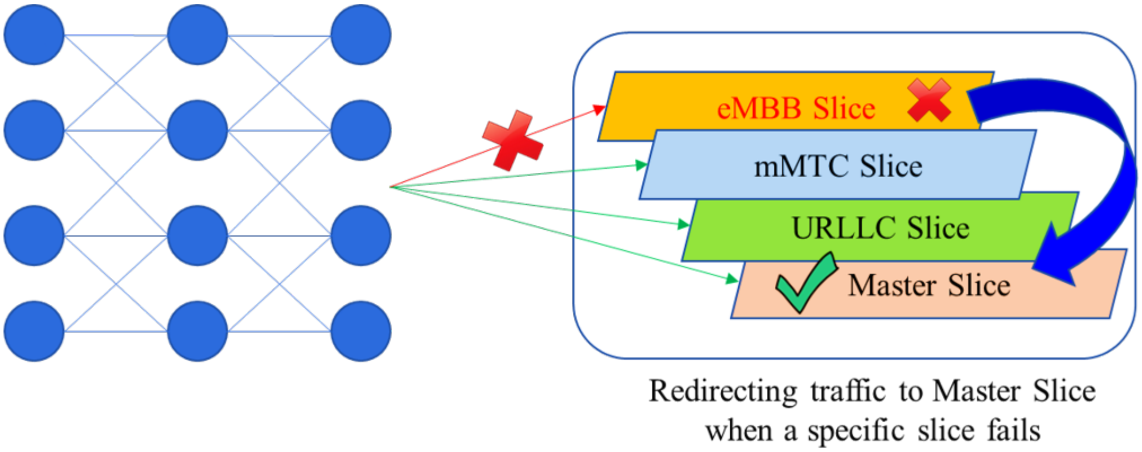

eMBB Slice
mMTC Slice
URLLC Slice
Master Slice
Redirecting traffic to Master Slice when a specific slice fails

Figure 10: Network Slice failure and re-direction to MasterThe graphs show that

slice for each of our network slices in terms of capacity and eMBB resources.

Network slicing in 5G is critical for next-generation wireless networks, mobile operators, and businesses. We have demonstrated the benefits of using DeepSlice for

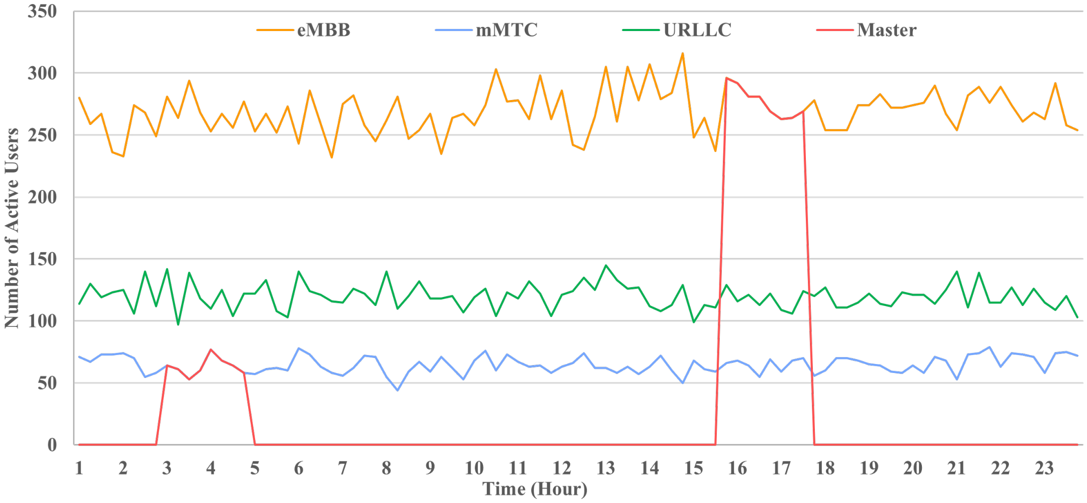


Figure 11: Network Slice failure demonstration

accurately predicting the best network slice and orchestrated the handling of network load balancing and network slice failure using neural network models.

2.6 Network Slice Security - *Secure5G* Model

Network Slicing will play a vital role in enabling many 5G applications, use cases, and services. Network slicing functions will provide end-to-end isolation between slices with an ability to customize each slice based on the service demands (bandwidth, coverage, security, latency, reliability, etc.). Maintaining the isolation of resources, traffic flow, and network functions between the slices is critical in protecting the network infrastructure system from Distributed Denial of Service (DDoS) attacks. The 5G network demands and new feature sets to support ever-growing, complex business requirements have made existing approaches to network security inadequate. In this work, we have developed a Neural Network-based *Secure5G* Network Slicing model as an extension to *DeepSlice* to proactively detect and eliminate threats based on incoming connections before they infest the 5G core network. *Secure5G* is a resilient model that quarantines the threats ensuring end-to-end security from device(s) to the core network and any of the external networks. Our designed model will enable the network operators to sell network slicing as-a-service to serve various services efficiently over a single infrastructure with high security and reliability.

3GPP Rel. 15 has covered several technical specifications; important ones include the new security measurement for rogue, false base stations by masking the permanent subscriber identifier (SUPI) so that the rogue base station cannot track the subscriber in the 5G network. 5G Globally Unique Temporary Identifier (5G-GUTI) is another improvement where UE and RAN have a mandatory requirement to refresh the GUTI from initial registration to mobility registration update. At the time of writing this paper, 3GPP Rel. 16, which is expected to be available mid-2020, and a list of work items being considered for standardization indicates the security requirements for Enhanced Network Slicing, URLLC for 5G Core, and Cellular IoT.

The vision of next-generation 5G networks is to improve the capacity, coverage, security, and connectivity of existing 4G networks. Network operators are designing the mmWave network to meet high capacity demand and relying on Sub 6 GHz 4G/LTE network for coverage. The current release of 5G NR is based on 3GPP Release 15, which takes advantage of sub-6 GHz and above 24 GHz to achieve substantial peak throughput and low latency. Current 5G deployment is on an overlay of the 4G LTE network, and different service providers are following a different approach. Mobile Network Operators (MNO) are deploying or planning to deploy 5G using two architectures - Non-Standalone (NSA) and Standalone (SA). NSA is an evolutionary step for network operators to offer 5G services without building a new dedicated 5G core network. In Non-Standalone, 5G-enabled smartphones will connect to 5G frequencies for data-throughput purposes but still use 4G/LTE for all control plane signaling to the cell towers and servers.

On the other hand, a Standalone will have its dedicated core, and a UE will also be able to use the 5G NR core for control plane signaling. SA will also support the growth of new cellular use cases such as Network Slicing, Control and User Plane Separation (CUPS), Virtualization, Multi-Gbps support, URLLC, and other aspects natively built into the 5G SA Packet Core architecture. We must recognize the threat, and security loopholes with the evolution of these new networks are growing substantially.

The threats and security challenges faced by the 5G ecosystem are the same as those encountered by 4G/LTE today. 5G networks, in addition, will have specific requirements on throughput, latency, and security to meet the service level agreements for diverse applications and services, especially with a diverse ecosystem for IoT devices. Fig. 12 shows today's various threat vectors we classify in the 5G Network. In a typical DDoS attack, a hacker floods the system by sending a huge amount of bad traffic, false ping, and connection requests to the targeted network, making bandwidth and resources

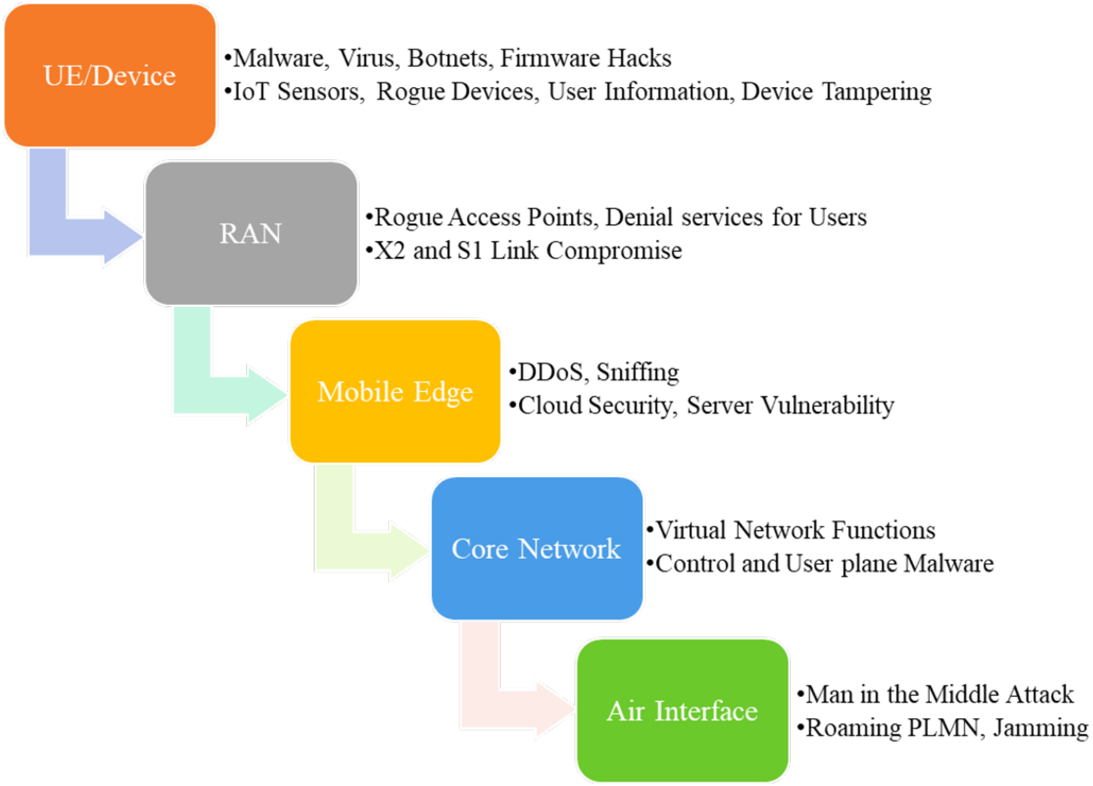

UE/Device
•Malware, Virus, Botnets, Firmware Hacks
•IoT Sensors, Rogue Devices, User Information, Device Tampering
RAN
•Rogue Access Points, Denial services for Users
•X2 and S1 Link Compromise
Mobile Edge
•DDoS, Sniffing
•Cloud Security, Server Vulnerability
Core Network
•Virtual Network Functions
•Control and User plane Malware
Air Interface
•Man in the Middle Attack
•Roaming PLMN, Jamming

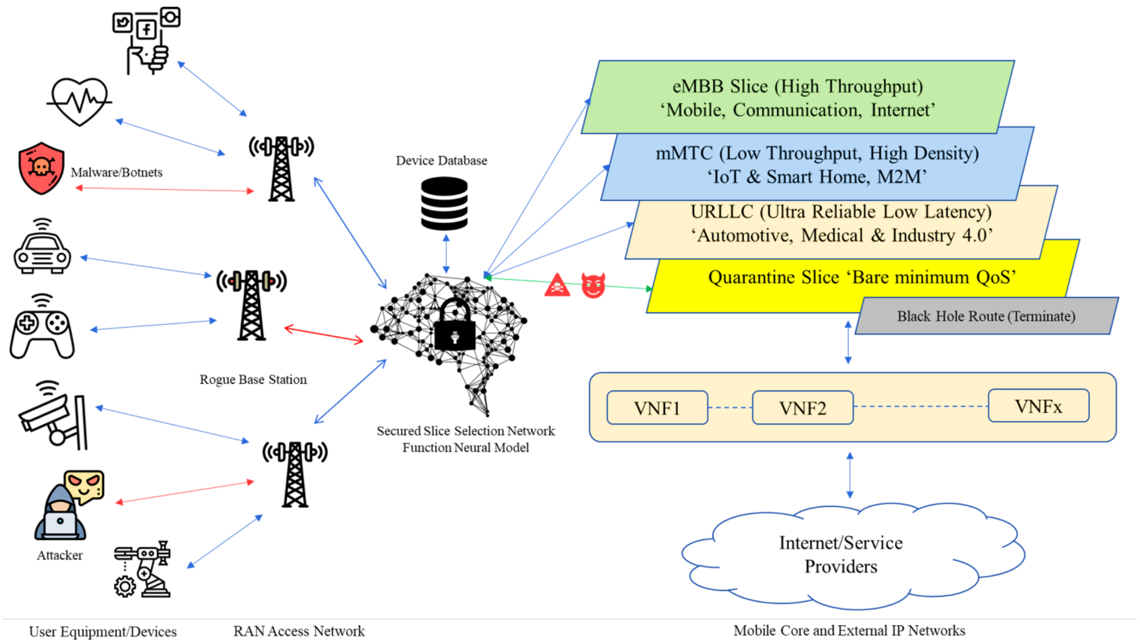
Figure 12: Common 5G Threats Vectors across Device and Network

unavailable or busy for normal traffic. UDP attack, for example, floods random ports on a remote host, making the host server busy and unresponsive; ICMP packets attack ping packets continuously without waiting for replies causing the system to clog; SYN flood exploits the 3-way handshake mechanism and does not close the connection after receiving acknowledgment from the server causing server bottleneck situation. Ping of death, which sends malicious pings with IP packets more than 65,535 bytes in length, is another form of cyber-attack to saturate and overwhelm the website or server and make resources unavailable for normal traffic.

Other network attacks include the International mobile subscriber identity (IMSI) catcher, where the IMSI of a device is sent unencrypted over the radio and eavesdropped on by hackers. The attacks on the user plane and control plane on the core network and radio interface are also common exploitation points for malicious actors. This could lead to data injection, such as the Man-in-the-Middle (MitM) attack. Radio Access Network (RAN) attacks include UE location tracking and malicious message insertions during the initial UE attach procedure. Bring your own device (BYOD) concepts are another threat concern for enterprise solutions, increasing floodgates and data leakage opportunities for attackers. With less control over BYOD devices, systems are more vulnerable to attacks and device tampering. Battery life is a key aspect of the 5G, LTE-M, and NB-IoT technology aims to power IoT devices for a battery life span of up to 30 years by disabling the power-saving abilities of these IoT devices through the injection of malicious code during the initial attack could drain battery drain five to ten times faster than expected life. F-secure threat report 2019 [4] indicates that 99.9% of the attack traffic comes from bots or some automation tool, IoT bot activity represented 78% of the malware network activity (detection events) across carrier networks, more than triple the rate seen in 2016 since the introduction of biggest DDoS attack in history with Mirai botnet. Mirai was a brute force password guessing attack on open telnet and SSH ports by scanning the internet for open Telnet ports, then attempting to log in default passwords.

SDN-based architecture is more prone to malicious attacks than monolithic core architecture because of the network function virtualization, which creates more entry points for attackers to infest into the network. With more slices, multiple network configurations and virtual devices, security and privacy concerns in the cellular ecosystem are at a critical juncture, resulting in the need for secure software and methods to build a robust and secure ecosystem. Attacks on the 5G network could have severe consequences on society in a broader aspect. Network Slicing in 5G can play a vital role in providing dynamic and flexible security architecture for isolated networks optimized for applications with varying needs from a security and privacy perspective by customizing independent firewall configuration(s), security policies, along-with slice-specific authentication schemes.

We have addressed the issue of DDoS attacks in 5G networks from the UE perspective. We have developed a *Secure5G* model based on DL techniques for Networking function in 5G as shown in Fig. 13, with an aim to (1) identify the incoming connection request and assign the most optimal slice based on the device type, (2) verification of the connection request if it is legit or a potential threat, and (3) implementation of an action,


Malware/Botnets
Device Database
eMBB Slice (High Throughput)
'Mobile, Communication, Internet'
mMTC (Low Throughput, High Density)
'IoT & Smart Home, M2M'
URLLC (Ultra Reliable Low Latency)
'Automotive, Medical & Industry 4.0'
Quarantine Slice 'Bare minimum QoS'
Black Hole Route (Terminate)
Rogue Base Station
Secured Slice Selection Network
Function Neural Model
VNF1
VNF2
VNFx
Attacker
Internet/Service
Providers
User Equipment/Devices
RAN Access Network
Mobile Core and External IP Networks

,

Figure 13: Proposed *Secure5G* Model

either assignment of appropriate network slice (valid request) or transfer to the quarantine slice (malicious request). The aim of the *Secure5G* is to mitigate the DDoS initiation attacks by UEs by ensuring UEs can access network slices only after being authenticated and authorized, minimizing the risk of denial of service. *Secure5G* protects the system in case of a Volume-based flooding attack and in the case where hackers mask the device identity and tries to exploit the network slice by requesting system access with low secured slice instance. The *Secure5G* model analyzes the overall traffic pattern and can predict future traffic so that it can allocate resources, in advance, to the most appropriate slice securely.

The *Secure5G* model primarily consists of User Equipment (UE) requesting service to the network for slice and resource allocation. Our model considered diverse input types and applications like smartphones, health devices, autonomous vehicles, AR-VR gaming, smart homes and cities, and Industry 4.0. We also included the Malware botnets and hackers, as shown in Fig. 4. We introduced a concept of a *Quarantine slice* along with the eMBB, mMTC, and URLLC standard slices. Quarantine slice has the network functions for a bare minimum QoS, which allows a device to communicate to the network with a very restricting set of requirements in case of slice attack, slice failures with bare minimum service to serve the user instead of abruptly terminating the connection. The black hole routing concept has been used to terminate the connection permanently after observing the repeated negative traffic pattern of the device(s).

For evaluation, we considered two scenarios: Volume Based Attacks (Flooding) and Masking Botnets. *Secure5G* can observe and learn the device request patterns and assign the best optimal slice, and the model will evolve and be able to predict future traffic patterns and can be used for the capacity forecast. The model also helps prepare the network for assigning slices to unknown (or new) applications or service requests unknown to the network and secure the system in case of a slice DDoS attack or Slice failure.

2.7 *Secure5G* Use-cases

2.7.1 Objective 4: Network Slice Volume Based Attack

In this section, we evaluate *Secure5G* DL model on how it can be used to proactively prevent DDoS attacks on a 5G network based on the incoming network connections before it even reaches the core network. We also observe the traffic pattern and QoS chNetworkingcs during slice allocation to detect anomalies. For instance, a device in an IoT slice whose traffic no longer matches IoT traffic patterns might trigger a warning for a potential attack. *Secure5G* knows if UE is accessing the unauthorized network slicing or requesting unauthorized operation, for example, changing QoS values for prioritizing eMBB over mMTC. The model can also detect abnormal behavior by the subscribed user, for example, requesting access to multiple slices simultaneously repeatedly compared to their previous usage or usage in general from similar devices on that slice.

Per 3GPP specifications, a device can access multiple Network Slices simultaneously; slices can have a diverse configuration, which increases the possibility of security loopholes between slices. For example, if UE exchanges a sensitive date in one slice (enterprise) and publishes data on another slice (consumer), data leaks between slices are possible. The administration of data exchange between UE and different slices is a high level of security concerns, and this impact needs to be studied further. Security policies need to be defined on the RAN or Core network slicing, as the UE has no notion or control over which slice to request a connection. The network operator should ask UE to re-authenticate for every network slice separately to check and validate if UE is meeting the SLA for every slice or not. Otherwise, a malicious UE can authenticate to a lower-level security slice and get access to other slices through common network functions and resource sharing.

The volume-based attack is one of the widespread forms of cyber-attacks. Hackers disrupt normal service flow, typically by flooding the target with a high volume of packets or connection requests, overwhelming networking equipment, servers, or bandwidth resources. We evaluated our *Secure5G* model by simulating the attack so that device(s) make multiple connection requests to the network simultaneously. Our model detects bad malicious traffic and blocking before it reaches the core network, as shown in Fig. 14. As shown, malware botnets and attackers are trying to flood the network by sending multiple requests. If such attackers are allowed into the network, the system may run out of capacity and crash, be unable to handle huge traffic, and become unresponsive to new requests.

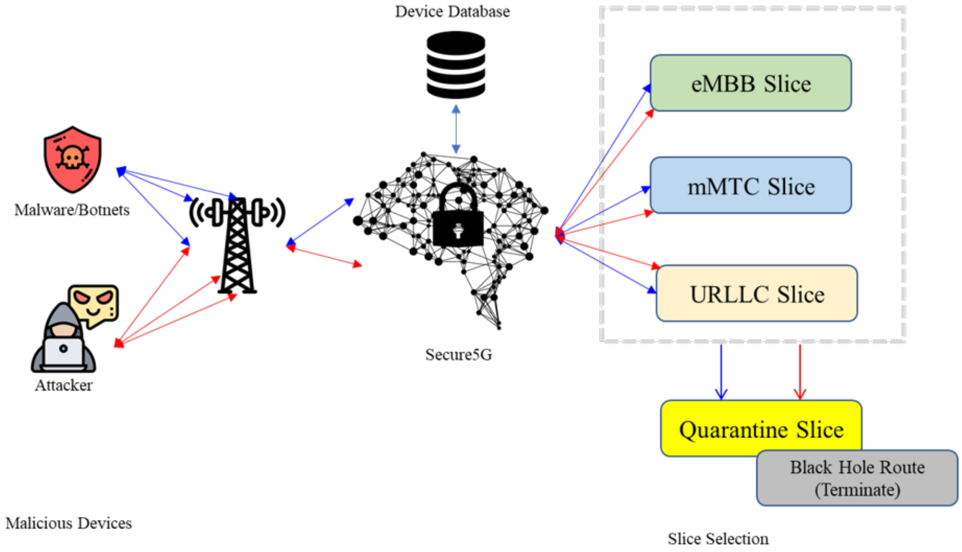


Figure 14: Volume-based Attack (Flooding) in Network Slicing

A smartphone can only make one control Radio Resource Control (RRC) signaling connection request to the network during the initial attach, and that too for a single network slice instance. However, suppose it makes multiple requests to multiple slices simultaneously. In that case, such an unusual behavior will be identified as suspicious and *Secure5G* model will detect this anomaly and quarantine these incoming connections. On the quarantine slice, such devices will only get the bare minimum service, eventually terminated if their malicious activity is confirmed.

The *Secure5G* model continues to learn and detect the incoming connections or the traffic pattern each time. If any known rogue (detected attacker) device continues its suspicious behavior by trying to flood the network, it will, ultimately, be denied any more service by the network. All its traffic will be moved to the black hole route, and this device will be marked as a possible threat to our database. After an initial attack, any device can make an immediate connection request to a different slice. We would not want any genuine user or an original connection request to be flagged as suspicious just because they make multiple requests divided by a short time interval. So, we'd like to consider a certain expiry timer on every incoming request to avoid false flagging. *Secure5G* model will only flag this as a threat if the device requests multiple slices after the timer have expired. Timer expiry can be customized, and here we have considered 5 seconds in our simulation, but network operators can choose to define their threshold based on slice requirements.

Fig. 15 represents the volume-based cluster simulated using *Secure5G*, providing

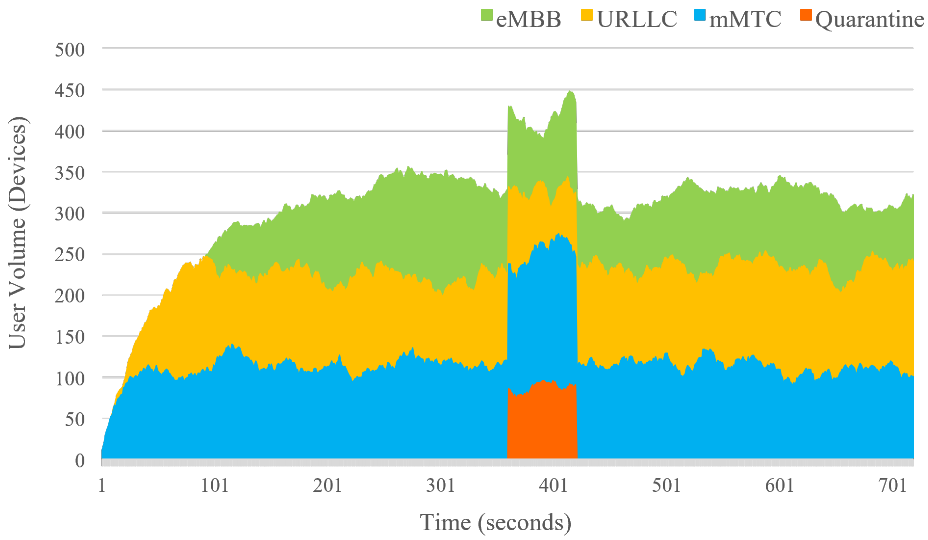


Figure 15: Volume-based Attack (Flooding) demonstration

the granular visibility of all incoming connections to different slices. The graph shows two scenarios combined; the normal and the attacker traffic. Normal traffic is when all devices follow the normal slice selection logic with DeepSlice, and the attack scenario shows the traffic with malicious connections where we randomly used ten malicious devices, requesting resources from all slice simultaneously. This is shown in the central region of the plot as a sudden increase in the number of users. Our model will quickly identify and transfer this malicious traffic to the quarantine slice.

Additionally, we evaluated the slice-centric attack scenario in our simulation as shown in Fig. 16, where ten identified IoT devices bombard the network with multiple requests for resources from eMBB and URLLC slice instead of their standard mMTC. *Secure5G* will kick in immediately after observing such a DDoS traffic stream as it identifies

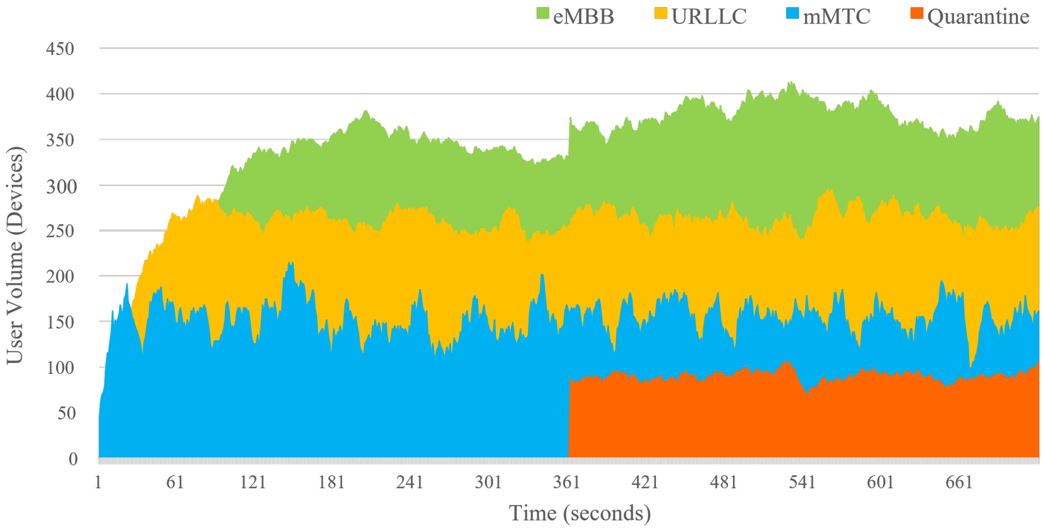


Figure 16: Slice-based Attack (Flooding by IoT devices)

an influx of packets with the suspiciously-identical device, making multiple slice connections that do not match a typical pattern. By tracking such minuscule abnormalities, the *Secure5G* will weed out malicious traffic without impacting regular (genuine) user flow.

2.7.2 Objective 5: Masking Botnets (Spoofing) Attack

We have evaluated the robustness of our *Secure5G* model by simulating the spoofing attack scenario by hackers. In simple terms, device or client masking is an impersonation of a user where the attacker disguises the source of an attack and allows the infected traffic to appear legitimate. Hackers commonly spoof DNS servers and IP addresses to spread the virus. Botnets are malware-infested devices that attackers use to generate massive traffic to consume the server capacity and multiple network connection requests to flood the system, resulting in server downtime, and most scenarios, even without their owners' knowledge.

In *Secure5G*, we have simulated spoofing attacks by masking as shown in Fig. 17; for example, a smartphone device appears to be an IoT device and makes a slice request for mMTC instead of eMBB. *Secure5G* model has an inbuilt database of devices and user patterns from learning, which maintains all original (and previous) connection requests made by any device. *Secure5G* assigns a unique global identifier when a UE tries to connect to the network first time. For example, when a smartphone device with a certain International Mobile Equipment Identity (IMEI), e.g., 123456789012345, requests a connection, the *Secure5G* model will assign a GUTI (e.g., 9a91abe4-baa2-4f55-b95e-7ab66040aec2) to this IMEI and this information is stored in our database. Every GUTI is mapped with the device type, IMSI, slice requested, etc. Suppose an attacker mask smartphone as an IoT device and requests an mMTC slice. In that case, the model will

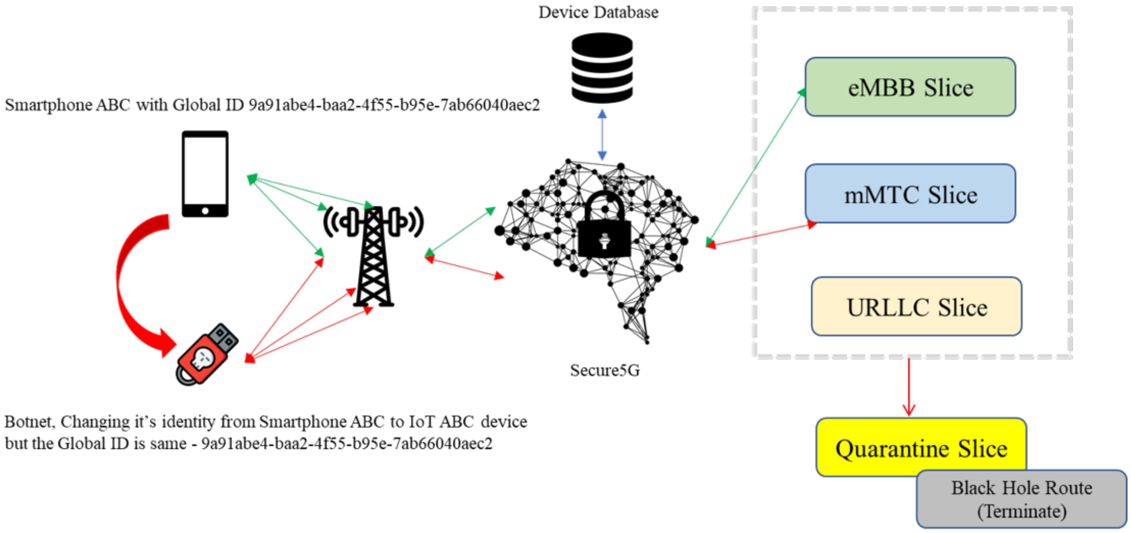

Device Database
Smartphone ABC with Global ID 9a91abe4-baa2-4f55-b95e-7ab66040aec2
eMBB Slice
mMTC Slice
URLLC Slice
Secure5G
Botnet, Changing it's identity from Smartphone ABC to IoT ABC device
but the Global ID is same - 9a91abe4-baa2-4f55-b95e-7ab66040aec2
Quarantine Slice
Black Hole Route
(Terminate)

Figure 17: Device Masking Attack in Network Slicing

understand this malicious request and flag it as a possible botnet by comparing it against IMEI and GUTI values in the database.

This paper has investigated the security concerns in the 5G network and presented a DLNN model to create a robust Network Slicing framework to combat DDoS attacks filtering the malicious UE connections to the 5G network. Volume-based flooding and spoofing attack scenarios were used as illustrations to evaluate the overall performance, and the detection accuracy was more than 98% with our limited dataset.

2.8 Conclusion

This research's primary objective is to develop methods and approaches for sustaining high availability and diverse service requirements for network slices in 5G networks. The *DeepSlice* research examines a framework that employs DL approaches to perform slice selection, slice load balancing, and slice failure schemes for network slices to achieve these objectives. We have extended the DLNN model to address the threats to the performance, availability, and robustness of B5G networks by proactively preventing and resolving threats to secure networks by understanding security threats and knowledge of protections and potential threat responses. The developed model is critical and future-proof in ensuring the end-to-end security of the 5G network and predicting the known and unknown applications/services which are not defined/developed today by utilizing the learning from a developed deep-learning model.

Published Work

DeepSlice: A. Thantharate, R. Paropkari, V. Walunj and C. Beard, "DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks," *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2019, pp. 0762-0767, doi: 10.1109/UEMCON47517.2019.8993066.* [19]

Secure5G: A. Thantharate, R. Paropkari, V. Walunj, C. Beard and P. Kankariya, "Secure5G: A Deep Learning Framework Towards a Secure Network Slicing in 5G and Beyond," *2020 10th Annual Computing and Communication Workshop and Conference (CCWC), 2020, pp. 0852-0857, doi: 10.1109/CCWC47524.2020.9031158.* [20]

CHAPTER 3

ADAPTIVE RESOURCE MANAGEMENT TECHNIQUES FOR NETWORK SLICING IN BEYOND 5G NETWORKS USING TRANSFER LEARNING

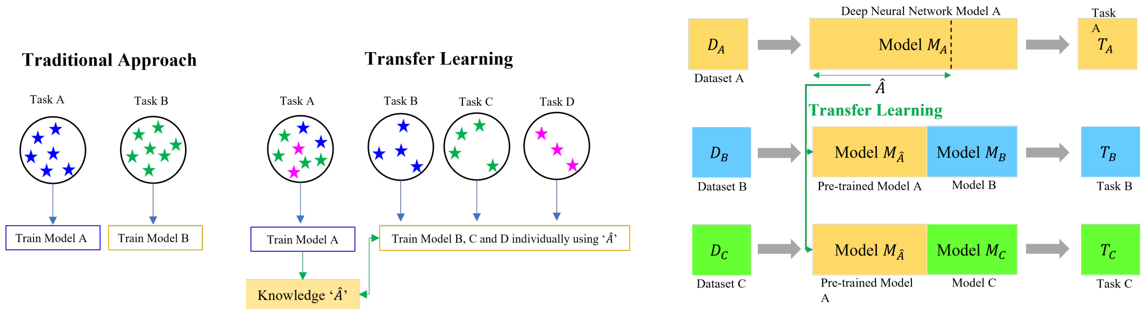
3.1 Introduction

The network functions and the core elements of the fifth-generation of mobile network, 5G, have transformed to serve and support various services such as enhanced Mobile Broadband (eMBB), Massive Internet of Things (mIoT), and Ultra-reliable low latency communication (URLLC). By allowing multiple virtual networks to be built within a single physical network architecture, network slicing can be instrumental in enabling verticals for mobile operators. Virtual networks, referred to as *slices*, would offer unique network functions to enable mobile network operators and service providers to develop innovations and business models. Optimizing cellular networks using the data collected from the end users and the core network is becoming relevant and raises more significant concerns over data privacy between users and operators.

Future intelligent wireless networks demand an adaptive learning approach towards a shared learning model to allow collaboration between data generated by network elements and virtualized functions. Current wireless network learning approaches have focused on traditional ML algorithms, which centralize the training data and perform sequential model learning over a large data set. However, training on a large dataset is inefficient; it is time-consuming and not energy and resource-efficient. Transfer Learning (TL) effectively addresses challenges by training based on a small data set using pre-trained models for similar problems without impacting neural network model performance. TL is a technique that applies the knowledge (features, weights) gained from a previously trained ML model to another but related problem. This work proposes an Adaptive Learning framework *ADAPTIVE6G*, a novel approach for a network slicing architecture for resource management and load prediction in data-driven B5G, 6G Wireless systems influenced by the knowledge learning from TL techniques. We evaluated *ADAPTIVE6G* to solve complex network load estimation problems to promote a more fair and uniform distribution of network resources.

We demonstrate that the *ADAPTIVE6G* model can reduce the Mean Squared Error (MSE) by more than 30% and improve the Correlation Coefficient 'R' close to 6% while reducing under-provisioned resources. TL can play a crucial role by utilizing an already trained model for general user behavior estimations, clustering of demographic profiles, and network environment analysis such as traffic, capacity, and resource management, which are critical for building a reliable and efficient next-generation network [21]. Then small amounts of specific user data can be used to target the model to that specific user without divulging large amounts of user data.

The study of collaborative ML, and TL specifically, in 5G systems for heterogeneous devices generating unusual data traffic, particularly with the network slicing feature, has yet to be investigated precisely. We used this opportunity to explore the optimum solutions tailored for network load estimation, thereby proposing a TL-influenced adaptive learning framework, *ADAPTIVE6G*. Fig. 18 (a) represents the current state of



(a) Typical Traditional and TL

(b) TL in *ADAPTIVE6G*

Figure 18: (a) Traditional ML vs. (b) Transfer Learning

Traditional ML, and Fig. 18 (b) depicts the general representation of TL. The conventional approach trains and constructs an individual model for each task, whereas the TL approach leverages prior knowledge from source tasks to improve the new target task performance. The TL flow comprises training a Global Model A (M_A) on a large dataset D_A to predict a task T_A on a different domain for a different source task but related problem. We capture the computed weights from the Global Model for each layer, which are then fed back to train new models using smaller datasets. The central concept here is to extract features using the weighted layers of the pre-trained model.

Traditional ML techniques presume that the training and testing data come from the same domain, with the same input feature space and data distribution. This is only sometimes true in real-world ML applications. Also, collecting training data can be costly or challenging in some cases. This is true for the Network Slicing use case, as the slicing feature has yet to be commercialized in the production 5G network. As a result, high-performance learners must be trained in various similar data domains. Deep Neural Networks are based on a reconfigurable architecture; many different hyper-parameters can be configured. Therefore, a knowledge transfer can be achieved by training or freezing Neural Network layers and classifiers, adding additional classification layers (Dense layers), fine-tuning, and retraining the new data on unfrozen layers based on our needs.

The primary challenges with traditional ML systems are:

- **Large Training and Insufficient Dataset:** Training a neural network from scratch is not practical if there are not enough labeled datasets. Also, getting a dataset for every domain is complex and not readily available.

- **Cost of Training:** Model training on a larger dataset is time and energy-consuming, not efficient.

- **Bandwidth Constraint:** Larger datasets demand larger bandwidth for uploading them to the server. TL uses smaller data and provides better, more reliable, and faster results using a pre-trained model.

- **Latency Constraints:** Smaller dataset helps with RF conditions caused by poor channels for sending data to the server, which helps analytical models to train quickly and predict.

- **Limited Resources:** Fast training using a small data set is critical to implementing ML in IoT devices with limited server and power capabilities.

- **Heterogeneity among end-devices:** Devices often generate data in a non-uniform, non-identically distributed (non-IID) fashion. Each device provides its unique feature KPIs for training purposes.

TL can overcome many of these challenges by reducing training time, improving the performance of neural networks, and using small amounts of data. TL utilizes model learning for the new task using pre-trained models by transferring the knowledge and extracting the feature learned from the initial training. TL is the process of enhancing learning in a new task by transferring knowledge from a previously learned related task. TL is an ML approach and technique that allows for time and resource savings by eliminating the need to train multiple ML models from scratch for similar tasks. By contrast, conventional ML algorithms have historically focused on discrete tasks. TL aims to address this by developing methods for transferring knowledge acquired in one or more source tasks and applying it to improve performance on a related target task.

The Network Slicing function without the *ADAPTIVE6G* framework takes slice selection based on SST type and S-NSSAI request from the UE and assigns appropriate slices upon availability. The current decision-making is based upon standards without intelligent insight into 5G RAN and CORE protocols and is naturally not the best. Considering these, we have simulated the network load across slices using the developed ML model *DeepSlice*. Network load varies throughout the day, and several factors drive network utilization, like geography, time, and events. During off-hours (midnight to 6 am), the slices have low utilization vs. utilization during peak hours (6 pm to midnight). This variation in usage creates a huge opportunity for a reconfigurable slicing strategy for resources and network management across network slices. We evaluated *ADAPTIVE6G* to estimate the loads on the entire network for individual slices based on our simulated data from the total network load using DeepSlice, which is further discussed in Section VI.

Definition: Domain and Task in Transfer Learning

In Transfer Learning, we define a **Domain '$\mathcal{D}$' and Task '$\mathcal{T}$'.**

Domain is defined by two components: Feature or Data space '$\mathcal{X}$' and a Marginal Probability Function '$P(X)$' of the dataset. Domains are modeled as probability distributions over an instance space, where a given dataset, $X = \{x_1, x_2, x_3, \ldots . x_n\}, X \in \mathcal{X}$

We can write Domain '$\mathcal{D}$' as:

$\mathcal{D} = \{\mathcal{X}, P(X)\}$;

Task consists of two parts: Label Space '$\mathcal{Y}$**'** and a predictive function 'θ' also referred as Objective function. Tasks associated to a domain (classification, regression, clustering).

The predictive function θ is an implicit one, which is expected to be learned from the sample data.

For a given dataset X, it's corresponding labels are Y = $\{y_1, y_2, y_3, \ldots . y_n\}, Y \in \mathcal{Y}$

Task can be written as:

$\mathcal{T} = \{\mathcal{Y}, \theta\}$, which can be also written in the form of conditional distribution $\{P\ (Y\ |X)\}$:

$\mathcal{T} = \{\mathcal{Y}, P\ (Y\ |X)\}$;

where, '$P\ (Y\ |X)$' is not observed yet but can be learned from input training data $\{x_i,\ y_i\}$, $i = 1, 2, 3, \ldots n$; $(x_i\ \in X)$ and $(y_i\ \in\ \mathcal{Y})$.

Transfer Learning Given the above definitions, a Source Domain $\mathcal{D}_S$ and its corresponding Task $\mathcal{T}_S$, and an objective function θ_S which has learning from $\mathcal{D}_S$ and $\mathcal{T}_S$. **Goal** is to get the target prediction function θ_T for Target Task $\mathcal{T}_T$ with Target Domain $\mathcal{D}_T$. Transfer Learning aims to improve the performance accuracy of θ_T by utilizing knowledge learned from θ_S where $\mathcal{D}_S \neq \mathcal{D}_T$ ***or*** $\mathcal{T}_S \neq \mathcal{T}_T$.

Transfer Learning can be denoted as:

$\mathcal{D}_S, \mathcal{T}_S\ \rightarrow \mathcal{D}_T\ \mathcal{T}_T$

3.2 Related Work

TL is a relatively new, under-explored paradigm in wireless communication systems, especially for a network function virtualization concept like network slicing. To our knowledge, we are the first to use ML to learn traffic patterns and apply the TL concept using the data from the individual slice toward network load prediction. The prior art and literature discussed in this section are related but not a direct comparison to the ADAPTIVE6G framework. DeepSlice [19], and Secure5G [20] were our first approaches to studying network slices in 5G systems by applying machine and DL techniques. We have demonstrated network slice selection for all UE types, including unknown devices, load balancing techniques in case of slice failure, and security of these slices in case of distributed denial-of-service (DDoS), flooding, and mask attacks. With ADAPTIVE6G, we have extended the part of the DeepSlice framework using TL toward load prediction across slices. We have categorized our related studies into the following four sections:

3.2.1 Transfer Learning and Traffic Prediction

Zeng, Q. et al. [22] proposed a wireless cellular traffic prediction model based on a cross-domain dataset containing SMS, call, and historical Internet records. The relationships of these data points with cellular traffic were studied and used against pre-trained models by adjusting the parameter values to improve the model accuracy. The experimental results showed a better performance of the model with the transfer learning capability than the model having no transfer learning. In contrast, in ADAPTIVE6G, we have used historical load to train the model and the learned weights to predict the total network load across slices. The authors in research [23] discuss the problem of predicting channel quality and average active user equipment when only a small amount of cell data is available. One-dimensional convolutional neural networks study several models with varying degrees of complexity overhead for prediction across 100 cells. The proposed framework's approach uses similar metrics as ADAPTIVE6G, where performance is compared to classical ML approaches in terms of accuracy and computation time. The results show that transfer learning outperforms non-transfer approaches, especially when the cell's data is limited.

The [24] survey paper discusses transfer learning, deep learning, and swarm intelligence for future wireless networks. The authors have summarized how TL uses DL features and applies a DNN trained in a different application instead of training a NN from scratch, which is also the basic modeling for ADAPTIVE6G. The authors discussed how DNNs with TL significantly improve the training process. In [25] the paper presents Transfer Learning based Prediction (TLP), a transfer learning-based framework for traffic prediction at the 5G edge that can achieve high prediction accuracy with limited and imbalanced data. Authors have developed a Similarity-based Elastic Weight Consolidation (SEWC) transfer learning technique that transfers a well-trained prediction model to a target edge node with limited data locally. Experiments on a real-world data set demonstrate that integrating TLP with SEWC can improve prediction accuracy by up to 57.9% compared to the current standard. In the proposed model, EWC analyzes the importance of different weights after training a NN, and Based on some analysis, EWC adds penalties to the loss function during future retraining.

M. Elsayed et al., in [26] evaluated three different ML approaches - Transfer Q-learning (TQL), Q-learning, and Best SINR association with Density-based Spatial Clustering of Applications with Noise (BSDC) algorithms and compared their performance under different scenarios to study the impact of network load in stationery and mobility scenarios. In [27], the authors proposed a novel framework based on transfer learning to address the problem of insufficient actual training data sets in contemporary networking platforms. By applying transfer learning, the agent can reuse experiential knowledge to aid in its action, resulting in a shorter training procedure than the conventional method and a reduction in energy expenditure.

3.2.2 Adoption of Deep Learning in the Wireless Network

This related work section discusses data-driven ML adopted and proposed by researchers in their work for 5G and Wireless Networks. Q. Zhao and D. Grace in [28] used a QoS-aware base station switching operation to reduce energy consumption and improve QoS.parameter learning algorithm has been developed that uses previously learned knowledge from spectrum assignment to make user association decisions. Also, to save energy, a tolerance range of system QoS was used to dynamically switch eNBs between active and sleep modes with user association load management. In contrast, in ADAPTIVE6G, we have used historical load data from base stations to perform load predictions. A transfer learning algorithm is used to train a smaller dataset from learned parameters, which also indirectly saves energy by reducing epochs and training time. In article [29], the authors discuss the design of Industrial IoT (IIoT) systems based on TL. The authors discussed how TL could alleviate the need for extensive data samples for training ML models in the

IIoT. Additionally, they classified TL systems for IIoT into two categories: TL for IIoT machinery anmethodsIoT networking. They also discussed the design components and challenges associated with each proposed category.

C. Zhang, P. Patras, and H. Haddadi in [30] have addressed how to customize DL models to broad mobile networking applications. In this survey, they identified areas where applying machine intelligence can be complicated and challenging in mobile network environments. The authors in surveys [31][32] have explored the role of AI in 5G wireless communication and networking by covering case studies, problems, and future research prospects. The authors also discussed network caching, task offloading, routing schedules, and resource allocation using reinforcement and deep reinforcement learning algorithms. C. D. Alwis et al. [33] have provided an overview of current 6G advancements by highlighting the socioeconomic and technological trends driving 6G. They also discussed the criteria for realizing 6G applications and standardization efforts. Q. Huang and M. Kadoch in [34] have proposed a reinforcement learning approach for radio spectrum resource scheduling; their evaluation suggested that the devised technique works well for mobile networks with a high spectrum load. The authors in [35] proposed a deep reinforcement learning model for 5G radio resource scheduling. Their experimental results show that it outperforms existing baseline approaches in many key performance indicators.

G. Zhu et al., [36] paper proposed supervised learning-based QoS assurance for 5G networks. Supervised ML can learn the network environment and adapt to changing conditions. They automatically reconstruct the relationship between historical QoS data and current QoS anomalies. After that, they suggest automatic mitigation. Supervised ML can also predict future QoS anomalies. The proposed framework architecture was validated using a decision tree-based case study for QoS anomaly root cause tracking. Y. Sun et al., in [37], discussed ML applications in resource management at the MAC layer, networking and mobility management in the network layer, and localization in the application layer. The performance of traditional procedures is compared to ML-based approaches. Theoretical advice for ML implementation, available data sets and academic platforms, and more are covered as part of the literature.

M. Karimzadeh et al., in [38], have used LSTM to predict the trajectory and traffic flow of moving items in cities. Their developed predictors consistently deliver satisfactory results over the state-of-the-art on two large-scale real-world datasets. B. Yang et al., in [39], proposed a privacy-preserving edge-CNN framework for 5G industrial edge networks. It can use existing image datasets to train the CNN, which is then fine-tuned using the limited datasets uploaded by devices.

3.2.3 Network Slicing in the Wireless Network

The work in [40] addresses the status of ML-related initiatives in standards bodies and industrial forums to design, build, deploy, operate, control, and manage 5G network slices. Zhou, H., Erol-Kantarci, M., and Poor, V. in [41] have proposed a Transfer Reinforcement Learning (TRL) scheme for joint radio and cache resource allocation for 5G RAN slicing. First, they have defined a hierarchical resource allocation architecture and proposed two TRL algorithms: Q-value TRL and action selection TRL (ASTRL). In the proposed schemes, learner agents learn from expert agents to improve their performance. The proposed algorithms are compared to model-free Q-learning and model-based PPF-TTL algorithms. QTRL and ASTRL have 23.9% less delay for Ultra Reliable Low Latency Communications and 41.6% more throughput for enhanced Mobile Broad Band, while Q-learning has significantly slower convergence. PPF-TTL has a 40.3% lower URLLC delay and almost twice the eMBB throughput. In our recent work [42], we use federated learning to solve complex resource optimization problems without collecting sensitive, confidential information from end devices. The evaluation results reflect more than 39% improvement in MSE, 46% better model accuracy, and more than 23% reduced energy cost for training the proposed FED6G against the traditional deep learning neural network model.

The authors in [43] proposed a convolutional neural network (CNN) and long short-term memory (LSTM) hybrid deep learning model, leveraging much learning from DeepSlice [19]. The CNN handles resource allocation, network reconfiguration, and slice selection, while the LSTM handles network slice statistics (load balancing, error rate). The models applicability is tested using unknown devices, slice failure, and overloading. McClellan, M.; Cervello-Pastor, C.; and Sallent, S. in survey paper [44] discuss the DL techniques for the wireless network, which is critical in helping 5G networks achieve eMBB, URLLC, and mMTC goals. They discussed how DL could predict user behavior and automate network resource management in 5G networks. It can improve user experience and future operational costs for telecom companies. To achieve an adaptive control strategy in unexpected network conditions, the authors in [45] have proposed a self-sustained RAN slicing framework. The authors have used TL to move from model-based control to autonomic and self-learning RAN slicing control. The proposed RAN slicing framework should significantly improve emerging service QoS.

J. J. Chen et al., in [46] developed a network slicing framework for IoT applications with varying needs using SDN technology. An electronic fence application illustrates the system's effectiveness in its evaluation. W. Wang in [47] discusses how physical node (PN) anomaly in substrate networks will degrade the performance of multiple network slices. The authors have proposed a cooperative anomaly detection scheme based on a transfer learning-based hidden Markov model for self-organizing network slice management. The PNs are first divided into four states. Then the hidden Markov model (HMM) captures the current states of PNs based on virtual node measurements. And based on learned network knowledge and PN similarity, HMM proposes a cooperative anomaly detection algorithm. The authors determined that the proposed TLHMM-based cooperative anomaly detection algorithm achieves an average detection accuracy of over 90% in simulations. The authors in [48] propose a model that allocates network costs to different deployed slices, which can later be used to price different E2E services. This is a network infrastructure provider's allocation. A resource allocation algorithm and a 5G network function (NF) dimensioning model are also proposed as inputs to the proposed model.

3.2.4 Other Load Prediction Studies with Machine Learning Applications

The authors in [49] have investigated a novel multimodal DL approach for Traffic Classification (TC). In particular, it can capitalize on traffic data heterogeneity, overcome the performance limitations of existing single-modality DL-based TC proposals, and solve various traffic categorization problems associated with various providers' needs.

Their proposal outperforms (a) current multitask DL architectures and (b) single-task DL baselines on a real dataset of encrypted traffic. A network survivability optimization framework and heuristic for the network provider is proposed in [50] to depict the network virtualization with several procedures and requires system slicing in 5G. The authors in [51] of this paper have reviewed recent literature, publications, and critical findings. They created a conceptual framework for cloud resource management, which they used to organize the state-of-the-art review. The authors have identified five challenges for future research based on our findings. These concerns include providing predictable performance for cloud-hosted applications, achieving global manageability for cloud systems, engineering scalable resource management systems, comprehending economic behavior and cloud pricing, and developing solutions for the mobile cloud paradigm. One of our research work [52] proposes Balanced5G, a data-driven traffic steering framework that takes proactive actions during the HO stage to steer traffic fairly between different frequencies (low, mid, and high). Balanced5G ensures that UEs (fixed or mobile) do not select a frequency solely based on the most robust signal strength but consider other network key metrics such as network load, active network slices, and the type of service for which the UE is requesting resources. And in Deep-Mobility, [53] we considered multiple parameters and interactions between system events along with the user mobility, which would then trigger a handoff in any given scenario, where network load also plays a critical role in making a handover decision, especially for load-balancing use-cases.

3.3 Proposed *ADAPTIVE6G* Model

Most ML algorithms operate in an iterative method where models learn from the sample data repeatedly and improve over time. As network slicing in 5G supports varieties of vertical use cases, one generic model does not set optimum model parameters for all data types generated by billions of diverse sets of devices. Consider network resource management as an example; a smartphone is more bandwidth and data-hungry than IoT devices like sensors, which are sensitive to power drain and generally resource-constrained. Augmented and Virtual reality (AR/VR) applications and mission-critical services, on the other hand, are more susceptible to latency. Also, many of these UEs are not always active; an IoT device checks into the server periodically and does not require high data connectivity.

Similarly, a connected car, Vehicle-to-everything (V2X) UE, can have many handovers during mobility. However, there is an opportunity to avoid those handovers if we know the car usage and the regular route and allow it to connect to sites with the least possible handovers and consistent, reliable links along its route. These miscellaneous network applications and services also generate quite different KPIs. Some UE types significantly generate more data than other UE types; this makes ADAPTIVE6G a unique learning model for wireless communication by creating an adaptive learning model for each of these unique slices for resource management problems.

Fig. 19 summarizes our *ADAPTIVE6G* proposed framework, which includes network load data from heterogeneous devices across slices A, B, and C. The term *ADAP-*

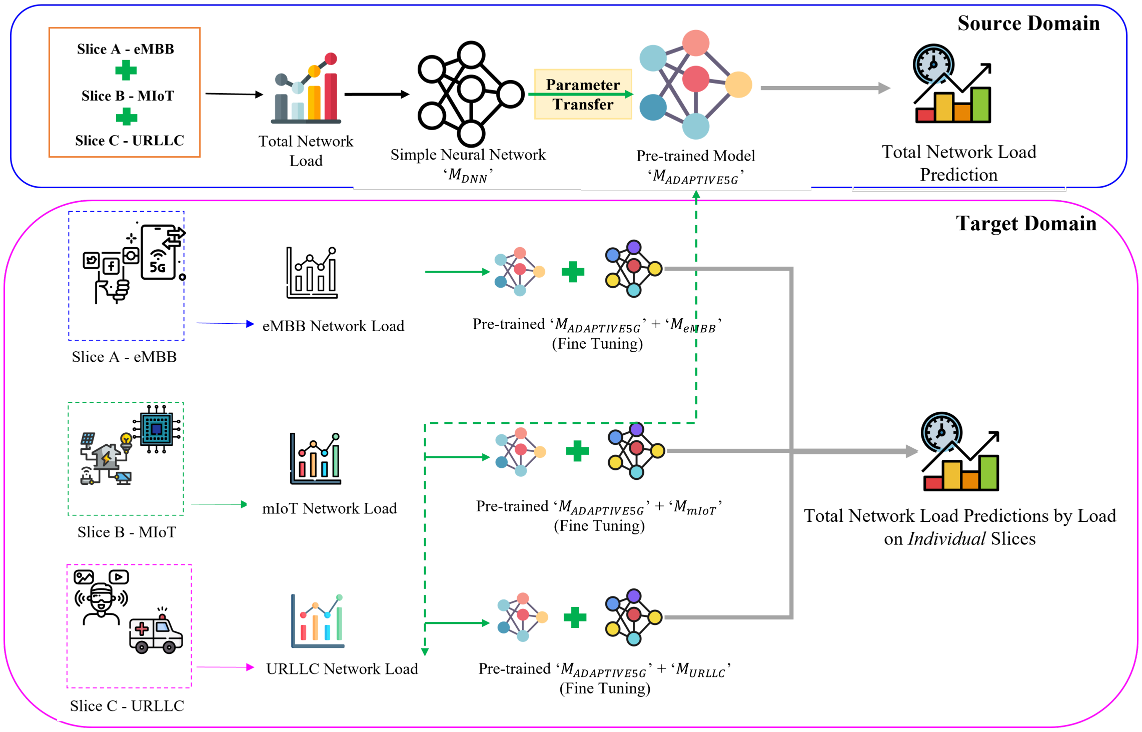

Source Domain
Slice A - eMBB
Slice B - MIoT
Slice C - URLLC
Total Network Load
Simple Neural Network ‘M_{DNN}’
Parameter Transfer
Pre-trained Model ‘$M_{ADAPTIVE5G}$’
Total Network Load Prediction
Target Domain
Slice A - eMBB
eMBB Network Load
Pre-trained ‘$M_{ADAPTIVE5G}$’ + ‘M_{eMBB}’ (Fine Tuning)
Slice B - MIoT
mIoT Network Load
Pre-trained ‘$M_{ADAPTIVE5G}$’ + ‘M_{mIoT}’ (Fine Tuning)
Slice C - URLLC
URLLC Network Load
Pre-trained ‘$M_{ADAPTIVE5G}$’ + ‘M_{URLLC}’ (Fine Tuning)
Total Network Load Predictions by Load on *Individual* Slices

Figure 19: Proposed ADAPTIVE6G Framework

TIVE6G reflects the adaptability of a single global model learned from a diverse dataset being applied to a smaller set of specific data for further predictions and estimations, which helps us achieve better performance with less training time and energy. In *ADAPTIVE6G*, Slice A represents enhanced Mobile Broadband (eMBB) services and includes traffic data generated from end devices such as Smartphones and typical mobile broadband application services. Slice B represents Massive Machine-type Communication (mMTC or mIoT) which focuses on traffic generated from devices like Industrial 4.0 and the Internet of Things (IoT). Slice C serves Ultra-Reliable Low Latency Communication (URLLC) data traffic like Augmented and Virtual Reality (AR-VR) and public safety. The data generated by each of these slices is diverse in load, service, and characteristics.

Our goal in using *ADAPTIVE6G* is to forecast network load using historical data for both scenarios, i.e., first using total network load (inputs from all three Slices A, B, and C) as *three* input vectors and predict network load. Then secondly, we take an individual slice as input (Slice A only, Slice B only, Slice C only) as *one* input vector to predict the load for that slice.

3.4 Proposed *ADAPTIVE6G* Step-by-Step Workflow

In this section, we detail the step by step the working of proposed ADAPTIVE6G as shown in Fig. 20:

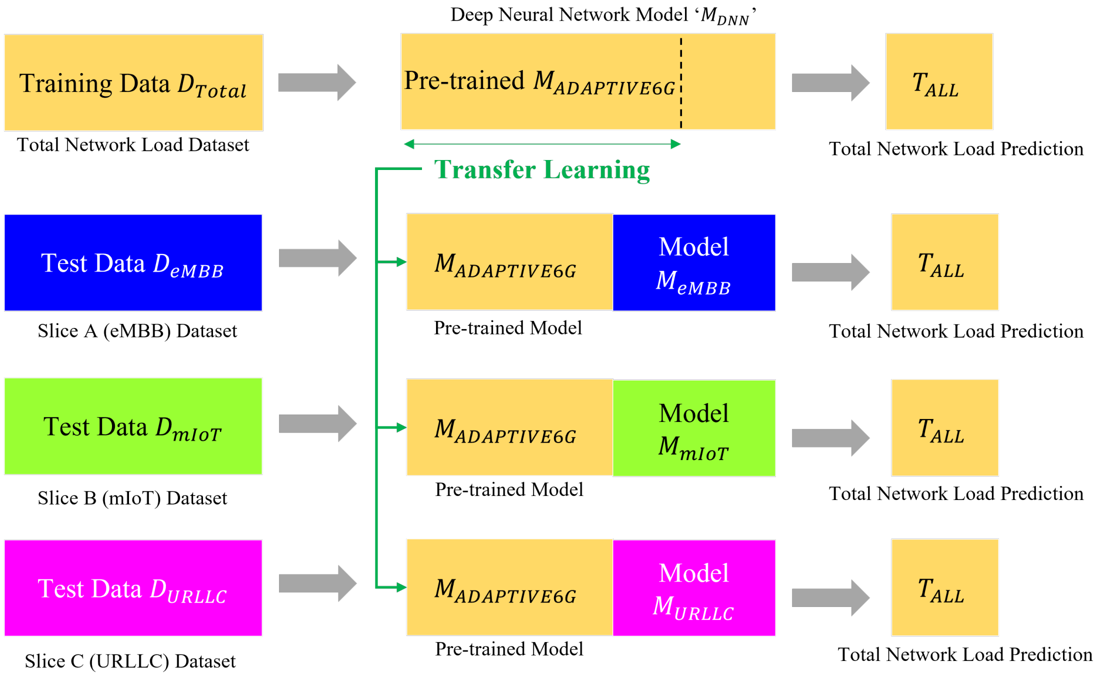


Figure 20: ADAPTIVE6G Framework

1. The ADAPTIVE6G framework initializes by training the traditional neural model M_{DNN} using observed network load data from all Slices - A (eMBB), B (mIoT), and C (URLLC), i.e., D_{TOTAL}. Network operators can deploy many slices; we are considering three standard slices for our evaluation per standardized 3GPP SST values. We have employed five-layer Deep Neural Networks: Input (features), 3 Hidden Layers, and Output (prediction). We have tuned the model hyper-parameters by changing the number of hidden layers, learning rate, activation function, and

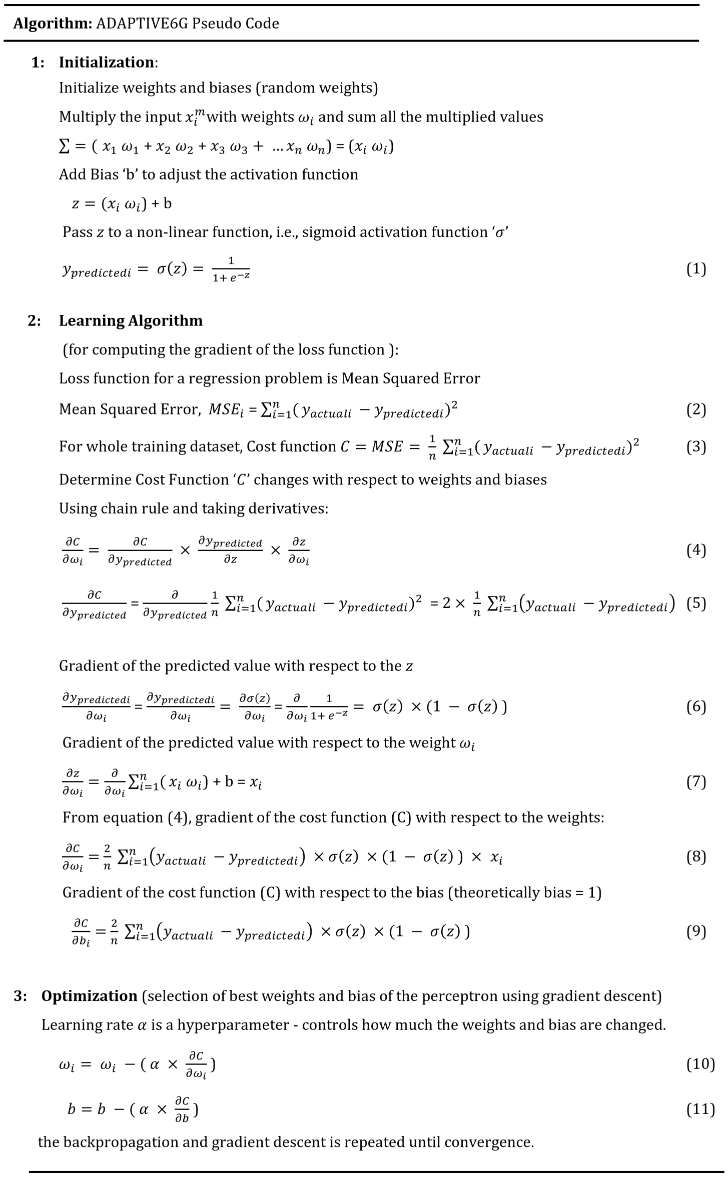
the number of epochs for the M_{DNN} model. Our goal is to validate the model performance between random weights and learned weights, so we kept the DNN modeling the same for both M_{DNN} and $M_{ADAPTIVE6G}$. The algorithm uses randomness to find a good set of weights for the data's specific input-output mapping function. Each time the training algorithm is run, a different network with a different model is fitted. The shuffling of the training dataset before each epoch also uses randomness, resulting in differences in the gradient estimate for each batch.

2. First, we train the M_{DNN} multi-layer model using a feed-forward back-propagation network with initialized random weights (stochastic gradient descent). A forward pass through the network is accomplished by iteratively computing each neuron in the subsequent layer until the output is achieved. We evaluate the output quality based on a cost function C and the desired result in the output layer. Mean squared error (MSE) is a loss function for evaluation.

3. A backward pass is then used to optimize the cost function C after the first result has been obtained by readjusting the weights and biases. We aim to optimize the output by adjusting the entire neural network. Based on this, we can calculate the total loss and determine the model's suitability (good or bad), and here weights are adjusted to obtain the least loss. After back-propagation, we capture each layer's computed weights (learned weights) for TL parameters and define these trained weights as $M_{ADAPTIVE6G}$.

4. Now instead training the $M_{ADAPTIVE6G}$ with *random weights*, $M_{ADAPTIVE6G}$ is

initialized using learned weights and re-train for smaller datasets D_{eMBB}, D_{mIoT}, D_{URLLC} from individual slices, which are subsets of D_{TOTAL} to predict total network load.



```
Algorithm: ADAPTIVE6G Pseudo Code

1: Initialization:
    Initialize weights and biases (random weights)
    Multiply the input $x_i^m$ with weights $\omega_i$ and sum all the multiplied values
    $\sum = ( x_1 \omega_1 + x_2 \omega_2 + x_3 \omega_3 + \ldots x_n \omega_n) = (x_i \omega_i)$
    Add Bias 'b' to adjust the activation function
      $z = (x_i \omega_i) + \mathrm{b}$
     Pass $z$ to a non-linear function, i.e., sigmoid activation function '$\sigma$'
    $y_{predictedi} = \sigma(z) = \frac{1}{1+ e^{-z}}$                                  (1)

2: Learning Algorithm
     (for computing the gradient of the loss function ):
    Loss function for a regression problem is Mean Squared Error
    Mean Squared Error, $MSE_i = \sum_{i=1}^{n}( y_{actuali} - y_{predictedi})^2$                  (2)
    For whole training dataset, Cost function $C = MSE = \frac{1}{n} \sum_{i=1}^{n}( y_{actuali} - y_{predictedi})^2$      (3)
    Determine Cost Function '$C$' changes with respect to weights and biases
    Using chain rule and taking derivatives:
    $\frac{\partial C}{\partial \omega_i} = \frac{\partial C}{\partial y_{predicted}} \times \frac{\partial y_{predicted}}{\partial z} \times \frac{\partial z}{\partial \omega_i}$          (4)
    $\frac{\partial C}{\partial y_{predicted}} = \frac{\partial}{\partial y_{predicted}} \frac{1}{n} \sum_{i=1}^{n}( y_{actuali} - y_{predictedi})^2 = 2 \times \frac{1}{n} \sum_{i=1}^{n}( y_{actuali} - y_{predictedi})$  (5)

    Gradient of the predicted value with respect to the $z$
    $\frac{\partial y_{predictedi}}{\partial \omega_i} = \frac{\partial y_{predictedi}}{\partial \omega_i} = \frac{\partial \sigma(z)}{\partial \omega_i} = \frac{\partial}{\partial \omega_i} \frac{1}{1+ e^{-z}} = \sigma(z) \times (1 - \sigma(z) )$     (6)
     Gradient of the predicted value with respect to the weight $\omega_i$
    $\frac{\partial z}{\partial \omega_i} = \frac{\partial}{\partial \omega_i} \sum_{i=1}^{n}( x_i \omega_i) + \mathrm{b} = x_i$                   (7)
     From equation (4), gradient of the cost function (C) with respect to the weights:
    $\frac{\partial C}{\partial \omega_i} = \frac{2}{n} \sum_{i=1}^{n}( y_{actuali} - y_{predictedi}) \times \sigma(z) \times (1 - \sigma(z) ) \times x_i$      (8)
     Gradient of the cost function (C) with respect to the bias (theoretically bias = 1)
     $\frac{\partial C}{\partial b_i} = \frac{2}{n} \sum_{i=1}^{n}( y_{actuali} - y_{predictedi}) \times \sigma(z) \times (1 - \sigma(z) )$         (9)

3: Optimization (selection of best weights and bias of the perceptron using gradient descent)
   Learning rate $\alpha$ is a hyperparameter - controls how much the weights and bias are changed.
    $\omega_i = \omega_i - ( \alpha \times \frac{\partial C}{\partial \omega_i})$                          (10)
     $b = b - ( \alpha \times \frac{\partial C}{\partial b})$                            (11)
  the backpropagation and gradient descent is repeated until convergence.
```

Figure 21: ADAPTIVE6G Pseudo Code

3.5 Fair and Uniform Load Forecasting using *ADAPTIVE6G*

The *ADAPTIVE6G* can act as an entity that can offer prediction and analytics from the learned data and assist network functions in making such decisions, as shown in Fig. 22. A good network slicing example would be to provide the slice-specific analytic data (e.g., network load, number of UEs, abnormal behavior, and alarms), which helps the system take the slice selection decision in real-time and helps the core network achieve better network efficiency and robust reliability. Awareness of load on the network, traffic congestion, and Quality of Service sustainability, especially for the individual slice instances, can drive more reliable network intelligence for service orchestration and network automation.

The proposed *ADAPTIVE6G* Learning framework has been extensively evaluated for resource optimization problems in a network slicing architecture. The goal is to forecast total network loads from Slices A, B, and C using a neural model assisted with TL. To emphasize the proposed network slicing framework's exemplary behavior, we first used a traditional neural model M_{DNN} as our benchmark model without incorporating any knowledge transfer (i.e., initialized with random weights) to predict network load across all slices. After M_{DNN}, we then used the pre-trained $M_{ADAPTIVE6G}$ (using the learned weights from M_{DNN} to predict the network load using the same set of validation data as M_{DNN}. We used pre-trained model $M_{ADAPTIVE6G}$ for the second round of evaluation, which incorporated learning from the M_{DNN} neural model to train and predict total network and traffic data inputs from each slice D_{eMBB}, D_{mIoT}, D_{URLLC} individually. The data pattern from each slice individually further provides a faster training time using

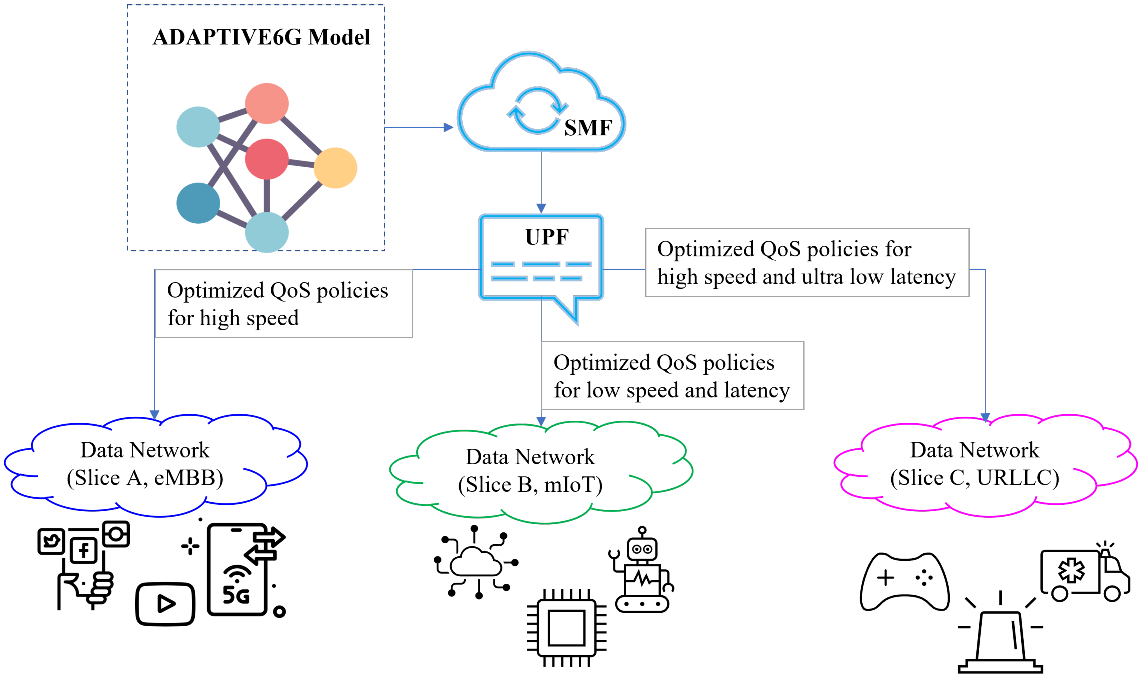

ADAPTIVE6G Model
SMF
UPF
Optimized QoS policies for high speed and ultra low latency
Optimized QoS policies for high speed
Optimized QoS policies for low speed and latency
Data Network (Slice A, eMBB)
Data Network (Slice B, mIoT)
Data Network (Slice C, URLLC)
5G

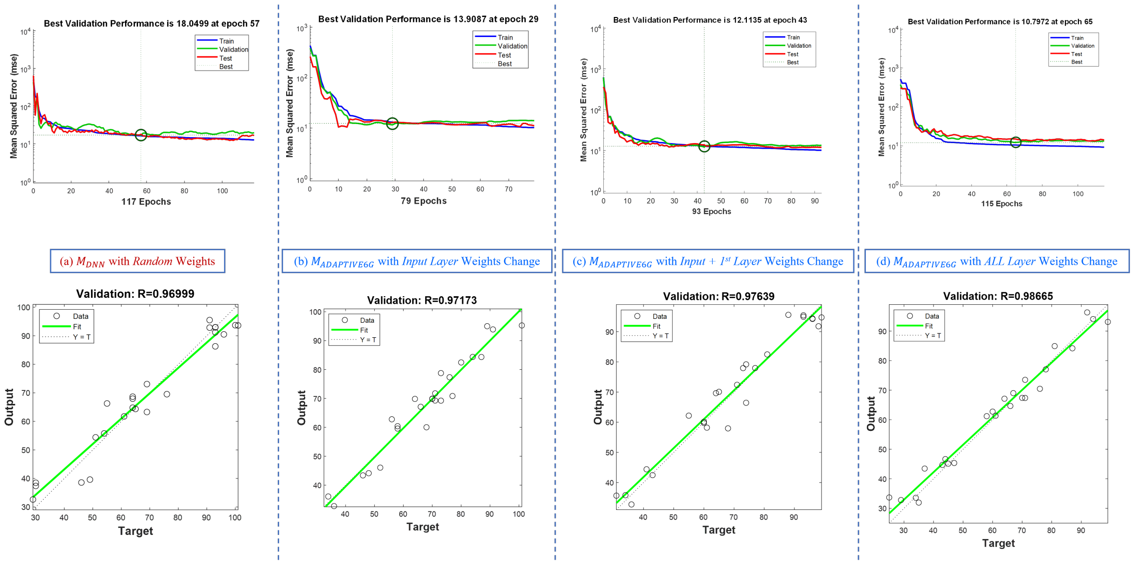
Figure 22: ADAPTIVE6G Framework in 5G CORE Systems

fewer data points and allows flexibility if computation and prediction are required for a particular slice, which removes redundancy to training the whole dataset every time.

Network slicing is still in its infancy, and no mobile operator has commercially deployed it in a production network. We evaluated ADAPTIVE6G using real-world data collected from a mobile network operator and augmented data. We conducted real-world measurements using commercially available 5G devices, replicating eMBB, mIoT, and URLLC-like services on live 5G (Sub-6 GHz and mmWave) networks to generate augmented data for slicing; details can be found in [19]. We used the DeepSlice framework to augment the data further to provide the load across network slices for our modeling. We believe that our developed *ADAPTIVE6G* dataset (available at [54]) will help bridge the gap in ML modeling, particularly for research communities interested in B5G and 6G systems for network slicing. We have carefully pre-processed the dataset to avoid over-fitting and applied techniques like pruning and regularization for good training.

3.5.1 Objective 1: Adaptive Load Forecasting using All Slice

We used feedforward modeling using the MATLAB Deep Learning Toolbox to train our neural model on the observed (actual) value dataset, splitting it into training (80%) and testing subsets (20%). We estimated the model loss using regression's 'Mean Squared Error' loss function metric, which is the average sum of the squared difference between the actual value and the predicted or estimated value. As a result, the value of MSE is always positive. Estimators and predictors with a close to zero MSE value will be more accurate. The predictor is perfect if the MSE is zero, and lower MSE values are preferable for good models. The Correlation Coefficient (R) is used to determine the strength of a relationship between data variables. It measures the closeness (strength and direction) of the association of the variables in a linear regression problem. R represents how well-predicted outputs match actual outputs; R-values range from -1 to 1. A value close to '1' represents a strong positive correlation (good model with better fit, thus better model accuracy), whereas '0' indicates no correlation, and a '-1' value indicates a strong negative correlation between variables. A trained model on a set of features with little or no correlation will produce incorrect results.



(a) M_{DNN} with *Random* Weights

(b) $M_{ADAPTIVE6G}$ with *Input Layer* Weights Change

(c) $M_{ADAPTIVE6G}$ with *Input + 1st Layer* Weights Change

(d) $M_{ADAPTIVE6G}$ with *ALL Layer* Weights Change

Figure 23: Model Performance for Total Load - Traditional DNN vs. ADAPTIVE6G

Fig. 23 and 24 show the performance validation and regression metrics plots of the neural model for both M_{DNN} and $M_{ADAPTIVE6G}$ models when trained and validated using total load from all slices. Fig. 23(a) represents the M_{DNN} model initialized with random weights. In Fig. 23(b), we have started with weights from Fig. 23(a) and then

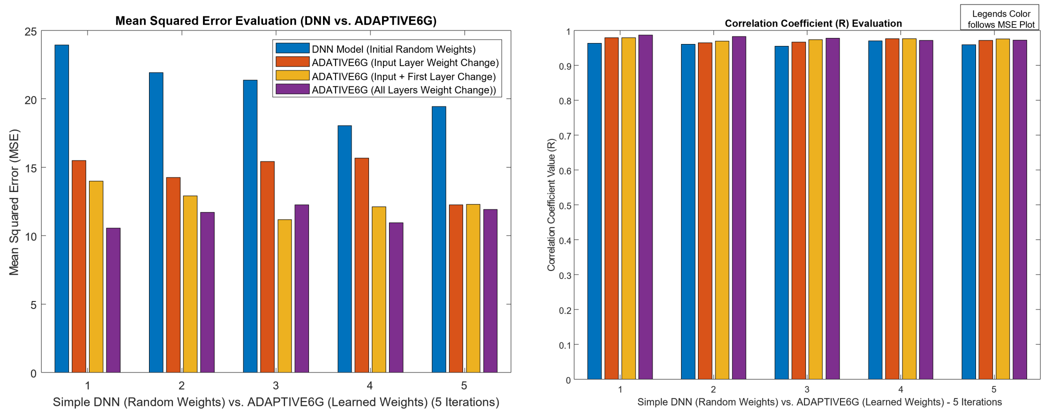


Figure 24: Metric Evaluation - Traditional DNN vs. ADAPTIVE6G

allowed just the input layer weights learned from M_{DNN} to be changed in the training process. We then observed improved MSE error and Correlation Coefficient R with a smaller number of epochs. In Fig. 23 and 24, we have further changed the weights for subsequent hidden layers, and as illustrated in the figures, the learned weight changes on each layer improve the model accuracy while decreasing MSE error and the total number of epochs; numerical details are discussed in Section VII. The dashed line in each plot represents the perfect result, i.e., outputs = targets. The solid line represents the best-fit linear regression line between outputs and targets. The R-value is an indication of the relationship between the outputs and targets. If R = 1, this indicates an exact linear relationship between outputs and targets. If R is close to zero, there is no linear relationship between outputs and targets. For this example, the training data indicates a good fit. The validation and test results also show large R values.

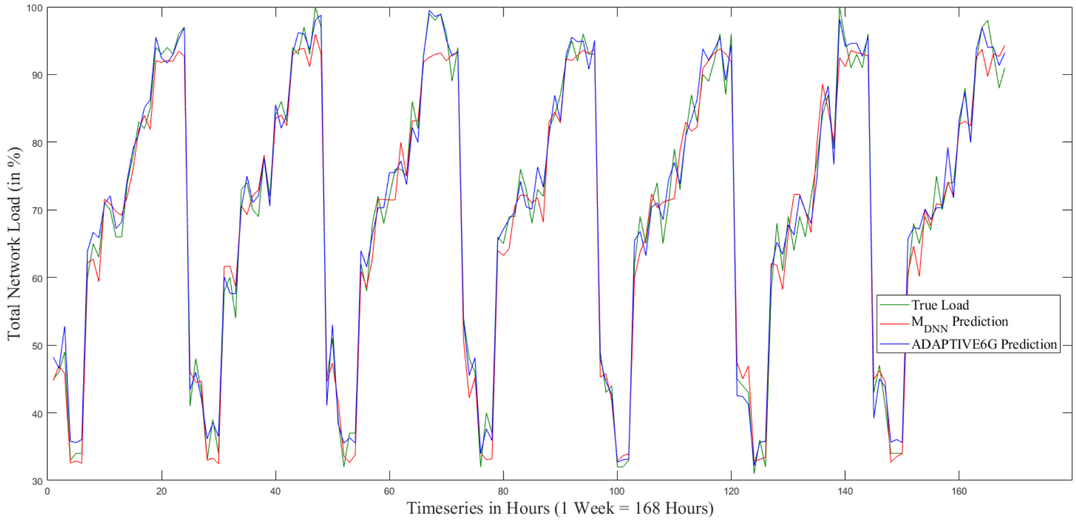


Figure 25: Predicted Network Load - Traditional DNN vs. ADAPTIVE6G

Fig. 25 shows the final predicted output over the period of a week, separated by hours (total 168 hours). For simplicity purposes, we have chosen the best-performed ADAPTIVE6G model with weights updated for all layers and compare it against the traditional DL model M_{DNN} and actual observed data. The results illustrate that the ADAPTIVE6G model is forecasting the traffic very closely with actual observed data using a smaller dataset, lower MSE error, improved Correlation Coefficient R, and in a smaller number of epochs.

Fig. 26 shows the error delta between Actual Load (i.e., '0' as baseline) and differences in predicted values between M_{DNN} and $M_{ADAPTIVE6G}$, indicating overprovisioned (above '0' baseline indicating overestimated load) and under-provisioned (below '0' baseline) resources. Under-provisioned resources yield a negative user experience for better resource modeling, which is observed significantly less in the predicted output from *ADAPTIVE6G*. The actual average load (baseline) over 168 hours for a week is observed as 69.78%, whereas our *ADAPTIVE6G* predicted 70.28% (0.72% overprovisioned) against 69.14% (0.91% under-provisioned) when using traditional ML. The *ADAPTIVE6G* approach will provide a significantly better user experience against traditional load prediction techniques with a sense of adaptability to dynamic load and traffic needs.

3.5.2 Objective 2: Adaptive Load Forecasting using Individual Slice

It is reasonable (though not desirable) for a network operator to schedule additional resources (to be over-provisioned) to accommodate load at all times in case of an unpredictable surge in data traffic that requires more physical resource block utilization. On the other hand, under-provisioned resources will result in a negative user experience with packet loss, interrupted data transmission, and higher latency, which would violate the agreed-upon SLA for users and degrade the Quality of Experience (QoE) overall. It is even more challenging for a network slicing architecture to make analytics predictions based on a single slice instance (one input vector).

With *ADAPTIVE6G*, we effectively addressed some of these challenges by learning features from a model trained on inputs from all three slices M_{DNN}; during individual slice modeling, we initialize the model with model learned weights sequentially during training to predict overall network load. Fig. 27, 28, and 29 shows the *ADAPTIVE6G* model performance validation and regression metrics for both M_{DNN} and $M_{ADAPTIVE6G}$ using slice data from slice A, slice B, and slice C individually. Isolated management of

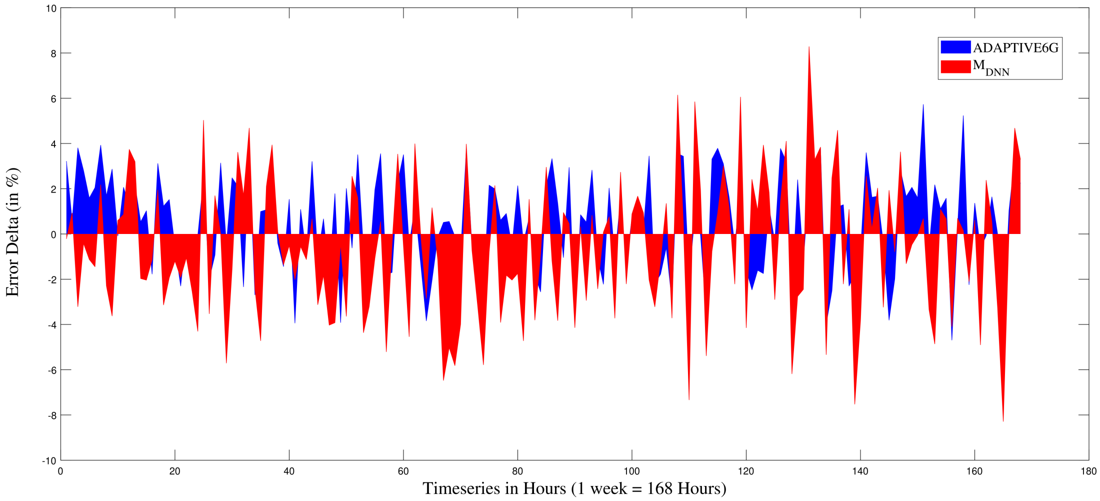


Figure 26: Error Delta - Actual Value vs. Traditional DNN vs. ADAPTIVE6G

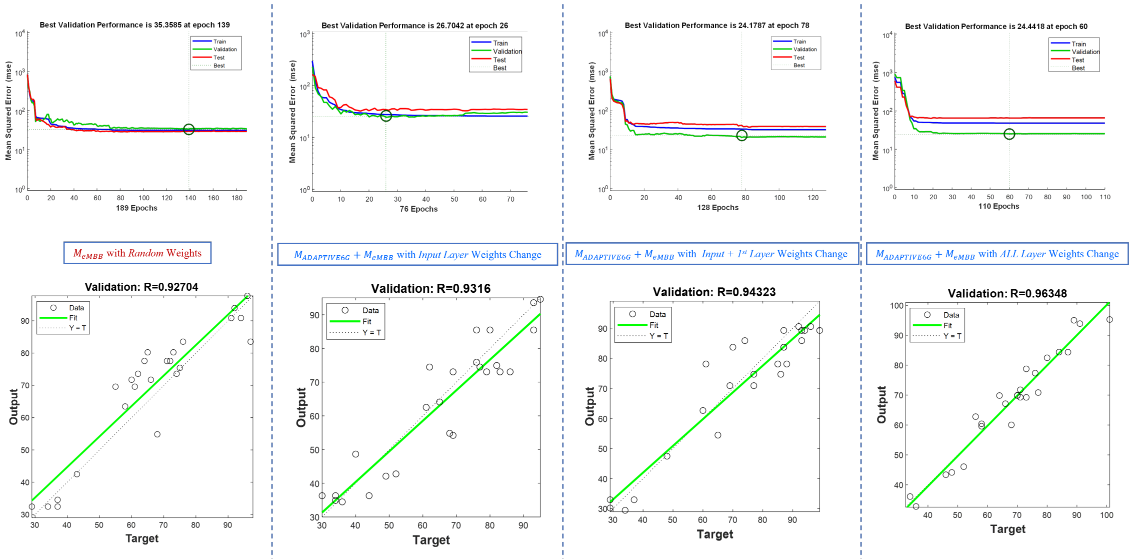


Figure 27: Model Performance (Slice A - eMBB): Traditional DNN vs ADAPTIVE6G

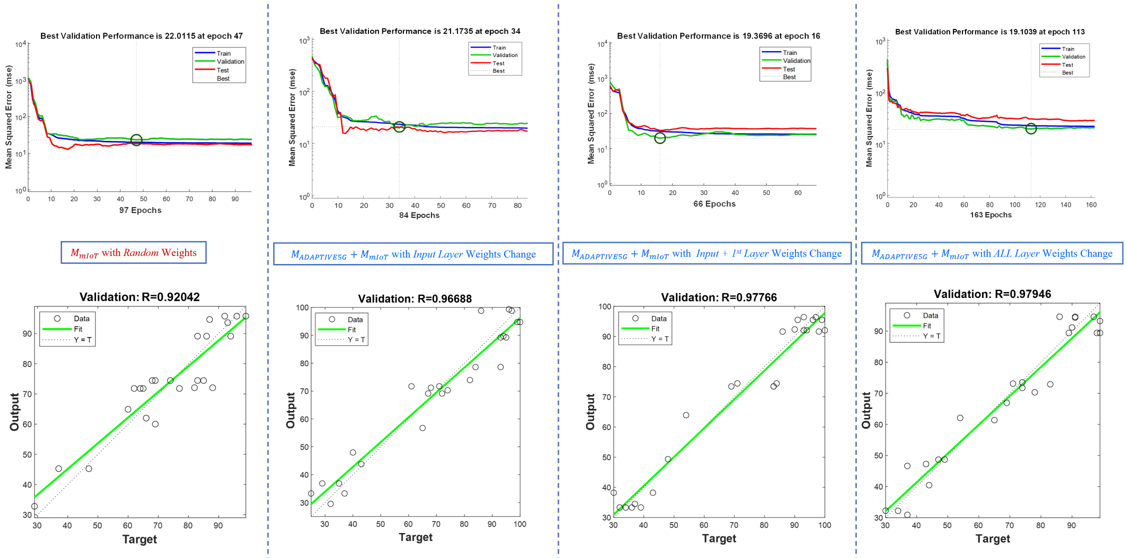


Figure 28: Model Performance (Slice B - mIoT): Traditional DNN vs ADAPTIVE6G

a specific slice is critical for future wireless systems. It is more difficult considering the time it takes to train a large dataset with additional energy and resources. We do not assume scarce resources; therefore, load prediction is exclusively concerned with matching

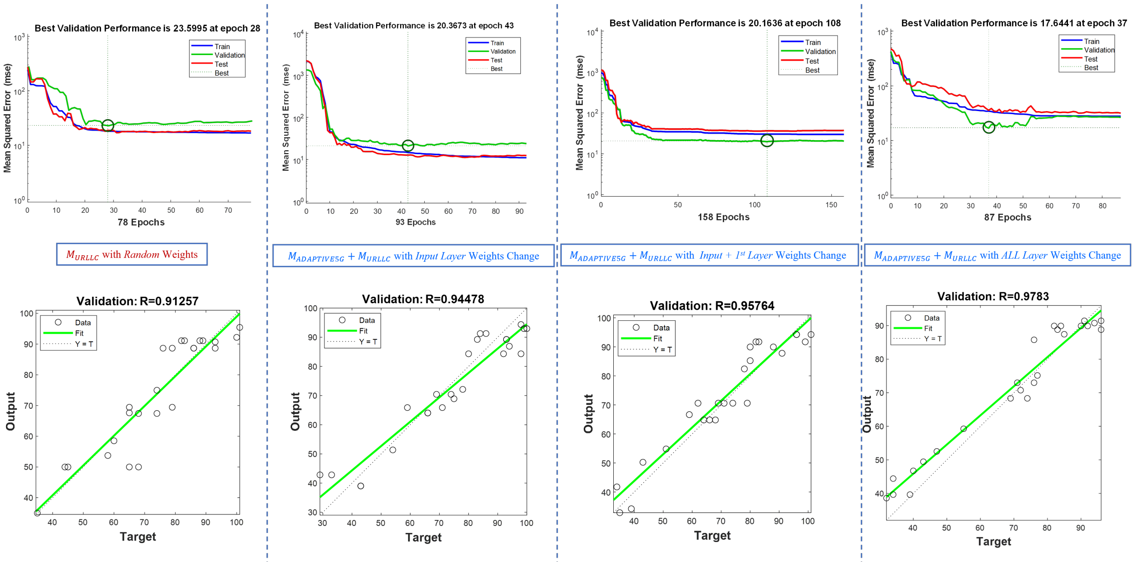


Figure 29: Model Performance (Slice C - URLLC): Traditional DNN vs ADAPTIVE6G

resource allocations to the predicted demand.

3.6 Objective 6: Numerical Evaluation of *ADAPTIVE6G*

We have summarized our ADAPTIVE6G and traditional neural model simulation using two metrics: Correlation Coefficient 'R' and Mean Squared Error 'MSE', as captured in Table 4. In the first experiment, we evaluated both models using inputs from all three slices. The second half of the experiment includes input from individual slices with the original aim of predicting the total network load. For simplicity purposes, we took an average from all three simulations with learned weights and evaluated it against the model initialized with random weights. Our Correlation Coefficient value 'R' evaluation indicated an improvement of 5.97% (0.8811 to 0.9338), 4.14% (0.9260 to 0.9643), and 3.76% (0.9167 to 0.9512) in predicted output when using individual slice data as input, i.e., Slice A (eMBB only), Slice B (mIoT only), and Slice C (URLLC only) respectively. An improvement of 1.4% (0.9614 to 0.9749) was observed in predicted output using ADAPTIVE6G for network load estimation using the Total Load (A+B+C) scenario. Both scenarios demonstrate that the ADAPTIVE6G model predicts the output strongly related to the inputs and a smaller dataset in a best-case scenario.

Also, MSE evaluation indicated a decrease of 32.82% (38.8357 to 26.0917), 7.17% (22.2998 to 20.7010), and 12.8% (24.0873 to 21.0053) across Slice A (eMBB), Slice B (mIoT), and Slice C (URLLC) respectively when using individual slice data as input using *ADAPTIVE6G* over the traditional neural network. Also, a decrease in MSE error of 38.58% (20.9447 to 12.8637) was observed using Total Load as input for predicting the network load. This evaluation also enables us to comprehend the less optimal

Table 3: Summary Results Traditional M_{DNN} and $M_{ADAPTIVE6G}$

Showing Metrics Evaluations resulting from 5 Iterations between M_{DNN} and $M_{ADAPTIVE6G}$	**Total Epochs**	**Best Epochs**	**MSE**	**R Value**
Total Load (A + B + C) - M_{DNN} Random Weights	154.4	104.4	20.9447	0.9614
Total Load (A + B + C) - $M_{ADAPTIVE6G}$ with Input Layer Weight Changes	76	39.2	14.6195	0.9716
Total Load (A + B + C) - $M_{ADAPTIVE6G}$ with Input + 1st Hidden Layer Weight Changes	89.8	48.6	12.4949	0.9747
Total Load (A + B + C) - $M_{ADAPTIVE6G}$ with All Layer Weight Changes	105.4	53.8	11.4767	0.9782
	Average of $M_{ADAPTIVE6G}$		**12.8637**	**0.9749**
Slice A (eMBB only) - M_{DNN} Random Weights	205.4	155	38.8357	0.8811
Slice A (eMBB only) - $M_{ADAPTIVE6G}$ with Input Layer Weight Changes	98	69	27.0036	0.9226
Slice A (eMBB only) - $M_{ADAPTIVE6G}$ with Input + 1st Hidden Layer Weight Changes	111.4	61.8	26.2053	0.9318
Slice A (eMBB only) - $M_{ADAPTIVE6G}$ with All Layer Weight Changes	118.4	70	25.0662	0.9469
	Average of $M_{ADAPTIVE6G}$		26.0917	0.9338
Slice B (mIoT only) - M_{DNN} Random Weights	91.2	61.2	22.2998	0.9260
Slice B (mIoT only) - $M_{ADAPTIVE6G}$ with Input Layer Weight Changes	86	52	21.2960	0.9602
Slice B (mIoT only) - $M_{ADAPTIVE6G}$ with Input + 1st Hidden Layer Weight Changes	90.2	52.2	20.7114	0.9649
Slice B (mIoT only) - $M_{ADAPTIVE6G}$ with All Layer Weight Changes	107.4	56.8	20.0957	0.9677
	Average of $M_{ADAPTIVE6G}$		20.7010	0.9643
Slice C (URLLC only) - M_{DNN} Random Weights	138.6	88.8	24.0873	0.9167
Slice C (URLLC only) - $M_{ADAPTIVE6G}$ with Input Layer Weight Changes	98.6	65.4	22.7898	0.9429
Slice C (URLLC only) - $M_{ADAPTIVE6G}$ with Input + 1st Hidden Layer Weight Changes	113.0	77.6	20.1191	0.9525
Slice C (URLLC only) - $M_{ADAPTIVE6G}$ with All Layer Weight Changes	116.2	66.2	20.1070	0.9584
	Average of $M_{ADAPTIVE6G}$		21.0053	0.9512

scenarios when considering updating weights on only a subset of hidden and input layers. As quantitatively demonstrated in the results, when we update only the input layer weights or the input with hidden layer weights, we certainly obtain better results than using '$RandomWeights$'. *ADAPTIVE6G* only needs data from a single slice for load forecasting on that slice. These experiments further prove that using a smaller dataset, the *ADAPTIVE6G* model converges fast and yields better results (fewer errors). One of the key design goals for B5G systems leading to 6G is to improve network scalability, reliability, latency, and efficiency while reducing operational costs. Using frameworks like *ADAPTIVE6G* will be critical to achieving that goal to a certain extent. It will help achieve prediction accuracy with less error and improve energy efficiency by reducing the time it takes to train the model.

As shown in Table 4, we have also evaluated ECO6G against classical ML algorithms. We observe a significant improvement with the ECO6G model in both total load and individual slice load prediction scenarios. On average, we computed a 41.42% improvement in MSE values and a 7.92% accuracy improvement in the case of the total load

Table 4: Summary Results Classical ML Models

Metrics Evaluations between Classical ML Models	**MSE**	**R Value**
Total Load (A + B + C) Random Forest	20.62	0.92
Total Load (A + B + C) Decision Tree	24.73	0.87
Total Load (A + B + C) Linear Regression	20.53	0.92
Average of ALL ML models	**21.96**	**0.90**
Total Load (eMBB Only) Random Forest	24.84	0.87
Total Load (eMBB Only) Decision Tree	25.14	0.84
Total Load (eMBB Only) Linear Regression	29.99	0.88
Average of ALL ML eMBB models	**26.66**	**0.84**
Total Load (mMTC Only) Random Forest	24.21	0.89
Total Load (mMTC Only) Decision Tree	24.44	0.88
Total Load (mMTC Only) Linear Regression	21.98	0.91
Average of ALL ML mMTC models	**23.54**	**0.89**
Total Load (URLLC Only) Random Forest	25.02	0.87
Total Load (URLLC Only) Decision Tree	25.22	0.86
Total Load (URLLC Only) Linear Regression	30.00	0.81
Average of ALL URLLC Only ML models	**26.75**	**0.85**

scenario. Furthermore, in the case of the individual slice scenario, we saw an 11.89% improvement in MSE values and 10.49% in model prediction accuracy. This result supports our ECO6G superiority over classical ML, traditional DNN, and statistical modeling.

3.7 Conclusion and Contribution

This research proposed a novel resource optimization framework for network slicing architecture in B5G and 6G systems, realized through the TL-based framework *ADAPTIVE6G*. The developed framework considered 'total load from all network slices' and 'load from individual network slices' to forecast the total traffic demand. The *ADAPTIVE6G* framework can enable network operators to configure slice resource automation more precisely, resulting in better management of network resources by avoiding excessively overprovisioned or under-provisioned resources in B5G systems. The simulated results demonstrate a considerable performance improvement and reduced error compared to a traditional neural network algorithm. To our knowledge, this is the first attempt to develop an adaptive framework that enables network resource management, especially for the network slicing architecture, which is a crucial technology for 6G.

The following specific research issues are addressed through *ADAPTIVE6G*:

- developed a novel TL approach to tackle the network load estimation problem using transfer learning in the context of network slicing using KPI information from individual slices.

- designed a knowledge-transfer framework that utilizes information from Radio Network Key Performance Indicators for network load estimation problems. These algorithms enable Mobile Network Operators to optimally coordinate their computational tasks in stochastic and time-varying network states and task arrivals.

- research topic is promising in future wireless communications for its potential to deliver optimized load forecasting for varying services while conserving energy by utilizing smaller data for training the model instead of a larger dataset and accurately estimating the future network load to avoid overestimation problems.

Published Work

ADAPTIVE6G: Thantharate, A., Beard, C. ADAPTIVE6G: Adaptive Resource Management for Network Slicing Architectures in Current 5G and Future 6G Systems. *J Netw Syst Manage 31, 9 (2023). https://doi.org/10.1007/s10922-022-09693-1* [55]

CHAPTER 4

ENERGY AND COST ANALYSIS FOR NETWORK SLICING DEPLOYMENT IN BEYOND 5G NETWORKS

4.1 Introduction

The 5G mobile communication network is a communication infrastructure that converges connectivity, intelligent edge, and the Internet of Things (IoT) from consumers to industries. 5G is revolutionizing businesses and society by enabling high-speed broadband with ultra-low latency, high capacity, massive connectivity, and reliability. To achieve sustainable development goals and create an environmentally conscious infrastructure to improve people's living standards, it is of utmost importance that the 5G network provides high speed and reduced latency with significantly lower network energy consumption. The 5G standard enables the Mobile Network Operator (MNO) to optimize the Quality of Service (QoS) and improve the Quality of Experience (QoE) for the end-users with the help of KPIs metrics such as network load, battery level, and signal strength. These 5G KPIs then guarantee both networks and device efficiency, which has always been the fundamental concern for MNOs and device manufacturers from the optimization standpoint. When combined with ML, 5G can further help grow businesses efficiently and grant consumers access to more information faster than ever. On the path to the future generation networks, we must develop an AI/ML-defined network infrastructure that is energy efficient and can learn from its dynamic environments [56].

5G is an inherently greener technology with more data bits per kilowatt (kW) energy than previous generations of wireless technology. However, the exponential growth in data traffic necessitates additional Energy Efficiency (EE) and Carbon dioxide (CO_2) reduction measures. The Global System for Mobile Communications Association (GSMA) found that the 5G data traffic has grown exponentially since its commercialization. By 2025, it is anticipated that the 5G data traffic will be eight times higher than fourth-generation (4G) / Long Term Evolution (LTE), and twelve billion devices will be connected to the 5G and IoT. These subscribers are expected to consume 5-10 times more than 4G (LTE) subscribers. The MNOs will need more ways to keep network energy consumption low as 5G services mature. According to the GSMA Intelligence Report, 67% of mobile service providers anticipate rising energy expenditures. Although 5G is more energy-efficient, increasing traffic demand and complicated use cases will increase the total energy consumption. On the positive side, the mobile industry has collaborated to build a climate action plan to attain net-zero greenhouse gas emissions by 2050, with over 30 percent of carriers making public commitments. The MNOs plan to optimize 5G networks for energy efficiency to reduce their carbon footprints and electricity bills using ML models to improve traffic prediction accuracy.

Developing 5G optimization strategies for EE that address data processing capacity and latency concerns is critical, especially for network slicing in the 5G architecture. Slicing a network refers to the process by which a network operator divides a single physical network into logically distinct networks. Networks are established to provide specialized networks for diverse service providers with varying characteristics. Currently, the third generation partnership project (3GPP) has defined three network slices:

enhanced Mobile Broadband (eMBB), massive Machine-type Communication (mMTC), and Ultra-Reliable Low Latency Communication (URLLC). To efficiently deliver these tailored services with varying KPI requirements, operators must employ more integrated and sophisticated methods than they did in 4G. Additionally, 5G's cloud-based architecture, which enables greater scalability and elasticity, is a significant differentiator from its predecessor, which allows operators to deploy new network functions (NFs) without incurring additional Capital Expenditure (CAPEX) to meet demand better.

For decades, the MNOs have prioritized throughput, coverage, and data latency for building networks. However, due to environmental and economic concerns, network energy efficiency has recently emerged as a significant factor for next-generation network deployments. With the advent of high-capacity traffic services, wireless data traffic has increased exponentially; this increase in wireless data traffic degrades the existing network's efficiency. To maintain QoS and per-bit cost, network operators must increase data traffic exponentially. Focusing on high-data-rate services increases the network's energy consumption, posing environmental and financial concerns. Hence, in modern wireless network operation and design, the MNOs must consider EE as one of the KPIs due to environmental, economic, and operational concerns.

4.2 Related Work

The 3GPP Release 17 [57] work item has limited use cases, requirements, and solutions for measuring the energy efficiency of 5G networks, including Next Generation RAN, core network, and network slices, for optimizing the energy efficiency or managing energy savings in 5G. Energy efficiency KPIs have been defined for network slices, including eMBB, URLLC, and IoT. However, V2X still needs to be addressed. Also, there needs to be a definition for the URLLC network slice reliability EE KPI in 3GPP reports today. DeepSlice [19], and Secure5G [20] studied the network slices in 5G systems by applying DNN techniques. We have demonstrated network slice selection for all UE types, including unknown devices, load balancing techniques in case of slice failure, and security of these slices in case of a DDoS attack. We have used various KPIs such as the 5G QoS Identifier (5QI), Packet loss rate, Packet Delay budget, UE types, Day, and time to simulate the models.

Using the cellular traffic data set consisting of three different traffic types (SMS, phone, and web), the authors in [58] trained LSTM to have an optimum slice resource allocation for vehicular networks to reduce the total system delay and improve traffic prediction accuracy. To further reduce the delay, the authors designed a Convolutional LSTM (ConvLSTM) based traffic estimation on estimating the traffic of complex slice services. Using a hybrid learning methodology, the authors in [59] put forth a 5G network slicing model in a step-by-step process starting with data collection comprising attributes viz device type, duration, PLR, packet delay budget, BW, delay rate, speed, jitter, and modulation type. It is then followed by optimal weighted Feature extraction (OWFE), further optimized using the combination of algorithms, Glowworm Swarm Optimization (GSO) and Deer Hunting Optimization Algorithm (DHOA) calS-DHOA, and finally slicing classification (eMBB, mMTC, and URLLC) using the Deep Belief Network (DBN) and NN. The authors targeted to prove better accuracy of the proposed algorithm. Our work differentiates from [59] in that we propose a TL-based DNN model for improving 5G energy efficiency (EE), ensuring lower learning time and faster convergence.

The paper [60] addresses the issue of maintaining QoS for the industrial IoT use case using network slicing. It implements a network-slicing architecture over the SDN-based LoRaWAN. The DDPG-based slice optimization algorithm enables the LoRaWAN to be autonomously aware of the different slice attributes viz transmission power and spreading factor to ensure there is no performance inefficiency and lack of resource availability for any network slices. The paper proposes a TL-based multi-agent DDPG (TMDDPG) algorithm for an accelerated learning process. The paper attempts to establish the superiority of its proposed algorithm by evaluating the different slice's performance concerning delay, EE, and PLR for DDPG, DQN (Deep Q Network), and its TMDDPG. Our work evaluates ARIMA, ETS, and DL models using random weights against the DL model using learned weights to investigate traffic forecasting for improving the 5G energy efficiency. We use ML techniques to estimate the total load prediction and use the predicted load to calculate energy efficiency.

The authors in [61] tackle the inherent disadvantage of slow convergence of DRL-based solutions for Radio Resource Management (RRM) related use cases RAN slicing, power and handover control, link adaptation, and packet scheduling that would drasti-

Table 5: Comparison of ECO6G against state-of-the-art methodologies

Sr. No.	Related Work	ECO6G Work
[58]	Use of the cellular traffic types (SMS, phone, and web), to train LSTM for slice resource allocation	Use of the network KPIs: RRC, RSSNI, and PDU to train a DNN for predicting total load estimation
[59]	5G network slicing model using the DBN and NN to improve accuracy	TL-based DNN model for improving 5G energy efficiency and ensuring faster convergence
[60]	DDPG slice optimization and TL based multi-agent DDPG (TMDDPG) for accelerated learning by evaluating delay, EE, and PLR for DDPG, DQN and TMDDPG	Evaluation of ARIMA, ETS, and DL models to investigate traffic forecasting for enhanced 5G energy efficiency
[61]	DRL based 5G RAN slicing resource allocation and TL to accelerate the learning and tackle slow convergence	Use of TL with DNN to estimate the network load using slicing KPIs, to estimate the energy efficiency and improved convergence rate.
[62]	TL-based A2C approach to increase network utility at the expense of reduced adaptability of the various network topologies.	TL approach to improving energy efficiency with an approximate OPEX savings of seven hundred eighty-six million for the MNOs in off-peak network load scenarios.
[63]	RAN slicing architecture for autonomous learning in interference affected and the TL approach to facilitate self-learning RAN slicing control.	The work in [63] targets autonomous RAN slicing, whereas our work uses the data-driven model trained on the network KPIs to estimate the EE of 5G networks.
[64]	Dynamic slicing resource allocation with an hourly dataset of live cellular network attributes recorded over five days for sites in dense urban areas fed directly to the GRU	Our dataset is captured on a real-world 5G base station using the MNO's proprietary software, including data for three sectors and network KPIs from each sector.
[65]	Comparative analysis of the transfer RL (TRL), Q-value TRL, and action selection TRL with model-free Q-learning and the model-based priority proportional fairness and time-to-live (PPF-TTL) to solve for slow convergence and lack of generalization of RL techniques	In contrast to [65], our work addresses the issue of slow convergence by proposing a comparative analysis of our ECO6G model with ARIMA, ETS, and DNN with random weights.
[66]	Use of techniques for enabling sleep mode methods in heterogeneous mobile networks with the aim of reducing power consumption.	Our work proposes to enhance the energy efficiency of the 5G network with an OPEX saving from the perspective of MNOs.
[67]	EE DRL based resource allocation for RAN slicing to improve computational and time complexity	Data-driven approach for improved OPEX savings against the conventional approaches for MNOS in varying load
[68]	DL based network slicing short-term traffic prediction for 5G transport network	Supervised ML model for forecasting traffic load and using the estimated load to evaluate EE and improve OPEX savings by a margin of 48.67% against other evaluated data-driven models

cally affect the end users QoE thereby violating the SLAs (Service Level Agreements). Therefore, the authors propose to eliminate this problem by developing a DRL-based approach to 5G RAN slicing resource allocation to investigate the exploration mechanism and reward convergence behavior of different DRL algorithms (DQN, AC, PPO, Dueling DQN, Double DQN, and A2C). In addition, the authors propose to accelerate the learning process and tackle slow convergence in DRL-based slicing resource allocation using a TL-based approach. Using TL (Transfer Learning) with our DNN model (ECO6G) also improves the convergence rate by accelerating the learning time of the algorithm. Our work employs ML methodologies to estimate the network load using slicing KPIs, subsequently used to estimate energy efficiency. Unlike the work in [61], which investigates the benefits of using a DRL-based approach to orchestrate network resources in network slicing, our work uses TL-based DNN to predict the network load and thereby estimate the energy efficiency. The work in [62] uses the A2C methodology to facilitate dynamic adaptation to changing environmental conditions. The work establishes the efficiency of the proposed algorithm (increase in the amount of the URLLC flows without degrading the eMBB performance flows) in terms of management of network resources regardless of the increased network density and service heterogeneity. The authors used a TL-based approach to increase the overall utility at the expense of reduced adaptability of the various network topologies. Our work uses the TL approach to improve energy efficiency with an approximate OPEX savings of seven hundred eighty-six million for the MNOs in off-peak network load scenarios. The TL approach aids in accelerating the learning process and ensures faster convergence with a faster epoch cycle.

To ensure a heterogeneous QoS in the B5G and 6G networks, it is necessary to accomplish effective RAN slicing. The authors address this issue by implementing a RAN slicing architecture in [63] that supports self-management of resources, optimizes KPIs (spectrum efficiency, energy efficiency, and QoS metrics), and autonomously learns in adversarial network conditions (interference in a multi-cell scenario). The authors additionally use the TL approach to facilitate self-learning RAN slicing control. The work in [63] targets autonomous RAN slicing, whereas our work uses the data-driven ECO6G model trained on the network slicing KPIs to estimate the EE of 5G networks. The authors in [64] propose a complete architecture for a big-date-based dynamic slicing resource allocation while maintaining the constraints of SLAs using DL. The slicing approach involves a vast hourly dataset of live cellular network attributes viz the OTTs traffic and the consumed RRC connected users licenses per vBBU recorded over five days for sites in dense urban areas fed directly to the Gated recurrent unit (GRU). Our dataset is captured on a real-world 5G base station using the MNO's proprietary software, including data for three sectors and KPIs such as RRC counts and the number of PDU network sessions, and network load from each sector.

4.3 Energy Efficiency Using Data-Driven Learning

Energy consumption is a considerable portion of network OPEX, and base stations are the radio access network's primary energy-consuming equipment. To achieve Radio Access Network (RAN) EE, turning off cells during off-peak hours is a way to reduce network energy usage; it would be ideal if the MNO could estimate the future load efficiently and configure resources accordingly. Predicting a network function overload or outage enables operators to take preventative measures (for example, avoid selecting a heavily loaded node for latency-sensitive/resource-intensive service) to ensure smooth network operation and improve the 5G customer quality experience. Other techniques for improving energy efficiency include adjusting a base station's coverage area based on its load level, favoring lightly loaded base stations to sleep, and load balancing by handing over the User Equipment (UE) to the micro or pico base station.

In contrast, network operators continue to deploy 5G and employ novel New Radio (NR) features like beam-forming, dynamic spectrum sharing, multiple-input multiple-output (MIMO), and network slicing, introducing complex system design and optimization challenges. The MNOs struggle with traditional hard-coded algorithms, which require human-machine interaction, which is error-prone, slow, costly, and cumbersome. AI, including ML algorithms, can help operators improve network management and user experience by analyzing and processing network KPIs and metrics. AI in 5G networks has captivated academia and industry to explore optimization methods for UE trajectory prediction, traffic steering, load balancing, energy-saving, and massive MIMO configuration optimization. AI and ML are enabling operators to gain new capabilities and efficiency gains. They enable network equipment to sense, reason, infer and bring novel solutions to technological issues. A holistic and end-to-end approach to AI and ML can provide a pervasive system-level approach to energy efficiency improvements spanning hardware, software, and algorithms. Energy management is a data-intensive operation; without AI, operators cannot efficiently process information and make real-time choices at scale. To implement adaptive energy management in the network slicing, the MNOs can assign different priority levels to differentiate services between slices, such as emergency services or service characteristics (e.g., number of end-users, location, average consumption).

Energy consumption is a significant issue, both environmentally (carbon footprint) and economically. The energy consumption of the 5G base stations is so high that electricity bills have become one of the most significant expenses for 5G providers. Costs to the MNOs are expected to increase significantly over the next five years. Studies suggest that, on average, mobile network operators spend twenty-five billion dollars annually on energy. Telco industry reports suggest energy efficiency and optimization are crucial for network transformation and climate action agendas. Energy is the only significant operational expense predicted to rise soon. 5G base station (BS) energy savings involve hardware and software, multiple power-saving features, small cell deployments, and new 5G architecture and protocols that can be combined to improve wireless network energy efficiency. Optimizing hardware architecture, production process, and integration of crucial core chips such as base-band processing, digital intermediate frequency, and radio-frequency modules reduces hardware energy consumption on AAU (Active Antenna Unit) and RRU (Remote Radio Unit).

Low-traffic areas account for 70% of network sites in most cases but carry only 25% of the total traffic. However, only 30% of network sites are in medium to high-traffic areas, yet they carry 75% of all traffic. Historically, the industry has prioritized high-traffic sites and neglected low-traffic networks. This is a massive opportunity for MNOs to use predicted load as one element to design network and energy optimization strategies, where BS resources must be scheduled according to service load to conserve energy. In, the authors have compared the load forecast for a single cell using several prediction techniques, as shown in Fig. 30.

The simulation results of load prediction are based on the consumption of fifty physical resource blocks (PRBs). Compared to each standalone sub-model, the ensemble learning model has significantly enhanced the accuracy of its predictions. The developed

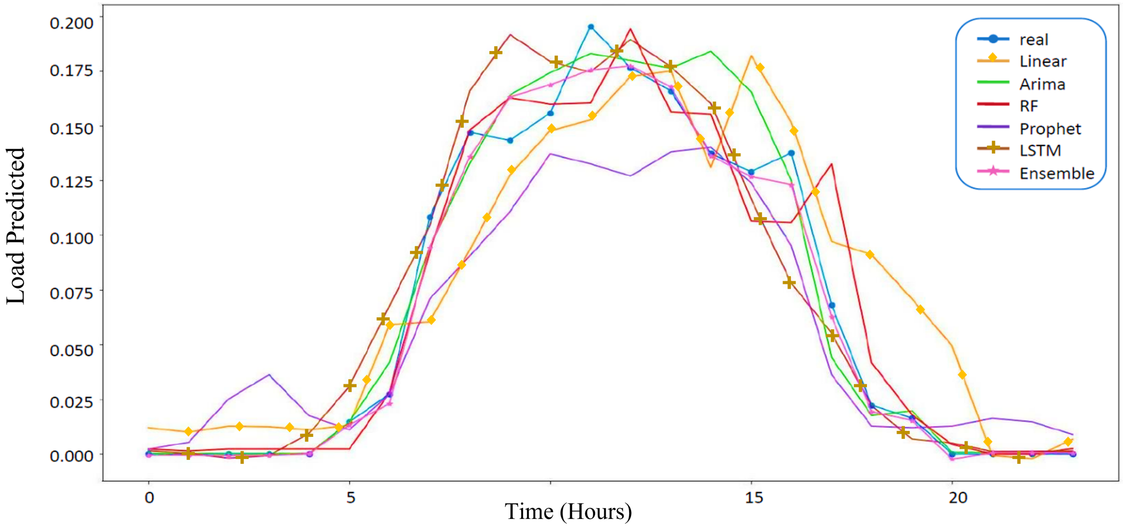


Figure 30: Comparative Study of Load prediction using different ML algorithms

ensemble learning method reduces the average Mean Absolute Error (MAE) by 0.008. The load prediction models include Arima, Prophet, Random Forest (RF), Long Short-Term Memory (LSTM), and Ensemble learning. The models use historical and current loads to predict future loads, so historical traffic loads are considered when building and training the ML model.

Because ML models can swiftly analyze substantial amounts of data, it improves the potential for network-wide energy savings. AI algorithms can be optimized to assess real-time demand, traffic patterns, and network resource availability and translate these data into actionable insights. In that case, more efficient resource management and network planning can be achieved, which is the primary motivation for this study.

4.4 Current Energy and Power Challenges in 5G Networks

Power saving has been a challenge since the second generation (2G) era of wireless communication. The massive MIMO and high output power needs of 5G have worsened this issue. Massive MIMO and high output power requirements to service the increasing number of connections and data traffic will further raise energy demands. Running redundant network resources ensures excellent network availability, even if other resources fail, but wastes much energy. Network traffic varies by time and place, so different elements of the RAN infrastructure in each area can be put to sleep for predetermined periods. The more components of that BS that are turned off, the more energy is saved. There is an opportunity to develop more profound and extended sleep periods when no or fewer data transmissions occur, lowering the network's overall energy consumption.

Currently, industries are experimenting with AI-powered solutions for simple operations like shutdown and sleep cycles for cells serving users based on estimated traffic patterns modeled. These models are built upon historical patterns, weather, local events, and other variables that can save energy by turning off power amplifiers, transceivers, and antennas. Such solutions can also help with load balancing, intelligent beam forming, interference reduction, and better spectrum utilization, among other things. In cellular networks, base stations consume the most power; studies show that base stations consume between 60 and 80% of all cellular network power even when it is not serving any users. The increasing traffic demand and complicated new 5G use cases mean that 5G consumes more energy than earlier wireless technologies. Thus, putting a Base Station

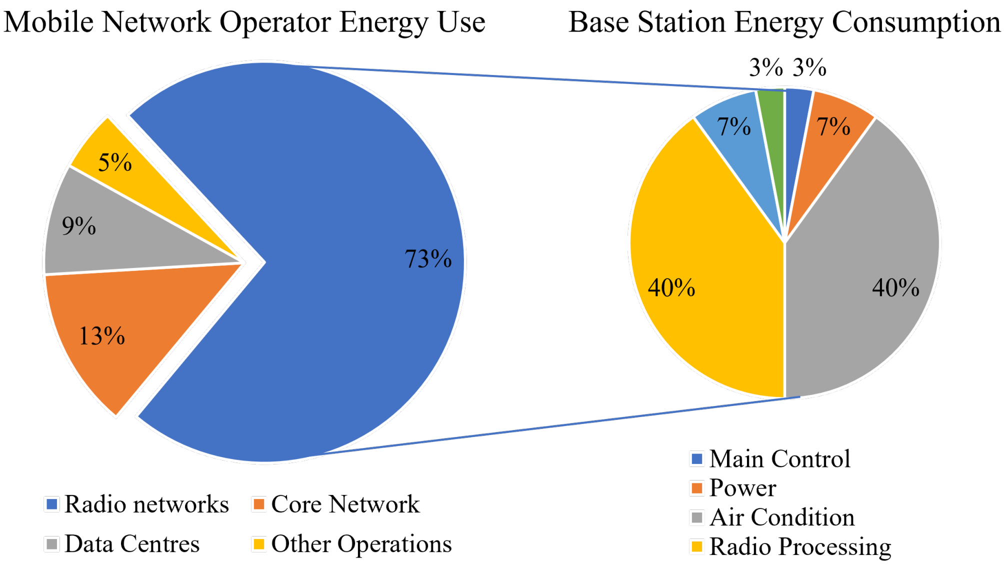


Figure 31: Power Consuming Elements in Mobile Cellular Networks

(BS) to sleep or turning it off entirely when there is little user traffic can help reduce cellular network power consumption. Also, BS experiments are application-layer friendly and do not necessitate network changes and standardization, making them less costly and easier to evaluate and implement.

As shown in Fig. 31, most network expenses are attributable to energy consumption (fuel and electricity). Base station sites are the primary energy consumers in a mobile network, requiring around 73% of a typical operator's total energy in 2021, according to a GSMA analysis of thirty-one MNOs. RAN energy consumption comprises the eNodeB (4G BS), gNodeB (5G BS), as well as the energy consumption of associated equipment, such as air-conditioning (AC), inverters, and rectifiers. The core network energy consumption comes from the network operations centers, value-added service platforms, and any energy consumption connected with backhaul transport. Furthermore, the energy spent by data centers includes the physical locations that host the infrastructure of operators, including Operational Support Systems (OSS) and Business Support Systems (BSS). It is important to note that the AC is still running and consuming the same amount of power, even when the network has low and medium load scenarios and other associated equipment.

The current NR design supports basic energy-saving measures such as a gNB that can turn capacity cells on/off to save energy. The gNB autonomously makes decisions without knowing the impact on neighboring nodes or the overall network energy consumption. When neighboring nodes make conflicting decisions, the situation worsens. Additionally, the current energy-saving tools are limited to cell deactivation. With NR beam-forming and multi-layered radio transmission structure, reducing the load in a coverage area or modifying the configuration of RAN nodes for coverage and capacity can reduce energy consumption. The optimal EE decision is conditional on many variables, such as node load, RAN node capabilities, KPI/QoS requirements, active UEs and mobility, and cell utilization; hence, optimizing EE at the RAN level using pre-defined and hard-coded rules is difficult.

ML can maximize the energy efficiency of a network by collecting pertinent data and taking the appropriate action. Utilizing a solution at the RAN level can reduce network energy consumption while maintaining coverage, capacity, and quality of service. The ML model could use internal node information, neighboring RAN node information, and UE assistance information to make an EE determination (such as offloading UEs, deactivating/activating capacity cells, and adjusting node configuration) and communicate it to neighboring nodes. Neighboring nodes can provide feedback on the energy efficiency of a decision, and UE may also indicate if performance requirements are not met, indicating that the network should modify EE. The potential for an ML-assisted solution can be enhanced by exchanging RAN-level metrics for energy savings/consumption. MNOs can introduce an energy status that can be communicated between RAN nodes. Such indicators can assist neighboring nodes in understanding a node's energy efficiency preferences, which can be considered when deciding on EE actions that may affect network energy consumption.

When a base station is powered on, its power consumption is proportional to the traffic volume. Fig. 32 demonstrates that around sixty percent of a BSs (Base Stations)

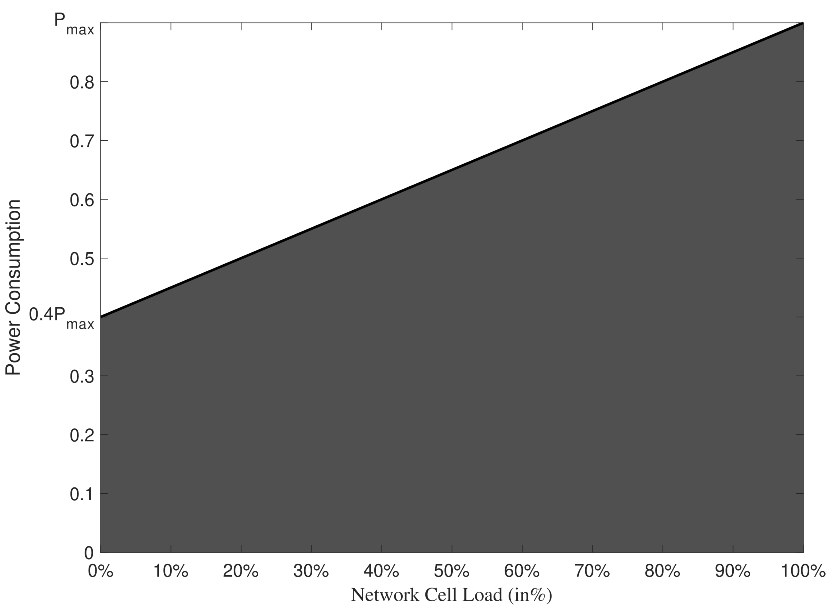


Figure 32: Power Consumption Factor with Traffic Load

radio power usage scales with traffic load. When the predicted traffic volume is below the threshold, the cells can be turned off, and the UEs can be moved to the new target cell. ML algorithms can train the relevant model and predict the next period state, especially traffic load. In Rel-16, a new mechanism for exchanging the current load status of RAN nodes was added, which is used as input for Mobility Load Balancing (MLB) / Energy saving algorithms. Additionally, based on our study and analysis, we believe that considering the predicted load status is beneficial, particularly for cells whose load status varies rapidly and follows a consistent pattern each day, especially in the case of network slicing, where logical networks can be managed independently.

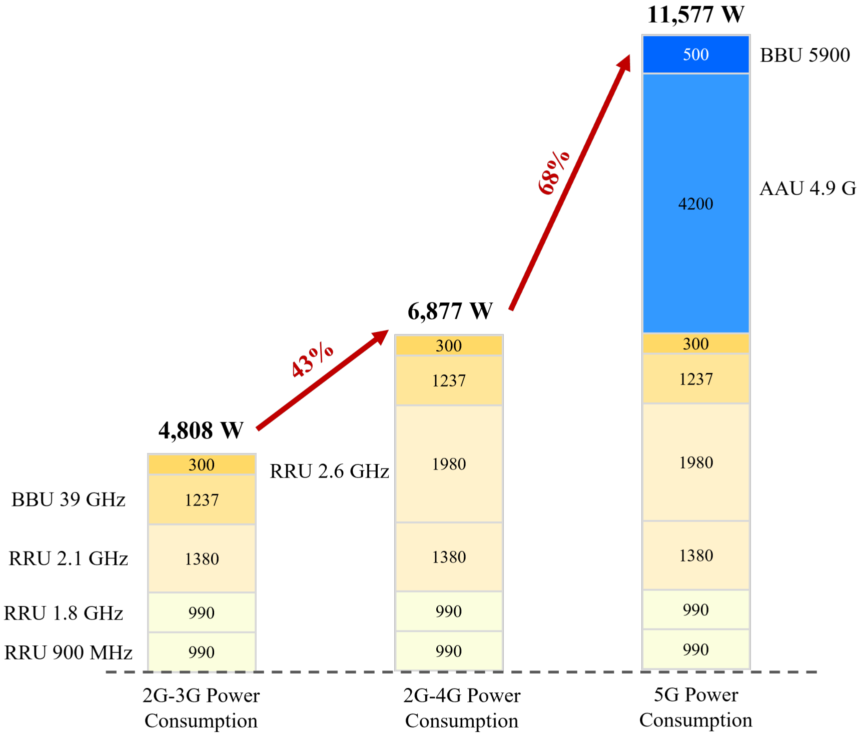


Figure 33: Power Consumption by different technologies in Cellular Networks

Network Equipment Manufacturer reports that compared to 4G, the power consumption per unit of traffic (Watt/bit) is drastically reduced, whereas 5G's power consumption increases. The percentage of sites with more than five frequency bands will rise from 3 percent in 2016 to approximately 43% in 2023. As shown in Fig. 33, the maximum power consumption of a 5G site will be greater than 10 kW and will be doubled if more than ten frequency bands are used. A typical 5G site consumes more than 11.5 kilowatts of power, around 70% more than a base station that uses a mix of 2G, 3G, and 4G radios. Network Equipment manufacturer forecasts that Massive MIMO alone can raise cell energy usage from 5-7 kW per 4G site per month to more than 20 kW per 5G site. China's 5G energy usage is projected to increase by 488% by 2035, reaching 297 billion kWh.

4.5 Proposed ECO6G Framework

The 5G NR standard was developed with an understanding of typical radio network traffic and the requirement for radio network equipment to support sleep states. The base station can be put to sleep when no traffic is present to conserve energy. Even in heavily loaded networks, base station resources are often unused. Most base transceiver station (BTS) hardware components remain active to transmit 4G or 5G mandatory idle mode signals like synchronization, reference, and system information [69]. The MNOs expect B5G deployment solutions to be low cost, can be deployed fast, with low energy, and simple Operations and Maintenance to improve carrier investment efficiency. To mitigate these challenges, we propose an energy optimization method using the learning from the predicted load, which we have simulated using Deep Neural Network (DNN), Transfer Learning (TL), ARIMA, and ETS models. The proposed ECO6G model is based on TL concepts, which utilizes a pre-trained model M_DNN, trained on a larger traditional DNN model.

The performance of any ML or DL model depends on the training data set size, quality, and relevance. Real-world data sets are disorganized and unstructured. Finding a balanced data set or working with an imbalanced data set is difficult, especially with the lack of field data for network slicing, which is not yet deployed in the production network. We believe that the field data is necessary for the ML model to function in real-world environments and for training, validation, and testing to ensure the validity and robustness of the model. Our ECO6G dataset is developed from a real-world 5G base station's measurements using MNOs proprietary software, which includes data from one base station with three sectors and KPIs such as RRC, number of PDU sessions, and the total network load [70]. Dataset was collected over 52 weeks, of which forty-seven weeks were used for training, and the remaining five weeks were used for testing and validation. As network operators have yet to deploy network slices, we do not have the availability of actual slicing data. We have used the 3GPP specification TS 28.554 [71] definitions to augment the data for network slicing KPIs such as RSSNI and PDU session counts.

Traditional statistical methods use linear processing, whereas ML methods use non-linear algorithms to achieve minimization objectives. This paper employs four primary approaches to achieve time-series forecasting methods and comparisons: ARIMA, ETS, DL model using random weights, and a DL model using learned weights. The most challenging aspect of time series problems is that they predict an uncertain future. Predictions are never accurate and are always subject to variance, and it is challenging to discover and learn underlying patterns in time series data. Typically, patterns are categorized as trends, seasonality, and cycles. In most time-series data, these patterns are strongly interconnected, and it is difficult to distinguish and locate them due to short data length, noise, and outliers. In the past, univariate time-series analysis and prediction problems were primarily addressed; however, multiple time-series data have gained prominence in recent years. We have performed a comparative study between ML and statistical modeling to rule out any issues between model superiority.

ECO6G utilizes TL, where weights are learned from a traditional deep neural network mode (M_DNN) trained on a larger data set comprising the three KPIs from each slice and the total network load from one base station. The knowledge transfer in the case of TL eliminates the need to train an ECO6G model from scratch, resulting in faster convergence using a smaller training sample size. Finding sufficient and high-quality training data is one of the most challenging tasks for conventional ML techniques. By leveraging the trained knowledge from similar domains with high-quality data, TL can circumvent this issue. Instead of learning from scratch, as with conventional ML approaches, the training process for ECO6G can be significantly accelerated by incorporating knowledge from a M_DNN model. Instead of maximizing the Quality of Service (QoS), we argued that better EE could be achieved by targeting satisfactory QoS levels. Furthermore, accurate prediction of estimated network load based using recent (more real-time) data, which is also smaller in size, can be used for predictive analytics.

In this section, we detail the working of proposed ECO6G as shown in Fig. 34:

Step I The ECO6G framework initializes by training the traditional neural model M_{DNN} using observed RRC, number of PDU sessions, and the total network load from all Slices - A (eMBB), B (mIoT), and C (URLLC), i.e., D_{TOTAL}. Network operators can deploy many slices; we are considering three standard slices for our evaluation per standardized 3GPP SST values. We have employed five-layer Deep Neural Networks: Input (features), 3 Hidden Layers, and Output (prediction). We have tuned the model hyper-parameters by changing the number of hidden layers, learning rate, activation function, and the number of epochs for the M_{DNN} model in MATLAB using Deep Learning Toolbox and Alteryx Analytics Automation tool running on

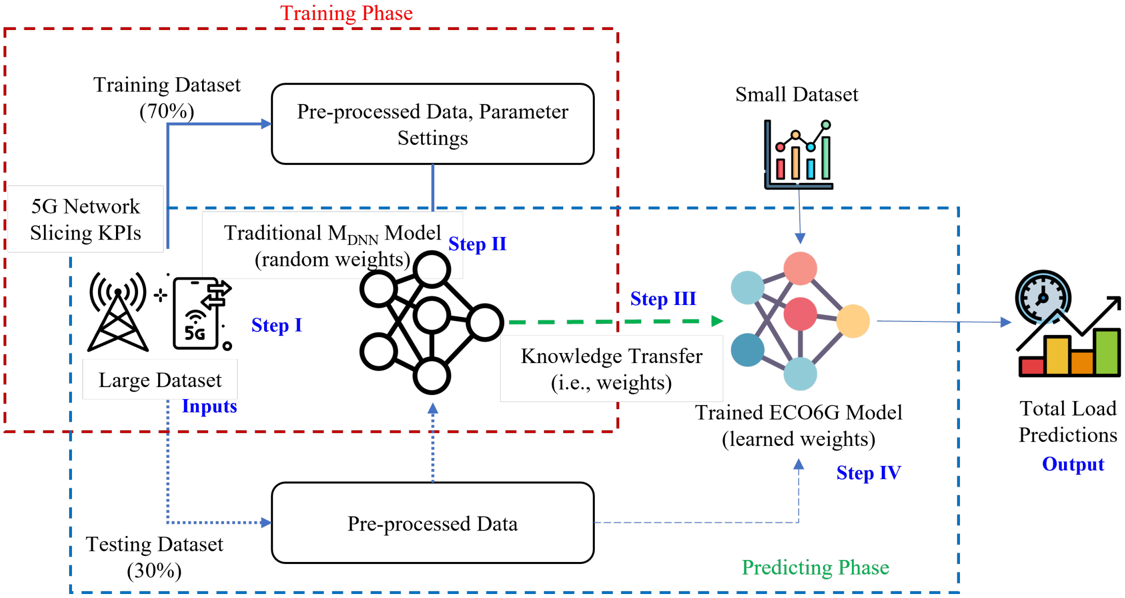


Figure 34: ECO6G Framework

Intel hardware and Windows 11 operating system. Our goal is to validate the model performance between random weights and learned weights, so we kept the DNN modeling the same for both M_{DNN} and M_{ECO6G}. The algorithm uses randomness to find a good set of weights for the data's specific input-output mapping function, such that each time the training algorithm is run, a different network with a different model is fitted. The shuffling of the training dataset before each epoch also uses randomness, resulting in differences in the gradient estimate for each batch.

Step II First, we train the M_{DNN} multi-layer model using a feed-forward backpropagation network with initialized random weights (stochastic gradient descent). A forward pass through the network is accomplished by iteratively computing each neuron in the subsequent layer until the output is achieved. We evaluate the output quality based on a cost function C and the desired result in the output layer. Mean squared error (MSE) is used as a loss function for evaluation.

Step III A backward pass is then used to optimize the cost function C after the first result has been obtained by readjusting the weights and biases. We aim to optimize the output by adjusting the entire neural network. Based on this, we can calculate the total loss and determine the model's suitability (good or bad), and here weights are adjusted to obtain the least loss. After backpropagation, we capture each layer's computed weights (learned weights) for TL parameters and define these trained weights as M_{ECO6G}.

Step IV Now instead training the M_{ECO6G} with 'random weights', we initialize M_{ECO6G} using learned weights and re-train for smaller datasets D_{eMBB}, D_{mIoT}, D_{URLLC} from individual slices, which are subsets of D_{TOTAL} to predict total network load.

In ECO6G, we are capturing weights on the final layer (i.e., output layer); however, we can capture weights in the initial layer and middle layer as referenced in [55]. The model's performance depends on the neural network architecture, the change in neurons, the hidden layer, hidden layer, influences the model performance, and energy consumption. The more time the model takes to converge, the more energy it consumes. The complexity of a NN-based algorithm primarily depends on the number of nodes in each NN layer, total training examples, M, and a number of epochs, N. The time complexity $\mathcal{T}$ of the ECO6G algorithm can therefore be approximated as

$$\mathcal{T} = O(M * N * \#nodesinlayer(i) * \#nodesinlayer(i - 1)) \tag{4.1}$$

ECO6G model pseudo-code can be written as:

Table 6: ECO6G Pseudo Code

Algorithm 1: ECO6G Training and Validation

```
1: Set parameters
2: $\theta \in (0,1)$ :weights/parameters
3: $b \in (0,1)$ :bias
4: $\alpha \in (0,1)$ :learning rate to control change in $\theta$ and $b$
5: $\sigma \in (0,1)$ :sigmoid activation function
6: $\sum = x_i\theta_i$ where $x_i$ is the input, $D_{train}$ consisting of RRC, RSSNSI and PDU of each of the three slices from the network and devices
7: Weighted sum value $z = x_i\theta_i + b$
8: $D_{train} \leftarrow$ Training data for the network load of size 7729 X 9
9: $D_{val} \leftarrow$ Training data for the network load of size 169 X 9
10: Initialization of the multi-layer model, $M_{DNN}$ consisting of parameters $\theta$ in [0,1]
11: Training of $M_{DNN}$ with $D_{train}$
12: Predicting the network load with error function $MSE_i = \frac{1}{n}\sum_{i=1}^{n}(y_{actual} - y_{predicted_i})^2$
    $MAPE_i = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{actual}-y_{predicted_i}}{y_{actual}}\right|$
13: Optimization of cost function $J(\theta)$ through backpropagation and gradient descent with $J(\theta)$ in step 12 until convergence
14: Selection of the learned parameters ($\theta^{(1)}$, $\theta^{(2)}$, $\theta^{(N)}$) representing the pre-trained model as $M_{pretrained}$
15: Using the $M_{pretrained}$ parameters for validating $D_{val}$ where $D_{val}$ $\in$D$_{Total}$
```

4.6 Proposed ECO6G Framework Evaluation

With TL, most data are trained by other source domains before transferring the trained models to the target domain, reducing the computing requirements for target domain training. This is useful for wireless devices with hardware constraints, such as smartphones, IoT, and edge devices. Additionally, only knowledge, such as model weights and biases, must be transferred, reducing communication overhead [72]. Consequently, this can significantly improve the learning rate, which is especially important for developing applications with ultra-low latency for future wireless networks. Conventional ML training is computationally intensive.

ECO6G uses all the layers of a pre-trained M_DNN model for initialization; this strategy is anticipated to be advantageous because the initial layers capture more typical characteristics, and training only the final layers is more computationally efficient. ML and conventional statistical methods aim to enhance prediction accuracy by minimizing a loss function, such as the mean of squared errors. A high loss indicates that the model performed poorly, and a low loss indicates a good-fit model. Cross-validation is used in the modeling process to determine which model performs best while remaining robust to data not encountered during training. By sampling multiple pairs of training and test data from a limited data set, one can ensure that the performance goals are met and that the extent of training has been adequate while preventing over-fitting. There is no one-size-fits-all indicator for forecast accuracy. We have used Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) metrics for evaluation purposes across all four models to evaluate the goodness of predictions.

The objectives of ML and conventional statistical methods are to enhance prediction accuracy by minimizing a loss function, such as the mean of squared errors. A high loss indicates that the model performed poorly, and a low loss indicates a good-fit model. Cross-validation is used in the modeling process to determine which model performs best while remaining robust to data not encountered during training. By sampling multiple pairs of training and test data from a limited data set, one can ensure that the performance goals are met and that the extent of training has been adequate while preventing overfitting. There is no one-size-fits-all indicator for forecast accuracy. We have used Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) metrics for evaluation purposes across all four models to evaluate the goodness of predictions.

MSE is computed by squaring differences between the predicted and actual values and averaging the result. The range of MSE is between 0 and ∞; the lower the MSE value, the more accurate the prediction model. MSE is the loss function of linear regression by default in ML. The MSE for our models can be expressed as:

$$MSE_{M\_DNN} = \frac{1}{n}\sum_{i=1}^{n}(y_{actual} - y_{predicted_{M\_DNN}})^2 \tag{4.2}$$

$$MSE_{ECO6G} = \frac{1}{n}\sum_{i=1}^{n}(y_{actual} - y_{predicted_{ECO6G}})^2 \tag{4.3}$$

$$MSE_{ARIMA} = \frac{1}{n}\sum_{i=1}^{n}(y_{actual} - y_{predicted_{ARIMA}})^2 \tag{4.4}$$

$$MSE_{ETS} = \frac{1}{n}\sum_{i=1}^{n}(y_{actual} - y_{predicted_{ETS}})^2 \tag{4.5}$$

MAPE is more robust than MSE to outliers in the dataset, and it expresses accuracy as a percentage of the error and measures the forecast error concerning actual values. The lower the MAPE value, the more accurately the ML model predicts values. MAPE less than a value of 10 percent indicates highly accurate forecasting. The MAPE for our models can be expressed as:

$$MAPE_{M\_DNN} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{actual} - y_{predicted_{M\_DNN}}}{y_{actual}}\right| * 100 \tag{4.6}$$

$$MAPE_{ECO6G} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{actual} - y_{predicted_{ECO6G}}}{y_{actual}}\right| * 100 \tag{4.7}$$

$$MAPE_{ARIMA} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{actual} - y_{predicted_{ARIMA}}}{y_{actual}}\right| * 100 \tag{4.8}$$

$$MAPE_{ETS} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{actual} - y_{predicted_{ETS}}}{y_{actual}}\right| * 100 \tag{4.9}$$

ARIMA is a time series analysis model that is fitted to time series data to better forecast future time series points. ARIMA uses trends, cyclic, seasonal, and irregular changes to characterize time features and sequences in patterns. Forecasting techniques

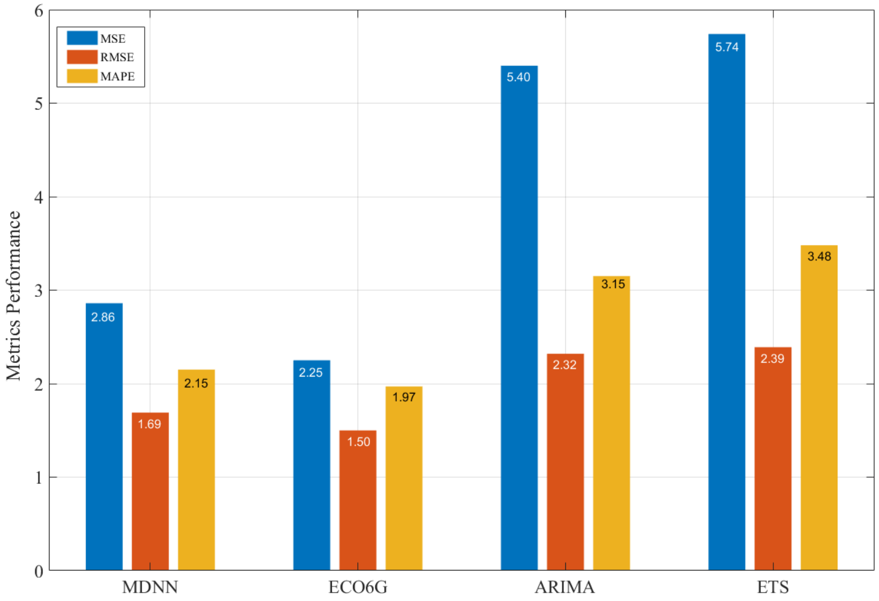


Figure 35: Model Evaluation and Metrics

based on ETS use a weighted sum of past observations, but the weights decrease exponentially. We have simulated network load for 168 hours (about one week) using all models as a comparative study.

Fig. 35 shows the performance of all models in terms of MSE, RMSE, and MAPE metrics. Our proposed ECO6G algorithm performs better than the other three algorithms in error and accuracy metrics for the given data set. In addition, our proposed algorithm has steady performance and converges faster because of pre-trained weights. Compared to the traditional neural network model (M_{DNN}), ECO6G yields 21% less error and 8.5 percent more accuracy. Table 7 also shows additional insight between various data-driven approaches and the superiority of the ECO6G model for the given data set.

Fig. 36 demonstrates all models' simulated forecasted network load results. The

Table 7: ECO6G Model Evaluation between Classical ML, Statistical Model and DNNs

Models	MSE	RMSE	MAPE	R (Accuracy)	Time to Run Models (x5)
M_{DNN}	2.86	1.69	2.15	0.9723	Approx. 16 minutes
ECO6G	2.25	1.50	1.97	0.9762	Approx. 3 minutes
ARIMA	5.40	2.32	3.15	na	Approx. 8 minutes
ETS	5.74	2.39	3.48	na	Approx. 8 minutes
Linear Regression	10.53	3.24	na	0.8810	Approx. 6 minutes
MATLAB Time-Series	15.99	4.00	na	0.8525	Approx. 8 minutes

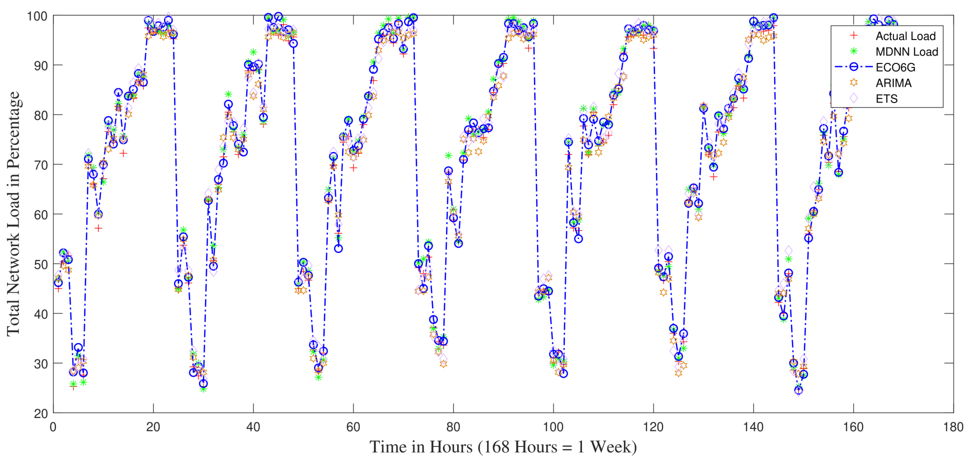


Figure 36: Simulation results of Network Load Prediction using Neural Network and Statistical Modeling

load change period is seven days and reflects the peak and off-hours variation through the day for a week, reflecting the real-world scenario. The figure also depicts the two prominent characteristics of mobile traffic and forecasting. First, the cell load is typically characterized by a strong periodicity, with periods of low load occurring from night to early morning. Second, the forecasting mechanism may produce non-negligible errors, meaning that deactivating cells at the incorrect time may significantly affect system performance. ECO6G closely follows the actual traffic load, which shows the model is reasonably accurate and provides reasonable confidence to use it against real-world network resource modeling. The difference between the average of all ECO6G estimates and the average of all actual values is only 1.10%, i.e., ECO6G is over-predicting by a little margin, which can be compensated against any spike in unusual traffic load to accommodate network resources during network planning.

Table 8: Simulation results of Average Load across all Models

Average Load %	Low load (6/24)	Medium Load (10/24)	High Load (8/24)
Actual	42.53	74.36	88.23
M_{DNN}	43.1	75.65	89.04
ECO6G	43.03	75.21	88.96
ARIMA	42.01	73.92	87.63
ETS	43.6	75.4	88.84

Table 9: Simulation results of Peak Load across all Models

Peak Load %	Low load (6/24)	Medium Load (10/24)	High Load (8/24)
Actual	73.86	94.50	99.60
M_{DNN}	74.44	95.32	99.59
ECO6G	74.48	95.23	99.78
ARIMA	75.39	92.96	96.86
ETS	75.15	94.21	99.61

Table 8 and 9 show the simulated average and peak load values across all models. Daily average traffic over 24 hours is modeled through three traffic loads (low, medium, and high) per ETSI ES 202 706-1 definition. For a week of validation, the ECO6G, M_{DNN}, and ETS models have predicted a positive delta (meaning the network would over-provision) in the case of average load against actual load for all three load scenarios. At the same time, ARIMA estimated a negative delta (under-provisioned). For a mobile network operator, it is moderately fair to over-provision to accommodate any spike in traffic but not by a large sum.

4.6.1 Objective 1: Energy Efficiency and Benefit-Cost Analysis

Load-aware metrics are crucial for the next generation of green communication networks. One of the primary objectives of 5G networks for enhancing energy efficiency is to match system capacity and power consumption with network load. The equations define the total system energy efficiency (EE) for different load scenarios in ETSI ES 202 706 and 3GPP TR 32.972 version 16.1.0.

$$EE_{global} = \sum_{low\ load} b_{low\ load} * EE_{low\ load} \tag{4.10}$$

$$EE_{global} = \sum_{med\ load} b_{med\ load} * EE_{med\ load} \tag{4.11}$$

$$EE_{global} = \sum_{high\ load} b_{high\ load} * EE_{high\ load} \tag{4.12}$$

EE (bits/joules) can be defined as the amount of traffic served per second by a base station (bits/s) divided by the power utilized by a base station to provide service

(Watt = Joule/s) multiplied by a weighting factor *b* based on the number of hours per day in each load condition. The ETSI TR load levels are 10%, 30%, and 50% for low, medium, and high loads, respectively. This weighting factor *b* takes on the value 6/24 for low load conditions in the last 6 hours of a typical day: low load, 10/24 medium load, and 8/24 high load. With the energy efficiency equations defined in 4.10, 4.11, and 4.12, the network energy efficiency can now be defined as the ability to minimize energy consumption relative to the provided traffic capacity. RAN EE measures the capability of RAN elements to sustain a much better mobile broadband data rate while minimizing BS energy consumption. The definition of RAN energy efficiency specified by the 3GPP is as follows:

EE (bits/joules) can be defined as the amount of traffic served per second by a base station (bits/s) divided by the power utilized by a base station to provide service (Watt = Joule/s) multiplied by a weighting factor *b* based on the number of hours per day in each load condition. The ETSI TR load levels are 10%, 30%, and 50% for low, medium, and high loads, respectively. This weighting factor *b* takes on the value 6/24 for low load conditions in the last 6 hours of a typical day: low load, 10/24 medium load, and 8/24 high load. With the energy efficiency equations defined in 4.10, 4.11, and 4.12, the network energy efficiency can now be defined as the ability to minimize energy consumption relative to the provided traffic capacity. RAN EE measures the capability of RAN elements to sustain a much better mobile broadband data rate while minimizing BS energy consumption. The definition of RAN energy efficiency specified by the 3GPP is as follows:

$$RAN_{EE}(bits/joules) = \frac{DataVolume}{Energy\ consumption} \tag{4.13}$$

Where; the unit of EE is bits/Joule, the unit for the data volume is bits/s/km^2, and the unit of energy consumption is Joules/km^2.

The typical and peak electrical power requirement for radio base stations (macro cell, micro cell, and pico or femtocell) related to aggregated RF power as defined in the ETSI ES 203 700 V1.1.1 is used for the energy consumption calculation. In the case of a complex macro base station, the peak power consumption is $P_{max} = 24kW$, which includes multiple frequencies across 2G/3G/4G/5G radios and MIMO configuration, and the typical consumption is $8kW$. The bits per watt can be calculated for all three loads as follows:

$$Bits\ per\ watts = \frac{Peak\ load_{low}}{P_{max} * P_{low\ load\ level}} * 10^6 \tag{4.14}$$

$$Bits\ per\ watts = \frac{Peak\ load_{medium}}{P_{max} * P_{medium\ load\ level}} * 10^6 \tag{4.15}$$

$$Bits\ per\ watts = \frac{Peak\ load_{high}}{P_{max} * P_{high\ load\ level}} * 10^6 \tag{4.16}$$

Therefore, concerning the power consumption vs. load (from Fig. 32), the power consumption values are $P_{lowloadlevel} = 0.46$, $P_{mediumloadlevel} = 0.58$, and $P_{highloadlevel} = 0.7$ respectively. Using Table 10, power consumption using the average load for a typical day across all load scenarios is calculated as shown in 11:

Table 10: Peak bits per Watts Calculation

Peak bits/Watts	Low load (6/24)	Medium Load (10/24)	High Load (8/24)	Weighted Avg for 24 hrs
Actual	6690.22	6788.79	5928.57	6477.41
M_{DNN}	6742.75	6847.70	5927.98	6514.89
ECO6G	6746.38	6841.24	5939.29	6516.87
ARIMA	6828.80	6678.16	5765.48	6411.59
ETS	6807.07	6767.96	5929.17	6498.14

Table 11: Energy consumption for a typical day

Power Consumption (in kW)	Low load (6/24)	Medium Load (10/24)	High Load (8/24)	Weighted Avg for 24 hrs
Actual	6.36	10.95	14.88	11.11
M_{DNN}	6.39	11.05	15.02	11.21
ECO6G	6.38	10.99	14.98	11.17
ARIMA	6.15	11.07	15.20	11.22
ETS	6.41	11.14	14.98	11.24

In most cases, as traffic volume and the number of utilized resources decrease, the energy consumed decreases linearly. Figure 11 depicts an analysis of typical energy consumption over a day by the BS based on the traffic pattern. The data analysis reveals that the difference between the minimum and maximum BS energy consumption is 4.15 kWh and 16.67 kWh for the ECO6G model. The graph establishes the superiority of our ECO6G model, as it can improve by 1.03% over the actual daily power consumption load for the given dataset.

Additionally, the average retail price per kilowatt-hour (kWh) in the US is USD 0.1177 for commercial uses, as of drafting this dissertation [73], so the OPEX cost to operate one BS for a day and 5 years can be calculated as in Table 12. Note how close the ECO6G calculation is to Actual (within \$4.31, only a 0.007% error).

Table 12: OPEX Cost (in $) for MNO to operate ’a’ BS for 5 years

OPEX cost per BS ($)	Low load (6/24)	Medium Load (10/24)	High Load (8/24)	Weighted Avg for 24 hrs
Actual	34,297.91	56,244.90	77,801.83	57,943.80
M_{DNN}	34,561.10	56,764.43	78,064.91	58,313.76
ECO6G	34,234.89	56,167.78	77,932.57	57,939.49
ARIMA	36,089.28	57,722.90	78,865.72	59,632.10
ETS	36,729.73	58,272.96	79,048.24	59,812.24

4.6.2 Objective 2: Plausible ECO6G Use Cases in B5G Implementation

A developed country like the United States of America has four major network operators. Suppose each operator deploys a hundred thousand 5G sites. In that case, the OPEX savings opportunity using the ECO6G load prediction model for weighted average load is close to two hundred and seventy eight million dollars over five years against other data-driven model predictions for each MNO. These savings will be multifold in billions if we consider global deployment from more than 750 mobile network operators deploying 5G where four hundred sixty-nine telecom operators from 140 countries/regions have already invested in 5G, while 182 telecoms from seventy-three countries/regions have started their own commercial 5G services.

We conducted five experiments using test data and concluded that the ECO6G model predicted 100.57% better OPEX savings for low-load, medium-load, and high-

Table 13: OPEX Cost Change (in Million $) for MNO to operate '100,000' BS for 5 years

OPEX Cost Change across models	Low load (6/24)	Medium Load (10/24)	High Load (8/24)	Weighted Avg for 24 hrs (also compared with ECO6G)
M_{DNN}	263.19	519.53	263.09	369.96 (+374.27)
ECO6G	-63.02	-77.12	130.74	-4.31
ARIMA	1791.37	1478.00	1063.89	1418.30 (+1,422.61)
ETS	2431.82	2028.06	1246.41	1868.45 (+1,872.76)

load scenarios for the given dataset against other data-driven models and accurately predicted the network load. Thus, utilizing ECO6G, we can improve the OPEX saving for different load levels. As shown in Table 13, an approximate saving of 374 million dollars against M_{DNN}, 1422 million dollars against ARIMA, and approximately 1872 million dollars against ETS can be achieved by using ECO6G in all load scenarios considering 100,000 BSs over a five-year period.

With 5G rapidly expanding globally and more sophisticated 5G-Advanced features planned in 3GPP Release-18, industry, standards bodies, and research organizations are setting the groundwork for the next generation's global sixth-generation (6G) communication standard. AI has the potential to become the foundation for the 6G air interface and network, making data, computing, and energy the new resources that can be used to achieve higher performance. As a result, 6G will have to deliver significantly more data at faster rates than current networks while also meeting extremely stringent EE goals to achieve a sustainable 6G system. This necessitates a significant reduction in the amount of energy needed to transmit a bit and the need for solutions that can be leveraged to attain energy-efficient next-generation networks.

The ECO6G model can be applied in multiple scenarios, such as enabling a 3GPP-compliant analytics service delivered in the form of statistics or predictions. Intelligence operational in real-time for Network Functions, Application functions (AFs), and operations, administration, and maintenance (OAM) services. The serving BS, for example, gets assistance data from RAN, such as load status, active UEs, QoS needs, and energy consumption status [71]. The serving node executes an ML algorithm on the collected data to choose an energy-saving action that maximizes network efficiency while maintaining service quality. The node may announce its intention to offload traffic to neighboring nodes to conserve energy. Additionally, a single analytics source in an environment with multiple vendors, especially with Open-RAN (O-RAN) concepts, could be beneficial. We are currently investigating ECO6G use cases in the core and the edge locations with application-aware output for User plane function (UPF) load, especially with Multi-Access Edge Computing (MEC).

4.7 Conclusion and Contribution

This paper investigates traffic forecasting models to enable network management to enhance the 5G OPEX savings. The report highlights the use of ML algorithms to predict the network load using network slicing KPIs and then uses the simulated predicted load to compute the OPEX savings per industry standards definition. We presented a comparative time-series study between Neural Network and Statistical Models and highlighted the proposed ECO6G superior metrics over other models. We are investigating the feasibility of ECO6G to supplement the 3GPP specified Network Data Analytics Function (NWDAF), introduced as part of Rel-16, which is intended to streamline how core network data is consumed to develop insights and take actions to improve the end-user experience. We firmly believe the ECO6G model can enable slice-level analytics and provide either statistics or forecasts of the performance of network load when used in conjunction with RAN systems, which can be further used to design and orchestrate energy-efficient network planning.

Our main contribution through this is a proposed ECO6G model using network slice KPIs to predict the network load and evaluate the OPEX savings. The contributions of our work are as follows:

- We have discussed the motivation for analyzing the energy efficiency using ML approaches and the challenges in current B5G networks in Sections 4.3 and 4.4.
- We have summarized various data-driven approaches in Network Slicing, 5G, and load forecasting. We also discuss how the proposed ECO6G model is novel and

different from the state-of-the-art.

- We have evaluated our proposed ECO6G model against the traditional DLNN with random weights and statistical time-series modeling, i.e., Auto-Regressive Integrated Moving Average (ARIMA) and Exponential Smoothing (ETS).

- We have modeled the CAPEX for an MNO according to the EE definitions defined in ETSI (European Telecommunications Standards Institute) TR 132 972 [74, 75] and highlighted the ECO6G load prediction usefulness towards the Operational Expenditures (OPEX) saving for MNOs in Section 4.6.1.

Published Work

ECO6G: Thantharate, A.; Tondwalkar, A.V.; Beard, C.; Kwasinski, A. ECO6G: Energy and Cost Analysis for Network Slicing Deployment in Beyond 5G Networks. *Sensors 2022, 22, 8614. https://doi.org/10.3390/s22228614* [76]

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

The key to realizing the promise and potential of 5G is network slicing as a service, but only if MNOs can overcome the complexity of establishing and managing many concurrent slices. This opportunity has substantial risks if not executed well. Service providers must carefully examine the level and type of control they expose to their customers while ensuring an effective monitoring and assurance system mitigates the risk. This research and dissertation discussed the approaches for delivering a resilient, secure, adaptive, resource-aware, and energy-efficient network slice infrastructure capable of supporting mission-critical applications under dynamic and unfavorable conditions without impacting communication network performance.

Our research proposed a novel resource management and network load prediction framework for network slicing architecture in B5G systems, realized through the DLNN and Transfer Learning-based data-driven methods. The developed *DeepSlice*, *Secure5G*, *ADAPTIVE6G*, and *ECO6G* framework can enable network operators to configure slice resource automation more precisely, resulting in better management of network resources by avoiding excessively over-provisioned or under-provisioned resources in B5G systems. The performed experiments using transfer Learning and derived results demonstrate a considerable performance improvement and reduced error compared to a traditional neural network algorithm, classical ML, and statistical models.

The conclusive summary of all proposed models is as follows:

- ***DeepSlice*** This research's primary objective is to develop methods and approaches for sustaining high availability and diverse service requirements for network slices in 5G networks. This research employs DL approaches to perform slice selection, slice load balancing, and slice failure schemes for network slices to achieve these objectives. We have demonstrated the benefits of using DeepSlice for accurately predicting the best network slice based on device key parameters and orchestrated handling network load balancing and network slice failure using neural network models. I am looking to extend and further improve this model to handle scenarios such as handovers, caching and predicting the future load, borrowing resources from other slices, and application-based slice management use cases.

  The developed model is critical and future-proof in ensuring the end-to-end security of the 5G network and predicting the known and unknown applications/services which are not defined/developed today by utilizing the learning from a developed deep-learning model. The proposed research will aid value in 5G-Advanced development, enabling greater control over device registration to network slices based on the applications running on those devices. It will optimize the formation of registration areas regarding network-slice utilization and accessibility. I am currently researching the feasibility of migrating *DeepSlice* to a containerized environment using Kubernetes and building an Automated deployment and management cloud-native container orchestration system for network slicing. This research can be extended further to improve this model to handle scenarios such as handovers, caching and predicting the future load, borrowing resources from other slices, and application-based slice management use cases.

- ***Secure5G*** This research has investigated the security concerns in the 5G network and presented a DLNN model to create a robust Network Slicing framework to combat DDoS attacks filtering the malicious UE connections to the 5G network. Volume-based flooding and spoofing attack scenarios were used as illustrations to evaluate the overall performance, and the detection accuracy was more than 98% with our limited dataset. The Secure5G implementation with DeepSlice will ensure the end-to-end security of the 5G network. I am considering several directions to improve further and extend the model to implement Secure5G into RAN, MEC, and Core Slicing. The future model will also include the on-device and traffic behavior learning to train the model in real-time using reinforcement and recurrent learning; this will help us achieve more detection accuracy for a secured 5G ecosystem.

  The proposed *Secure5G* can be used to protect the malicious actors from modifying the RAN or Core network slice configurations for already deployed slice instances. Network operators can update (add/delete/modify) the network functions and change the security policies while in use. These flexibilities open a floodgate for potential threats to slice operations, and malicious actors can modify the QoS/SLA for a targeted slice. I am exploring areas to implement Secure5G in Core and RAN slicing to mitigate some in-network security threats, which extends the zero-trust security paradigm to assure service continuity in cyberattack-related circumstances. Additionally, this can be further extended to minimize the network slice exhaustion problem. An attacker could potentially access the slice that could have lower-level security. For example, a slice for the consumer will have lower security compared to the security measures for Industrial or Enterprise IoT, and the malicious attacker can exhaust the resources in the consumer slice. However, the slices are virtually isolated if the network function resources are common to multiple slices (e.g., hardware resources: memory, processing power, or authentication). An ideal solution would be to pre-allocate security protocol resources for individual slices or ring-fenced resources so that a slice can run irrespective of exhaustion on other slices.

- ***ADAPTIVE6G*** This research proposed a novel resource optimization framework for network slicing architecture in B5G and 6G systems, realized through the Transfer Learning based framework. The developed framework considered *load from all network slices* and *load from individual network slices* to forecast the total traffic demand. The *ADAPTIVE6G* framework can enable network operators to configure slice resource automation more precisely, resulting in better management of network resources by avoiding excessively overprovisioned or under-provisioned resources B5G systems. The simulated results demonstrate a considerable performance improvement and reduced error compared to a traditional neural network algorithm. To my knowledge, this is the first attempt to develop an adaptive framework that enables network resource management, especially for the network slicing architecture, which is a crucial technology for 6G.

This research topic is promising in future wireless communications for its potential to deliver accurate load forecasting for varying services while conserving energy by utilizing more minor data for training the model instead of a larger dataset and accurately estimating the future network load to avoid overestimation problems. The proposed model will help build MNO's assessment of what the worst case looks like and plan toward that. That means ensuring the network is flexible enough to handle the revised view of what constitutes a worst-case scenario.

- ***ECO6G*** Sustainability is a crucial aspect of operational excellence, as more energy-efficient and environmentally viable networks provide more significant cost savings and fulfill the industry's expanding social responsibilities. This research evaluates an energy-saving method using data-driven learning through load estimation for B5G networks. The proposed *ECO6G* model utilizes a supervised ML and Transfer Learning approach for forecasting traffic load and uses the estimated burden to evaluate the energy efficiency and OPEX savings. The simulation results demonstrate a comparative analysis between the traditional time-series forecasting methods and the proposed ML model that utilizes learned parameters. Our ECO6G dataset 34 is captured from measurements on a real-world operational 5G base station (BS). Through simulations, I demonstrated that the proposed ECO6G model is accurate within $4.3 million over 100,000 BSs over five years compared to three other models that would increase OPEX cost from $370 million to $1.87 billion during varying network load scenarios against other data-driven and statistical learning models. I firmly believe the ECO6G model can enable slice-level analytics and provide either statistics or forecasts of the performance of network load when used in conjunction with RAN systems, which can be further used to design and orchestrate energy-efficient network planning.

The ECO6G model can be applied in multiple scenarios as future research direction, such as enabling a 3GPP-compliant analytics service delivered in the form of statistics or predictions, intelligence operational in real time for Network Functions, Application functions (AFs), and operations, administration, and maintenance (OAM) services. The serving BS, for example, receives assistance data from RAN, such as load status, active UEs, QoS needs, and energy consumption status [71]. The serving node executes an ML algorithm on the collected data to choose an energy-saving action that maximizes network efficiency while maintaining service quality. The node may announce its intention to offload traffic to neighboring nodes to conserve energy. Additionally, a single analytics source in an environment with multiple vendors, especially with Open-RAN (O-RAN) concepts, could be beneficial. We are currently investigating ECO6G use cases in the core and the edge locations with application-aware output for User plane function (UPF) load, especially with Multi-Access Edge Computing (MEC).

5G-Advanced is anticipated to dominate public and private networks beginning in 2025, necessitating a rethinking of network architecture, design, and deployment. It will provide robust support for mission-critical network applications via communication service providers (CSPs) or enterprise-grade private wireless networks. Following in the footsteps of previous generations, 6G will feature an array of innovative technologies that will shape the future of communication. I am researching the architecture, performance, and trustworthy requirements as we evolve from 5G-Advanced to 6G. As part of this survey research, I am emphasizing wireless networks' evolution from connecting humans to connecting things, the role of native intelligence in 6G architecture, open interfaces, existing practices, challenges, opportunities, and future research direction toward next-generation networks.

The current study focuses on defining mechanisms for deploying, orchestrating, and managing varied MNOs available in the network slice ecosystem, proposing an addition to the 3GPP-proposed network slice management capabilities. I am currently researching the network data analytics function (NWDAF) [77], which is defined in 3GPP R16 TS 29.500, 29.501, 29.520 as part of 3GPP Rel-15 and Rel-16, which uses standard interfaces from the service-based 5G architecture to streamline how core network data is consumed to develop insights and take actions to improve the end-user experience. Part of the research investigates the flow to collect data from other network functions for automation or reporting purposes where our proposed frameworks can complement the NWDAF function to predict traffic, improve resource optimization, and schedule resources.

The proposed research presents significant enhancement opportunities to the configuration, management, and control of network slices, enabling network operators to provide clients with the most granular service levels. The proposed research will aid value in 5G-Advanced development, enabling greater control over device registration to network slices based on the applications running on those devices. Unlocking the full value potential of 5G and Beyond communications will require resolving and proactively addressing the security, privacy, and trust challenge in communication networks. A more energy-efficient radio access network will increase the operating efficiency of 5G-Advanced networks, as will the enhanced slicing and analytics capabilities described above. In short, Network slicing technology is sophisticated and operational excellence is essential for constructing the sliced networks to meet the need for a connected society.

Bibliography

[1] Sunday O. Oladejo and Olabisi E. Falowo. “5G network slicing: A multi-tenancy scenario”. In: *2017 Global Wireless Summit (GWS)*. IEEE, Oct. 2017, pp. 88–92. DOI: 10.1109/GWS.2017.8300476.

[2] Lu Ma et al. “An SDN/NFV based framework for management and deployment of service based 5G core network”. In: *China Commun.* 15.10 (Oct. 2018), pp. 86–98. ISSN: 1673-5447. DOI: 10.1109/CC.2018.8485472.

[3] Ping Du and Akihiro Nakao. “Deep Learning-based Application Specific RAN Slicing for Mobile Networks”. In: *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*. IEEE, Oct. 2018, pp. 1–3. DOI: 10.1109/CloudNet.2018.8549243.

[4] Rohit Abhishek, Shuai Zhao, and Deep Medhi. “SPArTaCuS: Service priority adaptiveness for emergency traffic in smart cities using software-defined networking”. In: *2016 IEEE International Smart Cities Conference (ISC2)*. IEEE, Sept. 2016, pp. 1–4. DOI: 10.1109/ISC2.2016.7580854.

[5] Taewhan Yoo. “Network slicing architecture for 5G network”. In: *2016 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, Oct. 2016, pp. 1010–1014. DOI: 10.1109/ICTC.2016.7763354.

[6] Rahul Arun Paropkari, Aklilu Assefa Gebremichail, and Cory Beard. “Fractional Packet Duplication and Fade Duration Outage Probability Analysis for Handover Enhancement in 5G Cellular Networks”. In: *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, Feb. 2019, pp. 298–302. DOI: 10.1109/ICCNC.2019.8685530.

[7] Fabian Kurtz et al. “Network Slicing for Critical Communications in Shared 5G Infrastructures - An Empirical Evaluation”. In: *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. IEEE, June 2018, pp. 393–399. DOI: 10.1109/NETSOFT.2018.8460110.

[8] Rahul Arun Paropkari, Cory Beard, and Appie Van De Liefvoort. “Handover performance prioritization for public safety and emergency networks”. In: *2017 IEEE 38th Sarnoff Symposium*. IEEE, Sept. 2017, pp. 1–6. DOI: 10.1109/SARNOF.2017.8080381.

[9] Rohit Abhishek, David Tipper, and Deep Medhi. “Network Virtualization and Survivability of 5G Networks: Framework, Optimization Model, and Performance”. In: *2018 IEEE Globecom Workshops (GC Wkshps)*. IEEE, Dec. 2018, pp. 1–6. DOI: 10.1109/GLOCOMW.2018.8644092.

[10] Vinod Kumar Choyi et al. “Network slice selection, assignment and routing within 5G Networks”. In: *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*. IEEE, Oct. 2016, pp. 1–7. DOI: 10.1109/CSCN.2016.7784887.

[11] Claudia Campolo et al. “Towards 5G Network Slicing for the V2X Ecosystem”. In: *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. IEEE, June 2018, pp. 400–405. DOI: 10.1109/NETSOFT.2018.8459911.

[12] Rami Akrem Addad et al. “Optimization Model for Cross-Domain Network Slices in 5G Networks”. In: *IEEE Trans. Mob. Comput.* 19.5 (Mar. 2019), pp. 1156–1169. ISSN: 1558-0660. DOI: 10.1109/TMC.2019.2905599.

[13] Danish Sattar and Ashraf Matrawy. “Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices”. In: *2019 IEEE Conference on Communications and Network Security (CNS)*. IEEE, June 2019, pp. 82–90. DOI: 10.1109/CNS.2019.8802852.

[14] Peter Schneider, Christian Mannweiler, and Sylvaine Kerboeuf. “Providing strong 5G mobile network slice isolation for highly sensitive third-party services”. In: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, Apr. 2018, pp. 1–6. DOI: 10.1109/WCNC.2018.8377166.

[15] Anurag Thantharate, Cory Beard, and Poonam Kankariya. “CoAP and MQTT Based Models to Deliver Software and Security Updates to IoT Devices over the Air”. In: *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, July 2019, pp. 1065–1070. DOI: 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00183.

[16] Shahadate Rezvy et al. “An efficient deep learning model for intrusion classification and prediction in 5G and IoT networks”. In: *2019 53rd Annual Conference on Information Sciences and Systems (CISS)*. IEEE, Mar. 2019, pp. 1–6. DOI: 10.1109/CISS.2019.8693059.

[17] Anurag Thantharate, Cory Beard, and Sreekar Marupaduga. “An Approach to Optimize Device Power Performance Towards Energy Efficient Next Generation 5G Networks”. In: *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE, Oct. 2019, pp. 0749–0754. DOI: 10.1109/UEMCON47517.2019.8993067.

[18] Anurag Thantharate, Cory Beard, and Sreekar Marupaduga. “A Thermal Aware Approach to Enhance 5G Device Performance and Reliability in mmWave Networks”. In: *2020 International Symposium on Networks, Computers and Communications (ISNCC)*. 2020, pp. 1–5. DOI: 10.1109/ISNCC49221.2020.9297313.

[19] Anurag Thantharate et al. “DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks”. In: *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. 2019, pp. 0762–0767. DOI: 10.1109/UEMCON47517.2019.8993066.

[20] Anurag Thantharate et al. “Secure5G: A Deep Learning Framework Towards a Secure Network Slicing in 5G and Beyond”. In: *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. 2020, pp. 0852–0857. DOI: 10.1109/CCWC47524.2020.9031158.

[21] Murad Abusubaih. “Intelligent Wireless Networks: Challenges and Future Research Topics”. In: *J. Netw. Syst. Manage.* 30.1 (Jan. 2022), pp. 1–29. ISSN: 1573-7705. DOI: 10.1007/s10922-021-09625-5.

[22] Qingtian Zeng et al. “Traffic Prediction of Wireless Cellular Networks Based on Deep Transfer Learning and Cross-Domain Data”. In: *IEEE Access* 8 (2020), pp. 172387–172397. DOI: 10.1109/ACCESS.2020.3025210.

[23] Claudia Parera et al. “Transfer Learning for Channel Quality Prediction”. In: *2019 IEEE International Symposium on Measurements & Networking (M&N)*. 2019, pp. 1–6. DOI: 10.1109/IWMN.2019.8805017.

[24] Cong T. Nguyen et al. “Transfer Learning for Future Wireless Networks: A Comprehensive Survey”. In: *arXiv* (Feb. 2021). DOI: 10.48550/arXiv.2102.07572. eprint: 2102.07572.

[25] Xi Chen et al. “One for All: Traffic Prediction at Heterogeneous 5G Edge with Data-Efficient Transfer Learning”. In: *2021 IEEE Global Communications Conference (GLOBECOM)*. 2021, pp. 01–06. DOI: 10.1109/GLOBECOM46510.2021.9685204.

[26] Medhat Elsayed, Melike Erol-Kantarci, and Halim Yanikomeroglu. “Transfer Reinforcement Learning for 5G New Radio mmWave Networks”. In: *IEEE Transactions on Wireless Communications* 20.5 (2021), pp. 2838–2849. DOI: 10.1109/TWC.2020.3044597.

[27] Trung V. Phan et al. "Q - TRANSFER: A Novel Framework for Efficient Deep Transfer Learning in Networking". In: *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. 2020, pp. 146–151. DOI: 10.1109/ICAIIC48513.2020.9065240.

[28] Qiyang Zhao and David Grace. "Transfer learning for QoS aware topology management in energy efficient 5G cognitive radio networks". In: *1st International Conference on 5G for Ubiquitous Connectivity*. 2014, pp. 152–157. DOI: 10.4108/icst.5gu.2014.258141.

[29] Rodolfo W. L. Coutinho and Azzedine Boukerche. "Transfer Learning for Disruptive 5G-Enabled Industrial Internet of Things". In: *IEEE Transactions on Industrial Informatics* 18.6 (2022), pp. 4000–4007. DOI: 10.1109/TII.2021.3107781.

[30] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. "Deep Learning in Mobile and Wireless Networking: A Survey". In: *IEEE Communications Surveys & Tutorials* 21.3 (2019), pp. 2224–2287. DOI: 10.1109/COMST.2019.2904897.

[31] Youness Arjoune and Saleh Faruque. "Artificial Intelligence for 5G Wireless Systems: Opportunities, Challenges, and Future Research Direction". In: *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. 2020, pp. 1023–1028. DOI: 10.1109/CCWC47524.2020.9031117.

[32] Yichen Qian et al. "Survey on Reinforcement Learning Applications in Communication Networks". In: *Journal of Communications and Information Networks* 4.2 (2019), pp. 30–39. DOI: 10.23919/JCIN.2019.8917870.

[33] Chamitha De Alwis et al. "Survey on 6G Frontiers: Trends, Applications, Requirements, Technologies and Future Research". In: *IEEE Open Journal of the Communications Society* 2 (2021), pp. 836–886. DOI: 10.1109/OJCOMS.2021.3071496.

[34] Qian Huang and Michel Kadoch. "5G Resource Scheduling for Low-latency Communication: A Reinforcement Learning Approach". In: *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. 2020, pp. 1–5. DOI: 10.1109/VTC2020-Fall49728.2020.9348718.

[35] Faroq Al-Tam, NoÃ©lia Correia, and Jonathan Rodriguez. "Learn to Schedule (LEASCH): A Deep Reinforcement Learning Approach for Radio Resource Scheduling in the 5G MAC Layer". In: *IEEE Access* 8 (2020), pp. 108088–108101. DOI: 10.1109/ACCESS.2020.3000893.

[36] Guosheng Zhu et al. "A Supervised Learning Based QoS Assurance Architecture for 5G Networks". In: *IEEE Access* 7 (2019), pp. 43598–43606. DOI: 10.1109/ACCESS.2019.2907142.

[37] Yaohua Sun et al. “Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues”. In: *IEEE Communications Surveys & Tutorials* 21.4 (2019), pp. 3072–3108. DOI: 10.1109/COMST.2019.2924243.

[38] Mostafa Karimzadeh et al. “Reinforcement Learning-designed LSTM for Trajectory and Traffic Flow Prediction”. In: *2021 IEEE Wireless Communications and Networking Conference (WCNC)*. 2021, pp. 1–6. DOI: 10.1109/WCNC49053.2021.9417511.

[39] Bo Yang et al. “A Joint Energy and Latency Framework for Transfer Learning Over 5G Industrial Edge Networks”. In: *IEEE Transactions on Industrial Informatics* 18.1 (2022), pp. 531–541. DOI: 10.1109/TII.2021.3075444.

[40] Ved P. Kafle et al. “Consideration On Automation of 5G Network Slicing with Machine Learning”. In: *2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*. 2018, pp. 1–8. DOI: 10.23919/ITU-WT.2018.8597639.

[41] Hao Zhou, Melike Erol-Kantarci, and H. Vincent Poor. “Learning from Peers: Transfer Reinforcement Learning for Joint Radio and Cache Resource Allocation in 5G Network Slicing”. In: *ArXiv* abs/2109.07999 (2021).

[42] Anurag Thantharate. “FED6G: Federated Chameleon Learning for Network Slice Management in Beyond 5G Systems”. In: *2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. 2022, pp. 0019–0025. DOI: 10.1109/IEMCON56893.2022.9946488.

[43] Sulaiman Khan et al. “Highly Accurate and Reliable Wireless Network Slicing in 5th Generation Networks: A Hybrid Deep Learning Approach”. In: *J. Netw. Syst. Manage.* 30.2 (Apr. 2022), pp. 1–22. ISSN: 1573-7705. DOI: 10.1007/s10922-021-09636-2.

[44] Miranda McClellan, Cristina Cervelló-Pastor, and Sebastià Sallent. “Deep Learning at the Mobile Edge: Opportunities for 5G Networks”. In: *Appl. Sci.* 10.14 (July 2020), p. 4735. ISSN: 2076-3417. DOI: 10.3390/app10144735.

[45] Jie Mei, Xianbin Wang, and Kan Zheng. “An intelligent self-sustained RAN slicing framework for diverse service provisioning in 5G-beyond and 6G networks”. In: *Intelligent and Converged Networks* 1.3 (2020), pp. 281–294. DOI: 10.23919/ICN.2020.0019.

[46] Jen-Jee Chen et al. “Realizing Dynamic Network Slice Resource Management based on SDN networks”. In: *2019 International Conference on Intelligent Computing and its Emerging Applications (ICEA)*. 2019, pp. 120–125. DOI: 10.1109/ICEA.2019.8858288.

[47] Weili Wang et al. “Cooperative Anomaly Detection With Transfer Learning-Based Hidden Markov Model in Virtualized Network Slicing”. In: *IEEE Communications Letters* 23.9 (2019), pp. 1534–1537. DOI: 10.1109/LCOMM.2019.2923913.

[48] Asma Chiha et al. “Network Slicing Cost Allocation Model”. In: *J. Netw. Syst. Manage.* 28.3 (July 2020), pp. 627–659. ISSN: 1573-7705. DOI: 10.1007/s10922-020-09522-3.

[49] Giuseppe Aceto et al. “Encrypted Multitask Traffic Classification via Multimodal Deep Learning”. In: *ICC 2021 - IEEE International Conference on Communications*. 2021, pp. 1–6. DOI: 10.1109/ICC42927.2021.9500316.

[50] Rohit Abhishek, David Tipper, and Deep Medhi. “Network Virtualization and Survivability of 5G Networks”. In: *J. Netw. Syst. Manage.* 28.4 (Oct. 2020), pp. 923–952. ISSN: 1573-7705. DOI: 10.1007/s10922-020-09541-0.

[51] Brendan Jennings and Rolf Stadler. “Resource Management in Clouds: Survey and Research Challenges”. In: *J. Netw. Syst. Manage.* 23.3 (July 2015), pp. 567–619. ISSN: 1573-7705. DOI: 10.1007/s10922-014-9307-7.

[52] Anurag Thantharate et al. “Balanced5G - Fair Traffic Steering Technique using Data-Driven Learning in Beyond 5G Systems”. In: *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*. Vol. 1. 2022, pp. 01–07. DOI: 10.1109/ICACCS54159.2022.9785169.

[53] Rahul Arun Paropkari, Anurag Thantharate, and Cory Beard. “Deep-Mobility: A Deep Learning Approach for an Efficient and Reliable 5G Handover”. In: *2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*. 2022, pp. 244–250. DOI: 10.1109/WiSPNET54241.2022.9767158.

[54] Anurag Thantharate. *Transfer Learning 5G Dataset - ADAPTIVE6G*. [Online; accessed 18. Jul. 2022]. July 2022. URL: https://www.kaggle.com/datasets/anuragthantharate/transfer-learning-5g-dataset-adaptive6g.

[55] Anurag Thantharate and Cory Beard. “ADAPTIVE6G: Adaptive Resource Management for Network Slicing Architectures in Current 5G and Future 6G Systems”. In: *J. Netw. Syst. Manage.* 31.1 (Mar. 2023), pp. 1–24. ISSN: 1573-7705. DOI: 10.1007/s10922-022-09693-1.

[56] *The State of Mobile Internet Connectivity Report 2022 - Mobile for Development*. [Online; accessed 26. Oct. 2022]. Oct. 2022. URL: https://www.gsma.com/r/somic.

[57] *Release 17*. [Online; accessed 26. Oct. 2022]. Oct. 2022. URL: https://www.3gpp.org/specifications-technologies/releases/release-17.

[58] Yaping Cui et al. “Machine Learning-Based Resource Allocation Strategy for Network Slicing in Vehicular Networks”. In: *Wireless Commun. Mobile Comput.* 2020 (Nov. 2020). ISSN: 1530-8669. DOI: 10.1155/2020/8836315.

[59] Mustufa Haider Abidi et al. “Optimal 5G network slicing using machine learning and deep learning concepts”. In: *Computer Standards & Interfaces* 76 (June 2021), p. 103518. ISSN: 0920-5489. DOI: 10.1016/j.csi.2021.103518.

[60] Tianle Mai et al. “Transfer Reinforcement Learning Aided Distributed Network Slicing Optimization in Industrial IoT”. In: *IEEE Transactions on Industrial Informatics* 18.6 (2022), pp. 4308–4316. DOI: 10.1109/TII.2021.3132136.

[61] Ahmad M. Nagib, Hatem Abou-Zeid, and Hossam S. Hassanein. “Transfer Learning-Based Accelerated Deep Reinforcement Learning for 5G RAN Slicing”. In: *2021 IEEE 46th Conference on Local Computer Networks (LCN).* 2021, pp. 249–256. DOI: 10.1109/LCN52139.2021.9524965.

[62] Federico Mason, Gianfranco Nencioni, and Andrea Zanella. “Using Distributed Reinforcement Learning for Resource Orchestration in a Network Slicing Scenario”. In: *CoRR* abs/2105.07946 (2021). arXiv: 2105.07946. URL: https://arxiv.org/abs/2105.07946.

[63] Jie Mei, Xianbin Wang, and Kan Zheng. “An intelligent self-sustained RAN slicing framework for diverse service provisioning in 5G-beyond and 6G networks”. In: *Intelligent and Converged Networks* 1.3 (2020), pp. 281–294. DOI: 10.23919/ICN.2020.0019.

[64] Hatim Chergui and Christos Verikoukis. “Big Data for 5G Intelligent Network Slicing Management”. In: *IEEE Network* 34.4 (2020), pp. 56–61. DOI: 10.1109/MNET.011.1900437.

[65] Hao Zhou, Melike Erol-Kantarci, and Vincent Poor. “Learning from Peers: Deep Transfer Reinforcement Learning for Joint Radio and Cache Resource Allocation in 5G RAN Slicing”. In: *IEEE Transactions on Cognitive Communications and Networking* (2022), pp. 1–1. DOI: 10.1109/TCCN.2022.3204572.

[66] Fatima Salahdine et al. “A survey on sleep mode techniques for ultra-dense networks in 5G and beyond”. In: *Comput. Networks* 201 (Dec. 2021), p. 108567. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2021.108567.

[67] Qing Wang et al. “Energy-Efficient Priority-Based Scheduling for Wireless Network Slicing”. In: *2018 IEEE Global Communications Conference (GLOBECOM).* 2018, pp. 1–6. DOI: 10.1109/GLOCOM.2018.8647696.

[68] Qize Guo et al. “Proactive Dynamic Network Slicing with Deep Learning Based Short-Term Traffic Prediction for 5G Transport Network”. In: *2019 Optical Fiber Communications Conference and Exhibition (OFC).* 2019, pp. 1–3.

[69] Jin Ho Park et al. "A comprehensive survey on core technologies and services for 5G security: taxonomies, issues, and solutions". In: *Hum.-Centric Comput. Inf. Sci* 11.3 (2021).

[70] "ECO6G Dataset". In: (Oct. 2022). [Online; accessed 27. Oct. 2022]. URL: https://www.kaggle.com/datasets/anuragthantharate/eco6g.

[71] "ETSI, 5G; Management and orchestration; 5G end to end Key Performance Indicators (KPI)". In: (Jan. 2021). [Online; accessed 27. Oct. 2022]. URL: https://www.etsi.org/deliver/etsi_ts/128500_128599/128554/16.07.00_60/ts_128554v160700p.pdf.

[72] Cong T. Nguyen et al. "Transfer Learning for Future Wireless Networks: A Comprehensive Survey". In: *arXiv* (Feb. 2021). DOI: 10.48550/arXiv.2102.07572. eprint: 2102.07572.

[73] *Electric Power Monthly - U.S. Energy Information Administration (EIA)*. [Online; accessed 13. Nov. 2022]. Nov. 2022. URL: https://www.eia.gov/electricity/monthly/epm_table_grapher.php?t=epmt_5_6_a.

[74] "Environmental Engineering (EE); Metrics and measurement method for energy efficiency of wireless access network equipment". In: (Jan. 2021). [Online; accessed 27. Oct. 2022]. URL: https://www.etsi.org/deliver/etsi_es/202700_202799/20270601/01.06.01_60/es_20270601v010601p.pdf.

[75] "Environmental Engineering (EE); Sustainable power feeding solutions for 5G network". In: (Feb. 2021). [Online; accessed 27. Oct. 2022]. URL: https://www.etsi.org/deliver/etsi_es/203700_203799/203700/01.01.01_60/es_203700v010101p.pdf.

[76] Anurag Thantharate et al. "ECO6G: Energy and Cost Analysis for Network Slicing Deployment in Beyond 5G Networks". In: *Sensors* 22.22 (Nov. 2022), p. 8614. ISSN: 1424-8220. DOI: 10.3390/s22228614.

[77] In: (July 2020). [Online; accessed 14. Nov. 2022]. URL: https://www.etsi.org/deliver/etsi_ts/123200_123299/123288/16.04.00_60/ts_123288v160400p.pdf.

VITA

Anurag Thantharate is a member of the IEEE who received his MS degree in Electrical Engineering from the University of Missouri - Kansas City, US, in 2012. He is also a Senior Technical Manager at Palo Alto Networks, Santa Clara, California, where he works on 5G, Network Slicing, Multi-Access Edge Computing, Cloud and IoT Security. He has previously worked with Samsung Electronics, T-Mobile, Sprint, Qualcomm, and AT&T Mobility over the past ten years on a broader aspect of 5G, Wireless Networks, Computer Networking, and Communication Systems optimization. His research interests include Machine and Deep Learning applications in Wireless Communication, Greenfield technology, Network Security, 5G-Advanced, 6G, and Autonomous Network Management towards the Zero Touch Approach.