International Journal of Wireless Communications and Mobile Computing
2025, Vol. 12, No. 2, pp. 93-118
https://doi.org/10.11648/j.wcmc.20251202.14

Research Article

Slice-Specific Machine Learning Models for Intrusion Detection in 5G Telecommunication Networks

Vincent Andrew Akpan[1, *], Emmanuel Chinenye Njoku[2], Ebubechukwu Ifeoluwa Obi[3]

[1]Department of Biomedical Engineering, The Federal University of Technology, Akure, Nigeria
[2]ICT Department, Guaranty Trust Holding Company, Victoria Island, Nigeria
[3]Human Resources Unit, Human Family Foundation for Peace and Development, Osogbo, Nigeria

Abstract

The security challenges introduced by 5G network slicing demand tailored intrusion detection systems (IDS). Traditional intrusion detection systems (IDS) and intrusion detection and prevention systems (IDPS) frameworks, built for static network configurations, are inadequate for the dynamic and heterogeneous nature of 5G networks. To address this gap, this study develops and evaluates slice-specific machine learning models to enhance intrusion detection across different 5G slices, namely: enhanced Mobile Broadband (eMBB), massive Machine-Type Communication (mMTC), and Ultra-Reliable Low-Latency Communication (URLLC). Random Forest, Support Vector Machine (SVM), and Long Short-Term Memory (LSTM) models were applied to publicly available datasets representing each slice. These models are assessed based on their accuracy, precision, recall, F1-score, area under the receiver operating characteristic curve (AUC-ROC), confusion matrix and execution time. The results reveal that the LSTM model achieved the highest accuracy and AUC-ROC scores for the eMBB and mMTC slices, making it suitable for applications where detection accuracy is critical despite higher computational demands. In contrast, Random Forest demonstrated superior computational efficiency, making it the most preferred model for latency-sensitive URLLC slice, where real-time detection is essential. While the SVM model performed well in terms of accuracy, its high computational cost renders it less practical for real-time applications, particularly in URLLC environments. This research provides insights for enhancing 5G network security through the deployment of slice-specific machine learning models, thereby addressing the critical need for adaptable and efficient IDS frameworks.

Keywords

Fifth Generation (5G) Networks, Intrusion Detection System (IDS), Machine Learning Algorithms, Massive Machine-Type Communication (mMTC), Mobile Broadband (eMBB), Ultra-Reliable Low-Latency Communication (URLLC)

[*]Corresponding author: vaakpan@futa.edu.ng (Vincent Andrew Akpan)

Received: 26 September 2025; **Accepted:** 9 October 2025; **Published:** 26 November 2025

1. Introduction

The fifth generation (5G) of cellular networks represents a significant advancement in mobile communication networks. While previous generations of cellular networks have seen incremental improvements over their predecessors, 5G is disruptive, offering unprecedented capabilities in terms of data rates, latency, reliability, and device density. 5G is designed to support many use cases across various sectors including, but not limited to, healthcare, education, transportation, manufacturing, and entertainment. To cater for these use cases, the 3GPP defines three main service categories: enhanced Mobile Broadband (eMBB), massive Machine-Type Communication (mMTC), and Ultra-Reliable Low-Latency Communication [1]. Network slicing is one of the key techniques to dynamically manage logical and functionally separate networks on a common infrastructure. The concept of network slicing is illustrated in Figure 1. As shown in Figure 1, network slicing describes the process of accommodating virtual networks, typically composed of nodes and links with their respective requirements, into the main infrastructure [2]. The three main types of network slices as defined by 3GPP (3rd Generation Partnership Project) are: Ultra-Reliable Low Latency Communications (URLLC), enhanced Mobile Broadband (eMBB), and massive Machine Type Communications (mMTC) [3-5]. Each slice caters to specific needs, such as high-speed data transfer, extensive device connectivity, and ultra-reliable low-latency communication, laying the foundation for next-generation applications like virtual reality, autonomous vehicles, and the Internet of Things (IoT) [6, 7]. These slices, facilitated by technologies such as Software-Defined Networking (SDN) and Network Function Virtualization (NFV), operate independently with their own sets of network functions and security protocols [8]. This architecture allows the network to address the diverse needs of applications ranging from high-speed mobile broadband to industrial automation and large-scale IoT deployments.

However, these significant advancements introduce a more intricate security landscape. The centralized architecture of SDN and NFV, while essential for the flexibility of 5G networks, introduce new vulnerabilities, such as the risk of hypervisor attacks and side-channel exploits between Virtual Network Functions (VNFs) [9]. Furthermore, the sheer volume of connected devices - especially within the mMTC slice, which can accommodate up to one million devices per square kilometer - adds to the complexity of network security. Many of these devices, like IoT sensors, are resource-constrained and unable to implement robust security protocols, leaving the network exposed to large-scale distributed attacks [6].

Network Intrusion Detection Systems (NIDS) are critical for protecting 5G networks. However, traditional security mechanisms, such as signature-based and anomaly detection, struggles to keep up with the volume and variety of traffic in 5G. The disaggregation of networks into slices to serve specialized used cases has created the need to develop security solutions that are agile, adaptable and applicable to dynamic environments. The application of machine learning (ML) and deep learning (DL) models in anomaly-based detection systems presents a promising solution, especially for identifying previously unknown attacks. These ML/DL models analyze traffic patterns to detect deviations from the norm, identifying potential threats even in encrypted traffic without compromising user privacy [3-5].

Considering the diversity in performance requirements and threat landscapes across eMBB, mMTC, and URLLC slices, intrusion detection models be tailored to the unique characteristics of each slice. For instance, the eMBB slice is designed to deliver high data rates for applications such as 4K/8K streaming and virtual reality, the mMTC slice supports a massive number of IoT devices susceptible to device-level attacks due to the limited computational capacity of many IoT sensors. The most critical slice, URLLC, requires ultra-low latency and extreme reliability for applications such as autonomous driving, industrial automation, and remote surgery.

While 5G network slicing improves network performance and flexibility, it also necessitates advanced security mechanisms that can adapt to the specific needs of each slice. This study performs a comparative analysis of ML/DL models for intrusion detection across the eMBB, mMTC, and URLLC slices, examining how each approach can be customized to address the security challenges unique to 5G environments.

2. Literature Review

Numerous scholars have investigated security within the realm of 5G and network slicing, with emphasis on the role of machine learning (ML) in strengthening security frameworks. Network slicing has introduced increased flexibility and efficiency to cellular networks. The centralized nature of SDN and NFV, which are key facilitators of network slicing has led to an increase in the attack surface and introduction of new vulnerabilities [10]. The distinct performance requirements and security challenges of each slice further underscore the need for tailored, effective management strategies. Machine learning, as a tool, has shown promise for addressing these security challenges. Machine learning algorithms can efficiently detect anomalies and potential threats through real-time analysis of network traffic. This review examines the current state of machine learning integration into 5G security frameworks, analyses recent advancements and innovations, and addresses ongoing challenges in securing the complex landscape of network slicing.

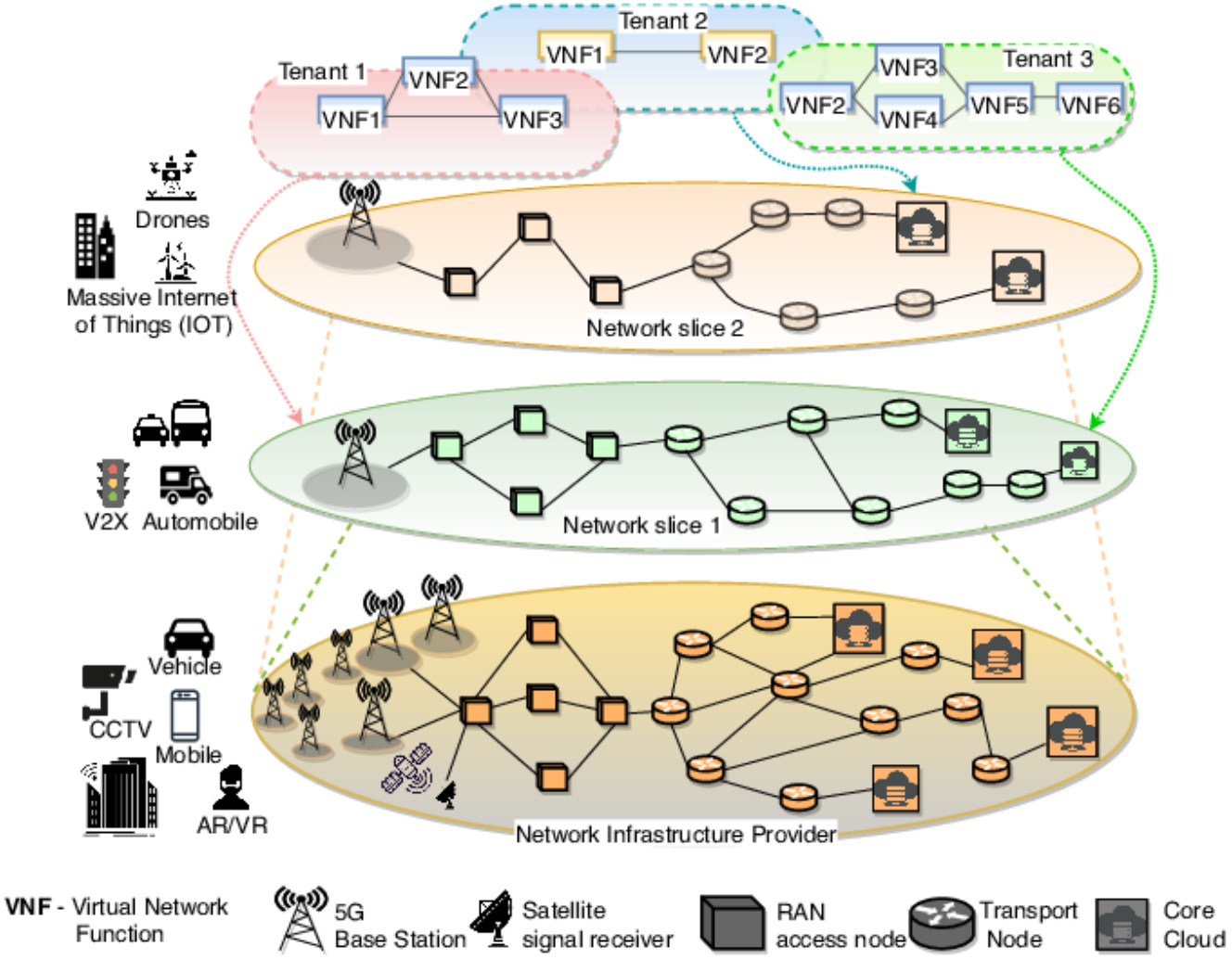


Figure 1. *Concept of network slicing.*

2.1. Network Slicing in 5G

Network slicing is a key technique in 5G networks that enables the creation of multiple virtual networks on a shared physical infrastructure. These virtual networks called slices can be customized to meet specific performance requirements, making it possible to serve many different use cases. This allows network operators to optimize resources and improve efficiency by tailoring each slice to applications, such as high-speed mobile broadband, massive IoT deployments, or ultra-reliable low-latency communications.

Barakabitze and co-workers highlights the transformative potential of 5G networks, particularly emphasizing the significance of network slicing in optimizing network resources through technologies like SDN and NFV [11]. According to their study, these technologies allow operators to create customized network slices that cater to specific applications, improving overall network efficiency and performance. Partel and Tripathy provided an extensive analysis of the primary network slices in 5G, identifying eMBB, mMTC, and URLLC as the three fundamental slices [3]. They outlined how eMBB supports data-heavy services like HD video streaming and virtual reality, while mMTC enables the massive connectivity required for IoT devices. URLLC, as they explained, is crucial for mission-critical applications, such as autonomous vehicles and real-time medical procedures.

Amogh, Pc and colleagues explored the impact of DDoS attacks on 5G network slicing [12, 13]. They introduce a unique threat model specifically designed for 5G networks and proposed a Quarantine Slice Manager architecture, which features a two-level hierarchical and distributed network design for effective slice management and rapid threat response. They utilized a simulated RAN and an emulated 5G core network to analyze the effect of DDoS attacks on three network slices: eMBB, mMTC, and URLLC. Gao and colleagues examined the immense potential of mMTC in supporting large-scale IoT deployments [14]. Their research highlighted how this slice enables connectivity for billions of devices, from smart meters to sensors, paving the way for innovations in industrial automation and smart homes. Furthermore, Cunha and co-researchers explored the technolog-

ical advancements that make 5G network slicing possible [7]. They emphasized the role of SDN and NFV in enabling operators to dynamically control and allocate network resources, which leads to improved performance and scalability across different sectors, including healthcare and manufacturing.

2.2. Security Challenges in 5G Networks

As 5G networks become the backbone of essential infrastructure and services, guaranteeing their security becomes critical. The architecture of 5G networks, with its emphasis on virtualization, software-defined networking, and network slicing, introduces new vulnerabilities that must be addressed. The increased use of software in network operations also expands the potential for software-based attacks, including malware infections and exploitation of software vulnerabilities. This expanded attack surface, coupled with the increased complexity of 5G networks, presents significant challenges to traditional security paradigms [15]. The examination on how the introduction of billions of internet of things (IoT) devices in 5G networks significantly expands the attack surface have been investigated [16, 17]. They emphasize that the diverse nature of devices creates numerous entry points for potential cyber attacks, making it difficult to implement uniform security measures across the network. The heterogeneous nature of devices, varying in capabilities and security requirements, poses additional challenges for the standardization of security measures. Some researchers have investigated the security implications of network slicing, a critical feature of 5G networks [18, 19]. Their research identifies that each slice can have different security requirements and vulnerabilities. They showed that improper isolation between slices can lead to cross-slice attacks, which can compromise one slice and thereby threaten others. Humayun highlighted the need for new security protocols specifically designed for 5G networks due to network security issues [20]. They argue that many existing protocols from earlier mobile network generations are inadequate for addressing the unique challenges posed by 5G, leaving systems vulnerable to various attacks.

Some researchers have focused on the potential for cross-slice attacks and the implications for security within network slicing [21-23]. These researches analyze how attackers can exploit security weaknesses in one slice to gain unauthorized access to resources in another. The results from these findings advocate for enhanced monitoring and intrusion detection systems tailored to dynamically assess threats across slices. Other researchers explored the trade-offs between resource allocation and security in network slicing [24, 25]. Their study revealed that aggressive resource allocation strategies aimed at maximizing performance can inadvertently compromise security. They propose a security-aware resource allocation framework that balances performance needs with adequate security measures to protect against potential threats. Cao and colleagues examined the importance of dynamic security policies in the context of network slicing; and it argued that static security measures may not be sufficient due to the dynamic nature of 5G services [26]. Their research presents a framework that allows for real-time adaptation of security policies based on traffic patterns and emerging threats, thereby enhancing the overall security posture of network slices.

2.3. Intrusion Detection Systems (IDS) in 5G Networks

In 5G networks, IDS takes on new dimensions of complexity and importance due to the enhanced capabilities of the network and diverse use cases. The fundamental principle of IDS remains consistent across generations of mobile networks: to act as a defender, constantly analyzing network data to detect and alert potential security threats. However, the implementation and capabilities of IDS have significantly evolved to meet the unique challenges posed by 5G architecture [27].

In another development, Sharma and co-woorkers provide a critical overview of the existing challenges in implementing NIDS within 5G networks [28]. They identify issues such as the heterogeneity of devices, rapid network changes, and the vast amount of data generated as significant barriers. The authors propose future research directions, including the need for more robust algorithms, cross-layer detection techniques, and improved collaboration between different network components. Bhelkar highlight the need for a holistic security framework to address various vulnerabilities inherent in the seven network security layers of 5G [29]. The study identifies key security issues such as authentication and authorization problems, including denial-of-service attacks, man-in-the-middle attacks, and eavesdropping. It evaluates both hardware solutions (like unmanned aerial vehicles and field-programmable gate arrays) and software solutions (such as artificial intelligence, machine learning, blockchain, and statistical process control) for mitigating these threats. Ouiazanne research group examined the hybrid NIDS that combine signature-based and anomaly-based detection techniques for better performance in 5G networks [30]. Their research demonstrates that a hybrid approach can leverage the strengths of both methodologies, allowing for faster detection and fewer false positives. They also present a framework for implementing such hybrid systems that can adapt to the unique characteristics of 5G traffic. Gyamfi and Jurcut explored the integration of edge computing with NIDS in 5G architectures [31]. They argued that deploying NIDS at edge nodes can reduce latency and improve response times to detected threats. Their study highlights a distributed approach to intrusion detection, which can significantly enhance the scalability and performance of security measures in 5G networks.

Almazroi and co-workers proposed a comprehensive framework for NIDS tailored for 5G environments [32]. They address the unique challenges posed by the high data rates and

low latency requirements of 5G, which necessitate real-time detection capabilities. The authors emphasize the importance of integrating machine learning algorithms to enhance the accuracy and efficiency of threat detection in 5G networks. Radoglou-Grammatikis and co-worker investigated the application of artificial intelligence (AI) techniques in constructing NIDS for 5G networks [33, 34]. Their research discusses the use of deep learning models for traffic classification and anomaly detection, with a focus on their ability to identify sophisticated attacks that traditional methods might miss. They present a comparative analysis showing that AI-based systems are often more effective in identifying intrusions in dynamic 5G environments. Research on the incorporation of blockchain technology into NIDS for enhanced security in 5G networks has been extensively carried out by some researchers [35, 36]. The research outcome reasoned that blockchain's decentralized nature can improve the integrity of data used for intrusion detection and facilitate secure information sharing among various network elements. Their results indicate that blockchain integration substantially increases resilience against common attacks, such as data tampering.

2.4. Machine Learning in 5G IDS

The complexities and dynamic nature of 5G environments necessitate intelligent solutions capable of effectively identifying and mitigating threats. The integration of machine learning into network intrusion detection systems represents this advancement. Several researchers have also explored various machine learning techniques for anomaly detection in 5G networks [37, 38]. They evaluate supervised, unsupervised, and semi-supervised learning methods to classify network traffic and identify abnormal patterns indicative of possible intrusions. The authors conclude that ensemble learning techniques, which combine multiple classifiers, show the best performance in terms of accuracy and false positive rates. Other researchers investigated the application of deep learning models, such as convolutional neural networks (CNNs), for security analytics in 5G networks [39, 40]. Their results emphasized that deep learning can capture complex patterns in high-dimensional data that traditional methods may overlook. Their results indicate that deep learning models significantly outperform conventional NIDS regarding both detection accuracy and processing speed. Furthermore, Gong and co-workers focused on the importance of feature selection in optimizing the performance of ML algorithms for NIDS in 5G environments [41]. They propose a framework that identifies the most relevant features from network traffic data, enhancing the effectiveness of classification algorithms like support vector machines (SVMs) and random forests. Their findings suggest that effective feature selection can help reduce computation costs while improving detection rates. Uszko, Hamroun and co-workers proposed a hybrid NIDS that integrates machine learning and traditional rule-based detection methods [42-44]. Their research reveals that while ML can adapt to evolving threats, rule-based systems can provide a reliable foundation for known attack signatures. The authors present a framework for combining these approaches to enhance detection capabilities in dynamic 5G environments, thereby reducing both false negatives and false positives.

Latif and colleagues explore the potential of federated learning, a distributed ML approach, for enhancing NIDS in 5G networks [45, 46]. This technique allows for collaborative training of models using data from multiple sources without compromising user privacy. The authors demonstrate that federated learning maintains high detection accuracy while allowing individual entities to keep their data localized, addressing privacy concerns associated with centralized data processing. Bocu and Lavich focused on the implementation of real-time intrusion detection systems using machine learning for 5G [47]. They discuss the need for low-latency detection mechanisms that can handle the rapid data flows characteristic of 5G networks. Their study presents a lightweight classifier that can be deployed at edge devices, achieving timely threat detection without compromising performance.

Several researchers have tried to address the challenge of detecting terabits-per-second (Tbps)-level Distributed Denial-of-Service (DDoS) attacks in 5G networks while maintaining low latency [12, 13, 27, 48-50]. They emphasize the significance of feature selection within machine learning (ML) for DDoS detection, showing that effective feature selection reduces computational complexity and improves detection performance. Their experiments reveal that utilizing feature selection enhances the real-time detection of large-scale DDoS attacks, making it a critical component for efficient ML-based security measures in 5G core networks.

Some researchers have proposed a deep learning-based solution specifically designed for detecting DDoS attacks in network slicing, a vital technology in 5G and beyond [49, 50]. Their innovative approach isolates and mitigates attackers by establishing a sinkhole-type slice with limited resources. Evaluated through a 5G prototype built on Open Air Interface (OAI), their method achieved nearly 97% detection accuracy, with a false positive rate of less than 4%. Additionally, it demonstrated a significant reduction in malicious users' network throughput by a factor of 15 while preserving high throughput for legitimate users.

In another development, Kasongo and co-workers introduced a Feed-Forward Deep Neural Network (FFDNN)-based intrusion detection system (IDS) tailored for wireless networks, enhanced by a Wrapper Based Feature Extraction Unit (WFEU) [51]. The WFEU employs the Extra Trees algorithm to generate an optimized feature vector, thereby improving the system's overall performance. Their approach was extensively tested using the UNSW-NB15 and AWID intrusion detection datasets. Results indicated that the WFEU-FFDNN outperformed standard machine learning algorithms, achieving detection accuracies of 87.10% and 77.16% for binary and

multiclass classification on the UNSW-NB15 dataset, and 99.66% and 99.77% on the AWID dataset.

Agrafiotis and co-workers introduced a novel strategy for detecting malware traffic within 5G networks [52, 53]. Their methodologies involved a network packet preprocessing toolkit that converts packets into images or embeddings, facilitating improved analysis and representation. They leverage Long Short-Term Memory (LSTM) Autoencoders to generate embeddings, combined with a fully connected neural network for classification. This approach is designed to be protocol-agnostic and adaptable to emerging threats. The proposed system showcases enhanced accuracy in identifying malware traffic in 5G environments and holds promise for future malware defense, including in potential 6G networks.

Park and colleagues have recently proposed a distributed learning-based intrusion detection system aimed at the decentralized environments expected in future 6G networks [54]. Their approach utilizes split learning to efficiently train models across various systems with distinct computing resources. This methodology addresses the limitations of centralized systems, making it suitable for distributed architectures. Evaluations using 5G network data demonstrate the system's effectiveness in bolstering network security for forthcoming mobile generations. Fahmin and colleagues focused on enhancing the performance of intrusion detection systems (IDS) through a comparative evaluation of various machine learning techniques [10]. Their study aimed to boost accuracy, reduce false alarms, and improve detection rates. Fahmin and co-workers also compared multiple algorithms, including multilayer perceptron, support vector machine (SVM), random forest, and extreme learning machine (ELM) using the NSL-KDD dataset, recognized as a benchmark for intrusion detection evaluation [10]. Their results revealed that ELM outperforms other techniques, demonstrating superior efficiency in analyzing larger datasets and overall performance enhancement of IDS.

2.5. Importance of Feature Selection in Machine Learning IDS

Feature selection is an important process in the development of machine learning models. Selecting the most relevant features helps enhance the accuracy and performance of ML models, minimize the computational resources required for training and inference, and reduce the risk of overfitting. Alazab and Fong in their comparative study, investigated various feature selection techniques applied to intrusion detection systems, including filter, wrapper, and embedded methods [41, 55-57]. They emphasize the significant improvements in performance metrics such as accuracy, recall, and precision achieved through effective feature selection, advocating for standardized practices in IDS design. The introduction of a hybrid feature selection approach which combines both filter and wrapper techniques have been explored by some researchers [56, 57]. The authors validate their method using machine learning classifiers on the KDD dataset, showcasing marked improvements in detection rates and reductions in false positives, thereby reinforcing multiple importance of robust feature selection in modern IDS. Narayan and co-researchers proposed a multiple feature selection mechanism that leverages entropy on the control plane to detect anomalous activities [58]. They utilized MLP, CNN, and LSTM models, with the LSTM model outperforming the other models achieving an accuracy of up to 99.83%.

Khan and Sadiq recently proposed an ensemble feature selection method tailored for intrusion detection in IoT environments [59]. They address the challenges presented by data volume and feature relevance, demonstrating that their ensemble approach significantly enhances model robustness and accuracy compared to traditional single-method techniques. Salama and Ghaleb employed genetic algorithms for optimized feature selection in network intrusion detection [60]. The authors illustrate how leveraging evolutionary algorithms can efficiently identify optimal feature subsets, leading to enhanced detection accuracy and reduced computational overhead. Zhou and Liu present a novel deep feature selection method that harnesses deep learning techniques for feature identification in IDS [61]. Their findings indicate that not only does this approach improve detection accuracy, but it also enhances the interpretability of the model's decision-making process. Bouke and Abdullah assessed various machine learning models for intrusion detection in 5G networks focusing on a data leakage-free methodology [62]. Their work emphasizes the importance of avoiding data leakage during model development to ensure the generalization of the models to real-world scenarios. The results indicate that robust feature selection, notably using the BukaGini algorithm, enhances detection accuracy while reducing unnecessary data complexity.

2.6. Slice-Specific IDS

5G networks are engineered to support a variety of use cases through network slicing, where each slice can be customized to meet specific service requirements. Existing intrusion detection systems typically apply general detection methods that do not capture the nuances of different slices. Recent studies illustrate that a tailored approach can significantly improve detection accuracy across slice-specific traffic patterns [63]. Employing machine learning models designed specifically for the diverse characteristics of each slice can make the security framework better identify and mitigate threats unique to each application. Current intrusion detection approaches often rely on static models that may struggle to adapt to evolving conditions or varied traffic loads. For instance, the fluctuation of users in eMBB services during peak hours necessitates a model that can adjust its parameters based on real-time data. Implementing slice-specific machine learning models that can accommodate these changes could significantly improve the effectiveness of intrusion detection

models. Amogh, Pc and co-workers further explored the impact of DDoS on 5G network slicing, highlighting the security risks associated with 5G technology [12, 13]. They introduced a unique threat model specifically designed for 5G networks and proposed a Quarantine Slice Manager architecture, which features a two-level hierarchical and distributed network design for effective slice management and rapid threat response. They utilized a simulated RAN and an emulated 5G core to analyze the effect of DDoS attacks on eMBB, mMTC, and URLLC slices. Sattar and Matrawy proposed the use of a mathematical model to provide on-demand slice isolation for mitigating DDoS attacks and increasing the availability of slices [27]. In a related development, Khan and co-workers demonstrated the impact of DoS/DDoS attacks on performance metrics using a novel dataset collected from a simulated network slicing testbed. Using a deep learning based bidirectional LSTM model, Slice Secure, they achieved an accuracy of 99.99% on the newly created dataset [64].

3. Materials and Methodology

3.1. Materials

The materials employed for this research are listed in this sub-section and briefly described with their roles in the realization of the results.

Jupyter Notebook

Jupyter Notebook is used for carrying out data analysis, machine learning model development, and documentation in the project. The interactive environment allowed for code execution and visualization, enhancing both data preprocessing and exploratory analysis of 5G network traffic. Its integration with libraries like Pandas, NumPy, and TensorFlow helped facilitate comprehensive analysis and model development, while the documentation features ensured clear communication of methodologies and results.

Matplotlib

Matplotlib is an essential tool for creating static and interactive visualizations that enhanced a deeper understanding of the network traffic data. It is a versatile plotting library for Python, which provided powerful capabilities to visualize patterns, anomalies, and correlations within large datasets. This library was useful in plotting trends and understanding results from data analyses and models.

Seaborn

Seaborn was used to enhance the visualization of statistical relationships and patterns in network traffic data. Seaborn, integrated with Matplotlib, offers additional features such as customized plot styles, color palettes, and high-level statistical plotting functions. This integration helped create visually appealing and informative visualizations that provided deeper insights into the distribution of network traffic and the occurrence of anomalies across different 5G network slices.

Keras

Keras is a high-level deep learning API running on top of TensorFlow. It is used extensively for building and training deep learning models.

TensorFlow

TensorFlow is used for developing and training complex machine learning models to detect security threats in 5G networks through vast deep learning architectures like CNNs and LSTMs. Its flexibility allowed for tailored neural network designs suited to specific 5G slices, while seamless integration with Python, Pandas, and NumPy streamlined data preprocessing and model training.

NumPy

NumPy is instrumental in facilitating efficient data processing by enhancing computational capabilities with its support for large, multi-dimensional arrays and matrices. Its functions enabled operations for normalizing traffic data and computing correlation matrices. NumPy's optimized array handling and manipulation significantly expedited data preprocessing tasks, ensuring data was efficiently structured for analysis. Additionally, its seamless integration with machine learning libraries like scikit-learn and TensorFlow facilitated the transition of preprocessed 5G network data into robust models for anomaly detection.

Pandas

This library was used to facilitate descriptive statistical analysis and time series analysis, examining trends and patterns in network attacks. Its functionalities for handling missing values, filtering, and merging datasets ensured clean and consistent data for machine learning models.

Google Colab

Google Colab is a cloud-based platform that allows users to write and execute Python code in a Jupyter notebook environment. It provides free access to GPUs to facilitate efficient machine learning and data analysis in a collaborative workspace. It was essential for developing machine learning models through it support for GPU resources which accelerated the training of complex models such as LSTM.

Python

Python is the programming language of choice for the project. It was chosen for its versatility and extensive library support in data preprocessing, exploratory data analysis, and machine learning model development. The key libraries used includes Pandas, NumPy, Matplotlib, Seaborn, scikit-learn, and TensorFlow.

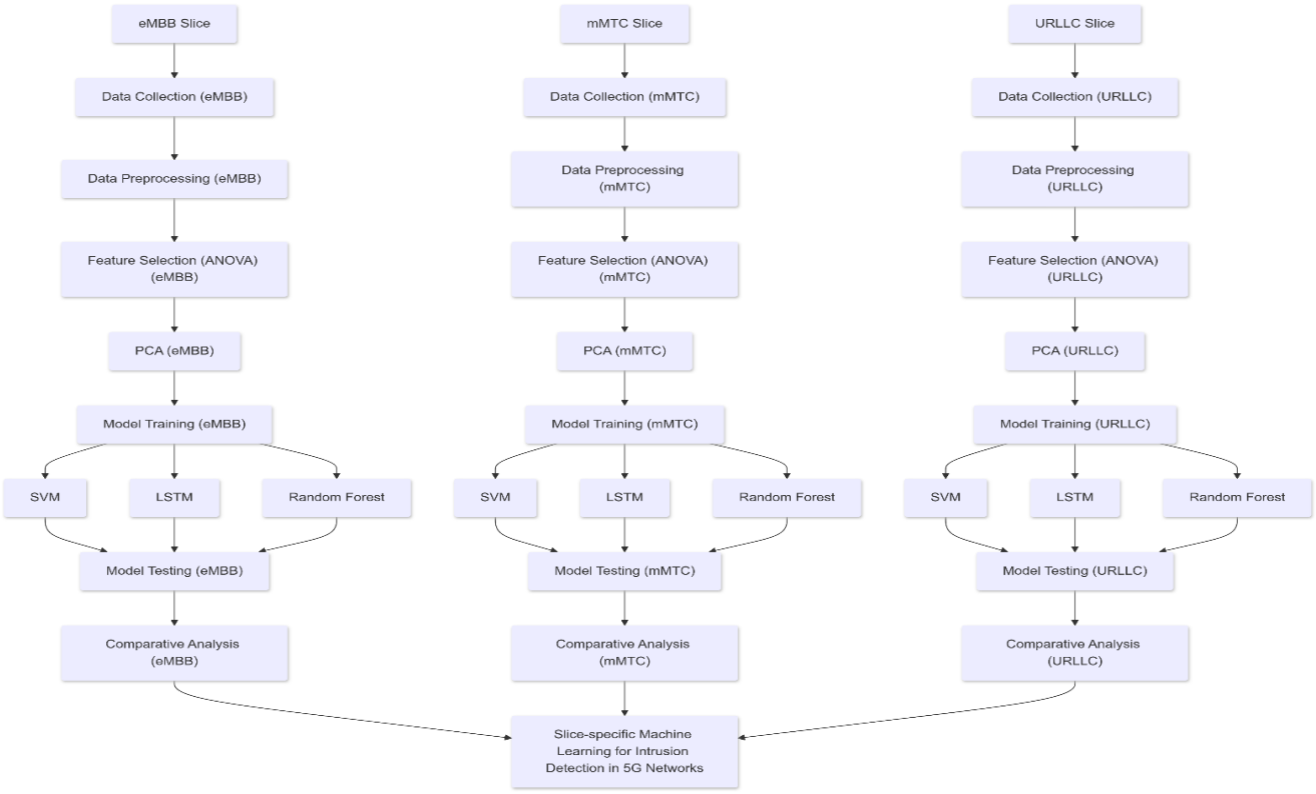


Figure 2. *Block diagram of the flow chat for the research design.*

3.2. Methodology

Several recent advances, taxonomy, huge requirements and open challenges as future directions have been reported [65]. However, this research at its core is a comparative study, focusing on how machine learning models perform in detecting network intrusions across different 5G network slices. Since eMBB, mMTC and URLLC slices have unique characteristics and use cases, this study analyses the effectiveness, accuracy, performance, and resource complexity of the models for each slice. Particularly, the models of concern are the Support Vector Machine (SVM), Random Forest (RF), and the Long-Short Term Memory Recurrent Neural Network (RNN-LSTM). For this study, a quantitative approach is adopted, encompassing data collection, statistical data analysis, preprocessing, feature selection, dimensionality reduction, model training and testing, and evaluation of model performance using predefined performance metrics as illustrated in Figure 2.

Data Collection

The importance of quality data in training machine learning models cannot be overemphasized. For this study, we utilize three publicly available datasets representing each network slice. These datasets capture network traffic peculiar to each slice, while including both benign and malicious traffic patterns.

3.2.1. 5G-NIDD Dataset

The 5G-NIDD dataset, used for the eMBB slice in this study, is a fully labeled dataset created using a functional 5G test network at the University of Oulu, Finland [66]. It captures network traffic from two base stations, each featuring an attacker node and several benign 5G users. The attacker nodes target a server deployed in the 5G Test Network Multi-access Edge Computing (5GTN-MEC) environment, simulating various attack scenarios, including Denial-of-Service (DoS) attacks - such as ICMP Flood, UDP Flood, SYN Flood, HTTP Flood, and slow rate DoS - and port scans, including SYN Scan, TCP Connect Scan, and UDP Scan. Additionally, the dataset includes non-malicious traffic, providing a comprehensive representation of network behavior. Each flow in the dataset is labeled as either attack or normal, with further labels indicating the attack type and the specific tool used, enabling multiclass classification.

The architecture of the 5G NIDD test environment is shown in Figure 3. The 5G-NIDD dataset consists of 1,215,872 samples, with 477,737 samples classified as “Normal” (0) and 738,135 samples classified as “Attack” (1). This distribution indicates an imbalance, with the "Normal" class making up approximately 39.4% of the samples, while the “Attack” class comprises around 60.6%. The focus of this work is on binary classification, specifically distinguishing between the “Normal” and “Attack” classes.

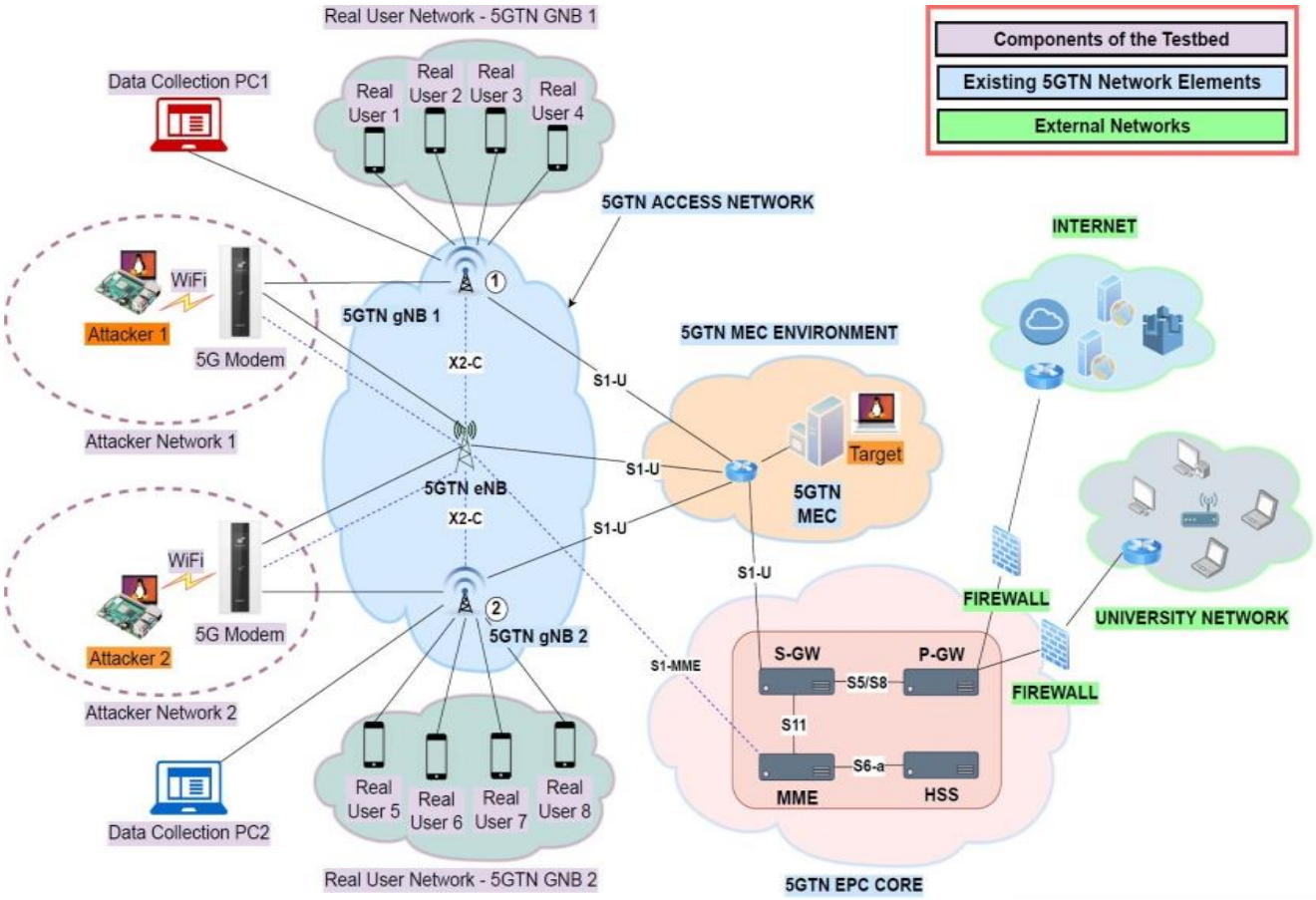


Figure 3. *Architecture of 5G NIDD test environment.*

3.2.2. CIC IoV 2024 Dataset for URLLC Slice

The CIC IoV 2024 dataset contributes significantly to cybersecurity for Internet of Vehicles (IoV), which is an important use case for ultra-reliable low-latency communication (URLLC) [67]. The CIC IoV 2024 dataset test environment is shown in Figure 4. It was developed to provide a realistic benchmark dataset that supports the development of new cybersecurity solutions. The dataset is derived from extensive experiments conducted on the Electronic Control Units (ECUs) of a 2019 Ford vehicle, ensuring a detailed representation of intra-vehicular communications. The dataset includes various features that are crucial for understanding the data transmitted within the vehicle. Key features consist of the ID, which indicates the priority of the message, and several data bytes (DATA0, DATA1, DATA2, etc.) that represent the actual data transmitted. These features provide a comprehensive view of the communication patterns and can be instrumental in identifying anomalies or malicious activities within the vehicle's network.

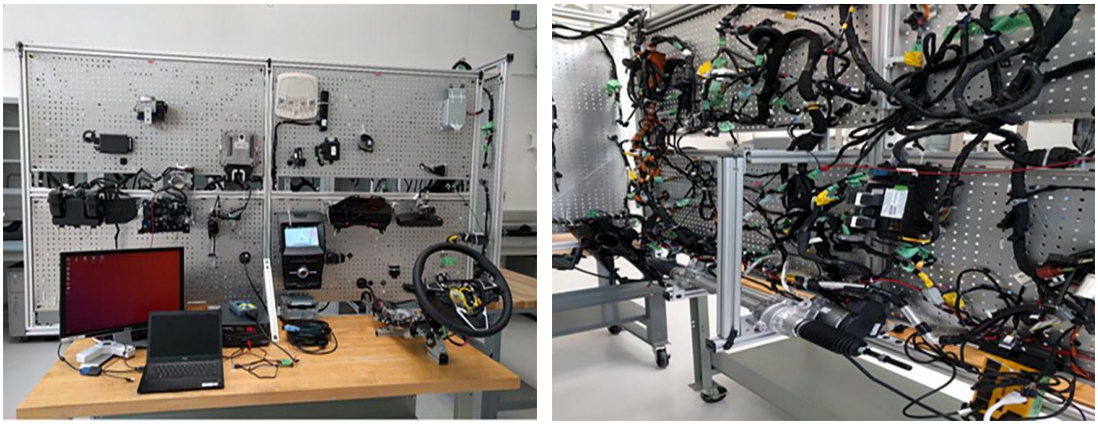


Figure 4. *CIC IoV 2024 dataset test environment.*

3.3.3. CIC IoT Dataset for mMTC Slice

The CIC IoT dataset was collected from an IoT topology with 105 devices where 33 distinct attacks were executed [68]. The CIC IoT 2023 datatest environment is shown in Figure 5. These attacks are classified into seven categories, namely Distributed Denial of Service (DDoS), Denial of Service (DoS), Reconnaissance (Recon), Web-based attacks, Brute Force attacks, Spoofing, and the notorious Mirai botnet. All attacks were executed by malicious IoT devices targeting other IoT devices. This dataset is useful for evaluating and developing machine learning models for mitigating cyber attacks in IoT environments.

3.3. Exploratory Data Analysis, Data Preprocessing and Feature Selection

3.3.1. Data Cleaning

The data collected was subjected to some preprocessing steps to ensure that the data was clean, consistent, and suitable as inputs for the machine learning models. The data was cleaned by the following strategies:

1) *Imputation:* where the number of missing values is relatively small, missing values were imputed using statistical methods. For numerical features, techniques employed involve mean, median.
2) *Removal:* Columns with predominantly null or missing values, for which imputation will introduce bias in the dataset are removed. Also, rows with duplicate entries are removed to avoid redundancy in the dataset which can lead to overfitting or skewed performance.

3.3.2. Exploratory Data Analysis, Feature Analysis and Selection for the eMBBSlice

The 5G-NIDD dataset used for the eMBB slice comprises 1,215,890 entries or rows of traffic data, each having 52 attributes [65]. Of the total data, 60.6% comes from malicious actors and 39.4% from benign nodes as shown in Figure 6. The attack type in the experimental set up was categorized into: *1).* Benign; *2).* UDP Flood; *3).* HTTP Flood; *4).* Slow rate DNS; *5).* TCP Connect Scan; *6).* SYN Scan; *7).* UDP Scan; *8).* SYN Flood; and *9).* ICPM Flood. These nine categorized attack types and the attack type dataset distribution for the 5G NIDD dataset is shown in Figure 7.

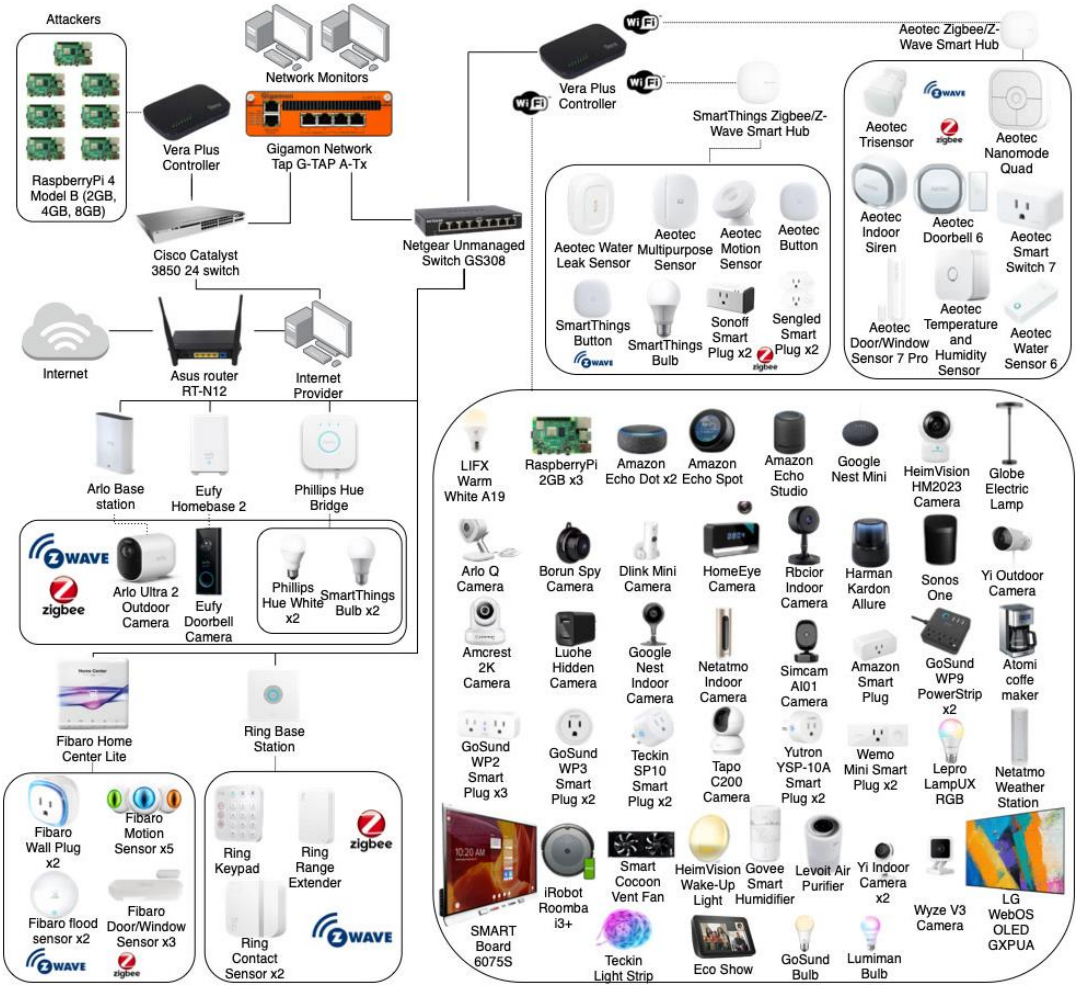


Figure 5. *CIC IoT 2023 data test environment.*

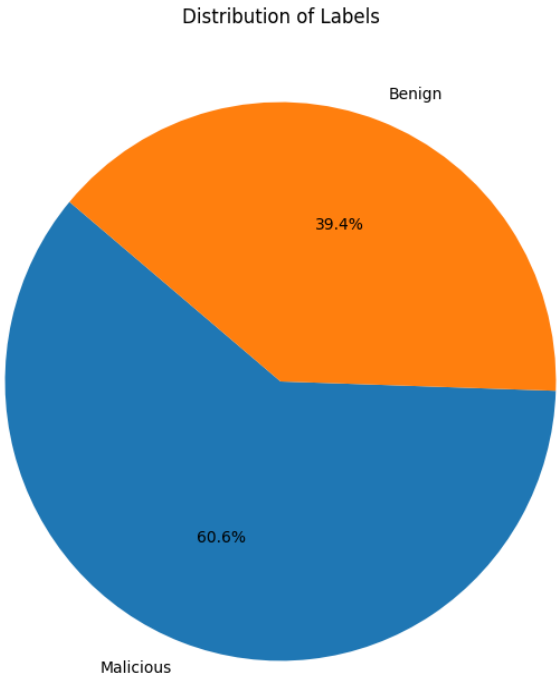


Figure 6. *The 5G NIDD dataset distribution.*

In the feature selection process, irrelevant columns from the dataset, particularly those unrelated to intrusion detection, were removed. Also, columns with predominantly null values were removed to prevent bias in the data, reducing the risk of overfitting and poor model performance. For columns with sparse missing values, imputation was applied using the strategies outlined above.

Before performing ANOVA (Analysis of Variance) on the dataset, categorical and numerical columns were extracted to enable easy encoding of categorical features. For encoding, label encoding and one hot encoding was employed to assign numerical value to categorical features. Label encoding involves assigning unique value to each category in the categorical value while one hot encoding creates a new binary column for each unique category in the categorical variable. The column corresponding to the category's value is set to 1 while other columns are set to 0.

Following data transformation, ANOVA was applied to assess the predictive importance of each feature. A p-value threshold of 0.05 was chosen, meaning features with p-values below this threshold were considered statistically significant. The 0.05 threshold is a standard convention, indicating less than a 5% likelihood that the results occurred by chance. Statistically significant features were retained for further analysis.

The retained features were then subjected to Principal Component Analysis (PCA), a statistical method used to reduce dataset dimensionality while preserving key information. PCA transforms a large set of correlated variables into a smaller set of uncorrelated variables known as principal components. The number of components retained is a hyper parameter that must be determined based on the desired level of dimensionality reduction and the ratio of variance explained. By setting the desired variance-explained ratio to 85%, the optimal number of principal components was determined to be 15. The data features were transformed into a 15-component space, and a new data frame was created using the resulting principal components. This transformed dataset will be used as input for the machine learning models.

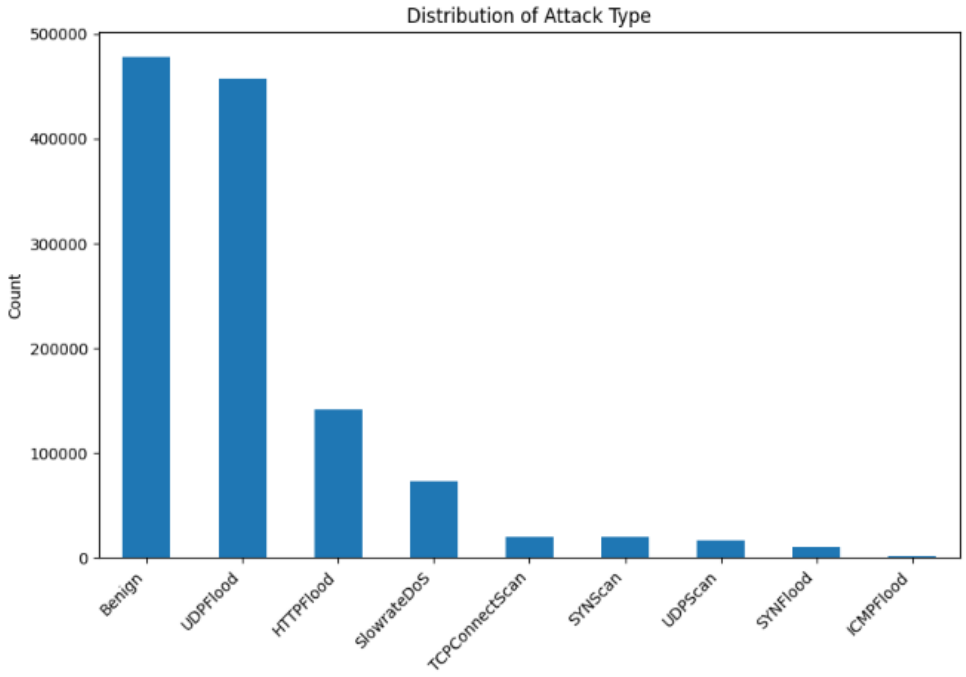


Figure 7. *The nine categorized attack types and the attack type dataset distribution for the 5G NIDD dataset.*

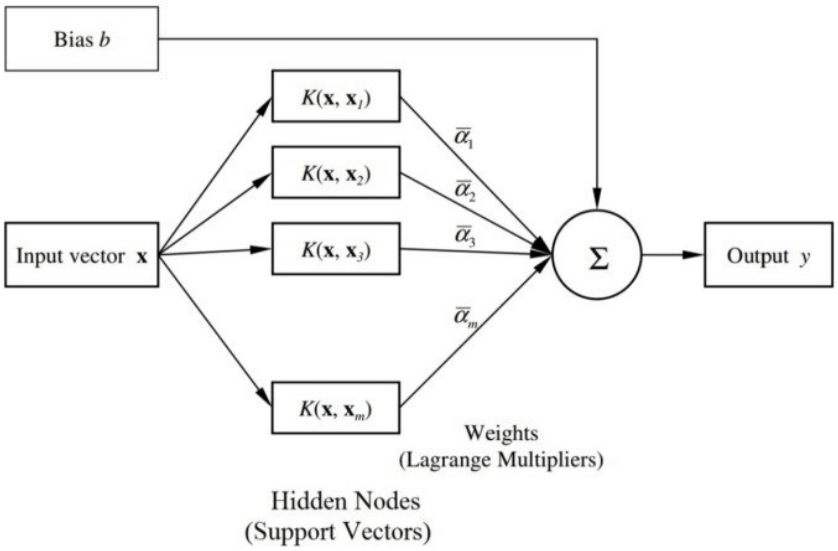


Figure 8. *Architecture of the support vector machine (SVM).*

3.4. Model Selection and Building

3.4.1. Support Vector Machine (SVM) Algorithm

Support Vector Machine (SVM) is a versatile supervised machine learning algorithm used for linear and nonlinear classification, regression, and even outlier detection tasks [10, 41]. The architecture of the support vector machine (SVM) is shown in Figure 8. The core objective of SVM is to find the optimal hyperplane that maximally separates different classes in the feature space, ensuring the largest possible margin between the nearest data points of each class. The dimensionality of the hyperplane corresponds to the number of features, becoming increasingly complex as the feature count grows.

Given the dataset

$$D=\left\{\left(x_i,y_i\right)\right\}_{i=1}^{N} \tag{1}$$

where $x_i\in\mathbb{R}^n$, $y_i\in\{-1,+1\}$ is the corresponding label, $y_i=-1$ represent the "Benign" class (0), $y_i=+1$ represent the "Malicious" class (1), and $N=1{,}215{,}872$ is the number of samples.

The goal is to find a hyperplane $w'x+b=0$ that maximizes the margin between the two classes while minimizing the classification error. The SVM optimization problem can be formulated as:

$$J=\min_{w,b,\xi}\frac{1}{2}|w|^2+C\sum_{i=1}^{N}\xi_i \tag{2}$$

subject to the constraints:

$$y_i\left(w'x+b\right)\geq 1-\xi_i,\ \xi_i\geq 0,\ \forall i=1,2,\ldots,N \tag{3}$$

where: $w\in\mathbb{R}^n$ is the weight vector defining the hyperplane; $b\in\mathbb{R}^p$ is the bias term; $\xi_i\geq 0$ are the slack variables that allow for some misclassification in the case of non-linearly separable data; and $C>0$ is the regularization parameter that controls the trade-off between maximizing the margin and minimizing the classification error.

Alternatively, the problem can be solved in the dual form:

$$J=\sum_{i-1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j x_i x_j \tag{4}$$

subject to the constraints:

$$\sum_{i=1}^{N}\alpha_i y_i=0,\ \ 0\leq\alpha_i\leq C,\ \ \forall i=1,2,\ldots,N \tag{5}$$

where: α_i are the Lagrange multipliers corresponding to each training sample.

Once the SVM model is trained, the decision function for predicting the class of a new sample is given by Equation (4):

$$f(x)=sign\left(\sum_{i=1}^{N}\alpha_i y_i x_i' x+b\right) \tag{6}$$

If $f(x)>0$, the attack is classified as "Malicious" (1); otherwise, it is classified as "Benign" (0).

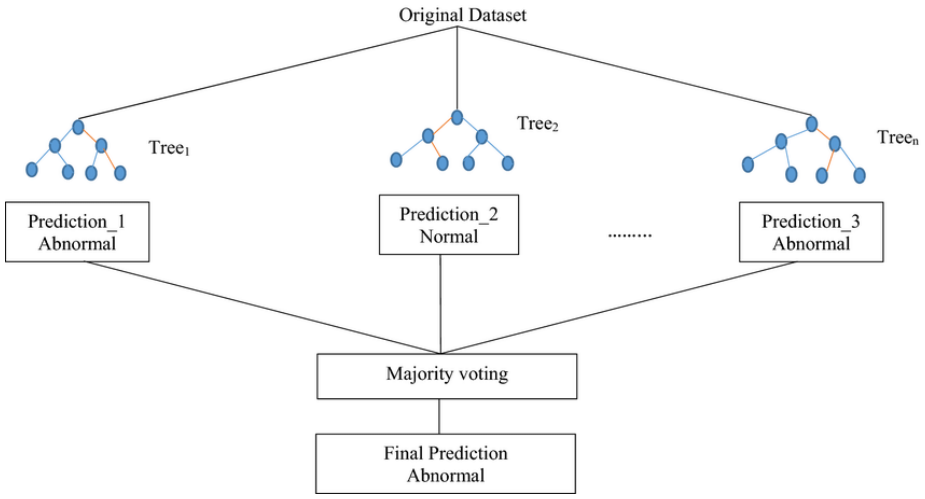


Figure 9. *Architecture of the random forest algorithm design.*

3.4.2. Random Forest Algorithm

The Random Forest algorithm is a robust machine learning technique that uses a collection of decision trees for model training and prediction as illustrated in Figure 9 [10, 41]. Each decision tree is built using a random subset of data and features, which introduces variability and helps reduce overfitting, enhancing overall predictive performance. During prediction, Random Forest aggregates the outputs of all trees (using voting for classification in this case) to provide stable and accurate results. Random Forest is employed for this work due to its characteristics of high accuracy, resistance to overfitting, ability to handle large dataset, and its built-in cross-validation.

Given a dataset as in equation (1), the goal is to build a model that can accurately classify a new sample as either "Normal" or "Attack" based on the ensemble of decision trees generated during training. Random forest use bootstrap sampling to create B different datasets D_b from the original dataset N_b by sampling with replacement, where D_b is given by:

$$D_b = \left\{\left(x_{ib}, y_{ib}\right)\right\}_{i=1}^{N_b}, \ b = 1, 2, \ldots, B \tag{7}$$

where $N_b \leq N$ is the number of samples in each bootstrap dataset.

For each bootstrap sample D_b, a decision tree T_b is grown using a subset of features. At each node, a random subset of features $F_b \subseteq \{1, 2, \ldots n\}$ is selected, and the best split is chosen based on the splitting criterion (Gini impurity in this case).

Gini impurity:

$$G(D) = 1 - \sum_{k=0}^{1} \left(\frac{N_k}{N}\right)^2 \tag{8}$$

where N_k is the number of samples in class k at that node.. Each tree T_b predicts the class label for a given input x such that:

$$y_b(x) = T_b(x), \ y_b(x) \in \{0,1\} \tag{9}$$

The final prediction $y(x)$ is obtained by majority voting over all trees in the forest:

$$y(x) = mode\left(\left\{y_b(x)\right\}_{b=1}^{B}\right) \tag{10}$$

where *mode* denotes the most frequent class label among the predictions from the B trees.

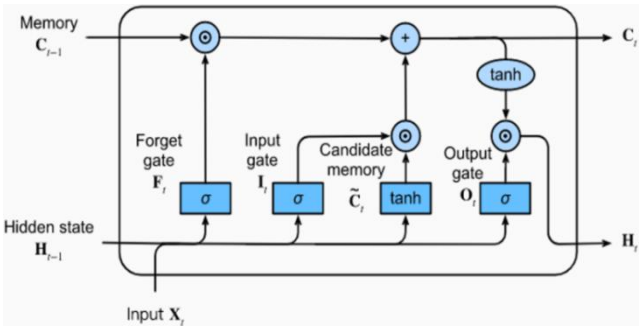


Figure 10. *The LSTM model architecture.*

3.4.3. Long-Short Temporary Memory (LSTM) Algorithm

The long-short temporary memory (LSTM) scheme is a type of recurrent neural network (RNN) aimed at dealing with the problem of vanishing gradient present in traditional RNNs [52, 53, 58]. The model architecture of the LSTM is shown in Figure 10. The LSTM uses a gating mechanism to control

information flow through a memory cell, making it possible for the network to capture long term dependencies in sequential data. Given the dataset:

$$D = \{(X_i, y_i)\}_{x=1}^{N} \quad (11)$$

where $X_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,T}]$ is a sequence of features for the *i*-th sample, with T time steps; $x_{i,T} \in \mathbb{R}^n$ is the feature vector at time step t for the *i*-th sample; $y \in \{0,1\}$ is the corresponding label, where $y_i = 0$ represent the "Normal" class and $y_i = 1$ represent the "Attack" class; and N is the number of samples.

The goal is to train an LSTM network to model the temporal dependencies in the sequences and accurately classify the sequence as either "Normal" or "Attack". Each LSTM cell computes the hidden state h_t and cell state c_t at each time step t based on the current input x_t, previous hidden state h_{t-1}, and previous cell state c_{t-1}.

Input gate:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (12)$$

Forget gate:

$$f_t = \sigma(W_f x_f + U_f h_{t-1} + b_f) \quad (13)$$

Output gate:

$$o_t = \sigma(W_o x_o + U_o h_{t-1} + b_o) \quad (14)$$

Cell state update:

$$c_t = f_t \odot c_{t-1} + i_t \odot tahn(W_c x_c + U_c h_{t-1} + b_c) \quad (15)$$

Hidden state:

$$h_t = o_t \odot tahn(c_t) \quad (16)$$

where i_t, f_t and o_t are the input, forget, and output gates respectively; W_i, W_f, W_o and W_c are the weight matrices for the inputs; U_i, U_f, U_o and U_c are the weight matrices for the hidden gates; b_i, b_f, b_o and b_c are the bias vectors, and $\sigma(\bullet)$ is the sigmoid activation function.

The output label:

$$y_i = \sigma(W_{out} h_T + b_{out}) \quad (17)$$

where W_{out} is the weight matrix for the output layer and b_{out} is the bias term.

The loss function is the binary cross-entropy loss, defined as:

$$L(y, \bar{y}) = -\frac{1}{N}\sum_{i=1}^{N}\left[\begin{array}{l} y_i \log(\bar{y}_i) + \\ (1-y_i)\log(1-\bar{y}_i) \end{array}\right] \quad (18)$$

To account for class imbalance, class weights can be applied to the loss function:

$$L(y, \bar{y}) = -\frac{1}{N}\sum_{i=1}^{N}\left[\begin{array}{l} w_1 y_i \log(\bar{y}_i) + \\ w_o (1-y_i)\log(1-\bar{y}_i) \end{array}\right] \quad (19)$$

3.5. Performance Evaluation of Models

The performance of all models in classifying the network traffic for each slice is evaluated based on certain parameters and metrics. In evaluating the performance of models, the focus is on the following criteria: accuracy, F1-score, recall, AUC-ROC, and computational complexity. Each metric provides a different perspective on the performance of the model, ensuring a holistic evaluation across the predictive and computational efficiency.

3.5.1. Accuracy

The accuracy of the models will be computed as the proportion of correctly classified instances to the total number of instances. For each model and network slice (eMBB, URLLC, and mMTC), accuracy serves as the primary indicator of the overall model performance. Accuracy is measured by the relation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (20)$$

where TP, TN, FP and FN represents the true positives, true negatives, false positives, and false negatives respectively.

3.5.2. Precision

Precision is the ratio of true positive predictions to the total predicted positives.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (21)$$

3.5.3. Recall (Sensitivity or True Positive)

Recall will be used to quantify the proportion of true positive cases correctly identified by the models. For critical applications like URLLC, where false negatives have significant consequences. Recall is given by:

$$Recall = \frac{TP}{TP + FN} \tag{22}$$

High recall ensures that the model minimizes false negatives, which is critical for applications that are sensitive to latency and require high reliability.

3.5.4. F1-Score

The F1-score is the harmonic mean of precision and recall. The F1-score will be used to provide a balanced measure of model performance. This metric is useful to handle the imbalance in class distribution and is especially relevant for URLLC and mMTC slices where certain classes may dominate the dataset. The F1 score is calculated as:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{23}$$

This metric is particularly important in situations where both precision and recall are important, as it balances the trade-off between them.

3.5.5. The Area Under Receiver Operating Characteristic Curve (AUC-ROC)

The area under receiver operating characteristic curve (AUC-ROC) is mathematically described as the integral of the True Positive Rate (TPR) with respect to the False Positive Rate (FPR) across all possible classification thresholds, and can be interpreted as the probability that a randomly chosen positive instance will be ranked higher than a randomly chosen negative instance. It quantifies the performance of a binary classifier across its entire range of operating points, with values closer to 1 indicating better performance. AUC-ROC can be expressed mathematically as:

$$\left.\begin{aligned} AUC - ROC &= \int_0^1 TPR\left(FPR\right) dFPR \\ &= \int_0^1 TPR\left(FPR^{-1}\left(x\right)\right) dx \end{aligned}\right\} \tag{24}$$

3.5.6. Confusion Matrix

A confusion matrix does not have a single formula but it is a format used to summarize the performance of a classification model by showing the actual versus predicted classes. For a two-class problem, it includes four components: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). These components are then used in formulas to calculate performance metrics such as the Accuracy (as defined in (20)), Precision (Positive Predictive Value, PPV given in (21), Recall (Sensitivity, True Positive Rate defined in (22)), Specificity (True Negative Rate), and the F1 score (given by (23)).

4. Results and Discussion

4.1. Results Overview

This section present the findings of this project, where three machine learning models, namely Random Forest, Support Vector Machine (SVM), and Long Short-Term Memory (LSTM), were tested for their effectiveness in detecting intrusion within a 5G sliced environment. The results are analyzed to assess the performance of all models across all slices using multiple metrics including accuracy, recall, precision, AUC-ROC, and execution time. Table 1 provides a detailed overview of the results obtained, examining the performance of the three models across the three slices on multiple metrics, including precision, recall, accuracy, F1-score, AUC-ROC, and execution time. A deep analysis of these metrics was done, shedding light on each model's strengths and weaknesses in the context of the given datasets. The outcome of this study is the importance of carefully selecting a model based on the computational resources and the slice-specific requirements.

Accuracy is an important metric in machine learning that shows the proportion of correctly predicted samples over the total number of samples. All models chosen for this study has been shown in previous works to give high accuracy values. In the context of the datasets used for this study, the LSTM model produced the highest accuracy across the eMBB and mMTC slices, while the SVM model came out top in the URLLC slice. Although all models achieved impressive accuracy scores above 95% across all slices.

Precision is another important metric which shows the proportion of correct identifications. In terms of precision, the LSTM model slightly outperform the SVM and Random Forest models in the eMBBslice having a precision score of 99.88%, Random Forest came out top in the mMTC slice with 99.51% precision, while all models achieved equivalent precision scores of 99.98% in the URLLC slice. Overall, all models achieved extremely high precision scores, showing their capability to correctly classify positive instances and minimize false positive errors.

The recall score measures how well the model correctly captures positive cases. It is the ratio of the true positives to the sum of the true positives and false negatives. In this study, the Random Forest model achieved an outstanding recall rate of 99.94% and 99.99% in the eMBB and mMTC slices respectively. The LSTM model follows slightly with 99.89% and 99.81% in both slices. The SVM outperformed the LSTM and Random Forest models in the URLLC slice recording a recall rate of 99.99%. These extremely high recall rates emphasize the effectiveness of these models identifying positive cases correctly, hence reducing false negatives.

F1-score represents the harmonic mean of precision and recall. In terms of the F1-score, the LSTM model outperforms the other models in both the eMBB and mMTC slices, but comes behind in the URLLC slice. All models achieved impressive F1-scores, signifying a balanced performance in

precision and recall.

The area under the receiver operating characteristic curve (AUC-ROC) evaluates the performance of a model across all possible classification threshold, balancing sensitivity and specificity. In this study, the LSTM model achieved the highest AUC-ROC score across all slices, boasting a near-perfect score of 99.99%.

Table 1. *Overall performance metrics of the three machine learning models.*

S/N	Slice	Models	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Execution Time (s)
1	eMBB	Random Forest	99.81	99.74	99.94	99.84	99.99	97.55
		SVM	99.08	99.21	99.27	99.24	99.75	1588.63
		LSTM	99.86	99.88	99.89	99.89	99.99	154.46
2	mMTC	Random Forest	96.37	99.51	94.77	97.08	98.99	107.20
		SVM	96.09	97.53	94.29	97.08	98.99	3586.39
		LSTM	97.56	99.31	96.89	98.04	99.34	383.31
3	URLLC	Random Forest	99.98	99.97	99.99	99.98	99.99	4.43
		SVM	99.99	99.98	99.98	99.99	99.99	15.54
		LSTM	99.92	99.98	99.81	99.89	99.99	83.60

The complexity of each model, measured by their execution time, is crucial, especially in scenarios where real time detection is critical. The Random Forest model gave the fastest execution time, recording 97.55, 107.2, and 4.43 seconds across the eMBB, mMTC, and URLLC slices respectively. Conversely, the SVM models recorded the highest execution time across all slices, suggesting poor computational efficiency.

Table 2. *Performance metrics for the eMBB slice.*

S/N	Metric	Random Forest	SVM	LSTM
1	Accuracy (%)	99.81	99.08	99.86
2	Precision (%)	99.74	99.21	99.88
3	Recall (%)	99.94	99.27	99.89
4	Fi-Score (%)	99.84	99.24	99.89
5	AUC-ROC (%)	99.99	99.75	99.99
6	Execution Time (sec)	97.55	1588.63	154.46

4.1.1. Performance Evaluation of the eMBB Slice

The performance metrics of the Random Forest, SVM, and LSTM models for the eMBB slice are shown in Table 2. It can be observed from Table 2 that the LSTM model performs slightly better in terms of accuracy (99.86%) and recall (99.89%) compared to Random Forest, which has an accuracy of 99.81% and a recall rate of 99.94%. SVM lags behind both with 99.08% accuracy and 99.27% recall. Although the Random Forest model offers high precision (99.74%) and F1-score (99.84%), the LSTM is slightly superior in both metrics. SVM again lags behind with 99.21% precision and 99.24% F1-score. Both Random Forest and LSTM models show strong discrimination capability with nearly perfect AUC-ROC values of 99.99%, SVM slightly trailing at 99.75%. In terms of complexity, Random Forest is bar far the most efficient with execution time of 99.55 seconds. SVM performs poorly in this metric having execution time of 1588.63 seconds, while LSTM falls in between, taking 154.46 seconds.

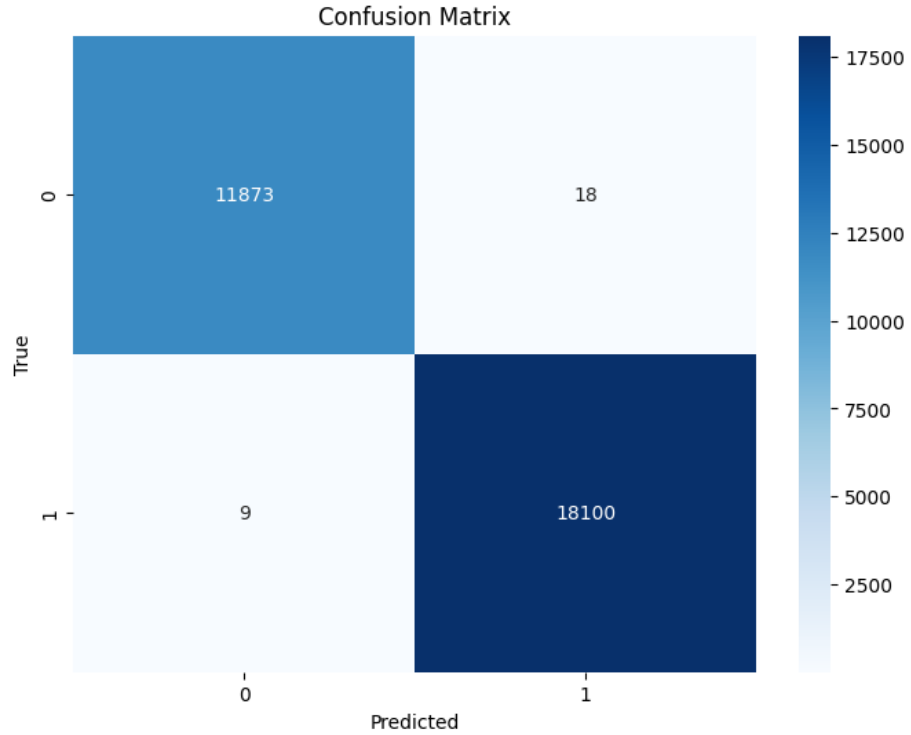


Figure 11. *Confusion matrix for the random forest model.*

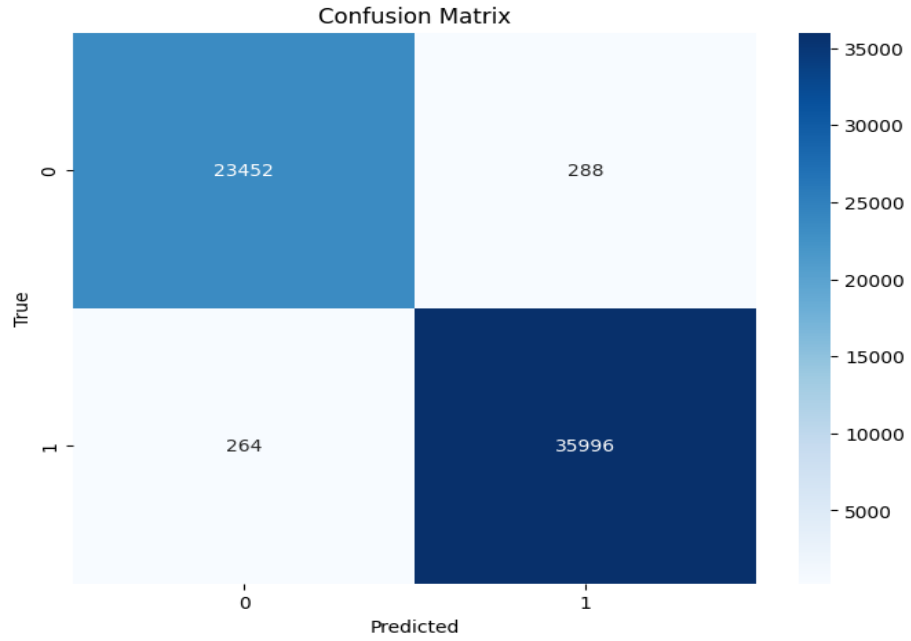


Figure 12. *Confusion matrix for the SVM model.*

The confusion matrices show that the models perform well in detecting both normal and malicious traffic in the eMBB slice. The confusion correctly identifies 11,873 normal instances (true negatives) and 18,100 attacks (true positives), demonstrating strong accuracy. The low count of 18 false positives indicates that there are few instances where normal traffic is mistakenly flagged as an attack. Also, the model misses only 9 actual attacks (false negatives), meaning that very few intrusions go undetected. This combination of high true positive and true negative rates, along with minimal false positives and false negatives, suggests that the model is effective for network intrusion detection in the eMBB slice, offering reliable protection while minimizing false alarms and missed threats. Figure 11 and Figure 12 show the confusion

matrices for the Random Forest and SVM models.

Overall, for the eMBB slice, the LSTM model delivers the best performance across all metrics. It can be also observed that Random Forest offers excellent performance with less computational complexity making it a more practical choice for time-sensitive applications.

From Figure 13, it can be seen that there is a significant drop in the training and validation loss of the LSTM model in the early stages of training. This indicates rapid learning by the model. For majority of the training process, the validation loss closely tracks the training loss, suggesting that the model is generalizing well. Although both losses show fluctuations, there are no signs of overfitting since the validation loss does not diverge from the training loss but remain in sync as training progress. Thus, it can be concluded that the near-perfect performance of the LSTM model is not due to overfitting.

Figure 13. *Plot of training and validation loss of the LSTM model for the eMBB slice.*

Table 3. *Performance metrics for the mMTC slice.*

S/N	Metric	Random Forest	SVM	LSTM
1	Accuracy (%)	96.37	96.09	97.56
2	Precision (%)	99.51	97.53	99.31
3	Recall (%)	94.77	94.29	96.89
4	Fi-Score (%)	97.08	96.85	98.04
5	AUC-ROC (%)	98.99	98.36	99.34
6	Execution Time (sec)	107.2	3586.39	383.31

4.1.2. Performance Evaluation of the mMTC Slice

The performance metrics for the Random Forest, SVM, and LSTM models for the mMTC slice are shown in Table 3. It can be observed in Table 3 that the LSTM again outperforms both Random Forest and SVM in terms of accuracy, achieving 97.56% as against 96.37% and 96.09% of the Random Forest and SVM models respectively. Random Forest achieved the highest precision (99.51%), while the LSTM and SVM trail at a slightly lower value (99.31% and 97.53% respectively). Again the LSTM model excels in recall and AUC-ROC metrics above the SVM and Random Forest, achieving values of 96.89% and 99.34% for recall and AUC-ROC respectively. Random Forest again shows significantly lower execution

time (107.2 seconds) compared to LSTM (383.31 seconds) and SVM (3586.39 seconds). Figure 14 and Figure 15 show the confusion matrix for the Random Forest and SVM models respectively.

Overall, the LSTM model once again delivers the most balanced performance with high accuracy, recall and AUC-ROC. Random Forest offers superior computational efficiency making it a good option for this use case due to the resource constraints of IoT devices. SVM is slow, computationally complex and thus less competitive across metrics.

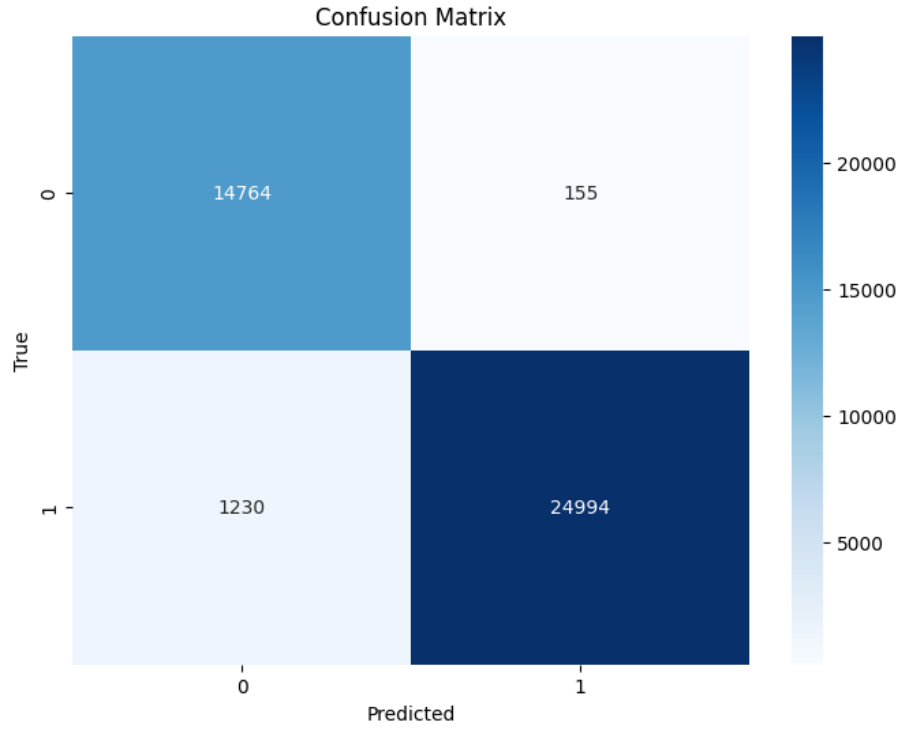


Figure 14. *Confusion matrix for the random forest model.*

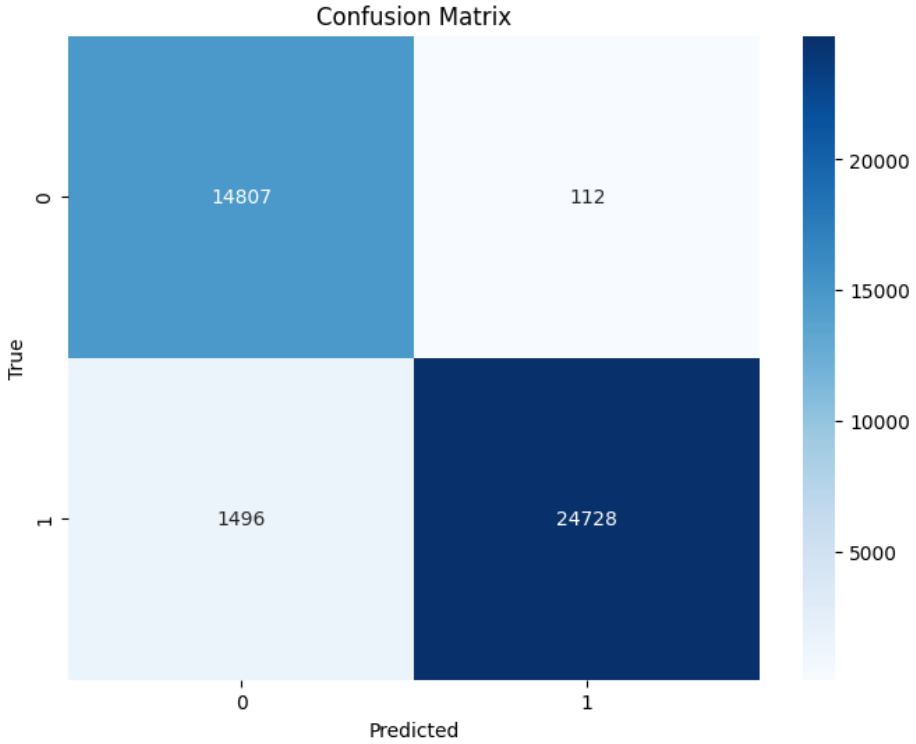


Figure 15. *Confusion matrix for the SVM model.*

4.1.3. Performance Evaluation of the URLLC Slice

The performance metrics of the Random Forest, SVM, and LSTM models for the URLLC slice are shown in Table 4. For the URLLC slice, as shown in Table 4, the SVM model has a slight edge in accuracy (99.99%) over the Random Forest (99.98%), but the difference is minimal. Both Random Forest and SVM exhibit identical precision, recall, AUC-ROC, and F1-score values. The LSTM model in the slice trails in all metrics. In terms of computational complexity, Random Forest outshines both the SVM and LSTM models, achieving an execution time of 4.43 seconds as against 15.54 seconds and 83.6 seconds for the LSTM and SVM models respectively. It can be concluded that although the SVM has slightly better accuracy than the Random Forest model, the computational advantage of the Random Forest model cannot be overlooked, particularly in the case of the URLLC slice where low-latency and reliability is critical.

Table 4. *Performance metrics for the URLLC slice.*

S/N	Metric	Random Forest	SVM	LSTM
1	Accuracy (%)	99.98	99.99	99.92
2	Precision (%)	99.97	99.98	99.98
3	Recall (%)	99.99	99.98	99.81
4	Fi-Score (%)	99.98	99.99	99.89
5	AUC-ROC (%)	99.99	99.99	99.98
6	Execution Time (sec)	4.43	15.54	83.6

4.2. Discussion of Results

Selecting the most suitable machine learning model for network intrusion detection in 5G networks requires a nuanced analysis of various critical factors, including model accuracy, computational efficiency, ease of deployment, and slice requirements or use case. These elements are critical in network security environment where the right balance can significantly impact the effectiveness and reliability of the detection systems.

In the eMBB slice, both Random Forest and LSTM give exceptional performance in terms of accuracy and precision, which implies that efficiently differentiate between normal and malicious traffic with a low false positive rate. They also achieved high recall values indicating that they can detect all true attacks, minimizing the chances of letting intrusions go undetected. Both models can practically be implemented or deployed for the eMBB slice, although Random Forest may be preferred for eMBB for time-critical applications where quick identification and response is necessary.

The mMTC slice is characterized by large device density with low data rates. Quick response to intrusion detection is critical. Packet latency for the mMTC slice may not be as critical as required for the URLLC slice or even the eMBB slice. However, with many devices generating traffic, processing time becomes a concern. LSTM provides better accuracy and overall performance metric, but its slower execution time might hinder its use in environments where quick response to potential intrusions is critical. Random Forest may still be favoured here for its computational efficiency, especially if traffic volume increases dramatically due to the number of connected devices. LSTM is preferable where more resources are available for analysis.

The URLLC slice demands extremely low latency and high reliability. Its use case includes mission-critical applications such as autonomous driving, remote surgeries, and industrial automation. Random Forest and SVM deliver extremely high accuracy, precision and recall, which implies that they are highly effective in detecting intrusions in low-latency environments. Random Forest is by far the fastest and is the optimal choice in mission critical applications where milliseconds matter. SVM, irrespective of other performance metrics, is unacceptable in low-latency environments where delay may result in catastrophic failures.

The ease of deployment is another key factor to consider, encompassing the complexity of training and tuning models, the need for ongoing maintenance, and the ability of the model to adjust to changes in network behavior. Complex models like Neural Networks require significant training, fine-tuning, and regular updates to maintain effectiveness, which demands considerable resources and expertise. On the other hand, simpler models are easier to deploy and maintain but may struggle to keep up with evolving attack patterns, potentially losing effectiveness over time unless continuously updated. Security administrators must assess the infrastruc-

ture's capability to support complex models without affecting other operations, evaluate the necessary security level against potential threats, and weigh the operational impacts of false positives and false negatives.

These results highlight the importance of understanding the trade-off between performance and computational efficiency when selecting an ML model. As indicated by our results, no model excels in all situations; each has its own strengths and weaknesses, and its performance can vary significantly depending on the context and the data used.

5. Conclusion and Future Direction

5.1. Conclusion

This study contributes to the field of 5G network security by presenting a comparative analysis of machine learning models tailored to different 5G network slices. The findings advance knowledge on the effectiveness of slice-specific IDSs and provide practical insights for selecting models based on slice requirements and operational constraints. The research also highlights the potential of LSTM for high-accuracy detection in complex environments and establishes Random Forest as a viable model for real-time applications.

This research has developed and evaluated slice-specific machine learning algorithms for intrusion detection in 5G networks, focusing on eMBB, mMTC, and URLLC slices. The study demonstrates that machine learning models can effectively address the security challenges introduced by 5G network slicing, tailoring intrusion detection systems to the unique characteristics of each slice. The results showed that LSTM performed best in terms of overall accuracy and AUC-ROC, particularly in the eMBB and mMTC slices, while Random Forest proved more efficient in computational complexity, making it suitable for real-time applications in URLLC. SVM showed high accuracy but lacked efficiency in terms of computational cost, making it less practical for real-time, low-latency environments.

5.2. Future Directions

Based on the results obtained in this study, the following recommendations can be considered as future directions to enhance the performance strategies proposed in this study: *1).* The Random Forest model is recommended for real-time and latency-sensitive applications like URLLC due to its low computational complexity and high accuracy. For environments with resource constraints or dense device connectivity, Random Forest offers a practical balance between performance and execution time; *2).* For enhanced Mobile Broadband (eMBB) and massive machine-type communication (mMTC) slices, Long Short-Term Memory (LSTM) provides superior performance in detection accuracy and recall. It is recommended for environments where computational resources are sufficient, and accurate detection is critical. However, the trade-off between performance and execution time must be considered; and *3).* While this study used controlled datasets, future work should validate these models with real-world data to ensure practical applicability across diverse 5G networks.

Despite the several challenges facing the 6G networks [69] and the promising vision of the 6G networks [70], the adoption of the slice-specific techniques presented in this paper can be deployment for improved performance of the 6G networks. Finally, in order to reduce computational cost and overload, the SVM algorithm can be implemented on high performance computational platform such as field programmable gate arrays (FPGS) so as to meet the real-time constraints in URLLC environment using techniques described in Akpan and co-workers [71-73].

Abbreviations

3GPP	Third Generation Partnership Project
eMBB	Enhanced Mobile Broadband
mMTC	Massive Machine-Type Communication
URLLC	Ultra-Reliable Low-Latency Communication
LSTM	Long Short-Term Memory
MVNO	Mobile Virtual Network Operators
NIDS	Network Intrusion Detection Systems
NFV	Network Function Virtualization
VNF	Virtual Network Function
SDN	Software-Defined Networking
AUC	Area Under Curve
ROC	Receiver Operating Characteristics
IDS	Intrusion detection systems
DDoS	Distributed Denial-of-Service
FFDNN	Feed-Forward Deep Neural Network
WBFU	Wrapper Based Feature Extraction Unit
UNSW	University of New South Wales
BN15	Nota Bene 2015 (Not Well or Take Notice)
CNN	Convolutional neural network
AWID	Aegean Wi-FI Intrusion Dataset
IDPS	Intrusion Detection and Prevention System
KDD	Knowledge Discovery in Database
ICMP	Internet Control Message Protocol
CIC	Carrier Identification Code
TCP	Transmission Control Protocol
HTTP	Hypertext Transfer Protocol
NIDD	Non-IP Data Delivery
MEC	Multi-access Edge Computing
5G	Fifth Generation
5GTN	5G Test Network
MEC	Multi-access Edge Computing
DL	Deep learning
ML	Machine learning
IoT	Internet of Things
SVM	Support Vector Machine
Tbps	terabits-per-second
OAI	Open Air Interface

AI	Artificial inteeligence
ELM	extreme learning machine
MLP	Multilayer percetron
GPU	Graphics Processing Unit
DoS	Denial-of-Service
UDP	User Datagram Protocol
SYN	Synchronization
RF	Random Forest
RNN	Recurrent neural Network
IoV	Internet of Vehicles
ECU	Electronic Control Unit
ANOVA	Analysis of Variance
PCA	Principal Component Analysis
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
NSL	No Secure Location
IP	Internet Protocol

Author Contributions

Vincent Andrew Akpan: Conceptualization, Formal Analysis, Investigation, Methodology, Project administration, Software, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing

Emmanuel Chinenye Njoku: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing - original draft, Writing - review & editing

Ebubechukwu Ifeoluwa Obi: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing - original draft, Writing - review & editing

Funding

This research is not supported by any external funding.

Data Availability Statement

The data supporting the outcome of this research work has been reported in this manuscript.

The 5G-NIDD dataset is a fully labeled dataset created using a functional 5G test network at the University of Oulu, Finland (see reference [66]).

The CIC IoV 2024 dataset contributes significantly to cybersecurity for Internet of Vehicles (IoV) for ultra-reliable low-latency communication (URLLC) (see reference [67]).

The CIC IoT dataset was collected from an IoT topology with 105 devices where 33 distinct attacks were executed (see reference [68]).

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] W. Lee, T. Na and J. Kim, "How to Create a Network Slice? - A 5G Core Network Perspective," *In the Proceedings of the IEEE 2019 21st International Conference on Advanced Communication Technology (ICACT)*, 17 - 20 February, 2019, PyeongChang, Korea, pp. 616 - 619, 2019. https://doi.org/10.23919/ICACT.2019.8701936

[2] M. Minardi, Y. Drif, T. X. Vu and S. Chatzinotas, "SAST-VNE: A Flexible Framework for Network Slicing in 6G Integrated Satellite-Terrestrial Networks," *IEEE Journal on Selected Areas in Communications*, vol. 43, no. 1, pp. 234 - 244, 2024. https://doi.org/10.1109/JSAC.2024.3460066

[3] W. Patel and P. Tripathy, "eMBB, URLLC, and mMTC 5G Wireless Network Slicing: A Communication-Theoretic View," *In the Proceedings of the 2023 International Conference on Artificial Intelligence and Smart Communication (AISC)*, 27 - 29 January, 2023, Greater Nolda, India, pp. 1291 - 1296, 2023. https://doi.org/10.1109/AISC56616.2023.10085623

[4] P. Pal, O. Misra and M. Shahid, "Network Slicing In 5G Architecture: Applications and Services," *NIET Journal of Engineering and Technology*, vol. 12, no. 1, pp. 34 - 40, 2023. https://doi.org/10.62797/njet.vol.12.issue.01W.005

[5] P. Popovski, K. F. Trillingsgaard, O. Simeone and G. Durisi, "5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View," IEEE Access, vol. 6, pp. 55765 - 55779, 2018. https://doi.org/10.1109/ACCESS.2018.2872781

[6] I. A. Bartsiokas, P. K. Gkonis, D. I. Kaklamani and I. S. Venieris, "A Federated Learning-Based Resource Allocation Scheme for Relaying-Assisted Communications in Multicellular Next Generation Network Topologies," *Electronics*, vol. 13, no. 390, pp. 1 - 18, 2024. https://doi.org/10.3390/electronics13020390

[7] J. Cunha, P. Ferreira, E. M. Castro, P. C. Oliveira, M. J. Nicolau, I. Nunez, X. R. Sousa and C. Serodio, "Enhancing network slicing security: Machine learning, software-defined networking, and network functions virtualization-driven strategies," *Future Internet*, vol. 16, no. 7, Article Number 226, pp. 1 - 36, 2024. https://doi.org/10.3390/fi16070226

[8] C. D. Alwis, P. Porambage, K. Dev, T. R. Gadekallu and M. Liyanage, "A Survey on Network Slicing Security: Attacks, Challenges, Solutions and Research Directions," *IEEE Communications Surveys and Tutorials*, vol. 26, no. 1, pp. 534 - 570, 2024.

[9] R. Pries, H. D. Morper, N. Galambosi and M. Jarschel, "Network as a Service - A Demo on 5G Network Slicing," *In the Proceedings of the 2016 28th International Teletraffic Congress (ITC 28)*, vol. 1, pp. 209 - 211, 2016.

[10] A. Fahmin, Y. C. Lai, M. S. hassain and Y. D. Lin, "Performance Modeling and Comparison of NFV Integrated with SDN: Under or Aside," *Journal of Network and Computer Applications*, vol. 113, pp. 119 - 129, 2018.

[11] A. A. Barakabitze, A. Ahmad, R. Mijumbi and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, no. 10698, pp. 1 - 19, 2020. https://doi.org/10.1016/j.comnet.2019.106984

[12] P. C. Amogh, A. Vashistha, Y. S. Rao, P. Yiyang and S. Sumei, "Defending 5G Networks Against DDoS Attacks Using Quarantine Slice Manager Architecture," *In the Proceedings of the 2024 IEEE VTS Asia Pacific Wireless Communication Symposium (APWCS)*, 21 - 23 August, 2024, Singapore, pp. 1 - 5, 2024. https://doi.org/10.1109/APWCS61586.2024.10679311

[13] A. Pc, A. Vashistha, Y. S. Rao, Y. Pei and S. Sumei, S. (2024). Defending 5G Networks Against DDoS Attacks Using Quarantine Slice Manager Architecture. *In the Proceedings of the 2024 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, pp. 1 - 5, 2024.

[14] S. Gao, R. Lin, Y. Fu, H. Li and J. Cao, "Security threats, requirements, and recommendations on creating 5G network slicing system: A survey," *Electronics*, vol. 13, no. 10, Article Number 1860, pp. 1 - 32, 2024. https://doi.org/10.3390/electronics13101860

[15] Z. Awada, M. E. Helou, K. Khawam and S. Lahoud, "Dynamic Multi-Tenant RAN Slicing for eMBB, URLLC, and mMTC in 5G Networks," *In the Proceedings of the 2023 26th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 19 - 22 November, 2023, Tampa, FL, USA, pp. 1 - 7, 2023. https://doi.org/10.1109/WPMC59531.2023.10338951

[16] I. Ahmad, T. Kumar, M. Liyanage, J. Okwuibe, M. Ylianttila and A. Gurtov, "5G security: Analysis of threats and solutions," 2017 IEEE Conference on Standards for Communications and Networking (CSCN), 18 - 20 September, 2017, Helsinki, Finland, 2017, pp. 193 - 199, https://doi.org/10.1109/CSCN.2017.8088621

[17] S. Park, D. Kim, Y. Park, D. Kim and S. Kwon, 5G Security Threat Assessment in Real Networks," *Sensors*, vol. 21, no. 16, Article Number: 5524, pp. 1 - 22, 2021. https://doi.org/10.3390/s21165524

[18] A. K. M. B. Haque, T. Nausheen, A. M. Shaan and S. A. Murad, "Security Attacks and Countermeasures in 5G Enabled Internet of Things," *In: Bhushan, B., Sharma, S. K., Kumar, R., Priyadarshini, I. (eds) 5G and Beyond. Springer Tracts in Electrical and Electronics Engineering*, Springer, pp. 1 - 311, 2023. Singapore. Available:: https://doi.org/10.1007/978-981-99-3668-7_7

[19] V. P. Singh, M. P. Singh, S. Hedge and M. Gupta, Security in 5G Network Slices: Concerns and Opportunities," IEEE Access, vol. 12, pp. 52727 - 52743, 2024. https://doi.org/10.1109/ACCESS.2024.3386632

[20] M. Humayun, B. Hamid, N. Z. Jhanjhi, G. Suseendran and M. N. Talib, "5G Network Security Issues, Challenges, Opportunities and Future Directions: A Survey," Journal of Physics: Conference Series - International Conference on Recent Trends in Computing (ICRTCE-2021), 20 - 22 May, 2021, Maharashtra, India, pp. 1 - 11, 2021. https://doi.org/10.1088/1742-6596/1979/1/012037

[21] C. I. Fan, Y. T. Shih, J. J. Huang and W. R. Chiu, "Cross-Network-Slice Authentication Scheme for the 5th Generation Mobile Communication System," IEEE Transactions on Network and Service Management, vol. 14, no. 8, pp. 1 - 12, 2021. https://doi.org/10.1109/TNSM.2021.3052208

[22] A. B. Sabastian, W. K. B. Mikkel and A. Birger, "5% Attacks and Countermeasures," In the Proceedings of the 25th International Conference on Wireless Personal Multimedia Communications, pp. 1 - 7, 2023. https://doi.org/10.1109/WPMC55625.2022.10014962

[23] A. Bajpai and S. Bhargava, "A comprehensive review on security threats and countermeasures in 5G network systems," AIP Conference Proceedings, vol. 3327, no. 020024, 2025. https://doi.org/10.1063/5.0289535

[24] H. Jmila and G. Blanc, "Towards Security-Aware 5G Slice Embedding," *Computer and Security*, vol. 100, no. 102075, 2021. https://doi.org/10.1016/j.cose.2020.102075

[25] M. Mahyoub, A. AbdulGhaffar, E. Alalade, A. Matrawy, "Security-Aware Network Function Sharing Model for 5G Slicing," *Security and Privacy*, vol. 8, no. 3, pp. 1 - 13, 2025. https://doi.org/10.1002/spy2.70039

[26] H. Cao, Y. Hu, S. Wu and Y. Guo, "Dynamic and Secure Resource Allocation Framework of Slices for 5G-Enabled Cyber Physical Systems," *In the Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2 - 5 May, 2022, New York, USA, pp. 1 - 6, 2022. https://doi.org/10.1109/INFOCOMWKSHPS54753.2022.9798374

[27] D. Sattar and A. Matrawy, "Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices," *In the Proceedings of the 2019 IEEE Conference on Communications and Network Security (CNS)*, pp. 82 - 90, 2019.

[28] H. Sharma, P. Kumar and K. Sharma, "Recurrent Neural Network based Incremental model for Intrusion Detection System in IoT," *Scalable Comput. Pract. Exp.*, vol. 25, pp. 3778 - 3795, 2024.

[29] M. S. Bhelkar, "Network Intrusion Detection System," *International Journal of Scientific Research in Engineering and Management*, vol. 8, no. 4, pp. 1 - 5, 2024.

[30] S. Ouiazzane, M. Addou and F. Barramou, "A Suricata and Machine Learning Based Hybrid Network Intrusion Detection System," In: Maleh, Y., Alazab, M., Gherabi, N., Tawalbeh, L., Abd El-Latif, A. A. (eds) Advances in Information, Communication and Cybersecurity (ICI2C 2021). Lecture Notes in Networks and Systems, vol 357, pp. 474 - 485, 2022. https://doi.org/10.1007/978-3-030-91738-8_43

[31] E. Gyamfi and A. D. Jurcut, "Intrusion Detection in Internet of Things Systems: A Review on Design Approaches Leveraging Multi-Access Edge Computing, Machine Learning and Datasets," *Sensors*, vol. 22, no. 10, Article Number 3744, pp. 1 - 33, 2022. https://doi.org/10.3390/s22103744

[32] A. Almazroi, N. Moustafa and S. Alsharif, "A Framework for Network Intrusion Detection Systems in 5G Networks," *IEEE Access.*, 2020.

[33] A. Almurshedi, A. V. Vasilakos and Z. Li, "A comprehensive survey on artificial intelligence for network intrusion detection: Challenges and solutions," *Computer Networks*, vol. 196, no. 108244, 2021.

[34] P. Radoglou-Grammatikis, G. Naks, G. Amponis, S. Giannakidou, T. Lagkas and V. Argyriou, "5GCIDS: An Intrusion Detection System for 5G Core with AI and Explainability Mechanisms," In the proceedings of the 2023 IEEE GlobecomWorkshops (GC Wkshps), 4 - 8 December, 2023, Kuala Lumpur, Malaysia, pp. 353 - 358, 2023. https://doi.org/10.1109/GCWkshps58843.2023.10464667

[35] V. Adat, I. Politis, C. Tselios, P. Galiotos and S. Kotsopoulos, "On Blockchain Enhanced Secure Network Coding for 5G Deployments," In the Proceddings of the 2018 IEEE Global Communication Conference (GLOBECOM), 9 - 13 December, 2018, Abu Dubai, United Arab Emirates, pp. 1 - 7, 2018. https://doi.org/10.1109/GLOCOM.2018.8647581

[36] M. S. Peelam, V. Chamola and B. K. Chaurasia, "Blockchain-Enabled Intrusion Detection Systems for Real-Time Vehicle Monitoring," *Vehicular Communications*, vol. 55, no. 100961, 2025. https://doi.org/10.1016/j.vehcom.2025.100961

[37] P. V. A. Alves, M. A. S. S. Goldbarg, W. K. P. Barros, I. D. Rego, V. J. M. T. Filho, A. M. Martins, V. A. D. Sousa, R. D. R. Fontes, E. H. D. S. Aranha, A. V. Neto and M. A. C. Fernandes, "Machine Learning Applied to Anomaly Detectio on 5G O-RAN Architecture," *Procedia Computer Science*, vol. 222, pp. 81 - 93, 2023. https://doi.org/10.1016/j.procs.2023.08.146

[38] G. K. Pavani and B. Veeramallu, "Hybrid Machine Learning Framework for Anomaly Detection in 5G Networks," *Journal of Information Systems Engineering and Management*, vol. 10, no 32s, pp. 733 - 739, 2025.

[39] Z. Lv, A. K. Singh and J. Li, "Deep Learning for Security Problems in 5G Heterogeneous Networks," *IEEE Networks*, vol. 35, no. 2, 67 - 73, 2021. https://doi.org/10.1109/MNET.011.2000229

[40] T. Saha, N. Aaraj and N. Jha, "Machine Learning Assisted Security Analysis of 5G-Network-Connected Systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 4, pp. 2006 - 2024, 2022. https://doi.org/10.1109/TETC.2022.3147192

[41] X. Y. Gong, Y. Yang, Y. Zhang, N. Li, Y. Guan and R. Jiang, "Feature Selection Method for Network Intrusion based on Hybrid Meta-Heuristic Dynamic Optimization Algorithm," Computer & Security, vol. 156, no. 104512, 2025. https://doi.org/10.1016/j.cose.2025.104512

[42] Y. Liu, Y. Chen and X. Zhang, "Adaptive Intrusion Detection System Based on Machine Learning Algorithms for 5G Wireless Networks," *Sensors*, vol. 21, no. 3, Article Number: 807, 2021.

[43] K. Uszko, M. Kasprzyk, M. Natkaniec and P. Cholda, "Rule-Based System with Machine Learning Support for Detecting Anomalies in 5G WLANs," *Electronics*, vol. 12, no. 11, Article Number: 2355, pp. 1 - 28, 2023. https://doi.org/10.3390/electronics12112355

[44] C. Hamroun, A. Fladenmuller, M. Pariente and A. G. Pujolle, "Intrusion Detection in 5G and WiFi Networks: A Survey of Current Methods, Challenges, and Perspectives," *IEEE Access*, vol. 13, pp. 40950 - 40976, 2025. https://doi.org/10.1109/ACCESS.2025.3546338

[45] Y. Li, Y. Chen, H. Zhang and Y. Wang, "Federated Learning for Edge Computing: A Review," *IEEE Access*, vol. 8, pp. 179835 - 179845, 2020.

[46] N. Latif, W. Ma and H. B. Ahmad, "Advancements in securing federated learning with IDS: a comprehensive review of neural networks and feature engineering techniques for malicious client detection," *Artificial Intelligence Review*, vo. 58, no. 91, pp. 1 - 63, 2025.

[47] R. Bocu and M. Lavich, "Real-Time Intrusion Detection and Prevention System for 5G and Beyond Software-Defined Networks," *Symmetry*, vol. 15, no. 110, pp. 1 - 15, 2023. https://doi.org/10.3390/sym15010110

[48] R. A. Khan, H. U. Khan, H. S. Alwageed, H. A. Hashimi and I. Keshta, "5G Network Security Mitigation Model: An ANN-ISM Hybrid Approach," *IEEE Open Journal of the Communication Society*, vol. 6, pp. 881 - 925, 2025. https://doi.org/10.1109/OJCOMS.2025.3529717

[49] B. Bousalem, V. F. Silva, R. Langar and S. Cherrier, "Deep Learning-based Approach for DDoS Attacks Detection and Mitigation in 5G and Beyond Mobile Networks," *In the Proceedings of the 8th International Conference on Network Softwarization (NetSoft 2022)*, Jun 2022, Milan, Italy. pp. 228-230, 2022.

[50] B. Farzaneh, N. Shahriar, A. H. A. Muktadir, M. S. Towhid and M. S. Khosravani, "DTL-5G: Deep transfer learning-based DDoS attack detection in 5G and beyond networks," *Computer Communications*, vol. 228, no. 107927, pp. 1 - 14, 2024. Available: https://doi.org/10.1016/j.comcom.2024.107927

[51] S. M. Kasongo and Y. Sun, "A deep learning method with wrapper based feature extraction for wireless intrusion detection system," *Computer & Security*, vol. 92, no. 101752, pp. 1 - 15, 2020. Available: https://doi.org/10.1016/j.cose.2020.101752

[52] G. Agrafiotis, E. Makri, I. Flionis, A. Lalas, K. Votis and D. Tzovaras, "Image-based Neural Network Models for Malware Traffic Classification using PCAP to Picture Conversion," *In the Proceedings of the 17th International Conference on Availability, Reliability and Security (ARES'22)*, 23 - 26 August, 2022, Vienna, Austria, pp. 1 - 7, 2022. Available: https://doi.org/10.1145/3538969.3544473

[53] G. Agrafiotis, E. Makri, A. Lalas, K. Votis, D. Tzovaras and N. Tsampieris, "A Deep Learning-based Malware Traffic Classifier for 5G Networks Employing Protocol-Agnostic and PCAP-to-Embeddings Techniques," *In the Proceedings of the 2023 European Interdisciplinary Cybersecurity Conference (EICC'23)*, 14 - 15 June, 2023, Stavanger, Norway, pp. 193 - 194. Available: https://doi.org/10.1145/3590777.3590807

[54] C. Park, K. Park, J. Song and J. Kin, "Distributed Learning-Based Intrusion Detection in 5G and Beyond Networks,", *In the Proceedings of the 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 6 - 9 June, 2023, Gothenburg, Sweden, pp. 490 - 495, 2023. Available: https://doi.org/10.1109/EuCNC/6GSummit58263.2023.10188312

[55] M. Alazab and S. Fong, "Feature selection for intrusion detection systems: A comparative study," *Information Systems*, vol. 103, no. 101825, 2021.

[56] M. Hussain and R. Mahmod, "A hybrid feature selection approach for intrusion detection using machine learning," *Journal of Information Security and Applications*, vol. 66, no. 103072, 2022.

[57] N. Moustafa and J. Slay, "A Hybrid Feature Selection for Network Intrusion Detection Systems: Central Points and Association Rules," In the proceedings of the 16th Australian Information Warfare and Security Conference, 30 November, - 2 December, 2015, Edith Cpwan University, Joodalup Campus, Perth, Western Australia, pp. 5 - 13, 2015. https://doi.org/10.4225/75/57a84d4fbefbb

[58] K. G. R. Narayan, R. Ganesula, T. S. Somasekhar, S. Mookherji, V. Odelu, R. Prasath and A. G. Reddy, "Attenuating majority attack class bias using hybrid deep learning based IDS framework," *Journal of Network and Computer Applications*, vol. 230, no. 103954, pp. 1 - 11, 2024. Available: https://doi.org/10.1016/j.jnca.2024.103954

[59] M. A. Khan and M. Sadiq, "An ensemble feature selection approach for intrusion detection in IoT environments," *Sensors*, vol. 23, no. 2, pp. 1050, 2023.

[60] A. A. Salama and F. A. Ghaleb, "Optimized feature selection for network intrusion detection using genetic algorithm. *Computer Networks*, vol. 203, no. 108674, 2022.

[61] W. Zhou and S. Liu, "Deep feature selection: A novel approach for intrusion detection systems," *ACM Transactions on Intelligent Systems and Technology*, vol. 12, no. 4, Article 44, 2021.

[62] M. A. Bouke and A. Abdullah, "An empirical assessment of ML models for 5G network intrusion detection: A data leakage-free approach," *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, vol. 8, no. 100590, pp. 1 - 9, 2024. Available: https://doi.org/10.1016/j.prime.2024.100590

[63] Y. Zhang, W. Li, Z. Xu and H. Zhou, "A Secured Data Sharing Framework in 5G Network for Machine Learning-Based Intrusion Detection," *IEEE Access*, vol. 8, pp. 200001-200015, 2020.

[64] M. S. Khan, B. Farzaneh, N. Shahriar, N. Saha and R. Boutaba, "Slice Secure: Impact and Detection of DoS/DDoS Attacks on 5G Network Slices," *In the Proceedings of the 2022 IEEE Future Networks World Forum (FNWF)*, pp. 639 - 642, 2022.

[65] L. U. Khan, I. Yaqoob, N. H. Tran, Z. Han and C. S. Hong, "Network slicing: Recent Advances, Taxonomy, Requirements, and Open Research Challenges," *IEEE Access*, vol. 8, pp. 36009 - 36028, 2020. https://doi.org/10.1109/ACCESS.2020.2975072

[66] S. Samarakoon, Y. Siriwardhana, P. Porambage, M. Liyanage, S. Y. Chang, J. Kim, J. Kim and M. Ylianttila, "5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network,", IEEE Dataport, 2nd December, 2022, https://doi.org/10.21227/xtep-hv36

[67] E. C. P. Neto, H. Taslimasa, S. Dadkhah, S. Iqbal, P. Xiong, T. Rahmanb, and A. A. Ghorbani, "CICIoV2024: Advancing Realistic IDS Approaches against DoS and Spoofing Attack in IoV CAN bus," Internet of Things, vol. 26, no. 101209, 2024. https://doi.org/10.1109/JIOT.2024.3522863

[68] M. Rabbani, J. Gui, F. Nejati, Z. Zhou, A. Kaniyamattam, M. Mirani, G. Piya, I. Opushnyev, R. Lu, A. A. Ghorbani. "Device Identification and Anomaly Detection in IoT Environments," IEEE Internet of Things Journal, vol. 12, no. 10, pp. 13625 - 13643, 2024. Available: https://www.unb.ca/cic/datasets/iot-diad-2024.html

[69] M. S. Akbar, Z. Hussain, M. Ikram, Q. Z. Sheng and S. C. Mukhopadhyay, "On Challenges of Sixth-Generation (6G) Wireless Networks: A Comprehensive Survey of Requirements, Applications, and Security Issues," *Journal of Network and Computer Applications*, vol. 233, no. 104040, 2025. Available: https://doi.org/10.1016/j.jnca.2024.104040

[70] P. Kamble and A. N. Shaikh, "6G Wireless Networks: Vision, Requirements, Applications and Challenges," *In the Proceedings of the 2022 5th International Conferences on Advances in Science and Technology (ICAST)*, 2 - 3 December, 2022, Mumbai, India, pp. 577 - 581, 2022. https://doi.org/10.1109/ICAST55766.2022.10039549

[71] V. A. Akpan, "Hard and soft embedded FPGA processor systems design: Design considerations and performance comparisons," *International Journal of Engineering and Technology*, 3(11): 1000 - 1020, 2013. Available: http://www.iet-journals.org/archive/2013/november_vol_3_no_11/71536713773485 26%e2%80%8f.pdf

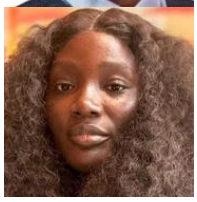
[72] V. A. Akpan, D. Chasapis and G. D. Hassapis, "FPGA Implementation of Neural Network-Based AGPC for Nonlinear F-16 Aircraft Auto-Pilot Control: Part 1 - Modeling, Synthesis, Verification and FPGA-in-the-Loop Co-Sim," *American Journal of Embedded Systems and Applications*, vol. 9, no. 1, pp. 6 - 36, 2022. Available: https://doi.org/10.11648/j.ajesa.20220901.13

[73] V. A. Akpan, D. Chasapis and G. D. Hassapis, "FPGA Implementation of Neural Network-Based AGPC for Nonlinear F-16 Aircraft Auto-Pilot Control: Part 2 - Implementation of Embedded PowerPC™440 with AGPC," *American Journal of Embedded Systems and Applications*, vol. 9, no. 2, pp. 37 - 65, 2022. Available: https://doi.org/10.11648/j.ajesa.20220902.11

Biography

Vincent Andrew Akpan obtained his B. Sc. (Hons) degree in Physics from Delta State University, Abraka, Delta State, Nigeria in 1997; a Master of technology (M. Tech) degree in Electronic Measurement & Instrumentation from The Federal University of Technology, Akure, Ondo State, Nigeria in 2003; and a Ph.D degree in Electrical & Computer Engineering from Aristotle University of Thessaloniki, Thessaloniki, Greece in 2011. He is currently a Professor with the Department of Biomedical Engineering, The Federal University of Technology (FUTA), Akure, Nigeria. His research interest is in artificial intelligence, advanced electronic instrumentation, intelligent adaptive control, automation and embedded systems engineering. He is the co-author of a book and has authored and/or co-authored more than 90 articles in refereed journals and conference proceedings. He is a reviewer of several local and international journals as well as conference proceedings. Prof. Akpan was one among two Nigerian recipients of the 2005 Bilateral Education Agreement (BEA) scholarship tenable in Greece for a P.hD programme. Prof. Akpan is a member of several local and international professional societies.

Emmanuel Chinenye Njoku obtained his B.Eng (Hons) degree in Information and Communication Engineering from the Federal University of Technology, Akure, Ondo State, Nigeria in 2024. He is currently a Product Management Intern at Guaranty Trust Holding Company, Nigeria, where he contributes to mobile banking product development. His research interests include AI applications in network security, electronic instrumentation, next-generation networks, and renewable energy systems.



Ebubechukwu Ifeoluwa Obi earned her B.Eng (Hons) degreein Information and Communication Engineering from the Federal University of Technology, Akure, Ondo State, Nigeria in 2024. With a growing passion for emerging technologies, she is currently delving into the intersection of lightweight/heavyweight blockchains and artificial intelligence. Her interests span decentralized solutions, electronic instrumentation, artificial intelligent systems, and the role of technology in driving innovation and sustainable progress.

Research Field

Vincent Andrew Akpan: artificial intelligence, electronic instrumentation, industrial networked control systems, intelligent adaptive control systems, real-time embedded systems, robotics/ automation.

Emmanuel Chinenye Njoku: artificial intelligence, electronic instrumentation, network security, next-generation networks, renewable energy systems.

Ebubechukwu Ifeoluwa Obi: artificial intelligence, electronic instrumentation, lightweight/heavyweight blockchains, decentralize and sustainable systems.