RÉPUBLIQUE ALGERIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE LE NSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ Saad Dahleb, Blida 1

Institut d’Aéronautique et des Études Spatiales IAES

Mémoire

Présentée pour obtenir le diplôme de

MASTER 02 LMD

SPÉCIALITÉ : TÉLÉCOMMUNICATION SPATIALES

An Intelligent Security Protocol for Enhanced Security in 6G Networks

Présenté par :

Boussaidi Rofeida
Messaoudi Tassadit

Encadré par :

Dr.Asmaa Iman Kaouachi
Dr.Rachid Benaissa

Promotion 2024-2025

Acknowledgement

Praise be to Allah, who taught man what he didn't know, and who, by His grace, paved for us a path toward our dream. His mercy was our companion, His protection our strength, and success the beautiful ending to a journey of effort and light.

To our beloved families, **Boussaidi** and **Messaoudi**,you were the heartbeat behind every ambition, the echo behind every prayer, and the flame when our light began to fade. To you, we offer gratitude that words could never truly express.

We extend our sincere thanks to **Dr. Asma Iman Kouachi** for embracing the idea from its inception and for her patience, guidance, and unwavering belief in our project until its completion. In addition, we thank our co-supervisor, **Dr. Rachid Benaissa**, for his support. Our sincere appreciation also goes to the esteemed Head of Department, **Dr.Sofiane Tahraoui**, for his encouraging spirit and his continuous dedication to opening doors of excellence for his students. And finally, to the honorable **members of the jury**,**Dr.Azine Houria and Dr.Krim Mohamed**, we offer our deep gratitude for your valuable time and insightful feedback, which have enriched this work and added to its significance.

To everyone who contributed, whether closely or from afar, you have our sincerest prayers and highest words of gratitude and appreciation.

الإهداء

﴿ وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ ﴾ – هود 88

الحمد لله الذي جبر قلبي كلما ضاق، ورفعني كلما ظننت أنني لن أقوى،الحمد لله على النور الذي غمرني في عزّ عتمتي، وعلى الفرج الذي جاء في الوقت الذي كدتُ أن أنطفئ فيه،الحمد لله الذي جعل في الطريق بركة، وفي الصبر ثمرة، وفي دعاء الأحبة سبيلاً للنجاة.

"حين مرّ الجميع عبر الأبواب، بنيتُ لي مدخلاً خاصاً... ونجحت".

"شكرًا لي، لأنني تعبت... جاهدت... ولم أخذل نفسي او عائلتي يومًا".

الى من ترك صلاته أكثر من مرة ليوصلني نحو حلمي، أبي الغالي يا من خبّأتَ تعبك خلف ابتسامة كي لا أضعف، ودفعتَ عني الحياة بصبرك وصمتك،كنتَ ظهري حين غابت السندات، وظلك كان أوسع من كل ظل.أحبك بامتنانٍ لا يُقال... بل يُرفع في الدعاء كل ليلة.

الى قرة عيني أمي،يا نبع الحنان الذي لا يجف، ويا قلبًا لم يخذلني يومًا،كنتِ لي وطنًا حين تاهت خطواتي، وستري حين خفَتَ صوتي،رأيتِ تعبي دون أن أتكلم، ومسحتِ همّي بكفٍّ من دعاء،أمي... رضاكِ هو الدرب، وحبك هو النور، ووجودك وحده نِعمة لا تُشبه شيئًا.

يا من كنتَ السند الأول، والظل الذي لا يغيب أخي وليد ،شكراً لأنك كنتَ تؤمن بي حتى عندما لم أكن أؤمن بنفسي.

إلى صاحب الروح رفيق المشاوير، وصاحب القلب الطيب، اخي حبيبي فارس كنتَ ترافقني كل يوم دون كلل، وكنتَ تقف بجانبي دون أن تنتظر شكرًا،وجودك اليومي كان أمانًا، وكان لُطف الله في أكثر لحظات ضعفي.

إلى صغيري محمد رفيق،كنتَ الفرح حين ثقل كل شيء، والحنان حين غابت الكلمات، ... كل شيء منك كان بلسمًا يريح نفسي.

إلى جدي (رحمه الله)،إلى من غاب جسده وبقيت روحه تحرسني ،رحمك الله كما كنتَ نوراً في حياتنا، وجعل قبرك روضةً من رياض الجنة.

إلى ابنة عمي، أختي الثانية مريم بوسعيدي ، انت نصفي الثاني و روح قلبي الهادئة احبك.

إلى تسعديت، زميلة التعب لم اكن لأكمل هذا وحدي لولاك الحمد لله اليوم نتوج تعبنا او. نأخذ بحياتنا الى نجاحات أخرى.

إلى كل افراد عائلة بوسعيدي وبوتريك، شكرًا لدعائكم و دعمكم الكبير و محبتكم لي. إلى كل من شجعني، أو آمن بي

... أهديكم هذا النجاح.

ROFI...

الإهداء

الحمد لله الذي بنعمته تتم الصالحات، وبفضله يُبلِغ الساعي مُراده، وبتوفيقه نكتب اليوم نهاية فصلٍ من مسيرة كفاح وعلم.

«أنا التي لم تنتظر أن يُقال لها: "أنتِ لها"…. بل قلتها لنفسي ومضيت».
«أنا الاستثناء الذي لا يشبه أحد، وكل نجاح يشبهني وحدي».

في لحظة الامتنان هذه، حين يصبح الحلم واقعًا، أُهدي تخرجي إلى من غادر الحياة وبقي حيًّا في قلبي… إلى والدي، أُهدي هذا التخرج لروحك الطاهرة، وأسأل الله أن يجعل هذا الإنجاز نورًا في قبرك وصدقةً جاريةً لك.

إلى أمي الحبيبة، نجمة عمري، وسيدتي الأولى في عالمي…شكرًا على كل لحظة خوف، وكل دعاء صادق، وكل تعب لا يُقَدَّر.لا أحد يشبهك، ولا أحد يفهمني كما تفعلين.لو كتبتُ عمري كلّه "شكرًا"، لما وفيت لحظةً من عطائك.

إلى أخواتي العزيزات: كنتنّ أماني ودفئي في كل لحظة.شكرًا لوجودكنّ الذي كان نعمةً لا تُنسى، وسندًا لا يزول.

إلى أخي الوحيد **محمد**، لم تكن فقط أخي، بل كنت الأب والسند والصديق.كل نجاحي هو ثمرة دعمك، فشكرًا لأنك عزوتي التي لا تنكسر.

إلى أعز أصدقائي: حياة، أماياس، وصحبي لكم مني كل الامتنان، وقلبي لكم بيت لا يُغلق.

وإلى زميلتي العزيزة **رفيدة**، شريكة التعب والخطى الثقيلة…
فخورة بكِ، وسعيدة أننا وصلنا معًا، كتفًا إلى كتف.

Tassadit

Abstract

Sixth-generation (6G) networks represent a major leap in modern telecommunications, delivering ultra-high speeds, minimal latency, and the ability to connect a massive number of devices. However, these technological advancements also introduce significant cybersecurity challenges, particularly Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) attacks, which threaten the stability, availability, and reliability of 6G infrastructures. This dissertation proposes an intelligent security protocol based on Large Language Models (LLMs), leveraging their advanced capabilities in data analysis and anomaly detection to safeguard network traffic. The PROGRES platform was adopted as a real-world use case to simulate DoS/DDoS attack scenarios within a realistic environment, using a modified version of the CICIDS2017 dataset tailored to the 6G context. The study evaluates both traditional machine learning (ML) models such as Random Forest (RF), K-Nearest Neighbors (KNN), Logistic Regression (LR), Decision Tree (DT), and Support Vector Machine (SVM)and advanced LLMs, including ELECTRA, DistilGPT-2, and DistilBERT. All models were assessed using a comprehensive set of metrics: accuracy, precision, recall, F1-score, throughput, latency, resource utilization, and energy consumption. Results revealed that DistilBERT outperformed other models in terms of detection accuracy and reliability, although it required higher computational resources. Conversely, traditional ML models showed greater efficiency in speed and resource usage but at the cost of lower accuracy. These findings underscore the potential of LLMs to enhance security in 6G wireless communication systems, while also highlighting the need to balance performance with computational efficiency for practical deployment.

Keywords: 6G Networks, Security, DoS/DDoS Attacks, System Availability, LLMs, DistilBERT.

Résumé

Les réseaux de sixième génération (6G) représentent un bond majeur dans les télécommunications modernes, offrant des vitesses ultra-rapides, une latence minimale et la capacité de connecter un nombre massif dappareils. Toutefois, ces avancées technologiques introduisent également dimportants défis en matière de cybersécurité, notamment les attaques par déni de service (DoS) et par déni de service distribué (DDoS), qui menacent la stabilité, la disponibilité et la fiabilité des infrastructures 6G. Cette dissertation propose un protocole de sécurité intelligent basé sur les modèles de langage de grande taille (LLM), tirant parti de leurs capacités avancées en analyse de données et détection danomalies pour sécuriser le trafic réseau. La plateforme PROGRES a été adoptée comme cas dutilisation réel pour simuler des scénarios dattaques DoS/DDoS dans un environnement réaliste, en utilisant une version modifiée du jeu de données CICIDS2017, adaptée au contexte 6G. Létude évalue à la fois les modèles traditionnels dapprentissage automatique (ML), tels que Random Forest (RF), K-Nearest Neighbors (KNN), la régression logistique (LR), larbre de décision (DT) et les machines à vecteurs de support (SVM), ainsi que les LLM avancés, notamment ELECTRA, DistilGPT-2 et DistilBERT. Tous les modèles ont été évalués selon un ensemble complet de métriques : exactitude, précision, rappel, score F1, débit, latence, utilisation des ressources et consommation énergétique. Les résultats ont révélé que DistilBERT surpassait les autres modèles en termes de précision de détection et de fiabilité, bien quil exige davantage de ressources informatiques. En revanche, les modèles ML traditionnels ont montré une meilleure efficacité en matière de vitesse et de consommation de ressources, mais au détriment de la précision. Ces résultats soulignent le potentiel des LLM pour renforcer la sécurité des systèmes de communication sans fil 6G, tout en mettant en lumière la nécessité déquilibrer les performances et lefficacité computationnelle pour un déploiement pratique.

Mots clés: Réseaux 6G, Sécurité, Attaques DoS/DDoS, Disponibilité du système, LLMs, DistilBERT.

ملخص:

تمثل شبكات الجيل السادس (6G) قفزة هائلة في عالم الاتصالات الحديثة، إذ توفّر سرعات فائقة للغاية، وزمن استجابة شبه معدوم، وقدرة على ربط عدد هائل من الأجهزة. ومع ذلك، فإن هذه التطورات التكنولوجية تفتح الباب أمام تحديات كبيرة في مجال الأمن السيبراني، لا سيما هجمات حجب الخدمة (DoS)والهجمات الموزعة لحجب الخدمة(DDoS) ، التي تهدد استقرار البنية التحتية لشبكات (6G) وتوافرها وموثوقيتها. تقترح هذه الأطروحة بروتوكولًا أمنياً ذكيًا يعتمد على نماذج اللغة الكبيرة(LLMs) ، مستفيدًا من قدراتها المتقدمة في تحليل البيانات وكشف الشذوذ لحماية حركة البيانات على الشبكة. وقد تم اعتماد منصة PROGRES كحالة استخدام واقعية لمحاكاة سيناريوهات هجمات DoS/DDoS ضمن بيئة واقعية، باستخدام نسخة معدّلة من مجموعة بيانات CICIDS2017 تم تكييفها لتلائم سياق الجيل السادس. تقيّم الدراسة كلاً من نماذج التعلم الآلي التقليدية مثل الغابة العشوائية(Random Forest) ، وأقرب الجيران(KNN) ، والانحدار اللوجستي، وشجرة القرار، وآلة الدعم الناقل (SVM) بالإضافة إلى نماذج اللغة المتقدمة مثل ELECTRA و DistilGPT-2و .DistilBERTتم تقييم أداء جميع النماذج باستخدام مجموعة شاملة من المعايير: الدقة، ومعامل الاسترجاع، ومعامل التكرار، ومعاملF1 ، ومعدل نقل البيانات، والكمون، واستهلاك الموارد، والطاقة. وقد أظهرت النتائج أن نموذج DistilBERT تفوق على النماذج الأخرى من حيث دقة الكشف والموثوقية، رغم أنه يتطلب موارد حسابية أعلى. بالمقابل، أظهرت النماذج التقليدية كفاءة أعلى من حيث السرعة واستهلاك الموارد، ولكن على حساب الدقة. وتؤكد هذه النتائج على إمكانات نماذج LLM في تعزيز أمان شبكات الاتصالات اللاسلكية من الجيل السادس، مع التأكيد على ضرورة تحقيق توازن بين الأداء والكفاءة الحسابية لضمان التطبيق العملي.

الكلمات المفتاحية: شبكات الجيل السادس (6G)، الأمان، هجماتDoS/DDoS ، توفر النظام، نماذج اللغة الكبيرة(LLMs) ، DistilBERT.

Contents

List of Figures

List of Tables

Acronyms

- 1G: The First Generation
- 2G: The Second Generation
- 3G: The Third Generation
- 4G: The Fourth Generation
- 5G: The Fifth Generation
- 6G: The Sixth Generation
- 3D: Three dimensional
- AES: Advanced Encryption Standard.
- AI: Artificial Intelligence
- AN: Access Network
- AR: Augmented Reality
- APE: Absolute Positional Encoding
- BMI: Body Mass Index
- BS: Base Station
- CAVs: Connected Autonomous Vehicles
- CIA: Confidentiality Integrity Availability
- CICIDS: Canadian Institute for Cybersecurity Intrusion Detection System
- CN: Core Network
- CPU: Control Processing Unit
- DDOS: Distributed Denial Of Service

- DL: Deep Learning
- DoS: Denial of Service
- DrLLM: Detection and Response using Large Language Models
- DTs: Decision Trees
- eMBB: Enhanced Mobile Broadband
- eSIM: Embedded Subscriber Identity Module
- ESAR: Enhanced Smart Access and Routing
- eNB: Evolved Node B
- EPC: Evolved Packet Core
- FBSA: Features Based Security Architecture
- FDMA: Frequency Division Multiple Access
- FiFo: First in First out
- FPGA: Field Programmable Gate Array
- GPS: Global Positioning System
- GPT: Generative Pretrained Transformer
- GPUs: Graphics Processing Units
- HTTP: Hyper Text Transfer Protocol
- ICMP: Internet Control Message Protocol
- IDS: Intrusion Detection System
- IoE: Internet of Everything
- IOST: Internet Of Space Things
- IoTs: Internet Of Things
- IP: Internet Protocol
- IRS: Intelligent Reflective Surface
- KNN: K-Nearest Neighbors
- LLaMA: Large Language Model Meta AI
- LLMs: Large Language Models

- LTE: Long Term Evolution
- M2M: Machine to Machine
- MESRS: Ministry of Higher Education and Scientific Research
- MIMO: multiple input multiple output
- MiTM: Man in The Middle
- ML: Machine learning
- MMS: Multimedia Messaging Service
- mMTC: Massive Machine Type Communication
- mmwave: Millimeter Wave
- NLoS: Non Line of Sight
- NOMA: Non Orthogonal Multiple Access
- NTN: Nom Terrestrial Networks
- PaLM: Pathways Language Model
- PKI: Public Key Infrastructure
- QC: Quantum Communication
- QR: Quick Response
- RAM: Random Access Memory
- RAN: Radio Access Network
- RSA: Rivest Shamin Adleman
- RPE: Regular Pulse Excitation
- SDN: Software Defined Networking
- SMS: Short Message Service
- SOPS: Standard Operating Procedures
- SVM: Support Vector Machine
- SYN: Synchronize
- TCP: Transport Control Protocol
- THz: Terahertz

- UAVs: Unmanned Aerial Vehicles
- UDP: User Datagram Protocol
- URLLC: Ultra-Reliable Low-Latency Communication
- UE: user equipment
- VoLTE: Voice over LTE
- VR: Virtual Reality
- XR: Extended Reality

Introduction

Each new generation of wireless technology has significantly transformed how we interact and communicate. The transition from fifth generation (5G) to sixth generation (6G) represents a substantial leap in redefining our interaction with the digital world, commonly referred to as cyberspace, and is closely associated with digital technologies and interconnected systems enabled by the Internet. These advances allow people to interact, work, learn, play, and share information across digital devices and networks. 6G is expected to become a vital connectivity force, driving digital transformation and expanding its applications beyond entertainment [1]. These applications and unprecedented communications also expand the scope of network attacks, making security a major concern. DoS and DDoS attacks pose a serious threat to the availability and reliability of the service. In these circumstances, current security measures must keep pace with 6G technologies [2].

In light of these new threats, the way forward for developing intelligent, adaptive, explainable security solutions lies with artificial intelligence (AI) integration, in particular, LLMs. LLMs can massively analyze large and heterogeneous data streams to detect stealthy attack patterns such as DoS and DDoS in real time. They furthermore carry the ability to explain reasoning transparently and adjust to emerging threats, making them a vital stone for next-generation 6G security frameworks [3].

6G Networks, with their ultra-fast speeds and massive connectivity, are increasingly exposed to cyber threatsespecially DoS and DDoS attacks that can disrupt availability and degrade service quality [4]. Traditional security methods lack the intelligence and adaptability to detect and respond to these attacks in real-time due to the complex and dynamic nature of 6G environments. To overcome these limitations, LLMs offer a promising solution by enabling real-time traffic analysis, anomaly detection, and intelligent mitigation of such threats [5]. This approach enhances the security, resilience, and reliability of 6G networks.

Our contribution is focused on detecting DoS/DDoS attacks in 6G using LLMs. This work presents a valuable contribution to the field of 6G network security through the following:

1. Proposing an intelligent security protocol based on LLMs for real-time detection of DoS/DDoS attacks.
2. Designing a realistic simulation scenario using the PROGRES platform to evaluate the proposed solution in an academic environment.
3. Creating a customized dataset that mimics 6G network behavior, based on CICIDS 2017.
4. Comparing the performance of LLMs with traditional ML models in terms of accuracy, speed, and resource efficiency.

This dissertation is structured as follows.

- Chapter 1: this chapter provides essential background information on 6G networks, followed by the fundamentals of security in 6G networks.
- Chapter 2: presents related work on the use of LLMs in network security and introduces the proposed methodology.
- Chapter 3: this chapter describes the simulation scenario, implementation, and evaluation of the proposed solution.

Chapter 1

Background on Security in 6G

Contents

1.1 Introduction

The evolution of wireless communications has witnessed rapid progress from the first generation (1G) to the upcoming 6G networks. Each generation has brought new capabilities and challenges, particularly regarding security. 6G promises higher speeds, lower latency, and smarter applications, but it also faces new types of cyber threats that need to be addressed [6]. Security is a critical aspect of 6G, ensuring the protection of data and services from attacks such as DoS, which can disrupt network availability [7].

This chapter provides a background on 6G technology, its architecture, and the importance of security in this new era, focusing on the challenges and threats related to DoS attacks.

1.2 Part 1: Background on 6G

The field of wireless communications has seen new developments in communication networks. The focus of each network changed according to the requirements at that time.

Integration of AI into 6G technology will revolutionize communication by enabling smarter networks to adapt and optimize without human intervention. However, these networks also present security challenges [1].

Cyberattacks and data security are becoming increasingly challenging in 6G, making AI-powered security solutions essential to protect networks from advanced threats and continuously monitor for potential anomalies and attacks. AI provides strategies to deal with the size and complexity of networks and the security challenges expected in 6G. So, while we faced the same security challenges in previous generations, AI was not as advanced and powerful as it is now [1].

This part explores the evolution from 1G to 6G, highlighting the need for 6G and its main features. It covers 6G architecture, technologies, applications, and challenges.

1.2.1 The Evolution of Wireless Communication Networks: From 1G to 6G:

Wireless networks have evolved significantly across generations, each introducing major technological improvements. According to [8, 9], the main developments from 1G to 6G are as follows:

1. The 1G focused on voice communications, using Frequency Division Multiplexing (FDMA) technology at 2 kbps. Therefore, the mobile service was very poor.

2. The second generation (2G) provided Short Message Service(SMS) and Multimedia Messaging Service(MMS) services at frequencies between 30 and 200 kHz, along with voice communications, expanding the capabilities of mobile networks.
3. With the advent of the third generation (3G), mobile data services became a key feature, enabling mobile internet access due to increased speeds of up to 2 Mbps with a frequency range of 15 to 20 MHz.
4. The fourth generation (4G) represented a qualitative leap, facilitating large-scale data consumption and introducing the first mobile broadband services via Long Term Evolution (LTE) networks.
5. The 5G has enhanced connectivity by integrating the Internet of Things (IoT) with industrial automation systems, driving innovation across various industries.
6. Following the significant development of the 5G, studies have begun to launch the 6G as its successor around 2030. 6G is human-centered, so security, confidentiality, and privacy must be higher in the implementation of 6G compared to other generations.

Fig.1.1 represents the evolution of cellular mobile communications.

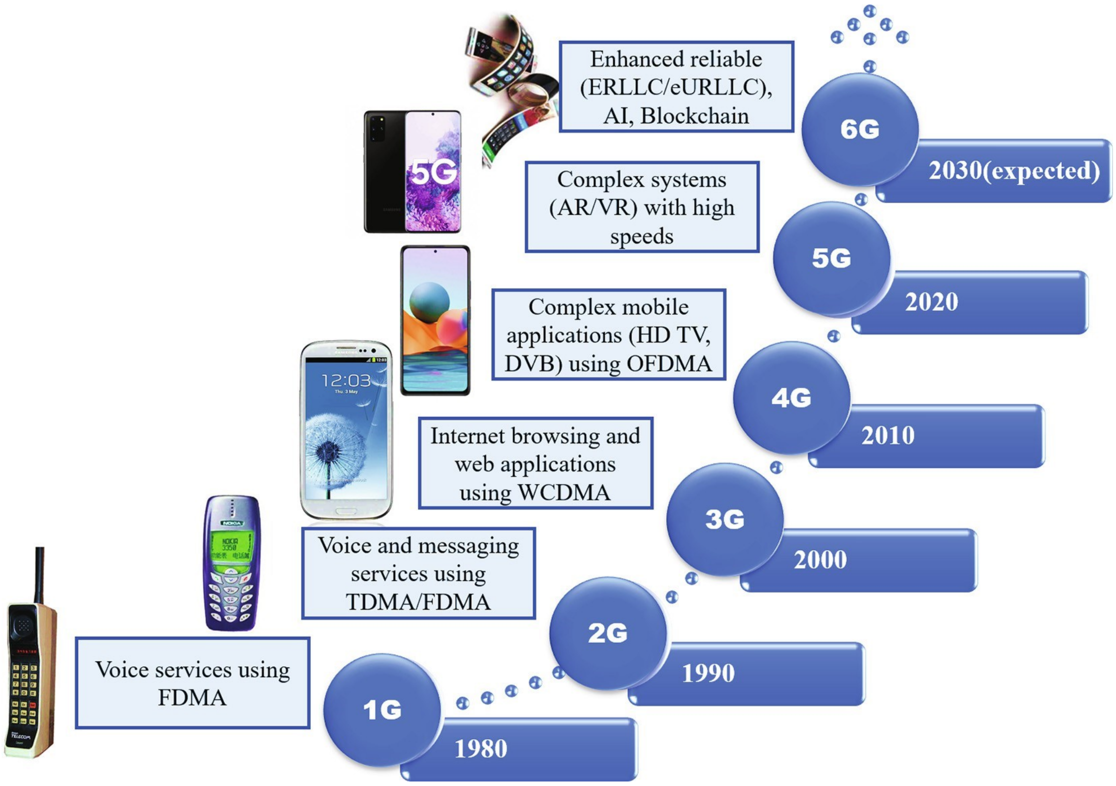

Figure 1.1: The evolution of cellular mobile communications .

1.2.2 The Need for 6G: Progress Beyond 5G Connectivity

According to [1], the growth of IoT and industry has significantly expanded the capacity of the network. 5G technology supports diverse and new applications such as deep-sea and space tourism, holographic communication, and the Internet of Everything (IoE) through high-frequency millimeter wave (mmwave) functions and speed-sensitive Enhanced Mobile Broadband (eMBB) services. eMBB services refer to one of the primary use cases of 5G technology that significantly increases data rates, capacity, and network coverage, especially in densely populated areas. It supports applications like high-definition video streaming, virtual and augmented reality experiences, and faster internet access on mobile devices [10].

However, the 5G network faces challenges in providing smart services due to its lack of some of the features that were taken and distinguished by the 6G technology, which is considered an evolutionary step and a qualitative shift, as it provides amazing features such as providing unprecedented speed, reliability, and safety.

The 6G technology has high-frequency bands, including the terahertz(THZ)frequency, for a faster data transfer rate and a wider bandwidth.

1.2.3 6G Definition

The 6G network is the latest technology in the field of cellular networks. It is a project currently under research and development and will be marketed in 2030.

The 6G vision focuses on longer battery life, reliability, and human-centric services, with potential applications like holographic calls, flying networks, and the tactile internet. Key technologies include THz communications, and integration of AI also improves network systems to be more robust and perform complex tasks such as optimization, classification, and clustering, which are used in multiple fields and industries. It can also help in network management and predict problems before they occur. This means better quality connections, faster responses, and improved security [9].

1.2.4 Comparison Between 5G and 6G

Based on the studies in [11]and [12], the key differences between 5G and 6G are summarized in Table 1.1.

Generation	5G	6G
Year	2020	2030
Data rate	10Gbps	1Tbps
Latency	1-5 ms	< 1ms
AI integration	Limited	Fully
Maximum Frequency	300 GHz	Up to 1 THz
Reliability	99.9%	99.999999%
Maximum spectral efficiency	30 bps/Hz	100 bps/Hz
Satellite integration	Limited	Fully
Density of devices	10^6 [devices/km^2]	10^7 [devices/km^2]
Maximum Bandwidth	1GHz	100Ghz

Table 1.1: Comparison between 5G and 6G technologies.

1.2.5 6G Architecture

According to [13], a simple taxonomy of the 6G architecture, based on four layers, is illustrated in Fig.1.2.

1. Intelligent Sensing Layer: This is the foundational layer responsible for collecting real-world data using IoT devices, sensors, and monitoring systems. It gathers environmental information, detects statuses, and measures various parameters for AI-driven automation.

2. Intelligent Edge Layer: This layer processes and analyzes data closer to its source to reduce latency and improve efficiency. It includes tasks like data filtering, knowledge discovery, feature extraction, and storage at the edge, ensuring real-time responsiveness.

3. Intelligent Control Layer: Acts as the hub for decision-making on the network, optimizing resources, and managing network functions. It includes parameter optimization, resource allocation, task scheduling, and policy learning, often using Software-Defined Networking (SDN) and cloud-based management.

4. Smart Application Layer: the top layer that delivers AI-driven services and applications for smart cities, intelligent transportation, smart industries, and healthcare. It focuses on automation, service provisioning, distributed services, and performance evaluation to enhance user experiences.

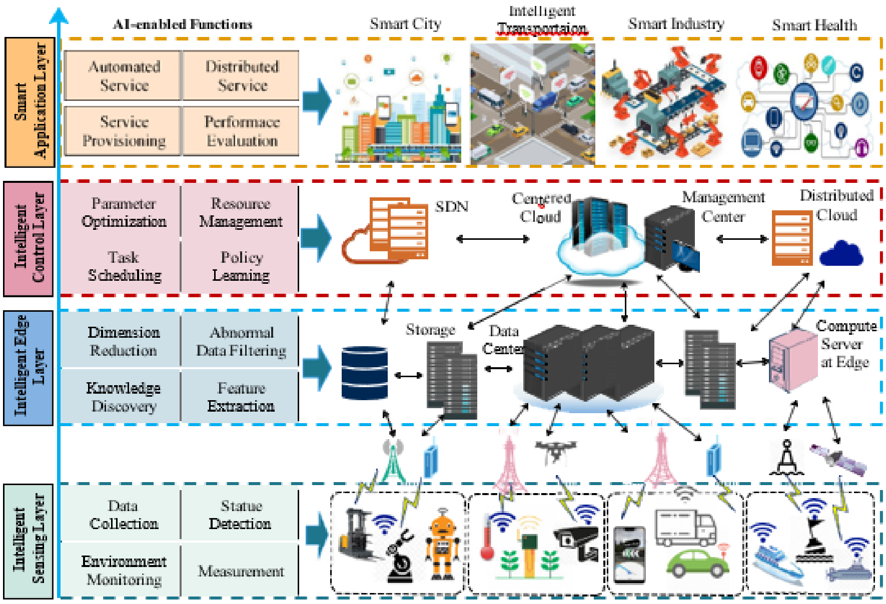

Figure 1.2: Intelligent 6G Architecture.

This dissertation focuses on the application layer within the architecture of 6G networks.

1.2.6 6G Applications

As shown in Fig.1.3 and supported by recent studies [14, 15, 16, 8], 6G is expected to support a wide range of innovative applications across different sectors, enabled by its advanced capabilities such as high speed, low latency, and intelligent connectivity. Among these applications:

1. Extended Reality(XR): 5G has improved Virtual Reality /Augmented Reality (VR/AR) experiences with high bandwidth and low latency, but issues like latency and file compression remain. 6G will address these by enabling real-time transfer of large, lossless files and enhancing dynamic VR/AR through advanced sensors.XR in 6G will combine ultra-reliable low-latency communication (URLLC) and eMBB, though security and privacy challenges persist. A new multiple-access approach has been proposed to support mass connectivity for multisensory XR applications in 6G networks [8].

2. Healthcare and remote surgeries: Medical health systems will also benefit from 6G wireless networks due to the innovations it aspires to such as AR, VR, holographic telepresence, and AI, which will help in building smart healthcare systems where even remote surgery becomes possible by using 6G communications [14]. 6G is designed to meet the highest and lowest access times required simultaneously, with 1 terahertz per second data speed and 99.99999% reliability, enabling the quick and reliable transmission of large amounts of medical data, all of which makes remote surgery possible and highly successful [15].

3. Holographic communication: As AR/VR apps evolve, we will soon realize that the virtual world does not provide access to quite as many elements of reality as we thought. The current COVID-19 pandemic has made teleconferencing a superior option to in-person meetings. Bandwidth and computations are required for this project to display an object or person in true VR. In other words, a visual in a virtual meeting could be a three-dimensional(3D), real-time simulation showing the audiovisual effect of a person or object [15].

4. Haptic communication: Haptic communication is a method that allows you to communicate using the sense of touch. Also, it is all about sensory perception and providing sensory feedback to some extent. Haptic systems implementations can take place due to the various features of 6G communication systems and networks like Massive Machine Type Communication (mMTC)[16], where users sitting far away will be able to enjoy haptic experiences through real-time interactive systems [15].

5. The IoE: it is a technology in which computing and sensor elements, data, processes, things, and people all become connected over the internet to facilitate seamless integration and autonomous coordination of devices of numerous types and forms.from standardization to commercialization of IoE. The 6G system will assist everything that connects. IoE can be used for smart society applications like smart cars, smart health, and smart industry [16].

6. Ultra-Smart Cities: A futuristic smart city scenario would require a 1 Tbps data rate,3D connectivity, 1cm localization, and 99.99% reliability, for

autonomous vehicles, e-health, or smart industry. Self-driving cars will interact with other vehicles and roadside sensors to synchronize with speeds as high as 1200 km/h. In another case, the drones would be required to follow the cars and either act as information relays or hover ground stations for cross-communication [15].

7. Industrial Automation: Industrial automation led by industrial development in the 6G aims to remove the human factor from manufacturing processes to reduce the potential damage caused by humans, which requires reliable machines to replace humans with extremely low latency and high reliability, which ensures fast responses with all system modules[8].

8. Internet of Space Things(IoST): The 6G will expand IoT into space by addressing the limitations it imposes on the flexibility and coverage of the infrastructure, especially in remote areas such as the North and South Poles. For flexible communication between remote areas and satellites, 6G mobile technology can be used because it increases the data transfer rate by up to 100 gigabits per second and has high bandwidth coverage [8].

9. Communication between brain and machine: The 6G technology will support brain-machine interfaces (BMI), which analyze brain signals transmitted to digital devices and interpret them into various procedures or commands, with high data transfer rates and reliability. The health sector is likely to benefit more from brain-machine interface systems, such as continuous monitoring of epilepsy patients, as seizures can cause permanent weakness if data transmission from brain cells and doctors is interrupted [8].

10. Autonomous Vehicles and Robotics: 6G will advance Connected Autonomous Vehicles (CAVs). By providing higher spectrum with shorter wavelengths, the THz band, a frequency between 100 GHz and 1 THz, includes remote driving assistance, raw sensor data perception, mobile edge intelligence, positioning, and 3D imaging. It allows for large antenna arrays to be installed in mobile devices and base stations because it has a short wavelength to achieve lower positioning error compared to the Global Positioning System (GPS) [8].

11. Drones: Drones help in many applications, such as transportation, parcel delivery, surveillance, and media production, so they require safety and reliability to ensure their safe control and operation. 6G relies on artificial intelligence to address various problems, such as communication problems in the sky, which is the biggest challenge for drones [8].

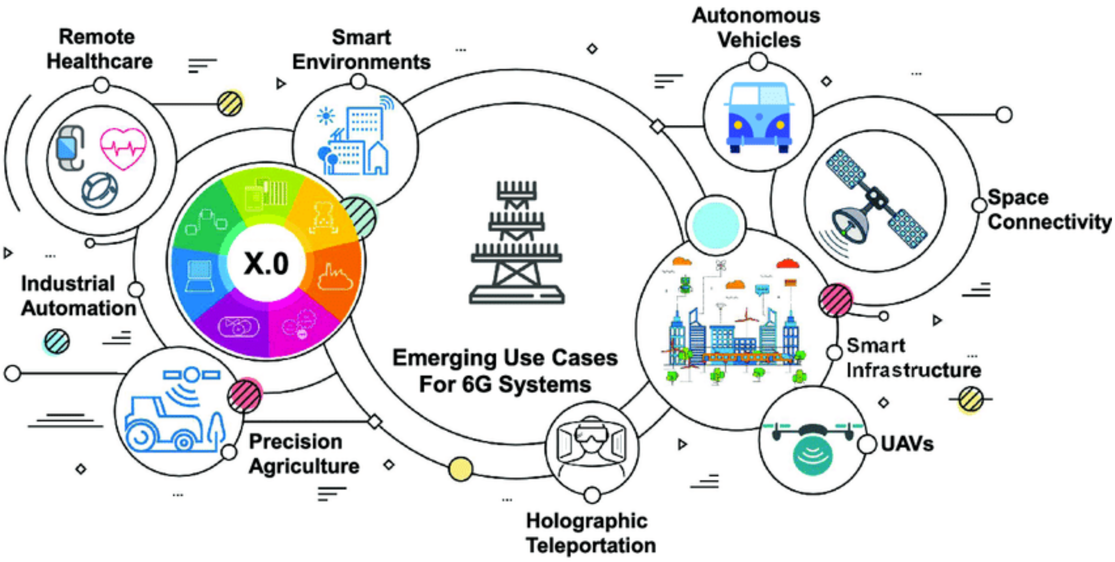

Figure 1.3: 6G Applications.

1.2.7 6G Technologies

As illustrated in Fig.1.4 and according to [8], 6G technologies are designed to enable intelligent, high-performance, and adaptive communication systems. Among these enabling technologies, we can cite the following:

1. AI: AI plays an important role in the advancement of 6G network architecture due to its features and applications in various fields such as architecture, computing, and storage. Also, intelligent analysis techniques using AI, such as descriptive analysis, predictive analytics, and diagnostic analytics, can be used to improve the security and efficiency of the network.

2. ML: ML is the most important self-learning and decision-making technology, it is a computational system that enables machines to learn from data without programming. This helps improve layer algorithms and increase data privacy and security by predicting threats. Deep learning also helps solve computational problems quickly based on the artificial neural network method, which contributes to solving optimization problems, reducing computational delay, and increasing energy efficiency.

3. The Thz Communication: it's operating between mmWave and infrared, offers high data rates and wide bandwidth, making it crucial for 6G. It supports applications like indoor networks and space communication, and can meet 6G's massive capacity demands with ultra-dense networks, high spectral efficiency, and enhanced security through narrow beams and reduced interference.

4. Intelligent Reflective Surfaces (IRS): IRS enhances 6G communication by reflecting electromagnetic waves in desired directions using low-cost, low-power passive elements. They improve signal quality in challenging environments like deep-fade or non-line-of-sight (NLOS) areas without altering existing infrastructure. IRS can be combined with technologies like deep learning and massive Multi Input Multi Output (MIMO) to boost spectral efficiency, beamforming, and multi-user communication, making it a key enabler for 6G systems.

5. Massive MIMO: Massive MIMO technology greatly enhances spectral efficiency and data transfer rate by using beamforming, allowing multiple users to share the same bandwidth without interference. MIMO technology was used in 4G with 8 antennas. The number of antennas increased to 256-1024 antennas in 5G. The 6G also uses more than 10,000 antenna elements with very narrow beams, which enhances capacity, reduces signal loss, and improves high-frequency transmission.

6. Quantum Communication (QC): QC in 6G offers unparalleled computational power for processing large, complex data sets quickly. It enhances communication security and efficiency, enabling advanced algorithms to operate faster, meeting 6G's demands for speed and reliability.

7. UAV/Satellite Communication: This technology integrates unmanned aerial vehicles (UAVs) and satellites with ground networks to provide wide-area coverage and reliable connectivity, especially in remote or hard-to-reach places [17].

8. Beamforming: Is a signal processing technique that directs wireless signals toward specific devices rather than broadcasting in all directions. This improves signal strength and reduces interference, enabling more efficient and reliable communication[18].

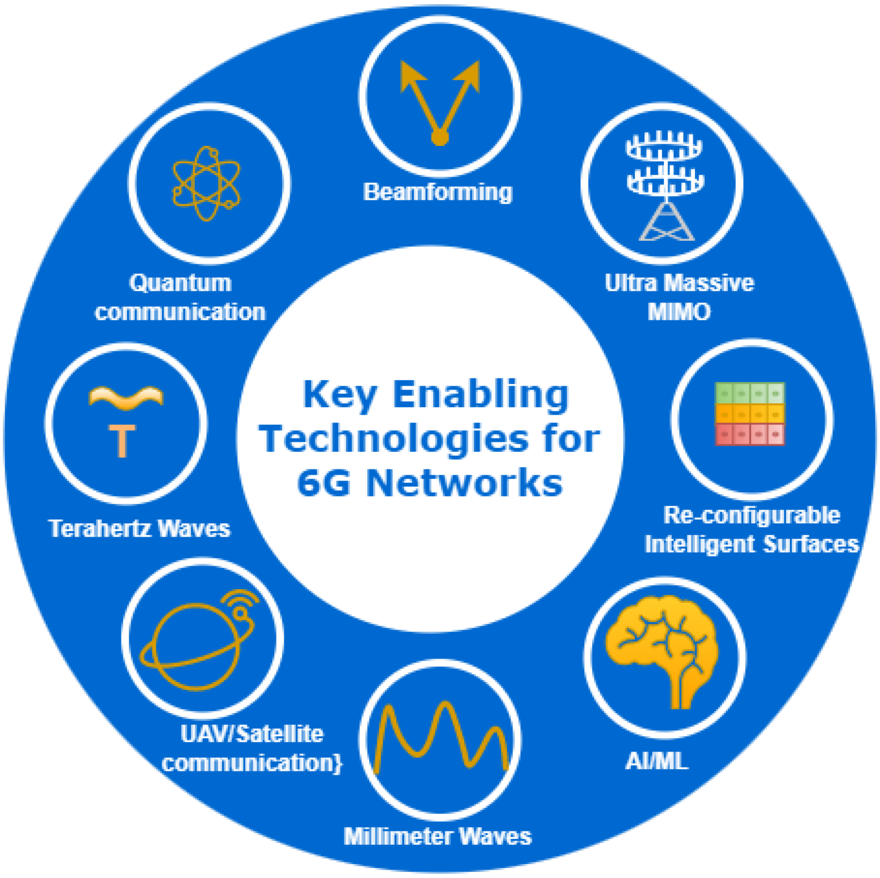

Figure 1.4: 6G Technologies.

In this dissertation, we will study AI and ML technology.

1.2.8 6G Challenges

According to [1, 8], this section examines the challenges in 6G arising from the integration of AI with future networks and proposes potential solutions:

1. Cost and efficiency: Organizations can rely on open-source software, edge-to-cloud collaboration, and leverage their existing infrastructure to reduce costs.

2. Complexity and integration challenges: Implementing complex AI models on edge devices is difficult due to limited processing power, logistical challenges, hardware, and software variability. The open-source community has developed communication protocols that address all of these issues. Examples of synaptic chips used in wireless communication technologies.

3. Data Governance: Researchers are utilizing Huffman coding and blockchain-based frameworks to enhance data quality, integrity, and storage efficiency while ensuring decentralized data management.

4. security: Integrating AI into 6G cloud environments poses various security risks, so AI/ML technologies improve resilience against attacks by introducing attack-like samples into training data. The security and integrity of AI-based systems can be improved by maintaining distributed ledgers in blockchain that make digital assets immutable and transparent.

5. Latency: To reduce latency, AI systems use edge caching to keep data closer to the edge of the network, and edge computing to reduce processing latency.

6. Scalability and Power Consumption: The performance and reliability of an AI system can be improved by distributed computing, integrating energy-efficient technology, and edge computing devices.

7. Transparency: Despite significant advances in AI-integrated IoT networks that improve infrastructure and address privacy and security concerns, the lack of explainability in AI decisions remains a challenge. Explainable AI models can help bridge this gap by providing insights into decision-making processes. Additionally, blockchain technology is being integrated into public service delivery, such as 6G networks, to enhance transparency, trust, and security, further supporting efforts to manage transparency and trust effectively.

8. Applications and Performance: 6G Technology will provide many advantages such as near-zero latency, long-range capability, and low power consumption. These advantages will be used in many applications, such as extreme reality, and will also help in the development of many fields, such as medicine, defense, and space. However, this development leads to the emergence of new challenges for this technology.

9. Integrating Systems: Seamless integration of AI, ML, and smart devices into 6G infrastructure is crucial. Ensuring flawless operation in real-time applications poses a significant challenge.

10. New Core Network Architecture: 6G will integrate terrestrial wireless mobile communications, satellite mobile communications, and short-range direct communications. The 6G architecture consists of a 3D core network that integrates terrestrial systems and satellites. It will therefore provide comprehensive global coverage but is subject to challenges, including smart mobility management, efficient use of network resources, and protocol interoperability. Solving these problems will provide high-speed connectivity on land, at sea, in the air, and in space.

11. Enabling satellites in 6G: 6G will use satellites for reliable, low-latency internet but faces challenges like Doppler shift, long transmission delays, path loss, and complex network integration. Efficient inter-satellite routing and seamless switching between satellite and terrestrial networks are also needed to ensure smooth communication. Addressing these issues is crucial for effective satellite-based 6G connectivity.
12. Peak Data Rates in 6G: 6G will use extremely high frequency bands to achieve high capacity, but these bands suffer from large path losses and short communication ranges. This requires dense networks with many access points, leading to economic challenges and frequent handovers for mobile users.
13. The Thz Wireless Communication in 6G: Thz communications are a key technology for 6G applications but are subject to several technical challenges, such as: terahertz antenna design, low-noise design, and accurate channel modeling and estimation to account for high-frequency selective path loss. Directional networks for terahertz wireless communications are also effective in enhancing range and reducing interference.

Fig.1.5 illustrates the challenges associated with 6G networks.

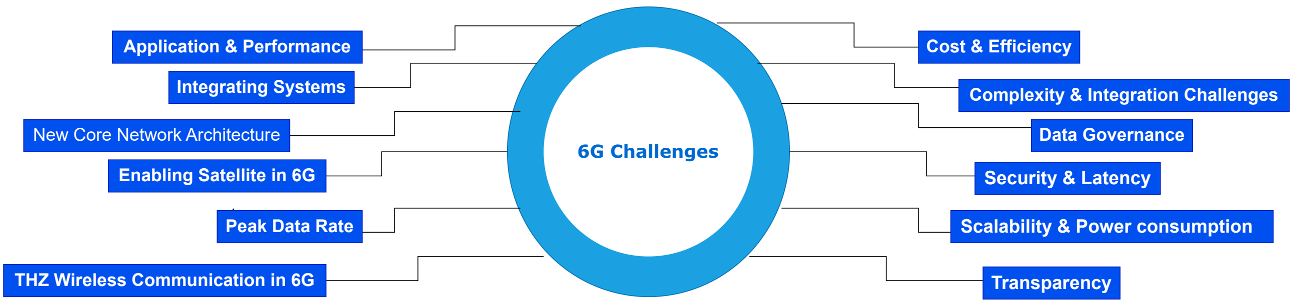

Figure 1.5: 6G Challenges

In this dissertation, we will address security challenges in 6G.

1.3 Part 2: Background on Security in 6G

6G technology will revolutionize mobile communications, providing global coverage by enabling high transmission rates, wider bandwidth, and ultra-low latency, fostering a fully interconnected civilization between humans and machines. The 6G communications environment will generate many fundamental security threats related to confidentiality, integrity, and availability (CIA). It will be easily traceable and vulnerable to hacking, such as impersonation, eavesdropping, DDoS attacks, Man in The Middle (MitM), and repudiation and replay attacks [19].

This part addresses the importance of security in 6G, presenting key threats, mitigation strategies, and AI-driven solutions. It also focuses on types of attacks (including DoS/DDoS).

1.3.1 Top 10 Reasons Why Security is Essential in 6G

AI is a key driver of change in 6G networks, as AI-powered security solutions will play a crucial role in protecting networks from advanced threats and ensuring data integrity. AI is now regarded as the core of 6G networks, providing dynamic solutions to evolving security challenges. According to [1], the following are the 10 most important reasons why security is crucial in 6G:

1. Protection of sensitive data: 6G handles large volumes of data at high speeds, which requires strong security to prevent unauthorized access, breaches, and misuse.

2. Safeguarding critical infrastructure: The 6G network infrastructure is designed to be highly integrated and AI-driven. It incorporates components such as User Equipment (UE), Access Networks (AN), Core Networks (CN), Edge and Cloud Computing, and Non-Terrestrial Networks (NTN), where 6G technology plays an important role in supporting critical infrastructure, making robust security measures essential to protect public safety, operational integrity, and societal well-being from cyber threats[20].

3. Global Connectivity: 6G aims to provide global coverage to all regions of the world, requiring strong security measures to protect against cyber threats.

4. Diverse Use Cases: 6G supports various applications like emergency services,AR, and IoT, necessitating a robust security framework to meet specific requirements such as enhanced encryption, AI and ML, quantum security, physical layer security, access control, privacy protection, comprehensive security frameworks, and user awareness and education.

5. Ensuring edge computing security and maintaining data integrity at the network edge are critical aspects of 6G. Edge computing processes data near the source, reducing latency, conserving bandwidth, and allowing immediate data handling. Unlike cloud computing, which relies on remote servers, edge computing processes data locally, minimizing the need to send vast amounts of data to distant data centers [21].

6. Global Connectivity and Handovers: 6G enables seamless global connectivity and network transitions but faces challenges like regulatory environments and cyber threats, requiring robust security measures.

7. AI-Driven Networks: While AI enhances security in 6G, it also introduces potential risks, making the security of AI models and the protection of user data privacy critical priorities.

8. Satellite Communication Protection: 6G uses advanced satellite communication for global coverage, demanding strong security to prevent interference, jamming, and unauthorized access.

9. Real-time application security: despite the low latency of 6G, it must prevent interruptions to benefit augmented reality and machine-to-machine communications.

10. Economic and Societal Impact: A secure 6G ecosystem is essential, as security breaches can result in significant financial losses and erode public trust. Security encompasses the measures and protocols designed to protect the system from unauthorized access and cyberattacks. Trust, on the other hand, reflects the confidence users and stakeholders have in the systems ability to safeguard their data and deliver reliable services [22].

    Fig.1.3.1 illustrates the top 10 reasons why security is essential in 6G networks.

Figure 1.6: Top 10 Reasons Why Security is Essential in 6G

1.3.2 Taxonomy for the Threat Model

According to [19], threat models can be divided into five main categories :

1. Threats against confidentiality: Confidentiality in 6G networks is exposed to many attacks and threats, the most important of which are: MiTM, eavesdropping, espionage, chosen text attacks, stalking impersonation, collaboration and disclosure, and other security threats that pose a risk to 6G networks. Among the most important of these attacks, we highlight the MiTM as the most dangerous due to the security vulnerabilities it causes, such as Fake Base Station (FBSA) attacks in modern mobile networks.MiTM attacks are often countered using public key cryptography, AI/ML-based security, and blockchain-based authentication.

2. Threats against integrity: Integrity threats in 6G networks are categorized into five types: message append, message alteration, tampering, session hijack, and data diddling. These threats are especially concerning in 6G enabled IoT envi-

ronments, such as smart ecosystems. Blockchain and ML-based approaches are being explored to enhance data integrity. Session hijacking and data diddling pose serious integrity risks, especially in fog computing and industrial IoTs. Solutions like QR code-based secret sharing have been proposed to counter these threats effectively.

3. Threats against availability: This classification includes seven major threat types: rerouting, physical threats, DDoS attacks, environmental threats, First in First out(FiFo) threats, free-riding threats, and artificial flooding threats. First, to identify DDoS attacks in 6G Machine-to-Machine (M2M) networks, research has proposed energy-efficient DDoS mitigation techniques. For rerouting and physical threats, virtual shadow networks can be used as a measure in the 6G security architecture. Threats and FiFo threats can be mitigated using machine learning-enabled intrusion detection systems. For environmental threats, intelligent interference mitigation can be implemented, and the network can be avoided in Radio Frequency(RF) saturated environments.

4. Threats against Authentication: The text discusses seven types of threats against authentication in 6G networks: brute force, replay, reuse, forgery-based, partial collision, password, and recovery-based threats. These threats aim to disrupt client-server authentication mechanisms. Password-based schemes are often targeted by attackers pretending to be legitimate, while partial collision attacks use cryptographic methods to extract secret keys or hashes. 6G's Radio Access Network (RAN) is also vulnerable to collision attacks on control signals. To address these issues, lightweight authentication schemes, such as those for maritime transport systems, have been proposed to prevent forgery attacks. Anonymous mutual authentication and key establishment protocols can enhance security against recovery-based attacks in IoT networks. Additionally, transitioning to 6G requires reevaluating traditional protocols like voice over LTE (VoLTE) to avoid reuse vulnerabilities. AI-based solutions and Public Key Cryptography, combined with embedded Subscriber Module (eSIM), offer protection against brute force and replay attacks. Overall, 6G networks need advanced authentication mechanisms to counter these evolving threats.

5. Threat against Access Control: The text outlines five key threats against access control in 6G networks: social engineering, data mining, birthday attacks, cloning, and phishing. These threats are particularly concerning for organizations handling sensitive or proprietary information, as they can lead to unauthorized access and significant damage. To counter these risks, robust computer-based access control systems are essential. Blockchain technology is proposed as a solution for secure access control in 6G ecosystems, offering decentralized and traceable security. Social engineering poses a serious threat to 6G enabled IoT systems,

while data mining attacks can exploit seemingly harmless data from heterogeneous networks. The post-quantum era of 6G requires algorithms resistant to birthday attacks, and quantum computing principles, like the no-cloning theorem, can help prevent cloning attacks. Additionally, adversarial learning algorithms are suggested to mitigate mmWave beam prediction attacks, and phishing attacks targeting AI-based models in spectrum management must be addressed.

Fig.1.7 illustrates the taxonomy of threat models.

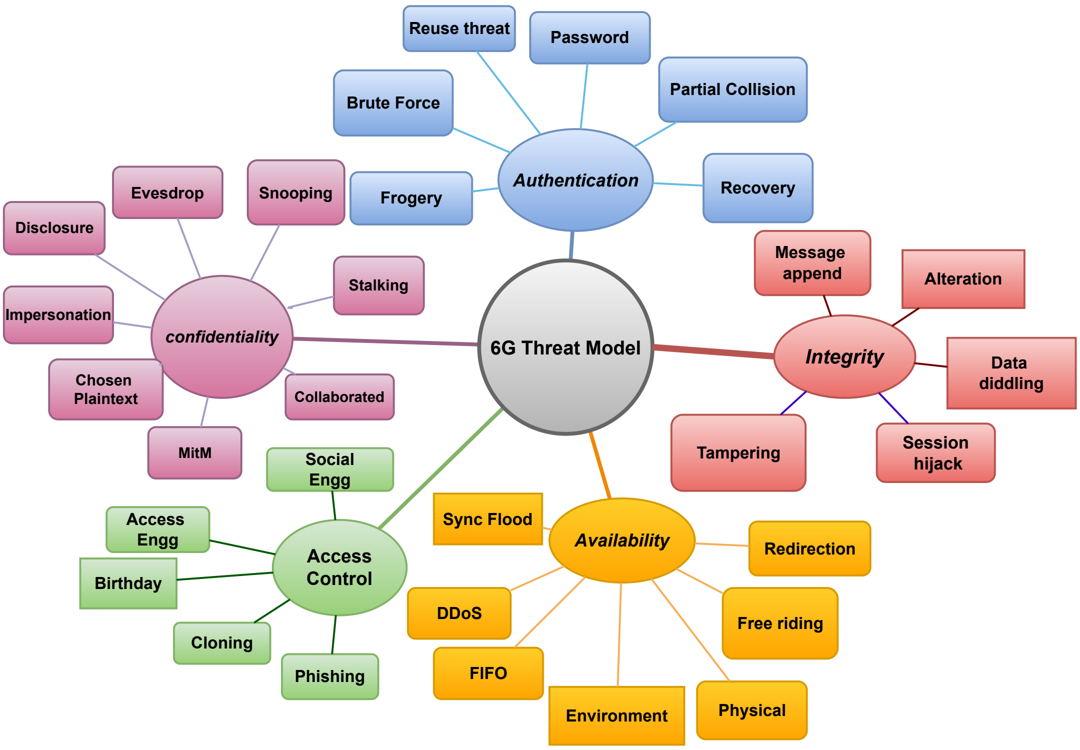

Figure 1.7: Taxonomy for The Threat Models

1.3.3 Approaches to Mitigating Threats in 6G Networks

According to [19], this section presents different ways to deal with threats in 6G networks, especially those related to authentication and privacy. The methods are divided into three main categories:

1. Cryptography Based Methods: To protect 6G mobile networks, both symmetric a asymmetric cryptographic techniques are needed. Public Key Infrastructure (PKI) is used for authentication, but traditional crypto systems like Rivest-Shamir-Adleman (RSA) and the Paillier cryptosystem may be vulnerable to quan-

tum attacks. Cybertwin, a unique network design, offers builtin cryptographic protection but faces problems with scalability and compatibility. Other methods include Advanced Encryption Standard (AES) based techniques, AI-enhanced solutions, Field-Programmable Gate Array (FPGA) based stream ciphers, and post-quantum cryptography. However, many of these methods need further testing to ensure they can handle 6Gs fast speeds and low delays, as well as resist various types of attacks.

2. Attribute-Based Methods: This group includes both traditional and newer approaches such as context-aware security, SDN-based intelligent interference management, and DDoS attack detection. Techniques like QR code-based authentication and quantum communication are also proposed. Context-aware systems can adjust to different situations, but they face issues like scalability and compatibility with older systems. Using quantum key distribution between IoT devices and servers is a promising solution, though it still needs more research.

3. Threat Detection Techniques: Researchers have developed several methods to identify and stop attacks in 6G networks. These include simplified threat models, Non-Orthogonal Multiple Access(NOMA), and Channel State Information(CSI) based authentication. AI and ML are used to detect threats like jamming, malware, and DDoS attacks. Although these tools help identify threats, most of them only work for wireless parts of the network and struggle with compatibility across different systems. Cognitive radio in 6G brings new risks due to its large bandwidth and varied connections. To improve protection, SDN-supported blockchain-based resource management and SDN-powered fog computing are proposed. Digital twins (virtual models of networks) can also be used for monitoring and threat detection. However, more research is still needed to fully protect the entire 6G system.

1.3.4 Comparison Between DoS and DDoS

According to [23, 24], Table 1.2 highlights the main differences between DoS and DDoS attacks in terms of structure, effect, and difficulty of mitigation.

Feature	DoS	DDoS
Definition	An attack that overwhelms a server with traffic from one source rendering a website or source inaccessible	Employs a number of computers or machines to overwhelm a focused resource
Attack source	Single machine or network	Thousands of infected devices (IoT, servers)
Common technique	• Synchronize(SYN)/User Datagram Protocol(UDP)/Internet Control Message(ICMP) Flood • HyperText Transfer Protocol(HTTP) Flood • Smurf Attack	• Protocol-based (Transmission Control Protocol (TCP) SYN floods) • Application-layer (HTTP floods) • IoT-based
Detection difficulty	Easier (single source Internet Protocol(IP))	Harder (traffic appears legitimate, distributed sources)
Mitigation strategies	• Rate limiting • Blacklisting attacker IP • Firewall rules	• Traffic scrubbing • Behavioral AI/ML detection • 6G-specific: Network slicing isolation, Zero trust architecture
Impact on 6G networks	May interfere with an individual 6G slice or edge node	Can overwhelm CN, disrupt network slicing and AI-based orchestration

Table 1.2: Comparison between DoS and DDoS

1.4 Conclusion

6G technology represents a major leap in communication, but it also brings heightened security challenges, notably DoS/DDoS attacks. This type of attack directly threatens one of the core pillars of cybersecurity: availability, leading to service disruption and loss of trust in the system. As networks grow more complex, it becomes essential to integrate intelligent, AI-driven security solutions to ensure service continuity and protect critical infrastructure.

Chapter 2

Proposed Approach

Contents

2.1 Introduction

6G networks aim to enable intelligent and secure communication environments. However, these ambitions are increasingly challenged by rising cybersecurity threats, particularly DoS and DDoS attacks [25]. In this chapter, we review previous works that have explored the use of large language models (LLMs) in the field of cybersecurity, with a particular focus on detecting DoS and DDoS attacks in 6G networks. We then present our proposed methodology, which leverages the PROGRES platform as an application model to simulate attacks within 6G networks and evaluate the effectiveness of AI-based defense mechanisms.

2.2 Part 1: Related Work

This section reviews recent studies on the use of LLMs in 6G network security, focusing on their role in detecting and defending against DoS and DDoS attacks. It highlights key functions, applications, and gaps in existing research.

2.2.1 Overview Of LLM

This section explains the main concept of LLMs and their families, followed by an overview of their architectures.

Definition

LLMs are AI systems capable of processing and generating human-like text, with applications ranging from chatbots to technical troubleshooting. In telecommunications, base station (BS) anomaliessuch as hardware failures or software issuescan disrupt services and are typically resolved manually, leading to inefficiencies. Previous research has explored the use of LLMs in telecommunications for tasks such as customer service and language understanding [26].

Figure 2.1 illustrates the hierarchical structure of AI fields, showing that LLMs fall under Generative AI, which is a subset of deep learning (DL), itself part of machine learning (ML), all within the broader domain of artificial intelligence (AI).

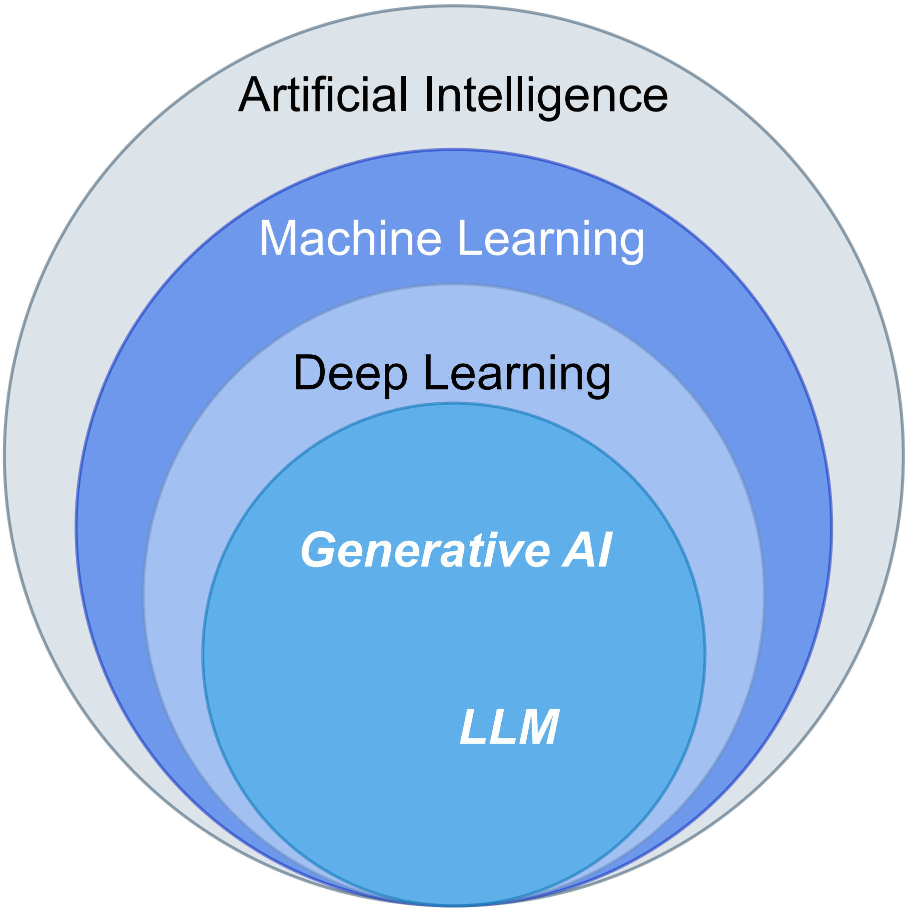

Figure 2.1: LLMs in the AI Hierarchy.

LLMs families

This section presents an overview of prominent LLM families, including Generative Pre-trained Transformer(GPT), Large Language Model Meta AI (LLaMA), and Pathways Language Model (PaLM).

1. GPT Family :

   (a) Developed by OpenAI, the GPT family started with GPT-3 and includes successors like ChatGPT and GPT-4.

(b) These models are based on transformer architecture and are pretrained on massive text corpora to generate human-like text.

(c) GPT models are primarily closed-source, with a focus on generative tasks such as text completion, translation, and dialogue.

(d) They are known for their large scale and strong few-shot learning capabilities[27] [28].

2. LLaMA Family :

   (a) Developed by Meta AI, LLaMA models were first released in early 2023 and have evolved through multiple versions, with LLaMA 4 released in April 2025.

   (b) LLaMA models range in size from 1 billion to 2 trillion parameters.

   (c) LLaMA emphasizes open-weight releases and more accessible licensing, fostering research and commercial use.

   (d) LLaMA 4 is notable for being natively multimodal, supporting text, images, and videos with an architecture that integrates these modalities into unified representations.

   (e) LLaMA models have been integrated into products like Facebook and WhatsApp virtual assistants[29] [28].

3. PaLM Family :

   (a) Developed by Google as part of the Pathways initiative, PaLM models focus on versatility and efficiency in language understanding and generation.

   (b) PaLM models use large transformer architectures, with versions up to 540 billion parameters.

   (c) They are largely closed-source, with limited public API access.

   (d) Specialized versions include Med-PaLM 2 (for medical text processing) and Sec-PaLM (for cybersecurity applications).

   (e) PaLM emphasizes multitask learning and scalability via the Pathways machine learning system.

   (f) PaLM currently lacks native multimodal capabilities and image generation, relying on external tools for visual tasks.[30] [28].

According to [31], Figure 2.2 illustrates the most popular families of large language models (LLMs).

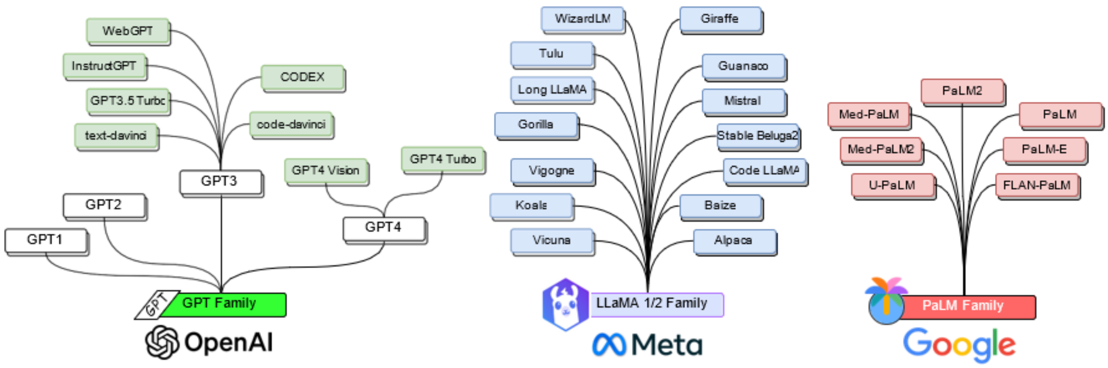

Figure 2.2: Popular LLM families

Comparison Between The Most Popular LLMs

According to the findings presented in [32], Table 2.1 highlights the fundamental differences between the most popular LLMs.

Version	Released By	Parameters	License	Context Window (token)
GPT-4	OpenAI	1.6T	Proprietary	32,768
GPT-3	OpenAI	175B	Proprietary	2,048
Gemini-pro	Google	137B	Proprietary	32K
BERT	Google	345M	Apache 2.0	512
PaLM	Google	540B	Proprietary	8K
LaMDA	Google	173B	Proprietary	/
LLaMA2	Meta AI	70B	Open-source	4,096
MPT	MosaicML	30B	Apache 2.0	8K
Mistral	Mistral AI	7.3B	Apache 2.0	32K
DistilBERT	Hugging Face	66M	Apache 2.0	512
ELECTRA	Google	110M	Apache 2.0	512
DistilGPT-2	Hugging Face	82M	Apache 2.0	1,024

Table 2.1: Comparison Between The Most Popular LLMs

LLM Architecture

According to [26, 33], LLMs rely on three core architectures, and each type balances language understanding and generation differently:

1. Encoder Models: These process input text and create meaningful vector representations, excelling at tasks like classification and sentiment analysis.

2. Decoder Models: Also known as autoregressive models, these focus on generating human-readable text, predicting the next word in a sequence for tasks like text generation and translation.

3. Encoder-Decoder Models: These integrate both understanding and generation, making them suitable for tasks such as translation and summarization.

Steps to Create LLMs

According to [34] and [28], we summarize the process of building LLMs, presenting the main steps as follows.

- **Step 01: Data Cleaning** Clean the raw data to improve quality, which directly impacts LLM performance. Common techniques include removing noise, handling outliers, deduplication, addressing class imbalances, and text preprocessing (e.g., removing special characters and stop words), as well as resolving ambiguities. *Example:* From the raw sentence Hellooo!!! How r u??, we clean it to Hello! How are you?.

- **Step 02: Tokenization** Convert text sequences into smaller units called tokens. This breaks down the input text into manageable segments for the model to process. *Example:* Hello world [Hello, world] or [101, 7592] using token IDs.

- **Step 03: Positional Encoding** Encode the position of tokens in the sequence to help the model understand word order. Techniques include Absolute Positional Embeddings (APE), Relative Positional Embeddings (RPE), Rotary Positional Embeddings, and Relative Positional Bias. *Example:* For the sentence I love AI, the model creates vectors like: Position 0 [0.0000, 1.0000, 0.0000, 1.0000].

- **Step 04: LLM Architecture Selection** Choose the model architecture type: encoder-only, decoder-only, or encoder-decoder. *Example:* BERT is encoder-only, GPT is decoder-only, and T5 uses an encoder-decoder structure.

- **Step 05: Model Pre-training** Train the model on a large, diverse dataset in a self-supervised manner to learn general language patterns, grammar, syntax, and semantics. This foundational training helps the model broadly understand language.

- **Step 06: Fine-tuning and Instruction Tuning** Further train the pre-trained model on specific datasets or tasks to adapt it to particular domains or use cases. This supervised fine-tuning ensures outputs align with human expectations. *Example:* Fine-tuning a general LLM on legal documents to accurately answer legal questions.

- **Step 07: Alignment** Guide the model to adhere to human values, goals, and ethical principles. This step mitigates harmful, biased, or misleading outputs and ensures safe, responsible deployment. *Example:* Reinforcement Learning from Human Feedback (RLHF) is used to discourage toxic or offensive responses.

2.2.2 Roles of Large Language Models in Securing 6G Networks

According to [25], LLMs offer powerful functions that can significantly enhance security protocols in 6G networks by enabling intelligent, adaptive, and automated defense mechanisms. Specifically, LLMs can:

1. Detect and mitigate network attacks such as DoS and DDoS and intrusion attempts by analyzing large volumes of network traffic data in real time to identify anomalous patterns indicative of threats.

2. Generate adaptive security policies or responses based on continuous monitoring of network conditions and threat intelligence, allowing the 6G network to dynamically adjust defenses without human intervention.

3. Automate vulnerability scanning and patch generation for 6G network components, accelerating the identification and remediation of security weaknesses in software and protocols.

4. Analyze vast amounts of unstructured data, including logs, alerts, and security reports to identify emerging threats and provide actionable insights for proactive defense.

2.2.3 LLM Defense Against DoS Attacks

According to [35], we summarize the key strategies of LLM defense against DoS attacks as follows:

1. Anomaly Detection: LLMs analyze network traffic to spot unusual spikes or patterns indicating a DoS attack.

2. Adaptive Mitigation: They generate dynamic responses like blocking suspicious IPs or adjusting firewall rules to reduce attack impact.

3. Input Filtering: LLMs help validate and filter incoming requests to prevent malicious overloads.

4. Continuous Monitoring: They monitor logs and alerts to detect early signs of attacks for quick response.

5. Resource Management: LLMs assist in scaling resources dynamically to maintain service availability during traffic surges.

2.2.4 Related Work on detecting DoS/DDoS attack in 6G networks

In this section, we focus on recent research that explores the use of large language models (LLMs) for detecting DoS and DDoS attacks in 6G networks.

From 2020 to the present, the most relevant studies have proposed concrete LLM-based or AI-driven solutions for detecting and mitigating DoS/DDoS attacksparticularly at the application layer and within the context of 6G. These studies include the following:

1. LLM-based Continuous Intrusion Detection Framework for Next-Gen Networks (2024) [36]

   This paper proposes an adaptive framework using LLMs for continuous detection, identification, and classification of emerging network attacks, designed for next-generation networks including 6G. Promising for adaptive and ongoing detection in 6G contexts. However, details on application-layer focus and mitigation strategies are limited in the available abstract.

2. LLMs in 6G Security: Challenges and Opportunities (2024) [25] This paper analyzes security vulnerabilities of LLMs themselves in 6G environments, proposes a threat taxonomy for adversarial behaviors, and discusses the integration of LLMs with blockchain for autonomous security solutions. While not a detection system, this paper is critical for understanding the security landscape of LLMs in 6G, including potential attack surfaces and defensive strategies relevant to your solutions robustness.

3. Proposed Explainable Interference Control Technique in 6G Networks Using LLMs (2024)[37] The authors investigate how explainable AI-based LLMs (e.g., Llama v3) can improve real-time interference detection and mitigation in 6G networks. This paper focuses on interference control rather than DoS/DDoS detection but demonstrates the potential of explainable LLMs for real-time network security tasks in 6G, which supports the feasibility of your application-layer defense approach.

4. DDoS Attack and Detection Methods in Internet-Enabled Networks (2023) [38] This survey provides a comprehensive survey of DDoS detection methods, including ML-based techniques, on internet-enabled networks. Useful for background and baseline ML methods but lacks LLM-specific or 6G-tailored solutions.

5. The paper [9] highlights the new technologies adopted in 6G, such as AI, holographic communication, and IoT. These advancements are expected to shape the overall future of communications. The design of 6G technology will have a major impact on communications in the 2030s, 6G will deliver reliability, ultralow latency, higher data rates, improved spectral efficiency, ultrafast speeds, more

connected devices, and enhanced security features [1]. Integrating AI into 6G networks creates challenges: cost and efficiency, complexity and integrations, data governance and management, security, latency, scalability and transparency, and federated learning or detected threats [1].

Most of these challenges can be addressed using blockchain technology. Blockchain enhances security by providing decentralized, tamper-proof data management, ensuring data integrity, and preventing unauthorized access. It streamlines authentication processes and secures IoT devices within the network [1].

In addition, among the most important challenges of 6G networks is security. Integrating AI into 6G environments poses significant security risks, including data breaches, cyberattacks, and privacy violations. AI and ML technologies, such as adversarial training and defensive distillation, which involve training models with attack-like samples to improve resilience and defensive distillation, which reduces vulnerability by smoothing out decision boundaries in training data, enhance resistance to attacks and improve resilience against attacks by introducing attack-like samples into training data. The security and integrity of AI-based healthcare systems can be improved by storing distributed ledgers in blockchain as a decentralized network that makes digital assets undecipherable [1].

6. The researchers in [13] provide an overview to understand the multiple opportunities and challenges of AI-based security and privacy. It also identifies future research directions by discussing challenges in AI-based security and privacy, such as AI-model attacks and IoT vulnerabilities, and proposing viable solutions such as federated learning, adversarial ML, and differential privacy.

7. The authors in [13] explore the role of AI in 6G networks, highlighting its benefits and security risks. AI is essential for intelligent network management, but its integration introduces vulnerabilities inherited from 5G and new threats from emerging technologies like blockchain, quantum computing, and the IoE. AI automation of future networks will enable proactive threat detection, intelligent mitigation techniques, and self-sustaining networks, making security systems more dynamic and adaptive.

8. This article [19] discusses the latest emerging technologies in 6G network security. It presents a CIA3-based threat model for 6G networks. We discuss the latest research related to each threat, classifying emerging security measures into three types, including encryption methods (PKI, RSA, DSS, ECC, AES, ESAR, and NGTP), and entity attributes (such as QR codes, 3D location, PLS, shared optical and frequency links, and reconfigurable surfaces). Countermeasures against IDS include several solutions, such as threat matrices, signature matrices, security system information (CSI), DL, ML, AI, and standard operating procedures

(SOPs).

We categorize authentication technologies in 6G communications into eight distinct types, including handover authentication, mutual authentication, physical layer authentication, deniable authentication, token-based authentication, certificate-based authentication, key agreement-based authentication, and multi-factor authentication. Future research directions were also discussed, including privacy preservation in 3D fog computing, privacy-protected smart infrastructures, augmented reality, secure architectures based on SDN networks, enhanced secure routing, physical layer security, THz spectrum, 6G security in quantum computing, and blockchain-based distributed security in 6G networks.

2.3 Part 2: Proposed Methodology

In this section, we explain in detail our proposed approach:

2.3.1 Use Case: PROGRES platform

The PROGRES platform (Progiciel de Gestion Intégré de l'Enseignement Supérieur). Its link is (https://progres.mesrs.dz/webauthentification/ It is an integrated management software system implemented by the Algerian Ministry of Higher Education and Scientific Research (MESRS). Its primary objective is to digitize and streamline various administrative and academic processes within higher education institutions, thereby enhancing governance, transparency, and efficiency.

The PROGRES platform is a centralized digital system implemented by the Algerian MESRS to manage various academic processes, including student enrollments, course registrations, and doctoral applications. This platform serves as a unified interface for students and universities, streamlining administrative tasks and data management.

Individual universities in Algeria maintain their internal databases to handle specific academic records and administrative functions. These internal systems are tailored to meet the unique needs of each institution. The PROGRES platform interacts with these university databases to facilitate processes such as doctoral admissions and student record management.

For instance, during the doctoral admission process, candidates submit their applications through the PROGRES platform. The platform then coordinates with the respective university's internal systems to process these applications, ensuring a seamless and efficient workflow.

Key Roles and Functions of the PROGRES Platform:

- Student Enrollment and Management: Facilitates online registration, course enrollment, and management of student records, reducing paperwork and administrative burdens.
- Doctoral Candidacy Applications: Enables candidates to apply for doctoral programs online, streamlining the submission and evaluation process.
- Recruitment Processes: Manages the recruitment of academic staff, ensuring transparency and equal opportunity in hiring practices.
- Diploma Authentication: Provides a platform for the verification and authentication of university diplomas, aiding in the prevention of credential fraud.
- University Services Management: Oversees applications for housing, scholarships, and transportation, improving the allocation and management of student services.

2.3.2 Use Case Architecture

The architecture of our use case contains:

- Users' smartphones, tablets, or laptops.
- Base station.
- Core Network.
- The University Server (Internal Database of the University) stores and manages student records for a specific university. Key Responsibilities:
  - stores student data (personal info, academic records, courses, enrollment history).
  - Handles authentication (verifies whether a student exists in the university database).
  - Processes enrollment requests (checks prerequisites, seat availability).
  - Updates records (if a student is successfully enrolled).
  - Returns responses to PROGRES about enrollment status.
- PROGRES Cloud Server (Middleware, Hosted Nationally): Acts as a secure intermediary between students and universities. Key Responsibilities:
  - Provides a unified interface for all universities in Algeria.
  - Authenticates users before sending requests to the university database.

- Forwards requests between students and their universities.
- Manages communication between students and universities.
- Ensures security (protects data flow from cyber threats).

Direct student-to-university communication is not allowed because:

- Security Risks: The university server is internal and should not be publicly exposed.
- Scalability: PROGRES manages thousands of students, ensuring that university servers do not get overloaded.
- Standardization: PROGRES ensures that all universities follow a unified system.

Fig. 2.3 represents our use case architecture.

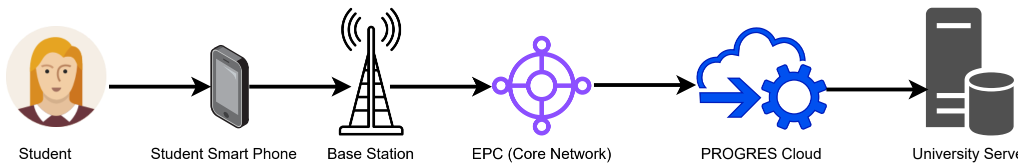

Figure 2.3: Use Case Architecture

2.3.3 Use Case in 4G

To improve security in 6G, let us first explain how a student accesses the PROGRES platform through mobile data (4G) in a traditional system.

Steps from the student to PROGRES Cloud (Request Path in 4G)

In 4G, a student using **a smartphone or tablet** with mobile data follows these steps to interact with the PROGRES platform:

1. The student turns on mobile data on his smartphone.
2. The smartphone connects to the nearest 4G base station (eNB - evolved Node B).
3. The base station forwards the request to the 4G Core Network (EPC - Evolved Packet Core).
4. The EPC determines the best route to forward the students request to the internet.
5. The request reaches the PROGRES platform, which is hosted in a cloud server.

6. The PROGRES cloud forwards the students request to their specific universitys internal database because the student logs into PROGRES using his/her university credentials.
7. The university (universitys internal database) retrieves the students records (e.g., academic profile, courses, etc.).
8. The university sends the data back to PROGRES, which displays it to the student.
9. The student selects a course (level) and submits his/her enrollment request.
10. PROGRES forwards the request to the university database for validation.
11. Once validated, the university database updates the students enrollment status.
12. PROGRES confirms the successful enrollment to the student.

Steps from PROGRES Cloud to the Student (Response Path in 4G)

Once the student submits their enrollment request, the data travels back to them. Let's break it down:

1. The PROGRES cloud server sends the enrollment request to the universitys internal database.
2. The university database checks if the student is eligible for the selected course (e.g., prerequisites met, seats available).
3. If the request is valid, the database updates the students record.
4. If there is an issue (e.g., course full, missing prerequisites), it sends an error response.
5. The universitys database confirms the enrollment (or rejection) to the PROGRES cloud server.
6. PROGRES Displays the Response to the Student (The PROGRES cloud server processes the universitys response and formats it. It generates an enrollment confirmation message or an error notification).
7. PROGRES Sends the Response Back Through the Network (The response travels through the internet EPC (Core Network) Base Station Student's Smartphone).
8. Student Receives Confirmation on Their Device (The student sees a success message (or error) on the PROGRES interface. If successful, they are officially enrolled in the selected course).

Fig.2.4 represents the Request and Response Path in the 4G.

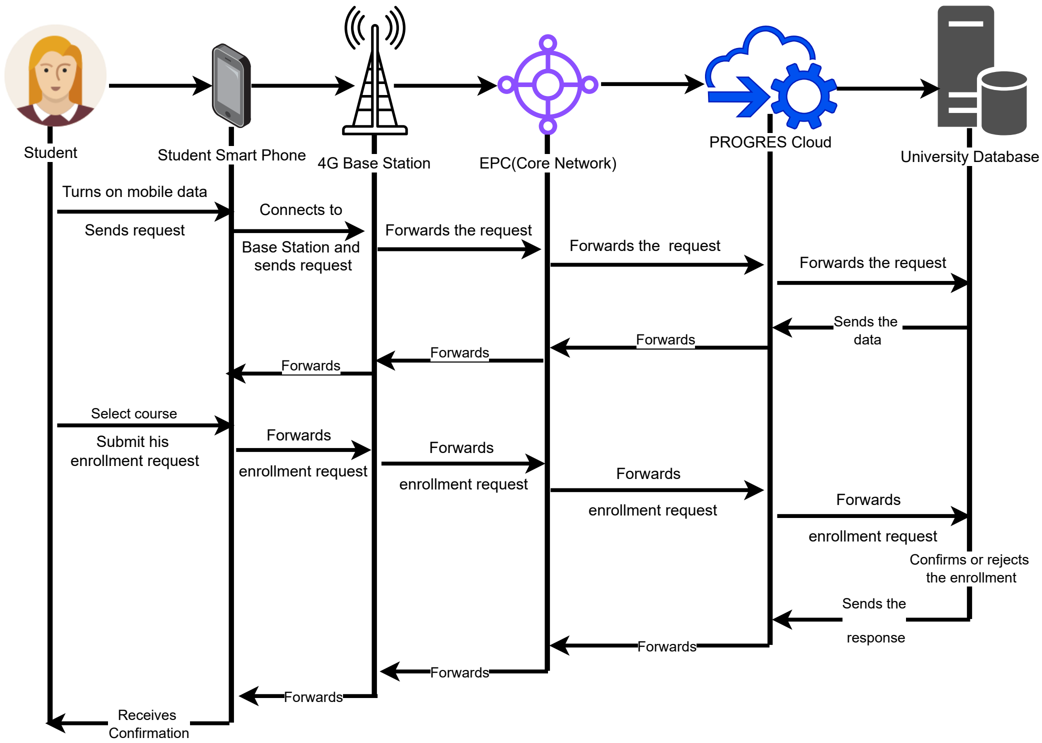

Figure 2.4: 4G Use Case

2.3.4 From 4G to 6G: Additional Features and Advancements

The transition from 4G to 6G introduces significant architectural, security, and operational enhancements to the student enrollment system. Below are the key differences between 4G-based and 6G-based use cases:

1. **Network Architecture Differences:**
    - **Core Network:** Centralized EPC in 4G vs. Distributed, AI-powered, cloud-native core in 6G.
    - **Base Stations:** Traditional eNBs in 4G vs. AI-enhanced smart base stations in 6G.
    - **Data Processing:** Centralized cloud processing (PROGRES Cloud Server) in 4G vs. edge computing with AI-based decision-making in 6G.
2. **AI and Automation Impact:**
    - **Role of AI:** Minimal AI usage in 4G (mostly for fraud detection) vs. AI-

driven authentication, traffic prediction, anomaly detection, and cybersecurity in 6G.

- **Anomaly Detection:** Basic rule-based methods in 4G vs. self-learning, real-time AI-based detection in 6G.
- **Decision-Making:** Manual decision-making by administrators in 4G vs. AI-powered autonomous decisions for enrollment verification in 6G.

3. **Performance and Scalability:**
   - **Latency:** Approximately 10 ms in 4G vs. ultra-low 1 ms in 6G.
   - **Reliability:** Centralized with potential bottlenecks in 4G vs. high availability through distributed computing in 6G.
   - **Bandwidth:** Limited in 4G vs. enhanced bandwidth using THz communication in 6G.
   - **Scalability:** Restricted by centralized processing in 4G vs. improved scalability with decentralized, AI-driven networking in 6G.
4. **Direct Communication and Edge Computing:**
   - **Direct Student-to-University Communication:** Not permitted in 4G due to security concerns vs. enabled in 6G via AI-based edge authentication.
   - **Edge Computing Role:** Minimal use in 4G (dependent on central servers) vs. local processing in 6G, reducing latency.
   - **Data Storage:** Centralized university databases in 4G vs. distributed storage with secure blockchain integration in 6G.
5. **Security Threats and Attack Surfaces:** *(Most critical aspect)*
   - **Attack Surface:** Limited to central servers and internet-based threats in 4G vs. expanded surfaces in 6G due to edge computing, AI integration, and intelligent network slicing.
   - **Authentication and Privacy:** Centralized and vulnerable in 4G vs. AI-powered Zero Trust Authentication in 6G, with exposure to adversarial AI attacks.
   - **DoS/DDoS Attacks:** Target centralized PROGRES Cloud Server in 4G vs. targeting edge nodes, smart base stations, and intelligent routing systems in 6G.

- **Man-in-the-Middle (MitM) Attacks:** Possible between user devices and the PROGRES Cloud in 4G vs. more sophisticated MitM risks in 6G due to vulnerabilities in quantum communication.
- **Eavesdropping and Data Interception:** LTE encryption in 4G (still vulnerable) vs. enhanced encryption in 6G, yet threatened by AI-based decryption methods.
- **Deepfake and Spoofing Attacks:** Low risk in 4G vs. high risk in 6G from AI-generated deepfakes targeting biometric systems.
- **Quantum Threats:** Not applicable in 4G vs. real risks in 6G unless quantum-secure cryptography is employed.

2.3.5 Threat Model

Fig.2.5 represents our threat model that outlines the sequence of steps from the student to the PROGRES Cloud (request path in 6G) and from the PROGRES Cloud back to the student (response path in 6G), highlighting the potential points where DoS/DDoS attacks could occur within our use case.

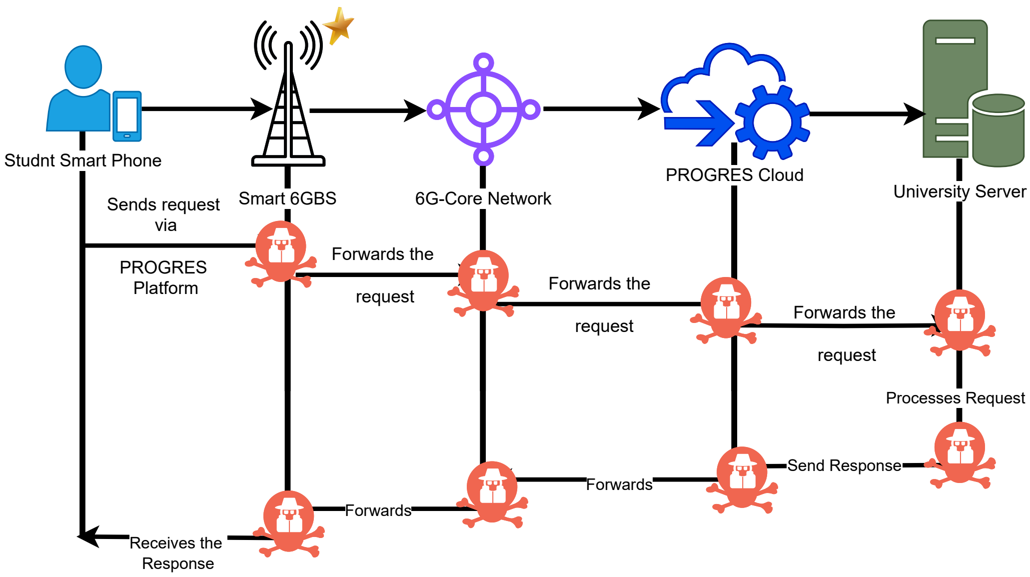

Figure 2.5: Threat Model

Steps from the Student to the PROGRES Cloud (Request Path in 6G)

1. **Student Device Initiates Request:** The student opens the PROGRES platform on their smartphone or tablet, enters their university credentials, and submits a request.

2. **6G Base Station Receives the Request:** The device connects to the nearest AI-enhanced 6G smart base station, where the request is received and optimized for low-latency transmission.

3. **Edge Computing Processes the Request:** The 6G edge computing node pre-processes the request. Basic verifications (e.g., authentication and request format) are performed locally to reduce dependency on the cloud.

4. **6G Core Network Routes the Request:** The 6G core network uses AI-driven intelligent routing to determine the optimal path for the request.

5. **PROGRES Cloud Receives and Verifies the Request:** Acting as a secure national middleware, the PROGRES Cloud Server performs:

   - **User Authentication:** Verification of login credentials.
   - **Request Validity:** Ensuring the request complies with university enrollment policies.

6. **PROGRES Cloud Forwards the Request to the University Server:** After successful verification, the request is forwarded to the university's database, where enrollment availability and student records are processed.

Steps from the PROGRES Cloud to the Student (Response Path in 6G)

1. **University Server Processes the Request:** The university checks:

   - **Student Eligibility:** Verification of enrollment criteria.
   - **Course Availability:** Ensures prerequisites are met and seats are available.

   If approved, the enrollment record is updated.

2. **University Sends the Response to the PROGRES Cloud:** The university returns an enrollment confirmation or error message. If rejected, the response includes the reason (e.g., missing prerequisites, course full).

3. **PROGRES Cloud Formats the Response:** The raw university response is converted into a user-friendly format and encrypted for secure transmission.

4. **6G Core Network Handles Data Transfer:** AI-driven traffic management ensures ultra-low latency. If required, edge computing assists with decryption and

final processing.

5. **6G Base Station Delivers the Response to the Student Device:** The response is transmitted from the core edge base station and received by the student device with latency under 1 ms.

6. **Student Receives the Final Response:** The PROGRES platform displays the enrollment confirmation or error message. If successful, the student is officially enrolled in the selected course.

In 6G networks, the term *edge* can refer to various components such as user devices, sensors, or local servers. In our scenario, we define the edge as the **Smart Base Station**, due to its central role in local data processing and real-time threat detection.

2.4 Conclusion

In conclusion, this chapter underscores the importance of adopting AI, particularly Large Language Models (LLMs), as an effective approach to addressing security challenges in 6G networks. Previous studies have demonstrated the strong potential of these models in the early detection of DoS/DDoS attacks, as well as in the intelligent and adaptive analysis of network behavior. Additionally, we presented the methodology adopted in this work.

Chapter 3

Simulation and Results

Contents

3.1 Introduction

The development of 6G networks introduces new challenges in terms of security, intelligence, and real-time data processing[39]. To address these, researchers increasingly rely on AI-based approaches, especially ML and LLMs, to detect and respond to cyber threats[40]. These models require high-quality datasets and efficient architectures to operate under the strict performance constraints of 6G environments.

In this chapter, we focus on dataset collection and preprocessing, the implementation of both ML and LLM-based models, and the evaluation of their performance in a simulated 6G.

3.2 Implementation Program

In this chapter, we use Python to simulate an anomaly detection system to find DoS and DDoS attacks in 6G networks. It is a popular programming language that is easy to learn and use. It is widely used in fields like data science and ML because it has many useful libraries and tools [41]. Python helps us build and test different models, including those based on LLMs, to detect unusual behavior in network traffic. Fig.3.1 represents python interface.

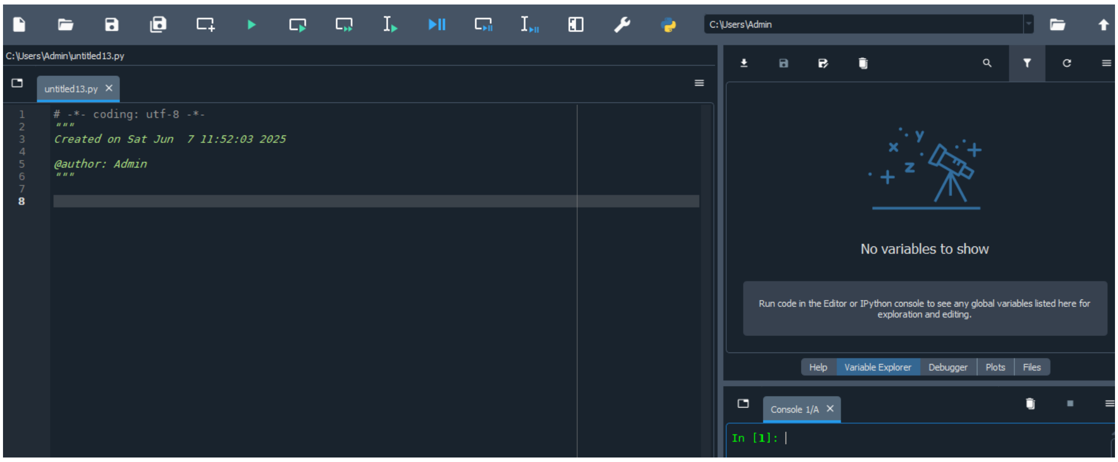

Figure 3.1: Python Interface

We use Google Colab https://colab.research.google.com/, a free online platform that lets us write and run Python code in the cloud as shown in Fig.3.2. Google Colab provides access to powerful hardware like Graphics Processing Units(GPUs), which makes it easier to train complex models without needing a strong computer locally.

It also supports many Python libraries, which helps us quickly develop and test our detection system.

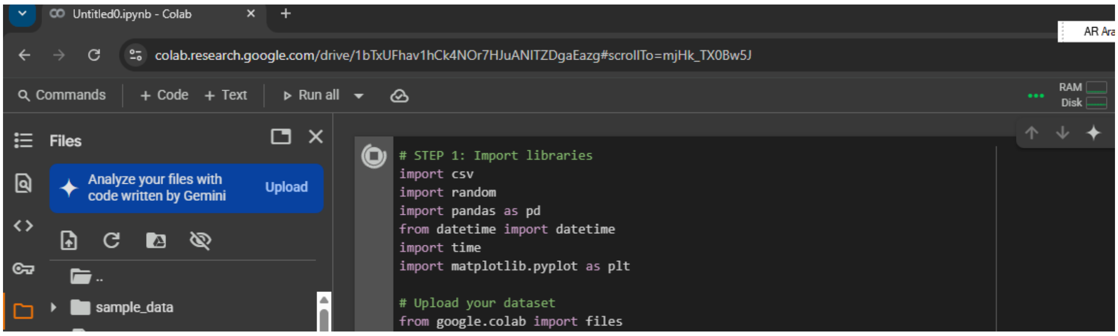

Figure 3.2: Google Colab Interface

3.3 Data Preparation and 6G Simulation Scenario

In this section, we present the datasets and the 6G simulation scenario used to build and test our detection models.

3.3.1 CICIDS 2017 Dataset Overview

1. The CICIDS2017 dataset was developed by the Canadian Institute for Cybersecurity (CIC) and is one of the most widely used and credible datasets for evaluating Intrusion Detection Systems (IDS). It contains realistic network traffic, both normal and malicious, generated in a controlled environment. The dataset includes various attack types such as DoS, DDoS, Brute Force and more[42].

2. We obtained the dataset from [43] and used it as a reference basis for training and evaluating our detection models. The dataset is labeled, rich in features (more than 80 columns), and suitable for ML-based traffic classification.

3. To reduce processing time and focus our work on a manageable dataset, we selected 5% with 20 columns of the original dataset using stratified random sampling. This allowed us to maintain the distribution of attack types while reducing the size of the data. We performed simulations and model training on this reduced dataset, focusing primarily on DoS attack detection. To implement this step, we implemented the following code using Python within the Google Colab environment:

```
from google.colab import files
import pandas as pd
```

```python
# Upload the original dataset
uploaded = files.upload()
# Load the dataset
df = pd.read_csv("cicids2017 (1) (1).csv")
# Clean column names
df.columns = df.columns.str.strip()
# Extract a random 5% sample
df = df.sample(frac=0.05, random_state=42)
# Keep only 20 relevant features + the label
columns_to_keep = [
    'Destination Port', 'Flow Duration', 'Total Fwd
  Packets', 'Total Backward Packets',
    'Total Length of Fwd Packets', 'Total Length of Bwd
  Packets',
    'Fwd Packet Length Mean', 'Bwd Packet Length Mean',
    'Flow Bytes/s', 'Flow Packets/s', 'Flow IAT Mean',
    'Fwd IAT Total', 'Bwd IAT Total', 'Fwd Header Length
  ',
    'Bwd Header Length', 'Min Packet Length', 'Max
  Packet Length',
    'Average Packet Size', 'Init_Win_bytes_backward',
    'act_data_pkt_fwd', 'Label']
# Save the reduced dataset
df.to_csv("cicids2017_5_percent.csv", index=False)
# Download the result
files.download("cicids2017_5_percent.csv")

```

3.3.2 6G Use Case Simulation

This code simulates 6G network traffic by modeling each flow as a student request passing through various components such as a smart base station, core network, cloud, and university server. The simulation classifies each request as either BENIGN or ATTACK, processes it accordingly, logs the results, and finally outputs performance statistics and saves them to a CSV file.

1. Step 1:Import libraries:Import necessary libraries for data handling, timing, and visualization.

```python
import csv
import random
import pandas as pd
from datetime import datetime
```

```
import time
import matplotlib.pyplot as plt
```

2. Step 2: Upload and read the CSV file containing 5% cicids2017 data for simulation.

```
# Upload 5% cicids2017 dataset
from google.colab import files
uploaded = files.upload()
# Load CSV
df = pd.read_csv("cicids2017_5_percent.csv")
```

3. Step 3: Define scenario components classes: Define the main components of the network and their roles in handling requests.

```
# STEP 3: Define Classes
class Student:
    def __init__(self, row):
        self.dst_port = row['Destination Port']
        self.packets_per_second = row['Flow Packets/s']
        self.flow_bytes_per_second = row['Flow Bytes/s']
        self.base_station = None
        self.attacker = self.detect_attacker()
    def detect_attacker(self):
        return "YES" if self.packets_per_second > 5000 or self.flow_bytes_per_second > 10000 else "NO"
class SmartBaseStation:
    def handle_request(self, student):
        return True
class CoreNetwork:
    def route_request(self, student):
        return True
class ProgresCloud:
    def verify_request(self, student):
        return student.attacker == "NO"
class UniversityServer:
    def process_request(self, verified):
        return "SUCCESS" if verified else "REJECTED"
```

4. Step 4: Simulation Scenario: Simulate each data flow through the network and log the processing results.

```
# STEP 4: Simulation Setup
bs = SmartBaseStation()
core = CoreNetwork()
```

```
cloud = ProgresCloud()
univ = UniversityServer()
results = []
base_stations_list = ["BS1", "BS2", "BS3", "BS4", "BS5",
    "BS6", "BS7", "BS8"]
total_requests = len(df)
simulation_duration = 15 * 60
interval = simulation_duration / total_requests
start_datetime = datetime.now()
print(f"\n Simulation Started at: {start_datetime.
   strftime('%Y-%m-%d %H:%M:%S')}")
for i, row in df.iterrows():
    student = Student(row)
    student.base_station = random.choice(
   base_stations_list)
    arrival_time = datetime.now().strftime("%H:%M:%S.%f"
   )[:-3]
    start_time = time.time()
    bs.handle_request(student)
    core.route_request(student)
    verified = cloud.verify_request(student)
    status = univ.process_request(verified)
    end_time = time.time()
    processing_time = round(end_time - start_time, 4)
    results.append({
        'base_station': student.base_station,
        'Destination Port': student.dst_port,
        'attacker': student.attacker,
        'status': status,
        'arrival_time': arrival_time,
        'processing_time': processing_time,
        'Flow Packets/s': student.packets_per_second,
        'Flow Bytes/s': student.flow_bytes_per_second
    })
    time.sleep(interval)
end_datetime = datetime.now()
print(f"\n Simulation Ended at: {end_datetime.strftime
   ('%Y-%m-%d %H:%M:%S')}")
print(f"Total requests processed: {len(results)}")
```

5. Step 5: Save results: Export the simulation results to a CSV file for analysis.

```
results_df = pd.DataFrame(results)
results_df.to_csv("simulation_results_cicids.csv", index
```

```
    =False)
print("\nResults saved to local file:
    simulation_results_cicids.csv")
```

This Fig.3.3 shows the resulting csv file containing 6G simulation results:

base_station	Destination Port	attacker	status	arrival_time	processing_time	Flow Packets/s	Flow Bytes/s
BS8	80	NO	SUCCESS	00:45:22.836	0.0	4.381983613	2764.865255
BS2	80	NO	SUCCESS	00:45:23.029	0.0	0.568570611	3.411423664
BS1	80	NO	SUCCESS	00:45:23.221	0.0	5.55029819	6453.331705
BS1	53	NO	SUCCESS	00:45:23.414	0.0	30.7503075	2367.773678
BS3	60219	YES	REJECTED	00:45:23.606	0.0	25974.02597	155844.1558
BS4	80	NO	SUCCESS	00:45:23.799	0.0	0.14193474	127.3372981
BS5	80	YES	REJECTED	00:45:23.991	0.0	18.10031193	23383.59187
BS8	52471	NO	SUCCESS	00:45:24.184	0.0001	0.221278485	161.4641444
BS2	80	NO	SUCCESS	00:45:24.377	0.0	0.148424888	133.1599594
BS4	80	NO	SUCCESS	00:45:24.569	0.0	1.06054773	6.363286383

Figure 3.3: 6G simulation results

3.4 Intelligent Detection Solutions

In this section, we apply ML and LLMs models on 5% of the CICIDS2017 dataset to detect attacks. These solutions are designed to operate within a smart base station in a 6G environment, focusing on real-time performance and accuracy.

3.4.1 Machine Learning Models for Detecting DoS/DDoS attacks

According to [43], five ML models were tested for detecting DoS/DDoS attacks. These models are:

1. Random Forest (RF): An ensemble method that builds multiple decision trees and combines their outputs through voting for more accurate and stable predictions.
2. K-Nearest Neighbors (KNN): Classifies data based on the majority label of the closest k points; simple but computationally expensive for large datasets.
3. Logistic Regression (LR):A linear model for binary classification that uses a sigmoid function to predict probabilities.
4. Decision Tree (DT): A tree-based model that splits data based on feature values to make decisions; easy to interpret but prone to overfitting.
5. Support Vector Machine (SVM): Finds the optimal hyperplane to separate classes and works well with both linear and non-linear data using kernel functions.

ML Implementation:

In this study, we applied five supervised machine learning algorithms: RF, SVM, KNN, LR, and DT on 5% of the CICIDS2017 dataset to evaluate their effectiveness in DoS/DDoS attack detection.

In this section, we explain how we generate solutions for detecting DoS and DDoS in a smart base station using RF. The same approach is applied for generating solutions using KNN, LR, DT, and SVM.

1. Step 1:This step sets up the working environment by importing all the required libraries for data processing, model training, and evaluation.

```
# Import pandas for handling data in DataFrame format
import pandas as pd
# Import numpy for numerical operations and array
    handling
import numpy as np
# Import train_test_split to split data into training
    and testing sets
from sklearn.model_selection import train_test_split
# Import LabelEncoder to convert categorical labels to
    numerical,
# and StandardScaler to normalize numeric features
from sklearn.preprocessing import LabelEncoder,
    StandardScaler
# Import RandomForestClassifier - a robust ensemble
    model for classification tasks
from sklearn.ensemble import RandomForestClassifier
# Import evaluation metrics to assess model performance
from sklearn.metrics import (
     accuracy_score, precision_score, recall_score,
    f1_score,
     confusion_matrix, classification_report)
# Import time and system libraries to measure execution
    time and resource usage
import time
import psutil
import os

```

2. Step 2: In this step, we connect to Google Drive to load the dataset file, read it into a DataFrame, and clean up the column names for consistency.

```
# Mount Google Drive to access files stored there
```

```
from google.colab import drive
drive.mount('/content/drive')
# Define the path to the dataset file in Google Drive
file_path = "/content/drive/MyDrive/Colab Notebooks/cicids2017_5_percent.csv"
# Load the dataset into a pandas DataFrame
df = pd.read_csv(file_path)
# Remove any leading or trailing spaces from column names (if present)
df.columns = df.columns.str.strip()

```

3. Step 3: Encode the Labels: Converts string class labels into numeric form. Required for classification algorithms that expect numerical targets.

```
# Convert string class labels into numeric form using LabelEncoder
# This is necessary because many machine learning models only accept numerical targets
le = LabelEncoder()
df_subset_selected["label_encoded"] = le.fit_transform(df_subset_selected["Label"])

```

4. Step 4: Split Features and Target Variable: Separates independent variables (X) from the target variable (y). Prepares the dataset for scaling and modeling.

```
X = df_subset_selected.drop(columns=["Label", "label_encoded"])
y = df_subset_selected["label_encoded"]

```

5. Step 5: Standardize Feature Values: Applies z-score normalization to ensure all features have the same scale. Improves model performance and stability, especially for distance-based or linear models.

```
# Keep only numeric columns (exclude text column and other non-numeric types)
X = X.select_dtypes(include=[np.number])
# Apply Z-score normalization to standardize feature values
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

```

6. Step 6:Split Dataset into Training and Testing Sets: Divides data into 80% for training and 20% for evaluation. Stratification ensures class balance is maintained in both sets.

```
    X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42, stratify=y)

```

7. Step7: Use of pretraining record resources:: Captures memory, CPU time, and wall-clock time before training begins. Enables measurement of model training cost and efficiency.

```
process = psutil.Process(os.getpid())
start_mem = process.memory_info().rss / (1024 ** 2)
start_cpu_time = process.cpu_times().user
start_train_time = time.time()

```

8. Step 8: Train Random Forest Model: Fits a Random forest model on the scaled training data. Provides a simple, interpretable baseline for binary or multi-class classification.

```
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

```

9. The final step evaluates the model's accuracy, security metrics, and resource usage to assess its performance in a 6G-like environment.

```
#Post-training Resource Usage
# Measure and calculate memory, CPU, and energy used during model training
end_train_time = time.time()
training_time = end_train_time - start_train_time
end_mem = process.memory_info().rss / (1024 ** 2)  # Memory in MB
memory_used = end_mem - start_mem
end_cpu_time = process.cpu_times().user
cpu_time_used = end_cpu_time - start_cpu_time
# Estimate energy usage assuming CPU power consumption is 20W
energy_consumed_wh = (cpu_time_used / 3600) * 20  # in Watt-hours (Wh)
```

```
# Model Prediction
# Measure inference time and predict labels on the test set
start_inf_time = time.time()
y_pred = lr.predict(X_test)
end_inf_time = time.time()
inference_time = end_inf_time - start_inf_time
# Classification Metrics
# Evaluate model using common classification metrics
acc = accuracy_score(y_test, y_pred)
prec = precision_score(y_test, y_pred, average="weighted", zero_division=0)
rec = recall_score(y_test, y_pred, average="weighted", zero_division=0)
f1 = f1_score(y_test, y_pred, average="weighted", zero_division=0)
# Compute confusion matrix and derived metrics (FAR and MR for binary classification)
cm = confusion_matrix(y_test, y_pred)
if cm.shape == (2, 2):
    tn, fp, fn, tp = cm.ravel()
    far = fp / (fp + tn + 1e-10)  # False Alarm Rate
    mr = fn / (fn + tp + 1e-10)   # Miss Rate
else:
far = mr = np.nan  # Not applicable for multi-class problems
#  6G-Like Performance Metrics
# Calculate throughput (samples per second) and latency (time per sample)
num_samples = len(X_test)
throughput = num_samples / inference_time
latency = inference_time / num_samples
#  Print Results
print("\n Classification Metrics ")
print(f"Accuracy              : {acc:.4f}")
print(f"Precision             : {prec:.4f}")
print(f"Recall (TPR)          : {rec:.4f}")
print(f"F1 Score              : {f1:.4f}")
print(f"False Alarm Rate (FAR): {far:.4f}")
print(f"Miss Rate (MR)        : {mr:.4f}")
print("\nConfusion Matrix:\n", cm)
print("\n6G Resource & Performance Metrics")
print(f"Training Time : {training_time:.2f} sec")
```

```python
print(f"Inference Time : {inference_time:.2f} sec")
print(f"Memory Used : {memory_used:.2f} MB")
#Make sure model_size_mb is defined elsewhere (see note
    below)
print(f"Model Size : {model_size_mb:.4f} MB")
print(f"Throughput: {throughput:.2f} samples/sec")
print(f"Latency : {latency:.6f} sec/sample")
print(f"Estimated Energy Used : {energy_consumed_wh:.4f}
    Wh")
```

3.4.2 LLMs for Detecting DoS/DDoS attacks in 6G

In this work, we selected three lightweight yet powerful LLMs: DistilBERT, DistilGPT-2, and ELECTRA to apply them on 5% of the CICIDS2017 dataset to evaluate their effectiveness in detecting DoS/DDoS attacks.

1. **DistilBERT** was chosen because it provides a smaller and faster alternative to BERT while retaining over 95% of its language understanding capabilities, making it well-suited for resource-constrained environments such as edge computing in 6G networks.

2. **DistilGPT-2** was selected to explore the generative modeling approach and its potential in anomaly generation or log prediction, offering a complementary perspective to classification-based models.

3. **ELECTRA** was chosen for its efficient training mechanism based on replaced token detection, enabling it to outperform BERT-like models while requiring fewer computational resources.

LLMs Implementation:

In this section, we present the approach for leveraging LLMs to detect DoS/DDoS attacks in 6G networks using DistilBERT. This approach can similarly be applied to DistilGPT-2 and ELECTRA.

1. **Install Required Libraries:** Install essential packages including:

    - `transformers` for working with pre-trained language models like DistilBERT,
    - `datasets` for managing and processing datasets,
    - `scikit-learn` for data preprocessing, splitting, and evaluation.

```
!pip install -q transformers datasets scikit-learn
```

2. **Load and Preprocess the Dataset:** Import tools for data handling and preprocessing:

    - Use `pandas` for loading and manipulating the dataset,
    - Use `LabelEncoder` from scikit-learn to convert class labels into numerical format,
    - Split the dataset into training and testing sets.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
```

3. Load the dataset into a DataFrame and Remove any extra spaces from column names to avoid errors.

```
from google.colab import files
uploaded = files.upload()
df = pd.read_csv("cicids2017_5_percent.csv")
df.columns = df.columns.str.strip()
```

4. Convert each row of numerical features into a single, descriptive sentence (text string) that can be fed into a language model.

```
def row_to_text(row):
 return (
     f"Dst Port: {row['Destination Port']}, Flow Duration: {row['Flow Duration']}, "
     f"Fwd Packets: {row['Total Fwd Packets']}, Bwd Packets: {row['Total Backward Packets']}, "
     f"Fwd Length: {row['Total Length of Fwd Packets']}, Bwd Length: {row['Total Length of Bwd Packets']}, "
     f"Fwd Mean Len: {row['Fwd Packet Length Mean']}, Bwd Mean Len: {row['Bwd Packet Length Mean']}, "
     f"Bytes/sec: {row['Flow Bytes/s']}, Packets/sec: {row['Flow Packets/s']}, "
     f"IAT Mean: {row['Flow IAT Mean']}, Fwd IAT Total: {row['Fwd IAT Total']}, Bwd IAT Total: {row['Bwd IAT Total']}, "
     f"Fwd Header Len: {row['Fwd Header Length']}, Bwd Header Len: {row['Bwd Header Length']}, "
     f"Min Len: {row['Min Packet Length']}, Max Len: {row['Max Packet Length']}, "
```

```
        f"Avg Packet Size: {row['Average Packet Size']}, Init Win Back: {row['Init_Win_bytes_backward']}, "
        f"Active Data Fwd: {row['act_data_pkt_fwd']}"
    )
df_subset_selected["text"] = df_subset_selected.apply(row_to_text, axis=1)
```

5. Encode string labels (e.g. "BENIGN", "ATTACK", etc.) into numeric values so they can be used as targets in training.

```
le = LabelEncoder()
df_subset_selected["label_encoded"] = le.fit_transform(df_subset_selected["Label"])
```

6. Split the dataset into training (80%) and testing (20%) sets, making sure the class distribution is preserved using stratify.

```
X_train, X_test, y_train, y_test = train_test_split(
    df_subset_selected["text"],
    df_subset_selected["label_encoded"],
    test_size=0.2,
    random_state=42,
    stratify=df_subset_selected["label_encoded"]
)
```

7. Tokenization:Load the tokenizer for the pre-trained DistilBERT model, which converts text into token IDs.

```
from transformers import DistilBertTokenizerFast
tokenizer = DistilBertTokenizerFast.from_pretrained("distilbert-base-uncased")
```

8. Convert the training and testing texts into tokenized format (with padding and truncation), ready for input into the transformer model.

```
train_encodings = tokenizer(X_train.tolist(), truncation=True, padding=True, max_length=256)
test_encodings = tokenizer(X_test.tolist(), truncation=True, padding=True, max_length=256)
```

9. Create PyTorch Dataset for Tokenized Inputs:We create a custom dataset class so that HuggingFace Trainer can use the tokenized data and labels proper.

```
import torch
from torch.utils.data import Dataset
```

```
# Define a custom dataset class to wrap tokenized data and labels
class AttackDataset(Dataset):
    def __init__(self, encodings, labels):
        self.encodings = encodings  # Dictionary of tokenized inputs
        self.labels = labels # List of encoded labels
    def __getitem__(self, idx):
        # Get one sample at a time as a dictionary of tensors
        item = {k: torch.tensor(v[idx]) for k, v in self.encodings.items()}
        item["labels"] = torch.tensor(self.labels[idx])  # Add label tensor
        return item
    def __len__(self):
        return len(self.labels)  # Return total number of samples
# Create dataset objects for training and testing
train_dataset = AttackDataset(train_encodings, y_train.tolist())
test_dataset = AttackDataset(test_encodings, y_test.tolist())
```

10. Define Training Arguments:We configure how the model will train: number of epochs, batch size, logging, and output directories.

```
from transformers import TrainingArguments
# Set up training configuration like batch size, epochs, and logging
training_args = TrainingArguments(
    output_dir="./results",  # Directory to save model outputs
    num_train_epochs=3,  # Train for 3 epochs
    per_device_train_batch_size=8, # Training batch size
    per_device_eval_batch_size=8, # Evaluation batch size
    logging_dir="./logs", # Where to save training logs
    logging_steps=10, # Log every 10 steps
    do_train=True, # Enable training mode
    do_eval=True  # Enable evaluation during training
)
```

11. Load Model, Train, and Monitor Resources:We load the pre-trained DistilBERT

model, train it using Trainer, and track resource usage like time, memory, and Central Processing Unit(CPU).

```python
from transformers import (
    DistilBertTokenizerFast,
    DistilBertForSequenceClassification,
    Trainer)
import time
import os
import psutil
import numpy as np
import shutil
# Load pre-trained DistilBERT for classification with 2 output classes
model = DistilBertForSequenceClassification.from_pretrained(
    "distilbert-base-uncased",
    num_labels=2  # Adjust this if you have more than 2 classes
)
# Set up Trainer to handle training and evaluation
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=test_dataset
)
# Record CPU and memory usage before training
process = psutil.Process(os.getpid())
start_mem = process.memory_info().rss / (1024 ** 2) # Memory in MB
start_cpu_time = process.cpu_times().user # CPU time in seconds
start_train_time = time.time() # Wall time
# Start model training
trainer.train()
# Record resource usage after training
end_train_time = time.time()
training_time = end_train_time - start_train_time
end_mem = process.memory_info().rss / (1024 ** 2)
memory_used = end_mem - start_mem
end_cpu_time = process.cpu_times().user
cpu_time_used = end_cpu_time - start_cpu_time
# Estimate energy consumption (very simplified)
```

```
energy_consumed_wh = (cpu_time_used / 3600) * 20  # Assuming CPU uses 20W
# Save trained model temporarily to measure size
trainer.save_model("temp_model")
# Calculate saved model size in MB
model_size_mb = sum(
    os.path.getsize(os.path.join(dirpath, filename))
    for dirpath, _, filenames in os.walk("temp_model")
    for filename in filenames
) / (1024 ** 2)
```

12. Evaluate Model and Report Metrics:We measure the accuracy, precision, recall, F1-score, confusion matrix, and performance metrics like inference time and throughput

```
from sklearn.metrics import (
    accuracy_score, precision_score, recall_score, f1_score,
    confusion_matrix, classification_report
)
# Measure inference time
start_inf_time = time.time()
preds = trainer.predict(test_dataset)
end_inf_time = time.time()
inference_time = end_inf_time - start_inf_time
# Convert prediction logits to class labels
y_pred = np.argmax(preds.predictions, axis=1)
y_test = np.array(test_dataset.labels)
# Compute evaluation metrics
acc = accuracy_score(y_test, y_pred)
prec = precision_score(y_test, y_pred, zero_division=0)
rec = recall_score(y_test, y_pred, zero_division=0)
f1 = f1_score(y_test, y_pred, zero_division=0)
# Confusion matrix and derived rates
cm = confusion_matrix(y_test, y_pred)
tn, fp, fn, tp = cm.ravel()
far = fp / (fp + tn + 1e-10)  # False Alarm Rate
mr = fn / (fn + tp + 1e-10)   # Miss Rate
# Inference throughput and latency
num_samples = len(test_dataset)
throughput = num_samples / inference_time
latency = inference_time / num_samples
# Display results
print("\n Classification Metrics")
```

```python
print(f"Accuracy : {acc:.4f}")
print(f"Precision : {prec:.4f}")
print(f"Recall (TPR) : {rec:.4f}")
print(f"F1 Score : {f1:.4f}")
print(f"False Alarm Rate (FAR): {far:.4f}")
print(f"Miss Rate (MR) : {mr:.4f}")
print("\nConfusion Matrix:\n", cm)
print("\nClassification Report:\n",
    classification_report(y_test, y_pred))
print("\n6G Resource & Performance Metrics ")
print(f"Training Time : {training_time:.2f} sec")
print(f"Inference Time : {inference_time:.2f} sec")
print(f"Memory Used : {memory_used:.2f} MB")
print(f"Model Size : {model_size_mb:.2f} MB")
print(f"Throughput : {throughput:.2f} samples/sec")
print(f"Latency : {latency:.6f} sec/sample")
print(f"Estimated Energy Used : {energy_consumed_wh:.4f}
     Wh")
# Clean up saved model directory
shutil.rmtree("temp_model")
```

3.5 Evaluation Metrics

To evaluate the performance of both traditional ML models (e.g., Random Forest) and LLMs in the context of DoS and DDoS detection in 6G networks, we consider a combination of ML, network-specific, and security-focused metrics. These are categorized and explained as follows [44, 45, 46, 47]:

The evaluation metrics used for assessing the DoS/DDoS detection models can be grouped into the following categories:

1. **Performance Metrics**
   These metrics evaluate the accuracy and effectiveness of the model in detecting attacks:

   - **Accuracy**: Overall correctness of the classification, it is calculated as [26]:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.1}$$

     Where:

     (a) TP = True Positives

(b) TN = True Negatives

(c) FP = False Positives

(d) FN = False Negatives

- **Precision**: Percentage of detected attacks that are actual attacks, it is calculated as [26]:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3.2}$$

- **Recall (True Positive Rate)**: Percentage of actual attacks correctly detected, it is calculated as[26]:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3.3}$$

- **F1 Score**: Harmonic mean balancing precision and recall, it is calculated as [26]:

$$F1\text{-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

- **False Alarm Rate (FAR)**: Proportion of benign traffic wrongly flagged as attacks.

- **Miss Rate (MR)**: Proportion of attack traffic that was missed by the model.

2. **Efficiency and Throughput Metrics**
   These metrics quantify the speed and throughput of the detection system:

   - **Throughput**: Number of samples processed per second.
   - **Latency**: Average time taken to process a single sample.

3. **Resource Utilization Metrics**
   These metrics capture the system resources consumed during training and inference:

   - **Memory Used**: Amount of Random Access Memory (RAM) used during model training.
   - **Model Size**: Storage size of the trained model.
   - **Training Time**: Time taken to train the model.
   - **Inference Time**: Time taken to classify the test dataset.

4. **Energy Metrics**
These metrics estimate the energy consumption associated with training/inference:

    - **Energy Consumed (Wh)**: Estimated power usage in Watt-hours; the amount of power (watts) used over a period of time (hours).

3.6 Evaluation results and Discussion

This section presents a summary of the experimental results obtained from the evaluated models.

Results of LLMs and ML Models on Performance Metrics:

After applying ML and LLMs models mentioned above to a sample representing 5% of the CICIDS2017 dataset, the performance evaluation results shown in the following table 3.1 were obtained:

Model Type	Model	Accuracy	Precision	Recall	F1 Score	FAR	MR
LLM	DistilBERT	1	1	1	1	0	0
	Electra	0.998	0.9964	1	0.9982	0.0045	0
	DistilGPT-2	0.996	0.92	0.9964	0.9964	0.0045	0.0036
ML	RF	0.996	0.996	0.996	0.996	0	0.0072
	KNN	0.996	0.996	0.996	0.996	0.0045	0.0036
	LR	0.97	0.9711	0.97	0.9699	0.0633	0.0036
	DT	0.996	0.996	0.996	0.996	0	0.0072
	SVM	0.964	0.9657	0.964	0.9638	0.0769	0.0036

Table 3.1: Results of LLMs and ML Models on Performance Metrics

From Table3.1, it is evident that the DistilBERT model achieved the best performance across all metrics, scoring 100% in Accuracy, Recall, Precision, and F1-score. In contrast, the LR model showed the weakest results, with the lowest precision (0.9711) and the highest relative error (MR = 0.0432). We can conclude that LLMs demonstrated significantly higher effectiveness in detecting DoS attacks compared to traditional ML models.

Results of LLM and LM on Efficiency and Throughput Metrics:

After applying ML and LLMs models mentioned above to a sample representing 5% of the CICIDS2017 dataset, Efficiency and Throughput results shown in the following table3.2 were obtained:

Model Type	Model	Throughput (samples/sec)	Latency (sec/sample)
LLM	DistilBERT	3.57	0.279
	Electra	1.55	0.643
	DistilGPT-2	3.66	0.273
ML	RF	81212.5	0.000012
	KNN	10627.05	0.000094
	LR	899679.1	0.000001
	DT	1374280.4	0.000001
	SVM	76583.1	0.000013

Table 3.2: Results of LLM and LM on Efficiency and Throughput Metrics

Table 3.2 shows that the Decision Tree (DT) model was the fastest in terms of processing, achieving the highest throughput and the lowest latency. In contrast, the ELECTRA model was the slowest, processing the fewest samples per second and exhibiting the highest response time. These results indicate that traditional machine learning models outperform LLMs in terms of speed, whereas LLMs tend to be slower due to their complex architectures.

Results of LLM and ML on Resource Utilization Metrics:

After applying ML and LLMs models mentioned above to a sample representing 5% of the CICIDS2017 dataset, Resource Utilization results shown in the following table 3.3 were obtained:

Model Type	Model	Memory Used(MB)	Model Size(MB)	Training Time	Inference Time
LLM	DistilBERT	1635.97	255.43	5720.75	139.95
	ELECTRA	4185.3	417.68	11780.2	321.75
	DistilGPT-2	474.8	312.49	6015.17	136.70
ML	RF	1.5	0.29	0.18	0.01
	KNN	0	0.35	12.86	0.05
	LR	0.67	0.001	12.86	0
	DT	0	0.0021	0.01	0
	SVM	0	0.0453	0.02	0.01

Table 3.3: Results of LLM and Ml on Resource Utilization Metrics

Table 3.3 reveals that the Random Forest model had the lowest memory usage and time consumption, requiring minimal resources for both training and execution. In contrast, the ELECTRA model consumed the most resources, particularly in terms of memory and training time. These findings highlight that LLMs require significantly more computational resources, which may limit their deployment in real-world systems unless adequate infrastructure is available.

Results of LLM and ML on Energy Metrics

After applying ML and LLMs models mentioned above to a sample representing 5% of the CICIDS2017 dataset,Energy results shown in the following table 3.4 were obtained:

Model Type	Model	Estimated Energy Used (Wh)
LLM	DistilBERT	30.7048
	Electra	63.8355
	DistilGPT-2	31.0624
ML	RF	0.001
	KNN	0.0004
	LR	0.0011
	DT	0.0001
	SVM	0.0001

Table 3.4: Results of LLM and ML on Energy Metrics

According to Table 3.4, the ELECTRA model recorded the highest energy consumption (63.83 Wh), reflecting its high computational complexity. In contrast, traditional models such as Random Forest (RF) and Decision Tree (DT) exhibited very low energy usage, consuming less than 0.01 Wh. This demonstrates that although LLMs may offer superior accuracy, they demand significantly more energy, a critical consideration for deployment in real-world 6G environments.

Conclusion of Results:

Based on the comparative analysis of traditional machine learning models and large language models (LLMs), DistilBERT emerged as the most effective in terms of accuracy for detecting DoS/DDoS attacks in 6G networks. However, this high performance came at the cost of increased computational demands, including longer processing times, higher memory usage, and elevated energy consumption. Conversely, models such as Random Forest (RF) and Decision Tree (DT) demonstrated exceptional efficiency in terms of speed, resource utilization, and energy consumption, but delivered comparatively lower accuracy.

ELECTRA, despite its advanced architecture, proved to be the most resource-intensive across multiple metrics, including memory and energy usage, making it less suitable for resource-constrained edge environments.

These findings highlight a critical trade-off: while LLMs are well-suited for intelligent and accurate threat detection in 6G networks, their deployment must consider infrastructure capabilities. Achieving an optimal balance between performance, efficiency, and scalability is essential to harness the benefits of LLMs without compromising the energy and resource constraints inherent in real-world 6G applications.

3.7 Conclusion

In conclusion, the integration of AI modelsboth traditional and language-basedoffers promising capabilities for securing future 6G networks. By leveraging high-quality datasets and advanced architectures such as DistilBERT, it becomes possible to detect and analyze threats in a more context-aware and efficient manner. These findings underscore the growing potential of large language models (LLMs) in communication systems, paving the way for intelligent, secure, and adaptive network infrastructures in the 6G era.

General Conclusion

In light of the rapid advancements in wireless communication technologies, 6G networks represent a significant leap forward in terms of speed, performance, and intelligence. However, they also introduce unprecedented security challengesparticularly DoS and DDoS attacks that target the core infrastructure of these next-generation systems. This dissertation proposed an innovative approach to strengthening 6G network security by leveraging Artificial Intelligence, specifically Large Language Models (LLMs), renowned for their advanced analytical and adaptive capabilities.

Throughout this study, we introduced an intelligent security framework utilizing LLMs for real-time threat detection and response. To validate this approach, we developed a realistic simulation inspired by the PROGRES platform, a widely adopted system in Algerias higher education sector. Using customized datasets derived from CICIDS2017, we trained a range of machine learning and language-based models to identify anomalous traffic patterns. The results demonstrated that the proposed models were highly effective in distinguishing between legitimate and malicious behaviors, with marked improvements in both resource efficiency and detection speed.

This research confirms that integrating LLMs into future network architectures is a promising path toward achieving proactive, interpretable, and adaptive cybersecurity solutions. It also emphasizes the growing need for AI-driven mechanisms to address the dynamic and complex threat landscape inherent in 6G environments.

Future Work:

1. Expanding the evaluation to cover a wider range of cyberattacks, including zero-day and multi-vector threats.
2. Utilizing more up-to-date and representative datasets that closely mirror real-world 6G deployment scenarios.
3. Integrating blockchain technologies and differential privacy techniques to enhance data integrity, transparency, and user trust.

Bibliography

[1] Navneet Kaur, Naresh Kshetri, and Purnendu Shekhar Pandey. 6ainets: Harnessing artificial intelligence for the 6g network security: Impacts and challenges. *arXiv preprint arXiv:2404.08643*, 2024.

[2] Michael Guastalla, Yiyi Li, Arvin Hekmati, and Bhaskar Krishnamachari. Application of large language models to ddos attack detection. In *International Conference on Security and Privacy in Cyber-Physical Systems and Smart Vehicles*, pages 83–99. Springer, 2023.

[3] Qi Qin, Xinye Cao, Guoshun Nan, Sihan Chen, Rushan Li, Li Su, Haitao Du, Qimei Cui, Pengxuan Mao, Xiaofeng Tao, et al. An llm-based self-evolving security framework for 6g space-air-ground integrated networks. *arXiv preprint arXiv:2505.03161*, 2025.

[4] Shaimaa A. Abdel Hakeem, Hanan H. Hussein, and HyungWon Kim. Security requirements and challenges of 6g technologies and applications. *Sensors*, 22(5):1969, 2022.

[5] Michael Guastalla, Yiyi Li, Arvin Hekmati, and Bhaskar Krishnamachari. Application of large language models to ddos attack detection. In *Security and Privacy in Cyber-Physical Systems and Smart Vehicles*, volume 552 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 83–99. Springer Nature, 2024.

[6] Walid Saad, Mehdi Bennis, and Mingzhe Chen. A vision of 6g wireless systems: Applications, trends, technologies, and open research problems. *IEEE network*, 34(3):134–142, 2019.

[7] Cheng-Xiang Wang, Xiaohu You, Xiqi Gao, Xiuming Zhu, Zixin Li, Chuan Zhang, Haiming Wang, Yongming Huang, Yunfei Chen, Harald Haas, et al. On the road to 6g: Visions, requirements, key technologies, and testbeds. *IEEE Communications Surveys & Tutorials*, 25(2):905–974, 2023.

[8] Mohammed Banafaa, Ibraheem Shayea, Jafri Din, Marwan Hadri Azmi, Abdulaziz Alashbi, Yousef Ibrahim Daradkeh, and Abdulraqeb Alhammadi. 6g mobile communication technology: Requirements, targets, applications, challenges, advantages, and opportunities. *Alexandria Engineering Journal*, 64:245–274, 2023.

[9] Shuping Dang, Osama Amin, Basem Shihada, and Mohamed-Slim Alouini. What should 6g be? *Nature Electronics*, 3(1):20–29, 2020.

[10] Ioannis P Chochliouros, Na Yi, Anastasia S Spiliopoulou, Alexandros Kostopoulos, Nathan Gomes, Uwe Herzog, Tao Chen, Athanassios Dardamanis, Olga Segou, Philippos Assimakopoulos, et al. Enhanced mobile broadband as enabler for 5g: actions from the framework of the 5g-drive project. In *Artificial Intelligence Applications and Innovations: AIAI 2019 IFIP WG 12.5 International Workshops: MHDW and 5G-PINE 2019, Hersonissos, Crete, Greece, May 24–26, 2019, Proceedings 15*, pages 31–45. Springer, 2019.

[11] Mostafa Zaman Chowdhury, Md. Shahjalal, Shakil Ahmed, and Yeong Min Jang. 6g wireless communication systems: Applications, requirements, technologies, challenges, and research directions. *IEEE Open Journal of the Communications Society*, 1:957–975, 2020.

[12] Abdulhalim Fayad, Tibor Cinkler, Jacek Rak, and Manish Jha. Design of cost-efficient optical fronthaul for 5g/6g networks: An optimization perspective. *Sensors*, 22(23):9394, 2022.

[13] Yushan Siriwardhana, Pawani Porambage, Madhusanka Liyanage, and Mika Ylianttila. Ai and 6g security: Opportunities and challenges. In *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, pages 616–621. IEEE, 2021.

[14] Shimaa A Abdel Hakeem, Hanan H Hussein, and HyungWon Kim. Vision and research directions of 6g technologies and applications. *Journal of King Saud University-Computer and Information Sciences*, 34(6):2419–2442, 2022.

[15] Ramraj Dangi, Gaurav Choudhary, Nicola Dragoni, Praveen Lalwani, Utkarsh Khare, and Souradeep Kundu. 6g mobile networks: Key technologies, directions, and advances. In *Telecom*, volume 4, pages 836–876. MDPI, 2023.

[16] Mostafa Zaman Chowdhury, Md Shahjalal, Shakil Ahmed, and Yeong Min Jang. 6g wireless communication systems: Applications, requirements, technologies, challenges, and research directions. *IEEE Open Journal of the Communications Society*, 1:957–975, 2020.

[17] Krishnakanth Mohanta and Saba Al-Rubaye. Towards 6g satellite–terrestrial networks: Analysis of air mobility operations. *Electronics*, 13(14):2855, 2024.

[18] Aarushi Sharma. Optimising beamforming techniques in 5g nr for 6g v2x communication. Master's thesis, Universitat Politècnica de Catalunya, 2023.

[19] Syed Hussain Ali Kazmi, Rosilah Hassan, Faizan Qamar, Kashif Nisar, and Ag Asri Ag Ibrahim. Security concepts in emerging 6g communication: Threats, countermeasures, authentication techniques and research directions. *Symmetry*, 15(6):1147, 2023.

[20] Carlos J Bernardos and Mikko A Uusitalo. European vision for the 6g network ecosystem. *The 5G Infrastructure Association*, 2021.

[21] David Kolevski and Katina Michael. Edge computing and iot data breaches: Security, privacy, trust, and regulation. *IEEE Technology and Society Magazine*, 43(1):22–32, 2024.

[22] Volker Ziegler, Peter Schneider, Harish Viswanathan, Michael Montag, Satish Kanugovi, and Ali Rezaki. Security and trust in the 6g era. *Ieee Access*, 9:142314–142327, 2021.

[23] Bomin Mao, Jiajia Liu, Yingying Wu, and Nei Kato. Security and privacy on 6g network edge: A survey. *IEEE communications surveys & tutorials*, 25(2):1095–1127, 2023.

[24] K Munivara Prasad, A Rama Mohan Reddy, and K Venugopal Rao. Dos and ddos attacks: defense, detection and traceback mechanisms-a survey. *Global Journal of Computer Science and Technology*, 14(7):15–32, 2014.

[25] Tri Nguyen, Huong Nguyen, Ahmad Ijaz, Saeid Sheikhi, Athanasios V Vasilakos, and Panos Kostakos. Large language models in 6g security: challenges and opportunities. *arXiv preprint arXiv:2403.12239*, 2024.

[26] Asma Iman Kouachi and Imed Eddine Mokrane. Leveraging large language models for base station in telecommunications. In *2024 1st International Conference on Electrical, Computer, Telecommunication and Energy Technologies (ECTE-Tech)*, pages 1–8. IEEE, 2024.

[27] Katikapalli Subramanyam Kalyan. A survey of gpt-3 family large language models including chatgpt and gpt-4. *Natural Language Processing Journal*, 6:100048, 2024.

[28] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

[29] Abhinand Balachandran. Tamil-llama: A new tamil language model based on llama 2. *arXiv preprint arXiv:2311.05845*, 2023.

[30] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[31] Badhan Chandra Das, M Hadi Amini, and Yanzhao Wu. Security and privacy challenges of large language models: A survey. *ACM Computing Surveys*, 57(6):1–39, 2025.

[32] Deepchecks Community Blog. Llm models comparison: Gpt-4, bard, llama, flan-ul2, bloom, March 2024. Accessed: 2025-05-23.

[33] Hao Zhou, Chengming Hu, Ye Yuan, Yufei Cui, Yili Jin, Can Chen, Haolun Wu, Dun Yuan, Li Jiang, Di Wu, et al. Large language model (llm) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities. *IEEE Communications Surveys & Tutorials*, 2024.

[34] Tasnim Ahmed, Nicola Piovesan, Antonio De Domenico, and Salimur Choudhury. Linguistic intelligence in large language models for telecommunications. In *2024 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1237–1243. IEEE, 2024.

[35] Kuofeng Gao, Tianyu Pang, Chao Du, Yong Yang, Shu-Tao Xia, and Min Lin. Denial-of-service poisoning attacks against large language models. *arXiv preprint arXiv:2410.10760*, 2024.

[36] Frederic Adjewa, Moez Esseghir, and Leila Merghem-Boulahia. Llm-based continuous intrusion detection framework for next-gen networks. *arXiv preprint arXiv:2411.03354*, 2024.

[37] H Ahmed Tahir, Walaa Alayed, Waqar Ul Hassan, and Amir Haider. Proposed explainable interference control technique in 6g networks using large language models (llms). *Electronics*, 13(22):4375, 2024.

[38] Kazeem B Adedeji, Adnan M Abu-Mahfouz, and Anish M Kurien. Ddos attack and detection methods in internet-enabled networks: Concept, research perspectives, and challenges. *Journal of Sensor and Actuator Networks*, 12(4):51, 2023.

[39] Marco Giordani, Michele Polese, Marco Mezzavilla, Sundeep Rangan, and Michele Zorzi. Toward 6g networks: Use cases and technologies. *IEEE Communications Magazine*, 58(3):55–61, 2020.

[40] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[41] Van Rossum. Python 3 reference manual. *(No Title)*, 2009.

[42] Ibrahim Sharafaldin, Arash Habibi Lashkari, and Ali A Ghorbani. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, pages 108–116. SciTePress, 2018.

[43] S Abiramasundari and V Ramaswamy. Distributed denial-of-service (ddos) attack detection using supervised machine learning algorithms. *Scientific Reports*, 15(1):13098, 2025.

[44] Ahmed Abdel Hakeem et al. Security requirements and challenges of 6g technologies and networks. *PMC*, 2022. Accessed 2025-06-06.

[45] Mengmeng Yang, Youyang Qu, Thilina Ranbaduge, Chandra Thapa, Nazatul Sultan, Ming Ding, Hajime Suzuki, Wei Ni, Sharif Abuadbba, David Smith, Paul Tyler, Josef Pieprzyk, Thierry Rakotoarivelo, Xinlong Guan, and Sirine M'rabet. From 5g to 6g: A survey on security, privacy, and standardization pathways. *arXiv preprint arXiv:2410.21986*, 2024. Accessed 2025-06-06.

[46] Authors Not Specified. Security and trust in the 6g era: Risks and mitigations. *Electronics*, 13(11):2162, 2024. Accessed 2025-06-06.

[47] 6G Flagship Consortium. 6g white paper: Research challenges for trust, security and privacy, 2023. Accessed 2025-06-06.