PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

University of Kasdi Merbah – Ouargla

Faculty of New Technologies of Information and Communication

Department of Computer Science and Information Technology

MASTER THESIS

to obtain the diploma of a Master's degree in Computer Science

Speciality : Administration and Network Security

Federated Learning in Internet of Medical Things (IoMT) Healthcare Applications.

Prepared by : **Semmadi Abdennour & Bahhou Tarek**

Publicly supported, on 10/06/2024, before the jury composed of :

Dr.Eushi Salah :	UKMO	- President
Dr.Messiaid Abdessalem :	UKMO	- Examiner
Dr.Boukhamla Akram Zine Eddine :	UKMO	- Thesis Supervisor

UKMO 2024


بسم الله الرحمن الرحيم

Acknowledgments

“

First and foremost, we thank God Almighty for His help and success. O God, praise be to You and thanks be to You first and last.

After that, we extend our sincere thanks to our parents, who were credited with our existence for their upbringing, support, and trust.

We also express our deep gratitude to our supervisor, ***Dr.Boukhamla Akram Zine Eddine****, for his unwavering support, wise advice and constant encouragement, which were helpful in solving our research problems and completing this thesis. We also thank the jury members for reviewing our work and our teachers for their invaluable teaching.*

Finally, we express our thanks to our family and friends, one by one, and to everyone who supported and encouraged us throughout this educational journey.

Thank you all

”

Dedication

“

On this special occasion, I dedicate my graduation to my honorable parents for their continued support for me, to the souls of my grandfathers and grandmothers, to my family, my family, my friends, my relatives, my colleagues, my teachers, my sheikhs, and everyone who has a right over us. We ask God Almighty to accept and work with this knowledge and to facilitate and guide us in the next stage, which will be full of challenges and challenges. Successes, God willing.

- Semmadi Abdennour -

I dedicate my graduation to my loving parents, whose hard work and generosity have been the cornerstone of my happiness. I also acknowledge the beauty of the world, the support of my siblings, the friendship of my dear friends, the guidance of my teachers, and the love of all my relatives. Together, they have shaped my experiences and supported me throughout, and I am immensely grateful for their presence in my life.

- Bahhou Tarek -

”

الملخص

تستكشف هذه الدراسة إمكانية التعلم الموحد لتعزيز أمن البيانات في أجهزة إنترنت الأشياء، مع التركيز على تطبيقاته في قطاع الرعاية الصحية، ولا سيما إنترنت الأشياء الطبية. حيث تعمل تقنية إنترنت الأشياء الطبية على ربط الأجهزة الطبية الذكية بالإنترنت، مما يتيح تبادل البيانات بكفاءة وتحسين العمليات. إذ يتيح التعلم الموحد تدريب نماذج الذكاء الاصطناعي على البيانات الموزعة عبر هذه الأجهزة دون نقلها إلى مراكز البيانات المركزية، مما يحافظ على خصوصية بيانات المرضى الحساسة ويقلل من مخاطر الانتهاكات. يعمل هذا النهج على تحسين جودة الرعاية الصحية بشكل كبير من خلال تمكين التبادل الآمن للبيانات وتقديم خدمات الرعاية الصحية الشخصية. وتؤكد الدراسة على أهمية دمج أجهزة إنترنت الأشياء الطبية مع التعلم الموحد لتحسين أمن البيانات والخصوصية في الرعاية الصحية، مدعومة بدراسة محاكاة توضح فعالية هذا التكامل في تقديم حلول رعاية صحية ذكية تحافظ على خصوصية و أمن البيانات.

الكلمات المفتاحية : التعلم الموحد ، إنترنت الأشياء ، إنترنت الأشياء الطبية ، أمن البيانات، الرعاية الصحية، نماذج الذكاء الاصطناعي.

Abstract

This research explores the potential of Federated Learning (FL) to enhance data security in Internet of Things (IoT) devices, with a focus on its applications in the healthcare sector, particularly the Internet of Medical Things (IoMT). IoMT technology connects smart medical devices to the internet, enabling efficient data exchange and process improvement. FL enables the training of AI models on data distributed across these devices without transferring it to central data centers, thereby maintaining the privacy of sensitive patient data and reducing the risk of breaches. This approach significantly enhances patient care quality by enabling secure data exchange and personalized healthcare services. The study underscores the importance of integrating IoMT devices with FL to improve Data security and privacy in healthcare, supported by a simulation study that demonstrates the effectiveness of this integration in delivering smart, privacy-preserving healthcare solutions.

Keywords : Federated Learning(FL), Internet of Things(Iot), Internet of Medical Things(IoMT), Data Security, Healthcare, AI Models.

Résumé

Cette recherche explore le potentiel de l'apprentissage fédéré (FL :Federated Learning) pour améliorer la sécurité des données dans les appareils Internet des objets, en mettant l'accent sur ses applications dans le secteur de la santé, en particulier l'Internet des objets médicaux . La technologie l'Internet des objets médicaux connecte les appareils médicaux intelligents à Internet, permettant un échange de données efficace et une amélioration des processus. FL permet la formation de modèles d'IA sur les données distribuées sur ces appareils sans les transférer vers des centres de données centraux, préservant ainsi la confidentialité des données sensibles des patients et réduisant le risque de violations. Cette approche améliore considérablement la qualité des soins aux patients en permettant un échange de données sécurisé et des services de santé personnalisés. L'étude souligne l'importance de l'intégration des appareils l'Internet des objets médicaux avec FL pour améliorer la sécurité et la confidentialité des données dans les soins de santé, appuyée par une étude de simulation qui démontre l'efficacité de cette intégration pour fournir des solutions de santé intelligentes et préservant la confidentialité.

Mots clés : pprentissage fédéré , Internet des objets , Internet des objets médicaux , Sécurité des données, Santé, Modèles d'intelligence artificielle

---

Table of Contents

List of Figures

List of Tables

List of Acronymes

- **IoT** : Internet of Things .
- **IoMT** : Internet of Medical Things.
- **EHRs** :Electronic Health Records .
- **SMPC** :secure multi-party computation .
- **IDS** : Intrusion Detection Systems.
- **DDoS** :A Distributed Denial of Service.
- **IA** : Intelligence Artificielle .
- **ML** :Machine Learning .
- **DL** :Deep Learning .
- **SGD** : Stochastic Gradient Descen .
- **ANN** : Artificial Neural Network .
- **CNN** : Convolutional Neural Network .
- **DNN** : Deep Neural Network .
- **RNNs** : Recurrent Neural Networks .
- **ReLU** : Rectified Linear Unit.
- **LSTM** : Long short-term memory.
- **FL** : Federated Learnin.
- **FedProx** : Federated Prox.
- **FedAvg** : Federated Averaging.
- **HFL** :Horizontal Federated Learning .
- **VFL** :Vertical Federated Learning .
- **FTL** :Federated Transfer Learning .
- **IID** :Independent and Identically Distributed.

General Introduction

Project Background

In light of the rapid technological advancements of the twenty-first century, digital technologies have become an integral part of our daily lives, contributing to the improvement of various aspects such as healthcare, transportation, industry, and education. Among these technologies, the Internet of Things (IoT) stands out as a key platform driving innovation and progress.

The Internet of Things represents a modern vision that relies on connecting smart devices to the internet, enabling them to interact and exchange data with each other and with other systems. This technology allows users to benefit from a wide range of applications and services that facilitate daily life and increase efficiency and productivity.

IoT applications span various fields, including smart homes, smart cities, healthcare, smart agriculture, and industry. Of particular interest to us in this context are its applications in healthcare, where IoT offers innovative solutions to enhance efficiency and security. It can improve medical information management, digital biomarkers, telemedicine, medication adherence monitoring, 3D scanning, and smartwatches. IoT technology also enables the creation of smart hospitals and facilities equipped with connected devices, thereby enhancing the quality and efficiency of healthcare. This technology is particularly relevant in remote areas.

Problem

Despite the significant benefits of IoT, it faces a range of challenges that hinder its widespread adoption. Among these challenges, security and privacy issues are paramount, as it becomes crucial to protect users' sensitive data from cyberattacks and breaches. Additionally, connected devices require continuous power, posing challenges in energy management and efficiency.

Proposed Solutions

Federated learning is a modern machine learning model that aims to improve security and privacy in data processing. This model relies on training AI models across multiple devices without the need to transfer raw data to a central data center.

Federated learning is particularly suitable for sectors that require sensitive data processing, such as healthcare. It allows for the development of robust models without compromising the privacy of personal data. For example, it can be used in developing medical diagnostic systems based on patient data without the need to share this data outside the hospital.

Combining the Internet of Things with federated learning is a natural step towards maximizing the benefits of both technologies. IoT can provide a vast amount of data from various sensors, while federated learning offers a secure and efficient framework for analyzing this data without compromising its privacy. This integration can lead to significant improvements in multiple fields, such as smart healthcare, where the aggregated data can be used to optimize operations and make more accurate decisions

Document Plan

This thesis is structured in 3 chapters as follows:

Chapter I: We provide an overview of the basic principles, various applications and current challenges of IoT and federal learning.

Chapter II: Delves into the technical details of the proposed integrated model, including the system's structure, data management mechanisms, security and privacy strategies.

Chapter III : presents the results of practical experiments and assessments conducted to verify the effectiveness of the proposed model in enhancing the efficiency and security of health care.

Finally, we conclude the thesis with a summary of key findings, discussion of benefits and challenges, and recommendations for future research.

Chapter 1

Overview on the Internet of Medical Things (IoMT)

1.1 Introduction

The integration of IoT in healthcare has sparked innovation, notably with the rise of the Internet of Medical Things (IoMT). IoMT links medical devices, wearables, and healthcare systems for real-time monitoring and personalized treatment. However, challenges like data security and privacy arise due to the abundance of sensitive patient data. Artificial intelligence offers solutions through advanced algorithms and applications. This chapter explores IoMT architecture, delving into AI's role in bolstering data protection. It examines various AI algorithms, discussing their benefits and challenges in healthcare. Additionally, future directions for AI in IoMT security and privacy are explored, aiming to foster healthcare innovation and trust among patients and stakeholders.

1.2 Definition of IoT

The internet of things, or IoT, is a network of linked objects that communicates with other IoT devices and the cloud to share data. IoT devices typically include consumer goods, digital and mechanical equipment, as well as hardware such as sensors and software [1].

1.3 Internet of Medical Things (IoMT)

The Internet of Medical Things (IoMT) refers to the interconnected network of medical devices, equipment, and software applications that utilize internet connectivity. This network enables the exchange of data and information among healthcare systems, devices, and providers, allowing for remote monitoring, diagnostics, and healthcare management. IoMT plays a crucial role in advancing healthcare by enhancing patient care, improving efficiency, and enabling innovative medical solutions [2].

1.4 IoMT Architecture

Most modern IoMT systems typically comprise four layers, covering the entire data lifecycle from biometric data collection to storage and visualization for physician analysis. The cloud layer also enables patients to access and view their overall health status [2].As shown in the following figureFigure 1.1 :

- **Sensor Layer:** This layer is the central part of the IoMT system, patient data from a range of sensors and transmitted to the portal or cloud. Triggers, controllers and sensors form their devices, ensuring accurate identification of health-related variables [3].
- **Gateway Layer:** The layer following the sensor layer, termed as the fog or edge layer, includes local servers, gateway devices, and transmission and service sublayers. It facilitates real-time data transfer, integrates diverse networks, and executes data preparation in edge computing, distinct from the cloud layer [4].
- **Cloud Layer:** The cloud layer in healthcare systems stores data from medical devices, conducts machine learning tasks, and facilitates epidemiological research. It offers a graphical interface for visualization and enables access to valuable insights for service providers and caregivers [4].

- **Application Layer:** The action layer in IoMT manages medical records using various applications and devices, divided into medical information and medical decision-making sublayers [4].

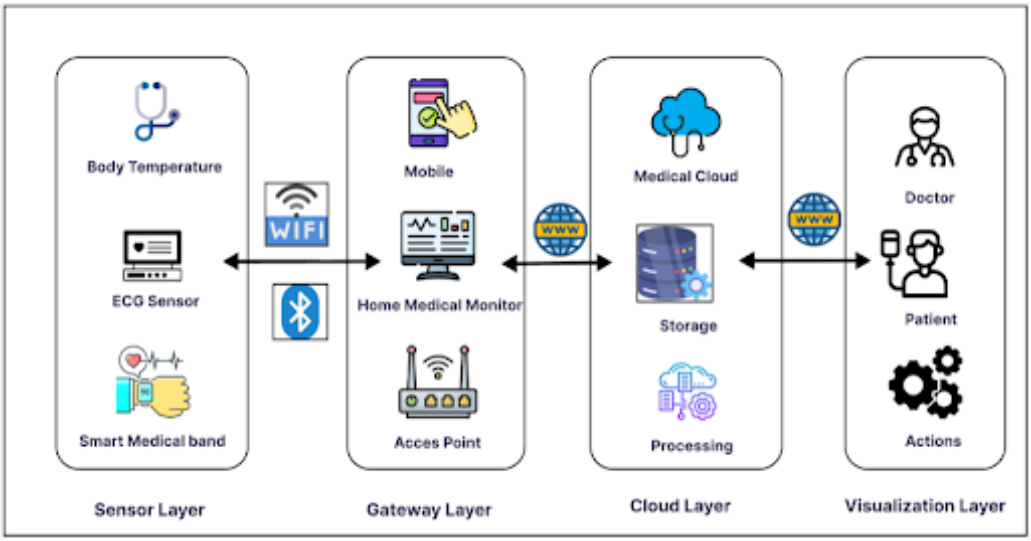

Figure 1.1: IoMT Architecture .

1.5 Types of IoMT Devices

In the realm of the Internet of Medical Things (IoMT), devices vary widely in their functionalities and applications, catering to different aspects of healthcare delivery and monitoring. Here are some common types of IoMT devices [5]:

1. **Wearable Health Trackers:** These devices are designed to be worn by individuals and typically monitor vital signs such as heart rate, blood pressure, activity levels, and sleep patterns. Examples include fitness trackers, smartwatches with health monitoring capabilities, and medical-grade wearable sensors .
2. **Home-based IoMT Devices:** This category includes a broad range of medical equipment, such as ventilators, infusion pumps, feeding assistance, emergency kits, therapeutic devices, baby care accessories, and diagnostic tools. Although these gadgets are meant to be used outside of conventional healthcare settings, they can link online to hospital-based gadgets and medical experts.
3. **Hospital and Clinical IoT Devices:** These devices are deployed within healthcare facilities to optimize operational efficiency, improve patient care, and enhance safety. Examples include asset tracking systems, smart beds, medication management systems, and IoT-enabled imaging equipment .

1.6 Protocols for Using IoMT in Interaction

The increasing proliferation of smart devices in IoT, especially in IoT systems, requires effective connectivity protocols. Various technologies such as ZigBee, Bluetooth and Sigfox have been developed to meet specific needs such as low consumption and wide range. Maintaining reliable communication between assorted devices requires an adaptable layered design. Different

approaches to IoT architecture have been proposed, often based on three basic layers similar to the OSI model. The following table summarizes the initial protocols used in IoT systems across these layers [3] .

Table 1.1: The primary protocols used in IoMT systems at various levels.

Layer	Communication Protocols	Range Type	References
Perception Layer	PersonnelleRFID, NFC, Bluetooth/BLE, Z-Wave, UWB	Short-Range	[17] [18]
Network Layer	- IPV4, IPV6 protocols (for network addressing) - TCP, UDP, 6LoWPAN, WIA-PA - For routing RPL, CARP, and CORPL protocols (for network routing)	N/A	[19] [20]
Network Layer	- IEEE 802.15.4 (ZigBee) - IEEE 802.15.1 (Bluetooth) - IEEE 802.11 (Wi-Fi) - NFC, RFID	Short-Range	[19] [20]
Network Layer	LPWAN (LoRaWAN and LTE-M)	Long-Range	[19] [20]
Application Layer	HL7, CoAP, DSS, MQTT, HTTP, HTTPS, TLS	N/A	[21] [22]

1.7 Applications of IoMT in Healthcare

The Internet of Medical Things (IoMT), has many useful and impactful applications in the healthcare field. These consist of :

- **Digital hospital:** The subject of medical information management offers a wide range of applications for the Internet of Things. Hospitals are currently in need of medical information management for identification, sample identity as well as recognition in medical records. Hospitals provide healthcare, which is one way the medical industry uses IoT. IoMT is making hospital medical work more intelligent, exacting, and productive [6].
- **Digital biomarkers:** These tools allow for real-time patient diagnosis and the early detection of illnesses before they manifest clinically. The health of animals can also be protected by using these sensors. A continuous monitoring of the patient's health can help with understanding and timely care [7].
- **Telemedicine and Virtual Consultations:** IoT facilitates telemedicine by enabling video conferencing, remote consultations, and virtual visits between patients and healthcare providers. This technology allows patients to access healthcare services from the comfort of their homes, particularly beneficial for individuals in remote or underserved areas, and reduces the need for in-person visits, saving time and resources for both patients and providers [6].
- **Medication Adherence Monitoring:** IoT devices can monitor patients' medication ad-

herence by tracking pill bottle usage, dispensing medication reminders, and sending alerts to patients or caregivers if doses are missed. This helps improve medication adherence, reduces the risk of medication errors, and enhances treatment effectiveness, especially for patients with chronic conditions [7].

- **3 dimensional scanning and printing :** In order to replicate hard and soft tissue characteristics in the mouth accurately and quickly, IoMT-based 3D scanners digitally record intraoral impressions. This eliminates the hassle that comes with using traditional impression materials [9].
- **Smart watches for depression :** There are a lot of people who fall victim to depression every year, and smart watches can help by guiding and counseling the patient on depression management techniques [7].
- **Smart Medication:** With electronic pharmaceuticals, a mobile sensing patch or other external body sensor is connected to an implantable sensor through the use of web pages or smartphone applications. The server houses the data.is used, for example, in individuals with chronic illnesses, to monitor the cardiovascular system. The mobile application can also be used to remind the user to take prescribed medication on time and to share information with family members or caretakers [7].
- **Body coagulation testing :** The IoT is used in the healthcare industry to periodically check blood coagulation levels, giving medical professionals useful information on patient behavior and assisting in the efficient tracking and management of their patient's condition. Continuous monitoring makes it possible to intervene when necessary and modify treatment regimens, which eventually improves patient outcomes and promotes healing [9].
- **Smart Hospitals and Healthcare Facilities:** IoT technology enables the creation of smart hospitals and healthcare facilities equipped with interconnected devices and systems for efficient patient care and management. Examples include smart beds that monitor patients' movements and vital signs, smart infusion pumps that deliver medication doses accurately, and environmental monitoring systems that ensure optimal conditions for patient comfort and safety [7].

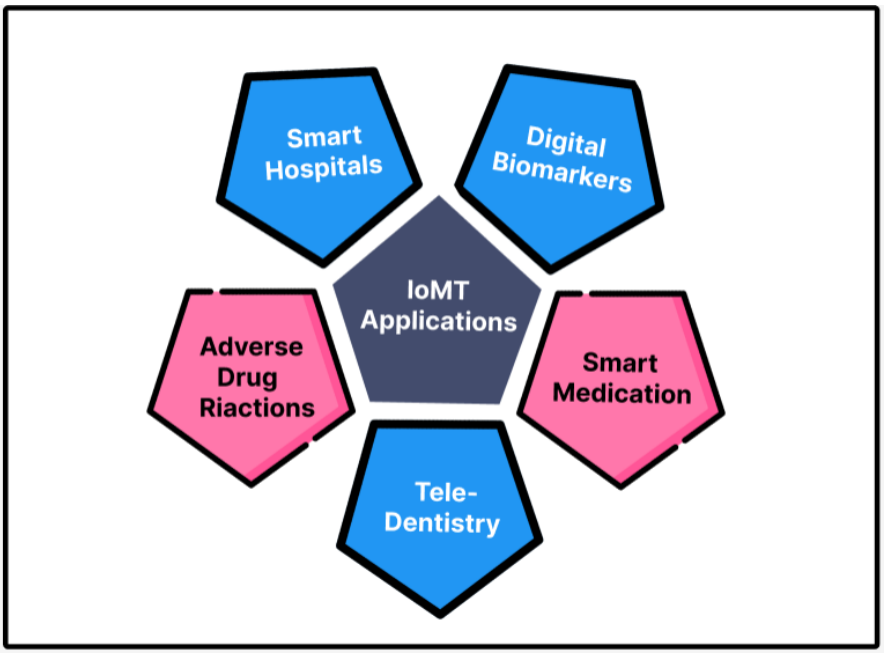

Figure 1.2: Applications in the IoMT Domain .

1.8 Security Requirements for IoMT

Security is paramount for medical devices in wireless and remote healthcare services. Weaknesses in authentication and access control can lead to data leaks in IoMT devices, posing significant risks. Most IoMT devices lack the ability to detect and prevent attacks independently, necessitating security measures at strategic network points. Therefore, security requirements for IoMT healthcare networks, including CIANA considerations, are outlined to ensure Confidentiality, Integrity, Availability, Non-Repudiation, and Authentication , We talk about the needs in the context of CIANA and other factors that are explained below [10].

1. **Confidentiality:** in relation to IoMT stands for safeguarding shared medical data between patients and healthcare providers. It is necessary to safeguard this data against unauthorized access, listening in, and improper use. Network access control and data encryption are required to maintain confidentiality in IoMT, even though standards offer basic guidance [10].
2. **Integrity:** is to protect against unwanted changes in data while transmitted, ensuring health and reliability in the context of IoMT. IoMT devices need protection from physical threats in order to protect patient safety and ensure high quality care. Unauthorized data manipulation poses serious risks to patients' outcomes through inaccurate diagnostics and inadequate treatment [4].
3. **Availability :** The capacity to maintain uninterrupted operation of IoMT systems is known as availability. It can be accomplished by maintaining the structure's current state, monitoring any changes in performance, providing backup storage or transmission channels in case of denial-of-service attacks, and swiftly fixing any problems [2].
4. **Non-Repudiation :** In order to avoid the cancellation of earlier agreements or activities, non-repudiation makes sure that authorized users are unable to retract their acts within a system. Assuring that entities accept accountability for their actions or repercussions, it verifies the reality or non-existence of an activity. The criterion can be satisfied by using digital signature techniques [10].
5. **Authentication :** Whereas message authentication verifies the source of the data, authentication in IoMT systems guarantees user identity verification during login. The safest type of authentication, known as mutual authentication, requires client and server authentication prior to data transfer. Because of memory and CPU limitations in IoMT devices, lightweight authentication algorithms are becoming more popular as a solution to classic protocols' cryptographic operation problems [11].
6. **Authorization :** As mentioned before, medical data is sensitive and must be shielded from unwanted access. Therefore, certain tasks, like giving commands to medical IoMT devices or upgrading their software and applying security updates, should only be allowed to be carried out by trustworthy individuals who have the necessary training or experience [2].
7. **Anonymity :** This stipulation guarantees the confidentiality of patient and physician identities during unapproved users' interactions with the system . It is not appropriate for a patient or a doctor to reveal their identities when communicating. The extent of passive attacks is observation of behavior without person identification [11].
8. **Privacy :** The IoMT system makes sure that patients' personal information is shielded from misuse and illegal access. The IoMT system conforms with regulations like GDPR (General Data Protection Regulation) and HIPAA (Health Insurance Portability and Accountability Act), which regulate the gathering and storage of health data and allow users to access private information securely [11].

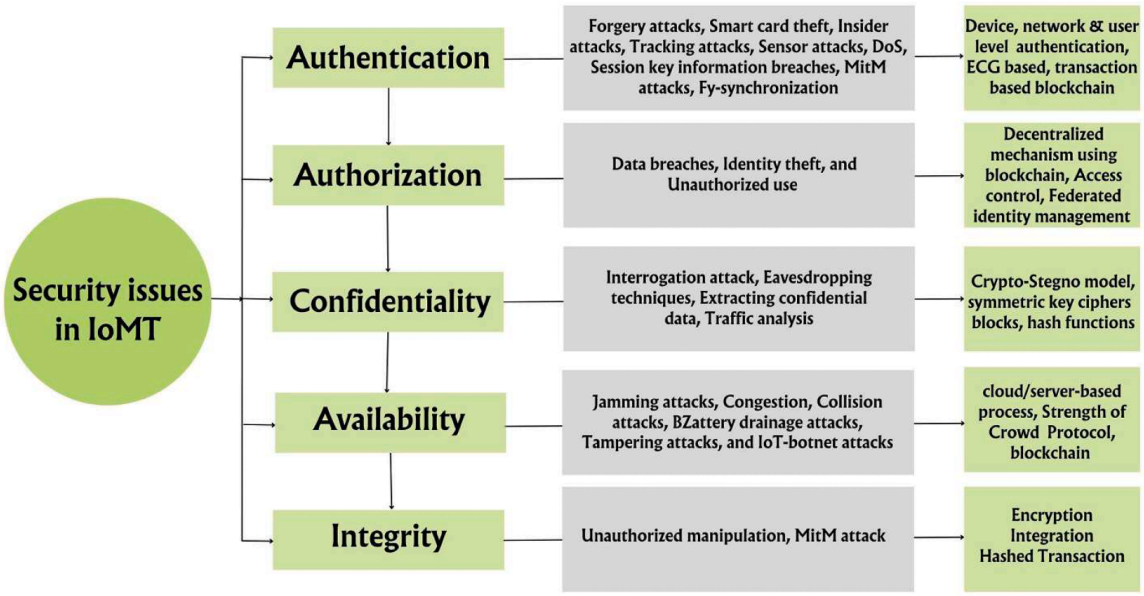

Figure 1.3: Key security issues in IoMT [4].

1.9 Categorization of Attacks in IoMT

IoT protocols frequently lack crucial security measures, which has resulted in security vulnerabilities caused by the growth of IoMT applications. A breach of these systems could lead to serious financial and reputational harm as well as compromise patient privacy [3].These attacks have been classified according to the layers that make up the device, as follows:

1.9.1 Sensor Layer

Common attacks in this layer include the following :

- **Tag cloning :** This attack allows attackers to exploit data generated by side-channel attacks or data cloned from previously used tags, such as Radio Frequency Identifiers (RFID). Using relatively simple techniques, this cloned data may allow access to sensitive information, such as medical records or unauthorized facilities [28].
- **Tampering devices :** This attack is used to alter data stored on an IoMT device, usually physical access is required. This modification can occur through methods such as manipulating the RFID tag or accessing the device via a communication link [27].
- **Sensor tracking :** Patient privacy may be jeopardized by insecure equipment, which gives hackers access to private information such as GPS data that has been manipulated or patient locations. Furthermore, sensors used in healthcare applications could be tampered with to reveal private patient data [29].
- **Side-channel attack :** Side-channel attacks exploit unintended information leakage in cryptographic systems, inferring sensitive data like cryptographic keys. They are sophisticated and difficult to detect, requiring robust countermeasures to protect against [25] [26].

1.9.2 Network Layer

Common attacks in this layer include the following :

- **DoS/DDoS :** A distributed denial of service (DDoS) attack involves multiple sources flooding a designated target with messages or connection requests in an attempt to prevent legitimate users from accessing the service, in contrast to denial of service (DoS) attacks, which are executed by a single node [32] [33] .
- **Man-in-themiddle :** This cyberattack aims to access the private data of two IoMT devices by interfering with their communication. In this attack, the attacker has the ability to listen in on or watch over the communication between the two devices.Before the intercepted data is sent to its intended location, the attacker can change it [31].
- **Eavesdropping :** In order to obtain data, an attacker tracks and intercepts the required hardware and communication. There are many uses for data that is obtained in this way (illegally) [28].
- **Selective Forwarding :** Selective forwarding is a wireless attack where a compromised node selectively forwards or drops certain packets, disrupting communication and compromising data integrity. It's particularly dangerous in hostile environments, and secure routing protocols and cryptographic techniques can help mitigate its impact [30] [35].

1.9.3 Application Layer

Common attacks in this layer include the following :

- **SQL injection :** An SQL injection attack happens when an application's backend database is compromised by a malicious SQL statement. Should this attack be successful, private patient information kept in the database may be jeopardized or changed [34] [38].
- **Brute Force :** Automated tools are frequently employed by attackers to repeatedly try different password combinations until they succeed. One particularly dangerous flaw in IoMT devices is the dictionary attack, which presents a serious security risk [36] [37].
- **Ransomware:** Ransomware is a type of malware that encrypts data and demands payment in order to unlock it. This allows attackers to store sensitive data, including medical records, until the ransom is paid [23].
- **Account hijacking:** IoT devices frequently use insecure encryption or transparent text for communication, which makes it possible for hackers to intercept packets sent during user authentication and take control of an account [24].

1.9.4 The Effects of These Attacks on Security Factors

These attacks can also be classified in terms of the security factors that affect them, and we have summarized this in the following table :

Table 1.2: The effects of these attacks on security factors

Attack	Affects	Type Layer	Ref
Tag cloning	Confidentiality , Authorization and Integrity	Sensor	[28]
Tampering devices	Confidentiality, Integrity	Sensor	[27]
Sensor tracking	Confidentiality, Authorization, Integrity and Privacy	Sensor	[29]
Side-channel attack	Confidentiality and Integrity	Sensor	[25] [26]
DoS/DDoS	Availability	Network	[32] [33]
Man-in-themiddle	Confidentiality and Authorization	Network	[31]
Eavesdropping	Confidentiality , Non-repudiation and Privacy	Network	[28]
Selective Forwarding	All	Network	[30] [35]
SQL injection	All	Application	[34] [38]
Brute Force	Confidentiality and Integrity	Application	[36] [37]
Ransomware	Integrity and Availability	Application	[23]
Account hijacking	Confidentiality and Integrity	Application	[24]

1.10 Challenges of IoMT Healthcare

The implementation of IoMT in healthcare is accompanied by several challenges, including :

1.10.1 Management and Equipment Challenges :

- **Data Quality and Quantity:** AI algorithms rely on large volumes of high-quality data for training and optimization. However, ensuring the availability of sufficient and diverse data sets from IoMT devices, electronic health records (EHRs), and other sources while maintaining data quality, consistency, and relevance poses challenges. Data may be incomplete, inaccurate, or biased, leading to suboptimal AI model performance and unreliable predictions [12] .
- **Data management :** The capacity to access, integrate, regulate, and oversee the flow of data information is known as data management. Only information that is useful for application and usage is provided via data filtering techniques including data synchronization, data integration, and data anonymization/privacy [12].
- **Environmental Resiliency :** Concerning environmental resilience, wearable or portable smart device sensing techniques can be applied both indoors and outdoors. Additionally, anticipated problems include using in the rain or bad weather are not thoroughly investi-

gated since these factors are out of our control. Systems may be made more dependable and research more repeatable by addressing these problems [7].

- **Energy Efficiency :** The device's usefulness, durability, and size and weight are some of the factors that limit how effectively energy is used. Therefore, a suggested IoMT technique need to lessen the IoMT devices' power footprint as well. For example, implanted devices should have a minimum battery life of ten to fifteen years in order to reduce the need for recurring surgery. The use of smart wearables is reduced when batteries need to be changed frequently [7] .
- **Power usage :** Another issue preventing IoMT devices from being used more frequently is power consumption. Since the majority of IoMT devices run on batteries, installing sensors necessitates either regular battery changes or the usage of high-power batteries. The current focus should be on creating energy-efficient healthcare devices that can run on their own or integrating renewable energy systems into the IoMT system to help alleviate the world energy issue [11] .
- **Interoperability and Integration:** Integrating AI applications with existing IoMT devices, healthcare IT systems, and clinical workflows requires interoperability standards, data exchange protocols, and seamless integration with electronic health records (EHRs). Achieving interoperability between disparate systems, platforms, and data formats is challenging due to the lack of standardized interfaces, communication protocols, and data models [11] .
- **Exorbitant costs of infrastructure:** A large initial investment is required due to the expense of the hardware, specialized IoMT IT infrastructure, cloud computing, and developing a consumer-facing app. The high infrastructure costs are an impediment to IoMT even though the ultimate return on investments is certain [12] .
- **Network Bandwidth:** The large volume of data generated by IoMT devices can strain network bandwidth, requiring upgrades to handle the increased traffic [12] .
- **User Acceptance and Training:** Healthcare professionals need proper training to effectively use and interpret data from IoMT devices. Building trust and understanding of this technology is essential for successful implementation [11] .
- **Scalability, upgradation, regulations and standardization :** Scalability is the capacity of a medical device to adjust to changes in its surroundings. Consequently, a highly scalable system is one that can operate smoothly and instantly while preserving consistency across all linked devices. A system that is highly scalable is more useful now and in the future. The necessity for frequent updates to the current device has arisen due to the ongoing development and progress of IoMT technology. In today's fast-paced world, this is still a difficulty [11].

1.10.2 Privacy and Security Challenges :

Data security and privacy issues in the IoMT environment are one of the challenges that require comprehensive strategies to effectively address potential problems. It is classified into:

1. **Vulnerabilities (Weaknesses):** IoMT devices may possess security weaknesses that malicious actors could exploit, acquiring unauthorized entry to confidential information or disrupting device operations [13].
2. **Protecting Sensitive Data :** Patient data collected by IoMT devices is highly sensitive, including medical records and personal information. Robust security measures are crucial to safeguard this data from cyberattacks and breaches.
3. **Multiple Entry Points :** The interconnected nature of IoMT creates more potential

vulnerabilities for hackers to exploit. Strong authentication and encryption protocols are essential to secure every access point.

4. **Data Privacy Concerns :** Information security in smart healthcare is growing increasingly problematic as a result of a rise in data security breaches. The goal of an attack is to obtain data without authorization so that it can be used against patients and medical personnel [7] .
5. **Threats and risks to data security :**

   Cyberattacks have a great potential to compromise healthcare data. Enhancing the current therapeutically relevant medical data pool with IoMT data adds a substantial risk of exposure. Data breaches are becoming more likely as more devices are linked to other systems and to one other [12]. The security of the medical Internet of Things is exposed to significant risks that require taking diligent measures to ensure patient safety and data integrity. These risks and threats are [13]:

   - **Patient Safety:** The compromised security of IoT devices may result in misdiagnoses, incorrect treatment administration, and potentially life-threatening outcomes for patients.
   - **Data Breaches:** Breaches exposing sensitive patient information could lead to identity theft, fraudulent activities, and violations of patient privacy.
   - **Ransomware Attacks:** Incidents of ransomware could disrupt healthcare operations by locking providers out of essential systems until ransoms are paid, potentially impacting patient care.
   - **Malware Attacks:** Malicious software has the capability to infiltrate and undermine the functionality of IoMT devices, thereby affecting their accuracy and efficacy.
   - **Device Hijacking:** Unauthorized access to control medical devices poses serious risks as it can result in unauthorized manipulation, endangering patient well-being.

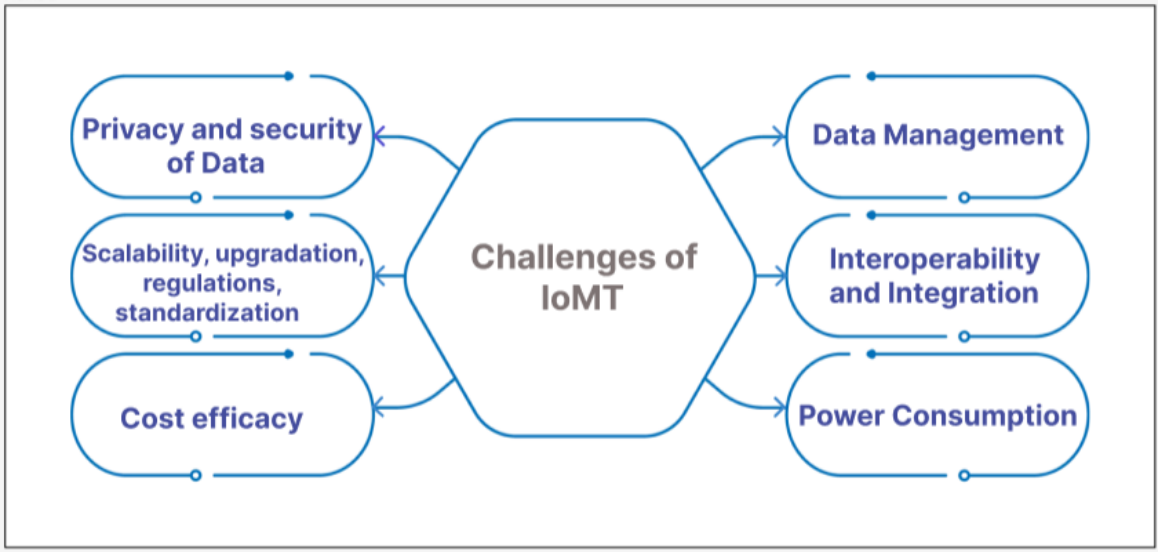

Figure 1.4: Challenges of IoMT Healthcare .

1.11 Future Trends for Data Security and Privacy in IoMT

Future studies on IoMT security and privacy may examine unexplored possibilities . Here are a few possible avenues for further research:

1.11.1 Security Assessment :

There are currently no standardized techniques for evaluating the efficacy of suggested solutions in IoMT security research. The assumptions and guiding principles of adversarial analyses, which are frequently employed in evaluation, make comparison difficult. It is essential to create a framework for assessing privacy and security levels. Initiatives such provide recommendations, but they do not evaluate the body of research or offer crypto-analysis. To guarantee the efficacy of IoMT security measures, future research should concentrate on thorough assessment techniques [15].

1.11.2 Blockchain :

Medical data can be securely stored in a decentralized manner across medical servers with the help of blockchain technology, which was first developed for secure financial ledger records. This improves IoMT healthcare systems' security and privacy. However, due to computational resource requirements, implementing blockchain on IoMT devices with limited resources may present difficulties. However, as demonstrated by initiatives, that concentrate on medical data access and permission management, blockchain is still capable of securing electronic health records kept on medical servers [15].

1.11.3 Privacy-Preserving Technologies:

AI model training and safe data sharing are made possible by privacy-preserving technologies like Federated learning and blockchain, homomorphic encryption, and differential privacy. These technologies also protect patient privacy. These developments enable decentralized data processing without revealing personal medical information or jeopardizing data privacy [15].

1.11.4 Artificial Intelligence :

1. **Trusted AI and Machine Learning for IoMT:** AI and machine learning technologies, which enable threat detection, anomaly detection, and behavior analytics, will be beneficial to real-time cybersecurity for IoMT systems. These tools assist in spotting questionable activity, anticipating security problems, and automating countermeasures to deal with changing cyberthreats [15].
2. **Deep Learning enhances IoMT:** Data analysis is essential in IoMT systems to find vulnerabilities and boost efficiency. Blockchain-enabled IoMT systems could benefit from new DL algorithms that can learn from raw data, unlike traditional ML methods that require extensive feature engineering. The various IoMT data types can be handled by DL techniques, which use RNN for time-series data and CNN for images. For effective data analytics in the future, blockchain-based IoMT systems might need to use several different DL algorithms [16].
3. **Possibility of Improving Privacy and Security:** Deep learning techniques, which are frequently employed in medical servers to diagnose illnesses, have the potential to enhance security and privacy in IoMT systems [15].
4. **AI-powered Federated Learning Security:** Federated learning with homomorphic encryption and secure multi-party computation (SMPC) are two new AI techniques that have the potential to improve federated learning privacy. Additionally, by examining

updates within the federated learning framework itself, AI can assist in identifying compromised IoMT devices [15].

1.12 Conclusion

In conclusion, Internet of Medical Things (IoMT) integration brings numerous benefits to healthcare, from improved diagnostic accuracy to simplified administrative tasks and improved patient engagement. However, challenges such as data security, privacy concerns, and interoperability issues must be addressed to fully realize the potential of the Internet of Things (IoMT). By working collaboratively, stakeholders can develop effective solutions and standards to overcome these obstacles, ensuring that IoMT optimally serves patient well-being while maintaining data integrity. Ultimately, through concerted efforts, IoMT technology holds the promise of revolutionizing healthcare delivery and providing better outcomes for patients around the world. In the next chapter, we will discuss one of the solutions used to reduce the challenges of data security and privacy using artificial intelligence, known as federated learning.

Chapter 2

Federated Learning

2.1 Introduction

In today's world, technology is rapidly advancing, leading to an increased reliance on digital applications in our daily lives. This digital transformation has made the use of artificial intelligence (AI) crucial for staying abreast of developments, enhancing processes, and delivering services more effectively. Machine learning, a key component of AI, enables systems to learn from data independently, making smarter decisions and executing tasks more efficiently. However, AI technologies face challenges, notably regarding data privacy. With the exponential growth in personal data collection, concerns about data security and confidentiality have heightened. Traditional approaches to data analysis and machine learning struggle to protect privacy, as they often require centralized data processing, increasing the risk of unauthorized access and data breaches. Collaborative learning, such as federated learning, offers a promising solution by allowing model training across distributed devices without sharing raw data. This decentralized approach preserves the privacy of sensitive information and mitigates the risks associated with centralized data processing.

2.2 Artificial Intelligence Overview

The world is witnessing a massive technological revolution driven by artificial intelligence. It is one of the foremost fields of scientific research in recent decades, aiming to develop systems endowed with intelligent capabilities akin to human learning, reasoning, and problem-solving. Artificial intelligence boasts tremendous advantages, making it a subject of widespread interest across various sectors. Its effectiveness has been demonstrated in numerous practical applications, such as image classification, speech recognition, autonomous vehicles, and computer vision.

Artificial intelligence techniques rely on sophisticated algorithms, such as machine learning and deep learning. These techniques enable computer systems to learn patterns from data and make decisions based on them. Federated learning, a novel approach to machine learning, allows models to be trained on distributed data across multiple devices without the need to aggregate the data in one location.

2.2.1 Machine learning :

Machine learning encompasses the creation of algorithms enabling computers to learn from data and make predictions without explicit programming. It includes supervised, unsupervised, and reinforcement learning. Supervised learning involves training models on input-output pairs to predict outputs for new inputs [45]. Unsupervised learning doesn't require labeled input data, operating solely on input data without output labels [45]. Reinforcement learning focuses on learning action strategies based on environmental observations and feedback. [48]

2.2.2 Deep Learning:

Deep learning, a branch of artificial intelligence, utilizes multi-layered artificial neural networks to learn complex data representations. It has shown significant success in tasks such as speech recognition, image recognition, and natural language processing. Within deep learning,

algorithms and techniques are employed in the field of machine learning and artificial intelligence. These algorithms use Deep Neural Networks (DNNs) to learn data and extract patterns and information from it. Among the famous algorithms in this field are:

2.2.2.1 Convolutional Neural Networks (CNNs):

which play a crucial role, especially in computer vision applications. CNNs consist of convolutional layers, pooling layers, and fully connected layers, each performing specific functions. Training involves forward and backward stages to adjust parameters based on computed gradients [46].

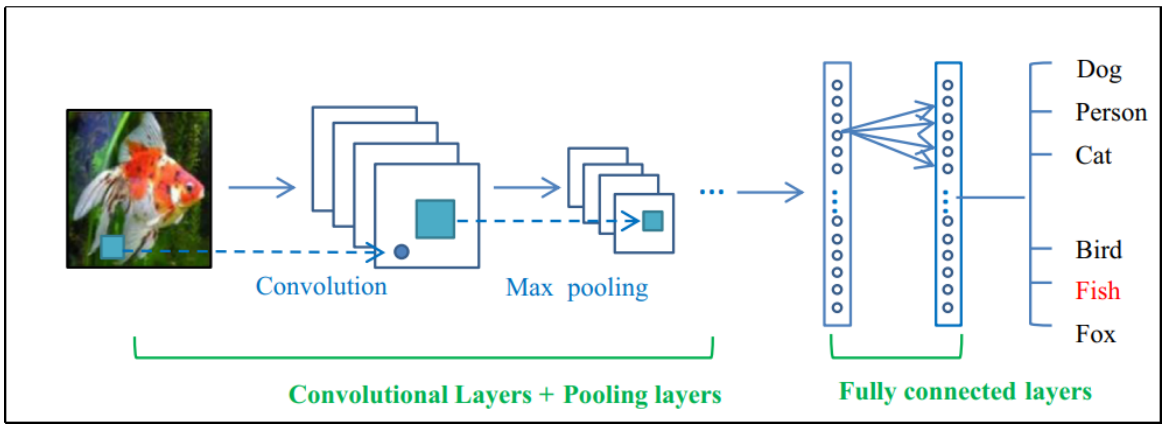

Figure 2.1: The general CNN architecture [46]

2.2.2.2 Recurrent Neural Networks (RNNs):

They excel in processing sequential data, making them valuable for sound and language taskse. RNNs leverage contextual information within data sequences for better understanding [47].

2.2.2.3 Long short-term memory (LSTM):

Deep learning uses (LSTM) a kind of artificial intelligence (DL) that is an improved form of a recurrent neural network (RNN) architecture. The LSTM is designed to avoid long-term dependency and contains feedback links. It has the ability to process entire knowledge sequences in addition to individual data pieces. For example, LSTM can be used for tasks like speech recognition, unsegmented data, connected recognizing patterns, anomaly detection in network traffic, or intrusion detection systems (IDS) [53].

2.3 Definition of Federated Learning (FL)

The concept of Federated Learning (FL) was first introduced by Google in 2016, where multiple devices collaborate to learn a machine learning model without sharing their private data under the supervision of a central server. This approach offers significant opportunities in critical domains such as healthcare, finance, and others, where sharing private user information is risky [39].

2.4 The Mechanism of Federated Learning

Federated learning is typically conducted in five steps, which are [40] :

- **Step 0: Initialize global model.**
  First, we configure the model on the server. We set the initial values for the model parameters, either randomly or by loading them from a previously saved checkpoint. This procedure is similar to that of traditional centralized learning.

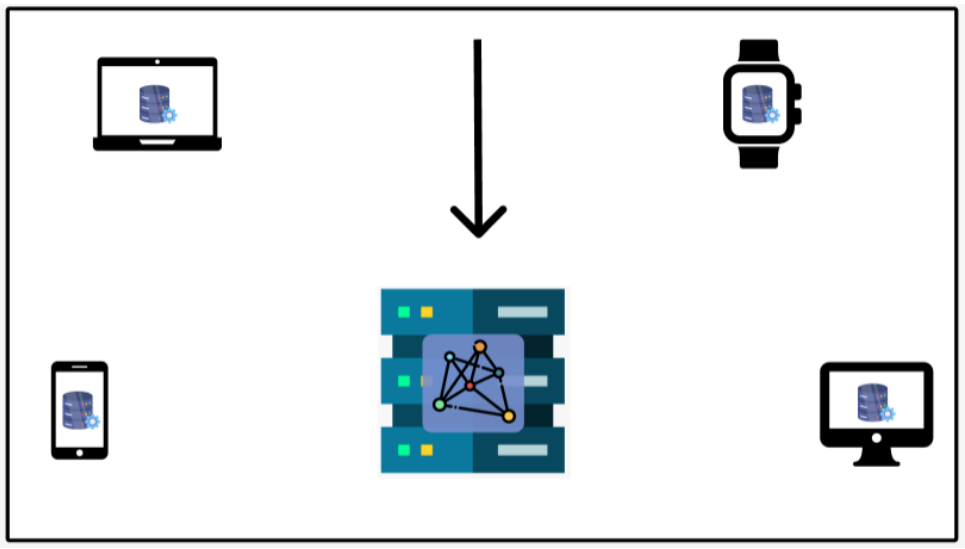

Figure 2.2: Initialize global model

- **Step 1: Send model to a number of connected.**
  organizations/devices (client nodes) In federated learning, the data handling differs as, after initializing the model on the server, a set of copies of this model (model parameters) is sent to the connected clients where the data resides. Each model is then trained locally with each dataset using the same model parameters. This is unlike machine learning, which gathers data from clients. Thus, federated learning preserves data privacy and does not incur storage costs [40].

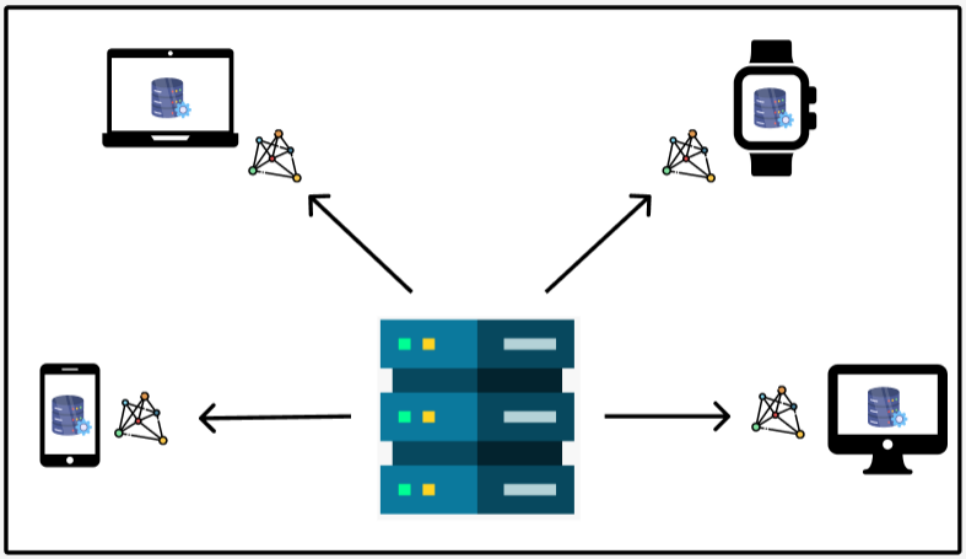

Figure 2.3: Send model to a number of connected.

- **Step 2: Train model locally on the data of each.**
  organization/device (client node) After all selected client nodes obtain the latest global model parameters, they commence local training using their respective local datasets. Instead of training the model until full convergence, short training sessions are conducted. These sessions may be limited to just one training epoch on the local data or even just a few steps (mini-batches) [40].

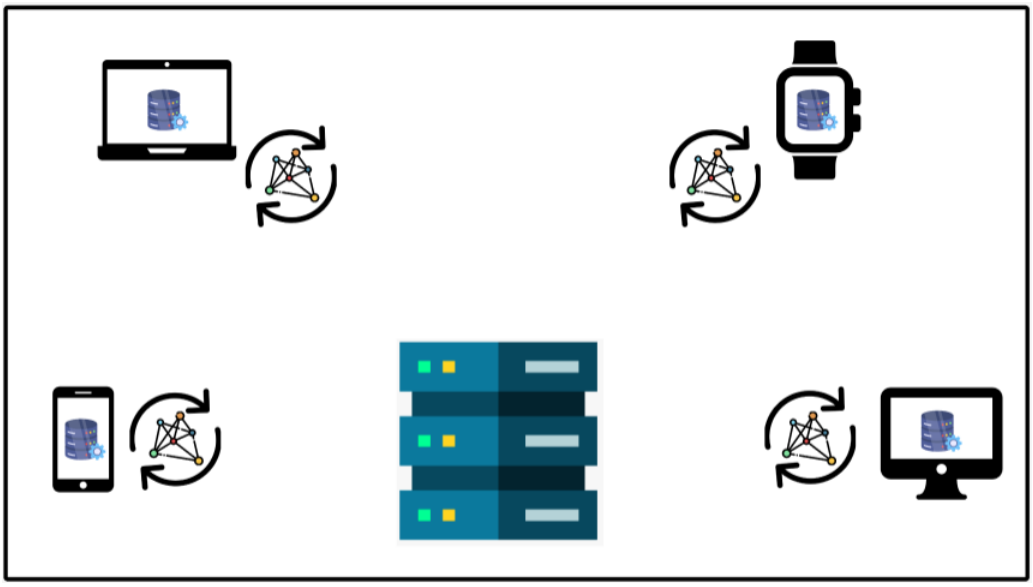

Figure 2.4: Train model locally on the data of each.

- **Step 3: Return model updates back to the server.**
  Each client node has a slightly different version of the original model parameters it was given after local training. Because every client node contains unique instances in its local dataset, the parameters are completely different. Subsequently, the client node relays these updates to the server once more [40].

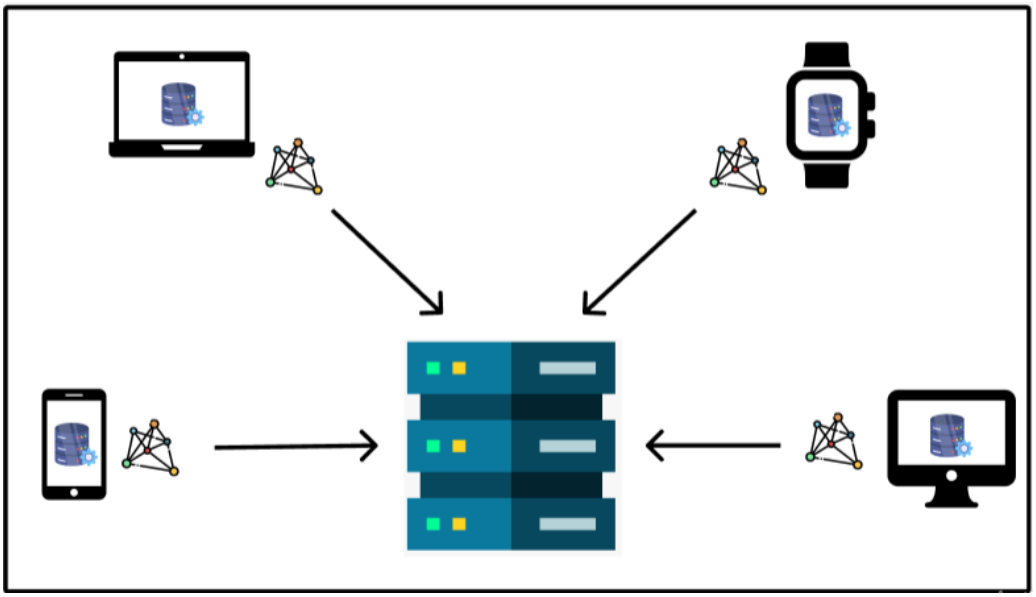

Figure 2.5: Return model updates back to the server.

- **Step 4: Aggregate model updates into a new global mode.**
  After the server receives model updates from the client nodes, these updates are aggregated to obtain a single model. This process is known as aggregation and is performed

using federated learning algorithms, with one of the simplest and most well-known algorithms being Federated Averaging. In this process, the average of the model updates received from all client nodes is calculated [40].

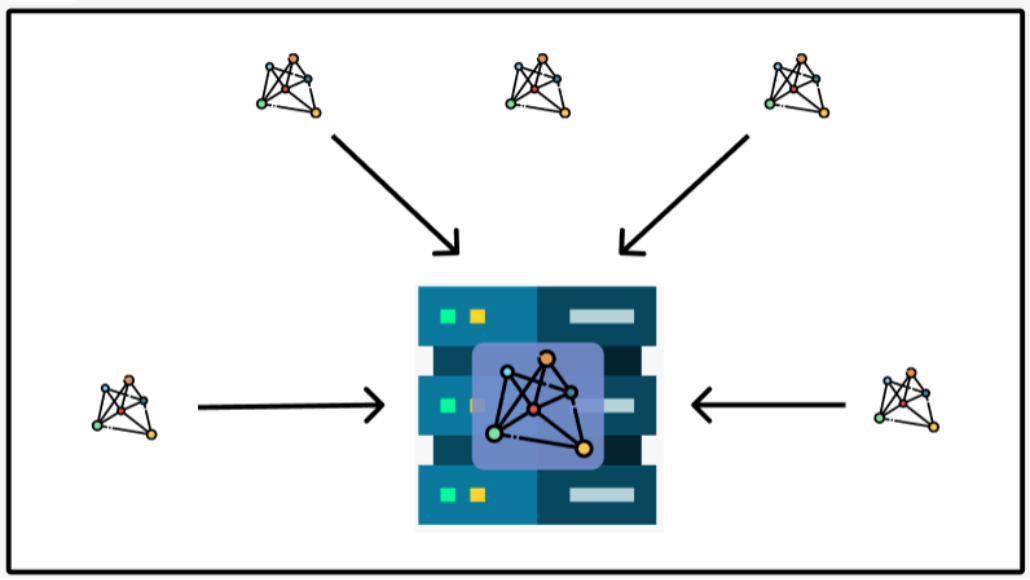

Figure 2.6: Aggregate model updates into a new global mode.

- **Step 5: Repeat steps 1 to 4 until the model converges.**
  The process is repeated (sending model copies to client nodes for training and then returning the copies to the server for aggregation) until we have a complete model for use, as training occurs through repeated short cycles until full convergence is achieved [40].

2.5 Types of Federated Learning

Federated Learning (FL) encompasses various types and approaches, each tailored to specific scenarios, challenges, and objectives. Here are some key types of Federated Learning:

2.5.1 Horizontal Federated Learning :

When data is dispersed over multiple nodes with similar properties but distinct sample spaces, it is referred to as horizontal federated learning (FL). The main focus of current FL algorithms is on IoT and smart device applications. This kind of FL is commonly observed in situations where the data may have similar feature space but considerable sample space differences. For example, because of the consistent feature dimension in the data, Google's federated model solution for Android mobile phone updates reflects a form of horizontal FL. [50].

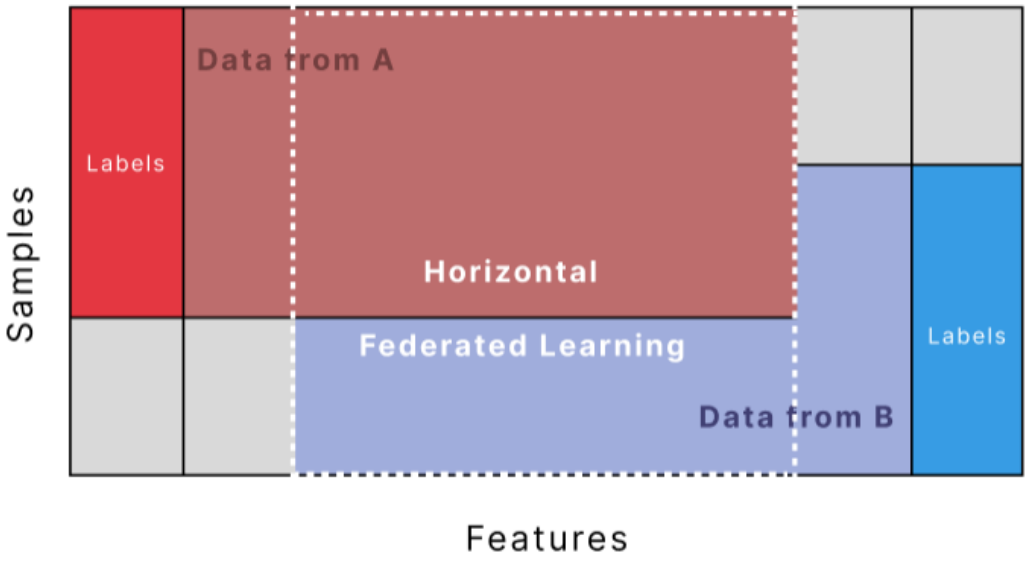

Figure 2.7: Illustration of Horizontal Federated Learning .

2.5.2 Vertical Federated Learning:

Vertical Federated Learning is a method used when data is divided vertically by feature dimension across multiple parties, each possessing similar yet partially overlapping datasets. For instance, in a medical scenario aiming to predict illnesses like type 2 diabetes, factors like age, weight, and medical history are considered. Vertical FL allows collaboration without sharing raw data, enabling analysis of additional data sources like smartphone apps tracking steps or diet. However, compared to Horizontal FL, it faces challenges in entity resolution and model aggregation due to differing data characteristics among owners. Techniques like token-based entity resolution and additive homomorphic encryption have been proposed for preprocessing and securing data in vertical FL. While current applications focus on simple models like logistic regression, there's still scope for improvement to extend vertical FL to more complex machine learning approaches [50].

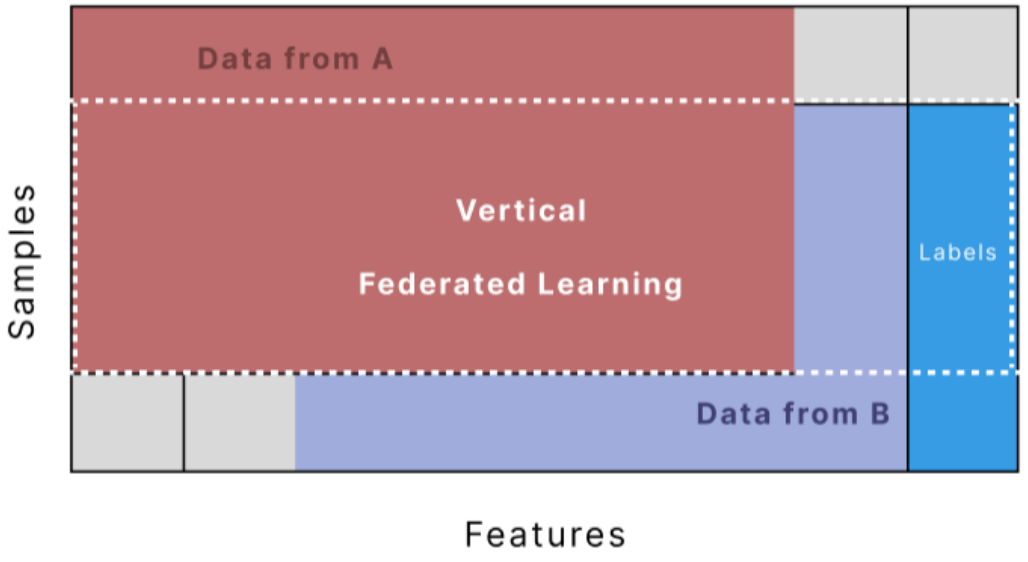

Figure 2.8: Illustration of Vertical Federated Learning .

2.5.3 Federated Transfer Learning:

Federated Transfer Learning (FTL) addresses scenarios where data lacks common sample or feature spaces, presenting challenges such as inadequate data labels and poor data quality. FTL leverages transfer learning to transfer knowledge from a source domain to a target domain, enhancing learning outcomes. This approach integrates transfer learning into federated learning, providing a comprehensive framework for training, evaluation, and cross-validation. Neural networks with additive homomorphic encryption ensure privacy preservation and comparable accuracy to non-privacy-preserving methods, although communication efficiency remains a concern. FTL is also used to extend federated learning applications to situations with limited intersection among parties. [50].

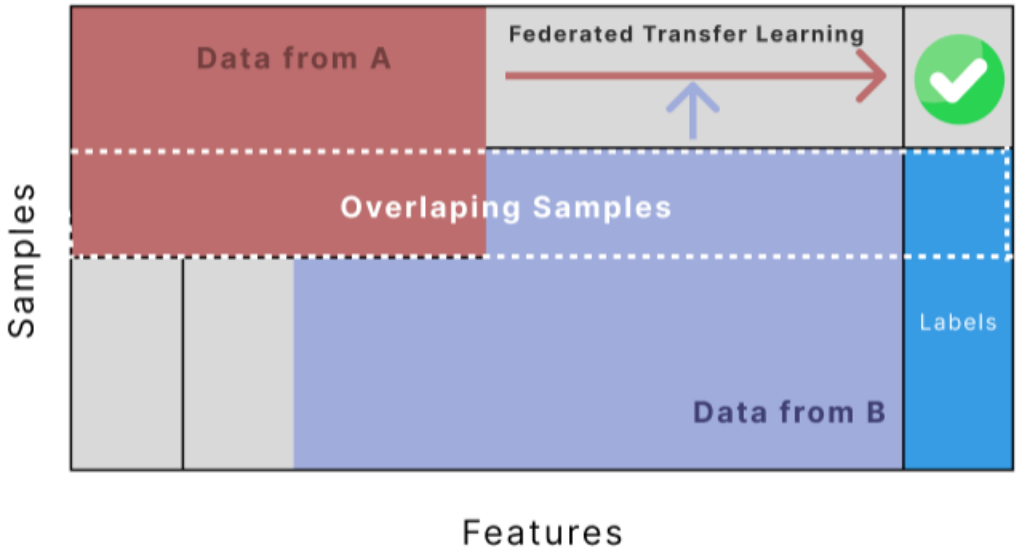

Figure 2.9: Illustration of Transfer Federated Learning.

2.5.4 Cross-Silo Federated Learning :

Cross-Silo Federated Learning is employed in scenarios where there are fewer participating devices accessible for all rounds. The training data can be structured in either horizontal or vertical FL format. This approach is predominantly utilized in organizational settings. For instance, they leverage Cross-Silo Federated Learning in developing their model [39].

2.5.5 Cross-Device Federated Learning :

This technique is applied in situations involving a large number of devices. Techniques such as client-selection and incentive designs are necessary to support this type of federated learning [39].

2.6 Federated Learning Algorithms

Federated learning algorithms play a crucial role in ensuring the effectiveness and security of the federated learning process. They aim to improve federated learning performance by

designing efficient methods for updating global models and guiding distributed training for clients. Among these key algorithms used in federated learning:

2.6.1 Federated Averaging (FedAvg):

The *FedAvg* algorithm is a federated learning algorithm used on the central server to aggregate updated parameters from clients to obtain the updated global model [42].

2.6.1.1 FedAvg Mathematical Description :

FedAvg entails a server storing the global predictive model, which is then distributed to a randomly selected subset (or federation) of devices. Each device within the federation trains the model using its local data and sends the parameter updates back to the server. The server aggregates these updates to generate a new global model . These aforementioned steps constitute one round of communication, and they are repeated for several iterations until the global model converges. The steps involved in *FedAvg* when applied to a fleet of assets are schematically depicted in Figure 2.10, where steps 1 to 4 represent one communication round. Consider a distributed system with devices, a total of n data points across the entire system, and P_k representing the set of indices of data points on device k. The finite sum objective function for the overall system can be expressed as [43]:

$$F(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w) \tag{2.1}$$

where $F_k(w)$ is the local objective function for the k^{th} device, and $n_k = |P_k|$ are the total number of data points on device k. The local objective functions may or may not be same as the global objective function [43].

In a specific communication round, let's consider a group of devices, with a size of C, forming a federation. Let s represent the total number of data points within this federation. Each device within this federation calculates the average gradient $g_k = \nabla F_k(w_t)$ using its local data for the current global model parameters w_t. Subsequently, the server combines the updates received from the devices and produces the updated global model for the subsequent communication round as [43]:

$$w_{t+1} \leftarrow w_t - \sum_{k=1}^{K} \frac{n_k}{s} g_k \tag{2.2}$$

The empirical convergence of the averaged model parameters from commonly initialized neural network models as described in (2.2) has been demonstrated. The expression $\sum_{k=1}^{K} \frac{n_k}{s} g_k$ is equivalent to $\nabla F_k(w_t)$. This mechanism allows the global model to learn from the data contributed by each device, where the weight of each device's update is proportional to the amount of data it carries. This constitutes the operational principle of *FedAvg* [43].

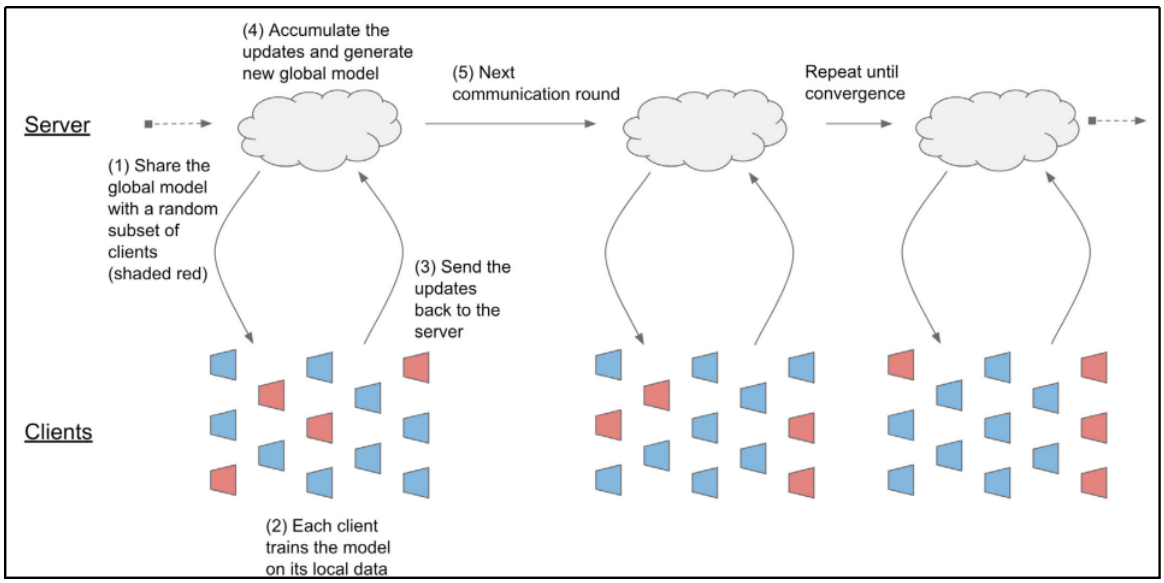

Figure 2.10: Schematic representation of $FedAvg$ for a client-server type networked system [43].

1. **Advantages of the FedAvg algorithm:**

   - **Communication-efficient:** The algorithm requires minimal data transfer between devices and the central server.
   - **Privacy preservation:** Local data is not shared with other devices or the central server.
   - **Scalable:** The algorithm can be used to train models on large datasets distributed across many devices.

2. **Disadvantages of the FedAvg algorithm:**

   - **May be slower than centralized training:** The algorithm requires multiple training iterations to achieve sufficient accuracy.
   - **Accuracy may be affected by the number of participating devices:** Accuracy may be low if the number of participating devices is small.

2.6.2 Federated Prox :

To address the challenges posed by data heterogeneity in Federated Learning (FL), conventional methods like $FedAvg$, which rely on local Stochastic Gradient Descent (SGD), may encounter convergence issues in practical scenarios, especially when participating devices execute numerous local updates. In response to this challenge, $FedProx$ has emerged as a solution tailored for addressing the empirical FL problem through the adoption of (imprecise) proximal point updates for local optimization [51].

2.6.2.1 FedProx Mathematical Description :

To elaborate, at each time step t, $FedProx$ randomly selects a subset $I_t \subseteq [M]$ of devices and formulates a proximal point Empirical Risk Minimization (ERM) sub-problem for each device ξ_t to facilitate local updates centered around the preceding global model w_{t1} [51]:

$$w_t \approx arg \min_{w \in Rp} \{Q^{\xi}_{erm}(w; w_{t-1}) := R^{\xi}_{erm}(w) + \frac{1}{2\eta_t}||w - w_{t-1}||^2\} \tag{2.3}$$

where $\eta t > 0$ is the learning rate that controls the impact of the proximal term. Then the global model is updated by uniformly aggregating those local updates from It as [51]:

$$w_t = \frac{1}{I_t} \sum_{\xi \in I_t} w^{\xi}_t \tag{2.4}$$

In the extreme scenario where t approaches infinity in equation (2.3), $FedProx$ essentially transitions to the $FedAvg$ framework when utilizing SGD for local optimization. Since its inception, $FedProx$ and its various iterations have garnered significant attention in research, emerging as a preferred algorithm in domains like autonomous driving and computer vision . From a theoretical standpoint, $FedProx$ offers convergence assurances predicated on a specific assumption regarding the bounded dissimilarity of local gradients, which encapsulates the statistica diversity of within the network [51].

1. **Advantages of the FedProx algorithm :**

   - **Enhanced Stability:** FedProx enhances the stability of local updates by explicitly guiding the local optimization process within the proximity of the current global model [51].
   - **Convergence Guarantees:** This method provides convergence guarantees for both convex and non-convex functions, even in scenarios involving partial participation and significant discrepancies in the number of local updates across devices [51].

2. **Disadvantages of the FedProx algorithm:**

   - **Complexity and Computational Overhead:** FedProx's proximal term makes the optimization problem more complex than it would be with a simpler method like FedAvg, sometimes requiring more computational resources and sophisticated optimization techniques [51].

2.7 Data Partitioning Strategies in Federated Learning

Federated learning requires careful consideration of data partitioning strategies to effectively leverage distributed data while addressing privacy, communication, and computational challenges. Understanding the nature of the data and the distribution across clients is critical in choosing the appropriate partitioning strategy and implementing robust federated learning systems [54].

2.7.1 Independent and Identically Distributed(IID)

In statistics and machine learning, the assumption that data points are Independent and Identically Distributed (IID) often simplifies analysis and model building. However, many real-world datasets violate these assumptions, leading to challenges in analysis and modeling. Here's a breakdown of what IID means :

- **Definition:** Data is partitioned such that each client receives a dataset that is representative of the overall data distribution [54].
- **Advantages:** Simplifies the learning process and usually leads to faster convergence since each client's local model is trained on a representative sample of the overall dataset.
- **Use Case:** Useful in scenarios where data distribution across clients is naturally balanced or where simulating a balanced data distribution is acceptable.

2.7.2 Non-Independent and Identically Distributed (Non-IID)

In statistics and machine learning, the assumption that data points are Independent and Identically Distributed (IID) often simplifies analysis and model building. However, many real-world datasets violate these assumptions, leading to challenges in analysis and modeling. Here's a breakdown of what non-IID means :

- **Definition:** Data is partitioned in a way that each client's data may not represent the overall data distribution. This more accurately reflects real-world scenarios where data on different devices can be biased or skewed [54].
- **Advantages:** Provides a more realistic evaluation of federated learning algorithms and is useful for understanding the robustness of models in heterogeneous environments.
- **Use Case:** Common in real-world applications where clients have data specific to their usage patterns, geographic locations, or device types.

2.8 Related Works

In this section, we will present some relevant previous works and studies that focus on developing IoT medical devices for healthcare to increase security and protect data privacy using federated learning algorithms and artificial intelligence. We have classified these works as follows:

Syreen et al [66].The study proposes classification and clustering techniques for FL-enabled IoT systems, addressing the vulnerability of these devices to cyber-attacks due to their resource-constrained nature.

Nguyen et al [67]. addressed this vulnerability by exploring a wide range of FL applications in IoT services, including data sharing, attack detection, and IoT privacy and security.

Similarly, Alam et al [68]. The study emphasizes the importance of resilient FL classifiers in

maintaining data confidentiality and privacy regulations, particularly in the context of Industry 4.0 and federated learning, which integrates AI and IIoT devices for healthcare training.

Elayan et al [69] [70]. The study developed an advanced federated learning system with transfer learning to detect skin disorders, achieving a high area under the curve (AUC) of 0.97, demonstrating strong performance. However, it did not address the issue of limited data availability.

Another noteworthy contribution by Singh et al [71]. The research introduces a blockchain-based distributed FL framework for IoT-enabled medical devices, enabling machine learning models to be trained at various device locations without transmitting patient data to the cloud.

Li et al [72]. The study introduced ADDetector, a diagnostic tool for early Alzheimer's disease detection, featuring three layers of privacy protection: users, clients, and gateway. It uses a classification model with differential privacy to protect data during aggregation. ADDetector achieved an accuracy of 81.9%.

Brisimi et al [73]. The study proposed a binary supervised learning system to predict hospitalizations due to cardiac events using distributed algorithms. It aimed to enable diverse data holders to collaborate on a shared predictive model without exchanging raw data.

Additionally, Blanquer et al [74]. The study suggested a hardware-enhanced architecture for encrypting sensitive data in memory and on disks, accessible only within trusted execution contexts. By leveraging federated clouds, they accumulated the necessary resources for high performance and security in their proposed approach.

Stephanie et al [75]. proposed a blockchain-based heterogeneous model for collaborative learning across healthcare institutions to protect user privacy. Hospitals used edge servers to train data, with models verified and stored in a private blockchain. Using Colon Pathology and Breast Cancer datasets, algorithms like centralized FL, FedAvg, and Ensemble-FedAvg achieved accuracies over 80%.

Similarly, Zhang et al [76].The study used cryptographic techniques like masks and homomorphic encryption to secure local models and prevent reconstruction attacks on medical data. Training on the HAM1000 dataset for skin lesion classification, their model achieved an accuracy of approximately 76.9%.

Khoa et al [77]. the researchers utilized an Encode Depth Convolutional Network for human activity recognition, serving as both server and client. They compared this approach with LSTM and CNN within the federated learning (FL) setting.

Lian et al [78]. The researchers introduced the DEEP-FEL model, trained in medical centers, maximizes topological ring structure and privacy, focusing on FL and machine learning techniques for COVID-19 CT scan images. Castillo et al [79]. The study introduced a diabetes-monitoring healthcare system with estimation and classification models. Employing federated learning (FL) and independent learning (IL), FL achieved a recall rate of 98.69%, while IL reached 97.87%. The research highlights FL's significance in IoT-enabled healthcare but notes challenges in data heterogeneity and model robustness.

Rahman et al [80]. introduced an FL framework tailored for health IoT, incorporating an edge layer for deep learning tasks and blockchain technology to enhance security and trustworthiness.

Similarly, the authors of [81] The study explores the use of FL and blockchain in sharing industrial IoT data, with promising results in healthcare applications, including clustering

techniques for clinical data generation [82]– [85].

Yu et al [86]. The study utilized a general object identification dataset to evaluate the efficacy of an abnormal weights-clipping federated learning algorithm, based on the Federated Average technique across various industries.

Bodagala H et al [55]. aimed to evaluate the effectiveness of federated learning approaches in contrast to traditional machine learning methods, particularly concerning cybersecurity in Internet of Things (IoT) systems. Utilizing the dataset derived from "Bot-IoT," the research showcases that federated learning techniques like "DNN, CNN, and RNN" outperform conventional centralized ML techniques. Employing federated learning enhances threat detection accuracy and ensures the privacy of data originating from IoT devices. Elayan H et al [56] . This paper introduces a Deep Federated Learning framework tailored for decentralized healthcare systems leveraging IoT while preserving user privacy by avoiding the sharing of private data. Additionally, an algorithm is proposed to streamline the training data acquisition process, facilitating a fully automated federated learning workflow. The implementation of deep federated learning in skin disease detection experiments is detailed, demonstrating improved AUC percentages post-federated rounds, sustained model accuracy, and favorable classification metrics. Despite the decentralized architecture increasing model conversion time, the objective of data non-sharing is achieved.

Borja T et al [57]. Their research assesses the Virtual Client Engine (VCE) architecture within the Flower open-source framework, aiming for efficient simulations with numerous users while reducing computational resource usage. They contrast it with traditional centralized machine learning techniques and Flower's prior Edge Client Engine (ECE) architecture, concentrating on key classification metrics like accuracy and loss. Findings reveal enhanced scalability alongside high accuracy, with approximately 91% accuracy for the MNIST dataset and 88% for the SPEECH evaluation setup involving 200 clients employing the VCE Flower architecture.

Pereira K et al [58]. Their study delves into federated learning, a distributed approach to machine learning. The proposed system involves a central server that aggregates features and weights from multiple nodes, reducing bias in prediction models while ensuring data decentralization for patient privacy. They develop an end-to-end application for distributed training of batch data, incorporating real-time visualization using sockets. Additionally, the application includes an inference service for classifying chest X-rays to detect signs of pneumonia based on image analysis.

Peta J et al [59]. The objective of the study is to introduce an automated system for disease diagnosis using federated learning and deep learning, aimed at streamlining and accelerating the process. The study outlines five key steps: image acquisition, encryption, optimal key generation, secure data storage, and disease classification. Python software is utilized for implementation, with experimental analysis conducted using the BreakHis Database. Simulation results demonstrate superior performance in accuracy (95.68%).

Otoum Y et al [60]. Their paper introduces a Federated Transfer Learning-based Intrusion Detection System (IDS) designed to enhance security for patients' healthcare-connected devices. The model employs a Deep Neural Network (DNN) algorithm for network training, transferring knowledge from connected edge models to construct an aggregated global model tailored to each edge device without compromising data privacy. Evaluation using the CICIDS2017 dataset demonstrates superior performance in terms of accuracy, detection rate, and average training time. Furthermore, the proposed model exhibits enhanced generalization and incremental learning capabilities compared to baseline ML/DL algorithms utilized in conventional

centralized learning schemes.

Rachakonda S et al [61]. The researchers developed a FL framework for scalability, monitoring services, and privacy. They evaluated the framework in a cross-silo setup and extended validation on a clinical use case. They found comparable performance in training a CNN in FL and in a non-FL setup, even for smaller dataset sizes. Soft decision trees were employed to train tabular data models in FL, which could be efficiently trained using streaming data. This could improve the deployment of FL models on edge devices with limited computation power and memory storage [89] .

Nair Akarsh K et al [62] . The paper proposes a privacy-preserving framework (Fed-Select) for IoMT-based big data analysis under Federated Learning (FL) to address privacy threats. Fed-Select uses alternative minimization to limit gradients and participants, employs hybrid encryption techniques, and uses Laplacian noise-based differential privacy for security enhancement. Experimental results show that the change in gradient volume and participant number is not proportional to system performance parameters. The framework is analyzed from a security perspective and compared with other schemes.

Zhao L et al [63].The paper introduces FedDIS, a method for image classification that shares hidden space data distribution information under privacy protection. This helps reduce non-IIDness across clients and improves image classification performance. Experiments were conducted on the Alzheimer's disease MRI image dataset and MNIST dataset, examining the effect of hidden space distribution types and parameters. The method has potential for improving traditional federated learning performance in privacy-preserving situations. Further research will explore improving performance with heterogeneous image data.

Makkar A et al [64] . In this article, a novel federated learning (FL) based aggregation approach called SecureFed is proposed to enhance privacy, fairness, and robustness in COVID-19 detection. The approach is validated using Chest X-ray data from 2100 positive cases, demonstrating superior performance compared to existing COVID-19 detection methods in terms of robustness and privacy preservation. SecureFed is compared with other FL aggregation methods such as FedAvg, FedMGDA+, and FedRAD, across various ratios of training and testing datasets, with SecureFed consistently outperforming the alternatives. Future plans include integrating the proposed aggregation method into different FL settings

AlSalman H et al [65]. emphasize the effectiveness of federated learning architectures, particularly utilizing Deep Convolutional Neural Networks (DCNNs), for breast cancer detection. Their approach demonstrates significant improvement in detection accuracy, achieving a rate of 98.9% across three major datasets: VINDR-MAMMO, CMMD, and INBREAST, while ensuring high standards of privacy and security.

A summary of the most prominent works is in the following table :

Table 2.1: A summary of the most prominent works

Authors	Dataset	Model	Algoritm Fl	Accrucy
Bodagala H et al [55].	Bot-IoT	DNN, CNN, and RNN	FL algorithms	98%
Elayan H et al [56].	Dermatology dataset	ML , ADAM optimizer	FedAvg	97.4%
Borja T et al [57].	MNIST and SPEECH	CNN	FedAvg	88%
Pereira K et al [58].	Kaggle's "Chest Xray Images"	CNN, RNN and LSTMs	FedAvg	91%
Peta J et al [59].	BreakHis	CNN, BiLSTM, DNN	FedAvg	95.68%
Otoum Y et al [60].	CICIDS2017	DNN	FL algorithms	88.27%
Rachakonda S et al [61].	MNIST	CNN	FL algorithms	99%
Nair Akarsh K et al [62].	MNIST	CNN	Fed-Select	94.75%
Zhao L et al [63].	MNIST , MRI	CNN	FedDIS	97.3%
Makkar A et al [64].	COVID-19 Medical images	CNN	FedAvg, FedMGDA+, FedRAD, SecureFed.	78.29% 65.3% 82.2% 88.3%
AlSalman H et al [65].	VINDR-MAMMO, CMMD, and INBREAST	DCNN	FedAvg	98.9%

2.9 The Advantages of Federated Learning for IoT

The distributed, collaborative, and privacy-preserving features of FL offer several significant benefits for IoT applications, as outlined below (refer to Figure 2.11) :

1. **Preserving User Data Privacy:**

   In an ideal FL scenario, each IoT device learns only the necessary information for its function. Raw data remains on the devices during federated training, with only model updates sent to the central server, reducing the risk of personal data exposure [41].

2. **Improving Model Performance:**

   Individual IoT devices may lack sufficient data to train high-quality models independently. Through FL, all devices collaborate to train a high-quality model, benefiting from each other's data without accessing private information. Additionally, periodic up-

dates to local models enable edge devices to continually improve their models, enhancing performance beyond individual capabilities [41].

3. **Flexible Scalability:**

   FL leverages computation resources from multiple IoT devices across different locations in parallel, enhancing scalability. With increasing edge device hardware capabilities and growing data sizes, centralizing data to a server can strain edge resources or communication networks. FL accommodates more devices without overburdening a centralized server, supporting scalable IoT networks. Furthermore, FL reduces the need for extensive data transmission, particularly beneficial for low-bandwidth IoT networks, thus enhancing scalability while minimizing communication costs [41].

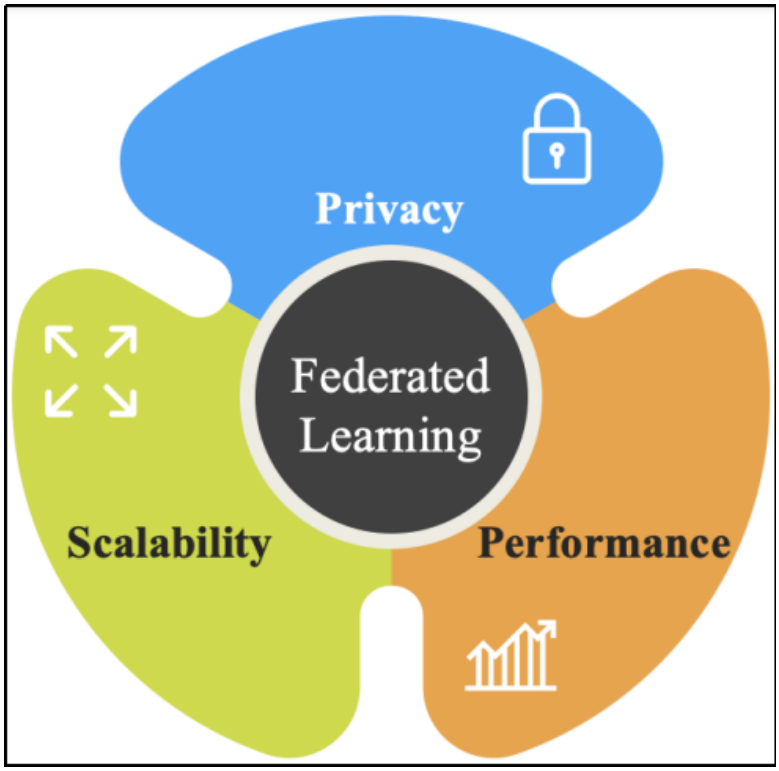

Figure 2.11: Advantages of Federated Learning for IoT [41].

2.10 Federated Learning and Its Integration with Emerging Technologies in Healthcare

Integration with emerging technologies can significantly enhance the capabilities of federated learning (FL) in various domains:

1. **Blockchain-based FL in Healthcare :**

   Blockchain technology can be used to enhance federated learning in healthcare. This is because federated learning preserves the privacy of client data and allows for collaborative model development. However, there are ongoing challenges. To overcome these challenges and enhance the security, scalability, and performance of federated learning, an effective alliance emerges in the form of blockchain technology. Blockchain provides a centralized and secure structure for storing health data and enhances security and privacy. There are innovative attempts to apply blockchain in healthcare, such as using encryption tech-

niques to maintain privacy in federated learning systems. The combination of blockchain and federated learning promotes the development of smart applications in healthcare, facilitating collaboration among individual units in healthcare systems [52].

2. **Health Monitoring with Federated Learning and IoT:**

   The emergence of the Internet of Things (IoT) has sparked a monumental revolution across various sectors, including healthcare. IoT has facilitated seamless communication, smart devices, and enhanced productivity. This has led to the integration of numerous distributed smart devices and sensors, allowing for the organization of real-time data flow across diverse applications. Healthcare is one of the sectors deeply impacted by the IoT revolution. The integration of IoT with federated machine learning in healthcare combines real-time data from IoT devices with machine learning models while preserving data privacy [52].

3. **Edge Computing Assisted FL for Healthcare:**

   Emerging technologies in healthcare, like edge computing and federated learning, are advancing rapidly. Edge computing improves cloud networks by bringing resources closer to data sources, enhancing network resilience, data availability, and bandwidth. FedHome's Federated Learning framework personalizes home health monitoring by integrating cloud and edge resources, achieving impressive accuracy improvements. Fog computing and models like BodyEdge also enhance processing speed and reduce internet traffic, making them applicable to healthcare. These technologies prioritize security and privacy, incorporating blockchain and edge computing techniques [52].

2.11 Challenges of Federated Learning

Federated learning presents several challenges that need to be addressed to ensure its successful implementation. Some of the key challenges include :

- **Privacy protection:** Federated Learning (FL) faces privacy challenges due to model gradients sharing data, potentially compromising privacy during training. Techniques like differential privacy and Secure Multiparty Computation address these concerns, but they can impact model performance and system efficiency. Balancing these trade-offs is crucial for creating effective private federated learning systems [44].
- **Communication cost:** Federated learning faces communication challenges, especially in networks with numerous devices like mobile devices. Developing efficient systems is crucial for its viability. Localizing data on devices can alleviate privacy concerns. Prioritizing iterative model updates and concise signals over transferring the entire dataset during training is essential for effective communication in federated learning [44].
- **Systems heterogeneity :** Federated networks have diverse communication and processing capabilities due to hardware and network connectivity differences. Only a small percentage of devices are active, presenting challenges for fault tolerance. Effective management requires resilience to offline devices and diverse hardware configurations. Federated learning strategies must be adaptable to accommodate varying user numbers, hardware setups, and occasional communication failures [44].
- **Unreliable model upload :** In Federated Learning (FL), nodes may purposefully or unintentionally give the server false information when the global models are being aggregated . By tampering with the global model's aggregation process, an attacker might

potentially interfere with model training through the use of malicious model parameters. Furthermore, mobile devices could mistakenly upload low-quality models in an unstable network environment, which would negatively impact FL. Preventing the submission of shaky local models is essential to maintaining the integrity of the FL process [44].

2.12 Future Directions

The field of federated learning is witnessing increasing research activity, with the emergence of several promising trends aimed at addressing previously raised challenges . In this section, we highlight some of these research directions and provide an overview of additional challenges that require further exploration [49]:

- **Extreme communication schemes :**

  Novel approaches to communication are being explored in federated learning, raising questions about the optimal level of communication required. It's widely acknowledged that machine learning optimization techniques can accommodate a degree of imprecision, this margin of error may even aid in generalization. While conventional data center setups have experimented with one-shot or divide-and-conquer communication strategies, their efficacy in large-scale or statistically diverse networks remains unclear. Similarly, in federated settings, one-shot or few-shot heuristics have been proposed but lack thorough theoretical analysis or evaluation at scale [49].

- **Communication reduction and the Pareto frontier :**

  Reducing communication and navigating the Pareto frontier are essential considerations in federated training. We explored various methods for communication reduction, such as local updates and model compression. To develop a practical federated learning system, it's crucial to comprehend how these techniques interact and systematically evaluate the trade-off between accuracy and communication for each method. Specifically, effective techniques should showcase advancements at the Pareto frontier—achieving higher accuracy compared to other approaches within the same communication constraints, ideally across diverse communication/accuracy profiles. Similar comprehensive analyses have been conducted for efficient neural network inference [e.g., 9], underscoring the necessity for meaningful comparisons of communication reduction techniques in federated learning [49].

- **Novel models of asynchrony :**

  There are two main approaches to communication widely explored in distributed optimization synchronous collective approach and asynchronous approach (assuming bounded delays). These strategies reflect real-world scenarios in data center environments, where worker nodes are often committed to specific tasks, enabling them to promptly request the next task after completing the previous one. In contrast, in distributed networks, devices are often uncommitted to ongoing tasks, with many remaining inactive during each iteration. Therefore, exploring the effects of a device-centric communication approach, where each device decides when to interact with the central server in an event-driven manner, is valuable [49] .

- **Granular privacy constraints :**

  It may be necessary to specify privacy on a more granular level, as privacy constraints may vary from one device to another or even among data points on a single device.

For example, researchers recently proposed privacy guarantees tailored to specific samples (rather than users), offering a weaker form of privacy in exchange for more accurate models. Developing methods to handle mixed privacy constraints (device-specific or sample-specific) is an intriguing and ongoing direction for future work [49].

2.13 Conclusion

Federated learning is an innovative and promising technique that relies on the collaboration of participating parties without the need to share actual data, allowing for privacy preservation and reducing communication costs. Despite the challenges it faces, such as privacy protection and communication costs, federated learning has multiple and diverse applications. The next chapter will delve into the practical aspect of this promising technique, where we will take a deep dive into how federated learning can be applied in fields such as healthcare and Internet of Things (IoT). We will explore innovative ideas and applications that can make a real difference in people's lives and in the world of technology.

Chapter 3

Implementation And Experimental Results

3.1 Introduction

This chapter focuses on the experimental results obtained in our work. In this chapter, we will present the components necessary for implementation, including a description of the datasets used, mechanisms, methods, algorithms, and classification functions that we adopted in building the centralized deep learning models. We will also detail the evaluation factors used to describe the results, and provide an overview of the tools, techniques, software, libraries and frameworks used that facilitated these achievements. We will also provide a comprehensive analysis of the experimental results obtained.

3.2 Project Structuring :

In this part we will provide an overview of the structure we followed to implement our project, as each of the following steps represents a significant contribution to the overall project of implementing unified learning :

1. **Import Libraries:** Setting up the environment with all necessary libraries is crucial for smooth development and execution.
2. **Load and Preprocess Data:** Proper data handling and preprocessing ensure that the model receives data in the correct format.
3. **Define Centralized learning Model:** Creating a robust Centralized learning model forms the backbone of our training process.
4. **Define Client Class:** Implementing the client class allows each client to independently handle its training and evaluation.
5. **Define Server and Training:** To define the server and train the model we need to create a server strategy and decide how to start the simulation. We must also determine a way to collect and store metrics in order to analyze performance.
6. **Split Data:** Distributing the data among clients ensures an even workload and simulates real-world federated learning scenarios.
7. **Start Federated Learning:** Initializing and running the federated learning process is the core of this project, leveraging Flower's capabilities.
8. **Evaluate and Visualize:** Evaluating and visualizing the model's performance helps in understanding the effectiveness of the federated learning approach and identifying areas for improvement.

3.3 Implementation

3.3.1 Environments and Libraries :

3.3.1.1 Programming Language:

- **Python :**

Python [96] is a high-level programming language with a strong emphasis on indentation, code readability, garbage collection, and dynamic typing, compatible with functional, object-oriented, and structured programming paradigms.

3.3.1.2 Development Environment :

- **Jupyter:**
  JupyterLab [97] is a powerful, versatile, and extensible IDE that significantly enhances the capabilities of the classic Jupyter Notebook. Its modular design, support for various file types, and integration with numerous tools and languages make it an excellent choice for data scientists, researchers, and educators who need a flexible and robust environment for their projects.
- **Google Colaboratory:**
  Google Colab [98] is a powerful tool for anyone involved in data science, machine learning, or educational fields. Its combination of cloud-based accessibility, powerful computing resources, and ease of use makes it an excellent choice for both beginners and experienced professionals.

3.3.1.3 Libraries :

- **Keras:**
  Keras [99] is an open-source package offers an artificial neural network Python interface. Initially developed independently, Keras was later included into the TensorFlow framework and expanded upon.
- **NumPy :**
  NumPy [100] serves as the cornerstone for scientific computing in Python. As a Python library, it furnishes a multidimensional array object alongside diverse derived objects like masked arrays and matrices. Moreover, it encompasses an array of functions tailored for swift array operations, encompassing mathematical, logical, shape manipulation, sorting, selection, input/output, discrete Fourier transforms, elementary linear algebra, fundamental statistical operations, random simulation, and beyond.
- **Tensorflow :**
  A large-scale machine learning system that performs well in a variety of settings is called TensorFlow [101]. TensorFlow represents computation, shared state, and the operations that change that state using dataflow graphs.
- **Pandas :**
  Pandas [102] is a library that provides advanced data structures and functions, aiming to streamline the handling of structured or tabular data, making it efficient, straightforward, and expressive. Since its inception in 2010, it has played a pivotal role in empowering Python as a robust and efficient data analysis platform. The core components of pandas include the DataFrame, tailored for column-oriented tabular data with labels for rows and columns, and the Series, a labeled one-dimensional array-like object.
- **Matplotlib :**
  Matplotlib [103] stands out as Python's leading library for generating 2D graphics and various visual representations. Crafted to deliver charts of publication quality, it remains unmatched. Despite alternative visualization libraries, matplotlib reigns supreme in popularity among Python developers, ensuring seamless integration within the ecosystem. I consider it a dependable default choice for visualization needs.
- **Pytorch :**
  PyTorch [104] is a machine learning library built on the Torch library and used for applications like computer vision and natural language processing, Originally created by Meta AI and now a part of the Linux Foundation.
- **Flower :**
  Flower [105] is an open-source framework designed for federated learning, a machine

learning technique where multiple servers train models with local data storage, facilitating task orchestration and scaling across environments.

3.3.2 Tools :

To implement this study, we relied on two computers, and we will present their specifications in the following Table 3.1

Table 3.1: Tools

Device Specifications	
PC Mark	HP Pavilion
Processor CPU	Intel(R) Core(TM) i5-4210U
RAM	8.00 GB
Hard Disk	750 GB HDD + 256 GB SSD
Graphics Card GPU	Intel(R) HD Graphics + NVIDIA GEFORCE 840 M
Operating System	Windows 10 Home(64-Bit)

3.4 The Experimental Dataset

3.4.1 The MNIST Dataset Discription

The MNIST (Mixed National Institute of Standards and Technology) database, introduced by LeCun et al. in 1998, serves as a widely used benchmark for testing various machine learning and pattern recognition techniques. With a total of 70,000 instances, 60,000 allocated for training and the remaining 10,000 for testing, the dataset draws from two primary sources: NIST's Special Database 1, sourced from high school students, and NIST's Special Database 3, sourced from Census Bureau staff. To ensure diversity, the training set includes samples from over 250 writers. Original images underwent comprehensive preprocessing to prevent redundancy, involving normalization to a 20 $\times$ 20 pixel box, followed by padding with blanks and anti-aliasing to convert them to grayscale and fit them into a larger 28 $\times$ 28 pixel box. [90]

3.4.2 Preprocessing Dataset

Preprocessing the MNIST dataset is a crucial step in preparing the data for effective training of machine learning models. These preprocessing steps collectively ensure that the dataset is well-prepared, leading to more efficient and effective training of machine learning models, particularly neural networks.Here are several key reasons why preprocessing is important:

- **Normalization:** Ensures consistent input range, aiding in faster and more stable training.

- **Reshaping:** Makes data compatible with CNNs and maintains uniform dimensions.
- **One-Hot Encoding:** Prepares labels for classification tasks and proper loss calculation.
- **Data Splitting:** Creates training, validation, and test sets to monitor performance and prevent overfitting.
- **Data Augmentation:** Enhances the dataset size and variety, improving model generalization and robustness.

3.5 Compilation Process :

The compilation process for training a centralized deep learning model involves configuring the model with the components needed for training. This includes the activation function, loss function, optimizer selection. Below is a breakdown of each component:

3.5.1 Activation Functions Used :

In artificial neural networks, activation functions play a crucial role in introducing non-linearity into the model. Each neuron's weighted sum of inputs plus a bias is used by them to calculate whether or not to activate the neuron, or whether to send the data to the network's next layer [91]. It contains several functions, and we have based our model on:

1. **Rectified Linear Unit (ReLU) Function :**
   The ReLU (Rectified Linear Unit) activation function, proposed by Nair and Hinton in 2010, has indeed become one of the most widely used activation functions in deep learning. Its popularity stems from several advantages it offers over other activation functions like Sigmoid and Tanh [92].The mathematical definition of the ReLU activation function is [93]:

$$G(E) = max(0, E) = \begin{cases} E & si \quad E \geq 0 \\ 0 & sinon \end{cases} \tag{3.1}$$

   To put it another way, ReLU applies a threshold operation to every input element, keeping positive values constant and setting negative values to zero. Deep learning applications benefit from this straightforward operation's computational efficiency and efficacy. And whose appearance is as follows:

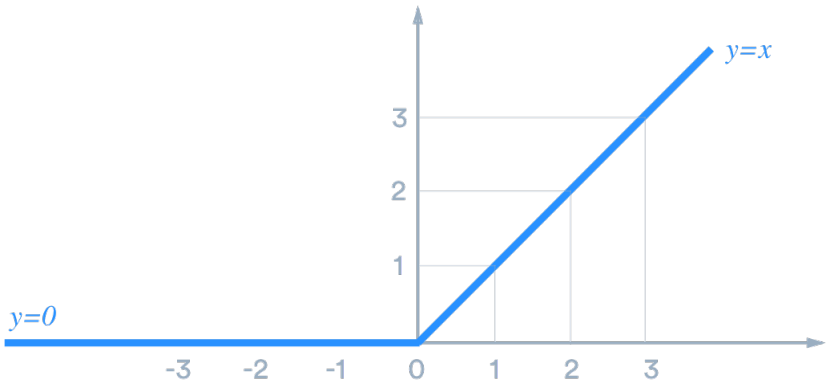

Figure 3.1: ReLU activation function

2. **Softmax Function :**

The Softmax function, a type of activation function in neural computing, is crucial for computing probability distributions from vectors of real numbers. It transforms inputs into a range of values between 0 and 1, ensuring that their sum equals 1. Employing the formula

$$G(e_j) = \frac{e^{e_j}}{\sum_i e^{e_i}} \tag{3.2}$$

, Softmax calculates probabilities for each class in multi-class models, identifying the class with the highest probability as the target. This function typically features in the output layers of deep learning architectures. Notably, Softmax is distinct from Sigmoid in its application: Sigmoid is for binary classification, whereas Softmax is for multivariate classification tasks [92].

3.5.2 Loss Functions Used :

loss function is a crucial component in training machine learning models. It quantifies how well the model's predictions match the actual target values. Since our models rely on multi-class classification, where each case belongs to one of many classes, we used the function Categorical Cross-Entropy Loss .

Categorical Cross-Entropy Loss : is commonly used for multi-class classification problems, where the goal is to classify an input into one of multiple categories. This loss function compares the predicted probability distribution over the classes to the true distribution (usually represented as a one-hot encoded vector) and calculates the loss based on the difference between them.The categorical cross-entropy loss (L_{CCE}) is computed as [106]:

$$L_{\text{CCE}}(y, p) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(p_{i,c}) \tag{3.3}$$

where:

- N is the number of samples.
- C is the number of classes.
- $y_{i,c}$ is a binary indicator (0 or 1) if class label c is the correct classification for sample i.
- $p_{i,c}$ is the predicted probability that sample i belongs to class c.

3.5.3 Optimization Algorithm Used:

1. **Adam Optimizer :** Adam (short for Adaptive Moment Estimation) is an optimization algorithm that combines the advantages of two other extensions of stochastic gradient descent: Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp). Adam computes individual adaptive learning rates for different parameters. The mathematical definition of the Adam Optimizer is [108]:

$$L = \frac{1}{N}\sum_{n=1}^{N}[y - y_p]^2 \tag{3.4}$$

   where y is the label, y_p the prediction and N the number of batches.
2. **Stochastic Gradient Descent (SGD):** is an optimization algorithm used to minimize objective functions, commonly the loss function in machine learning models. It is especially effective for training large-scale and complex models like neural networks. Unlike

standard Gradient Descent, which computes the gradient over the entire dataset, SGD updates model parameters using the gradient from a single data sample or a small batch, making it computationally efficient for large datasets .The update rule for SGD is [107] :

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta J(\theta_t; x_i, y_i) \tag{3.5}$$

where:

- θ_t are the parameters at iteration t.
- η is the learning rate.
- $J(\theta_t; x_i, y_i)$ is the loss for a single training example (x_i, y_i).
- $\nabla_\theta J(\theta_t; x_i, y_i)$ is the gradient of the loss with respect to θ_t for the training example (x_i, y_i).

3.5.4 Performance Evaluation Metric

Given that we're dealing with a balanced dataset, our experimentation results are presented using four evaluation metrics: Accuracy precision, recall, and F1 score. These evaluation metrics are calculated based on The confusion matrix factors.

3.5.4.1 The Confusion Matrix:

The confusion matrix is a tool used to evaluate the performance of a classification model by summarizing its prediction results. It breaks down correct and incorrect predictions by class and compares them with the actual values. This analysis helps to understand how the model is confused when making predictions, identifying the types of errors made. The matrix is structured with actual values marked as True (1) or False (0), and predictions marked as Positive (1) or Negative (0). Key metrics like True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are derived from the confusion matrix, providing insights into the model's classification capabilities.These key metrics indicate:

1. **True Positive (TP):** predicting and observing a positive outcome.
2. **False Positive (FP) :** arises when predicting a positive outcome, but the actual result is negative, akin to a Type 1 Error.
3. **False Negative (FN):** occurs when predicting a negative outcome, but the actual result is positive, similar to a Type 2 Error.
4. **True Negative (TN) :** is when predicting and observing a negative outcome.

3.5.4.2 Accuracy :

Accuracy measures the frequency with which the model correctly classifies samples, expressed as a percentage. It encompasses both true positives and true negatives, which indicate correctly predicted samples among all predicted data samples[9].The following formula is used to calculate it :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.6}$$

3.5.4.3 Precision :

Precision is the proportion of positive predictions that are accurate. It is calculated using the following formula:

$$Precision = \frac{TP}{TP + FP} \tag{3.7}$$

3.5.4.4 Recall :

The recall metric represents the ratio of true positives to the sum of true positives and false negatives. Mathematically, recall is expressed as:

$$Recall = \frac{TP}{TP + FN} \tag{3.8}$$

3.5.4.5 F1 score :

The F1 score, a key evaluation metric, combines precision and recall into a single measure, reflecting the overall performance of a model. It is calculated using the following formula:

$$F1score = \frac{2 \cdot precision \cdot recall}{precision + Recall} \tag{3.9}$$

3.5.5 Centralized Deep Learning Models :

In our project, we adopted several central deep learning models, represented by (ANN,CNN ,RNN,LSTM) for the purpose of comparison and choosing the most optimal model in terms of accuracy to be adopted in implementing federated learning. We will provide a description of each model:

3.5.5.1 ANN Model:

The model is an ANN tailored for digit classification on MNIST, comprising an input layer with 784 neurons, two hidden layers with ReLU activation (128 and 64 neurons), and an output layer with 10 neurons using softmax activation. Trained with Adam optimizer and categorical cross-entropy loss for 10 epochs with batch size 128, it achieves strong accuracy of 97.94 % , precision of 97.94% , recall of 97.91%, and F1 score of 97.92 % metrics on MNIST.

3.5.5.2 CNN Model :

A Convolutional Neural Network (CNN) is a specialized deep learning model designed for image analysis and recognition. The specified CNN model processes 28x28 pixel grayscale images. It features two convolutional layers with 20 and 50 filters, respectively, using ReLU activation and MaxPooling to extract and downsample features. Dropout layers are included to prevent overfitting. The model then flattens the output to a 1D vector, which is passed through dense layers, culminating in a softmax layer for classifying the images into 10 classes

(digits 0-9). The model is trained over 10 epochs with a training batch size of 64 and a testing batch size of 1000, using a learning rate of 0.01, The model's performance was evaluated using the metrics: accuracy of 99.25% , precision of 99.25%, recall of 99.25%, and F1 score of 99.25 % .

3.5.5.3 RNN Model:

The RNN model utilizes a SimpleRNN structure within RNN structure tailored for sequential data processing. With a lone SimpleRNN layer featuring 128 units and a subsequent Dense output layer containing 10 units for classifying into 10 categories, the model incorporates the ReLU activation function to instill non-linearity. Compiled with the Adam optimizer and categorical cross-entropy loss, the model assesses accuracy as its performance metric. The summary comprehensively delineates the architecture of the network, encompassing parameters and output shapes across each layer.The model's performance was evaluated using the metrics: accuracy of 97.53% , precision of 97.52%, recall of 97.50%, and F1 score of 97.50 % .

3.5.5.4 LSTM Model:

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem and capture long-range dependencies in sequential data. The model consists of LSTM cells with gated units that selectively retain and forget information over time. an LSTM model is built with 128 units and an input shape of (28, 28), suitable for processing sequences of data. The model is compiled using the Adam optimizer and categorical crossentropy loss function. Training the model over 10 epochs with a batch size of 64, it learns to classify input data into 10 classes. After training. The model's performance was evaluated using the metrics: accuracy of 98.83% , precision of 98.82%, recall of 98.81%, and F1 score of 98.81%.

3.5.6 Results Discussion Centralized Deep Learning Models:

We evaluate our models with (Accuracy,Precision, Recall, F1score), and the results are in Table 3.2 :

Table 3.2: Results discussion Centralized deep learning Models

Model	Accuracy	Precision	Recall	F1 score
ANN	97.94%	97.94%	94.91%	94.92%
CNN	99.25%	99.25%	99.25%	99.25%
RNN	97.53%	97.52%	97.50%	97.50%
LSTM	98.83%	98.82%	98.81%	98.81%

Choosing the most appropriate model involves considering the trade-off between performance metrics, the nature of the data, and the task. CNN outperforms all other models (ANN

, RNN and LSTM) in achieving the highest accuracy (99.19%) and almost perfect precision, recall, and F1-score. ,Convolutional Neural Network (CNN) emerges as a ,more effective model overall. For its crucial role in classifying and dealing with images and for its fit into the data set, which is images of numbers.

3.5.7 Summary of the CNN Model :

The following figure represents a comprehensive summary of the structure and formation of the layers of our CNN model .

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 20, 24, 24]             520
            Conv2d-2           [-1, 50, 8, 8]          25,050
            Linear-3                [-1, 500]         400,500
            Linear-4                 [-1, 10]           5,010
================================================================
Total params: 431,080
Trainable params: 431,080
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.12
Params size (MB): 1.64
Estimated Total Size (MB): 1.76
----------------------------------------------------------------
```

Figure 3.2: Summary of the CNN Model.

3.5.8 Plot Accuracy of the CNN Model :

Figure 3.3 in this subsection represents graphical curves showing the accuracy and loss values for training and testing our CNN model.

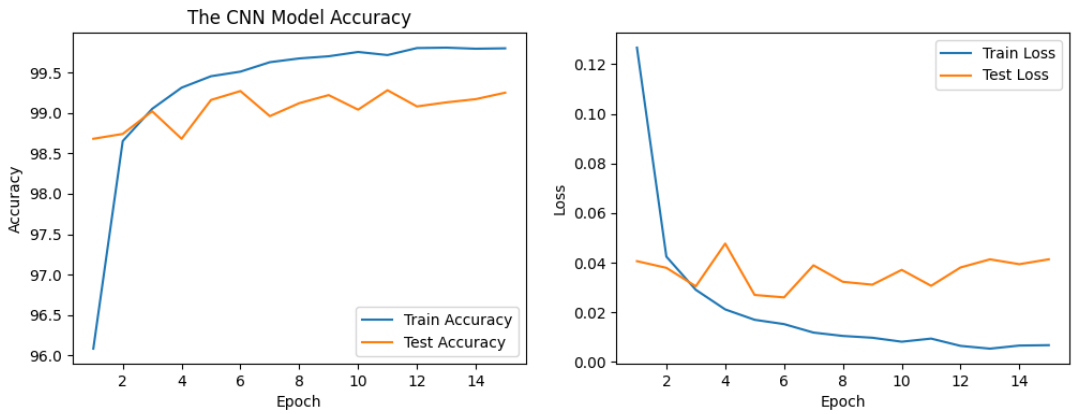

Figure 3.3: Plot Accuracy of the CNN Model.

3.5.9 Confusion Matrix of the CNN Model:

A confusion matrix provides a powerful tool for analyzing the performance of a CNN model for classifying handwritten digits. By systematically evaluating the matrix entries and deriving additional metrics, you can gain valuable insights to improve the model's accuracy and ability to differentiate between various digit classes. This scientific analysis paves the way for further optimization and refinement of the CNN model for the MNIST classification task. The next Figure 3.4 represents the confusion matrix of our CNN model .

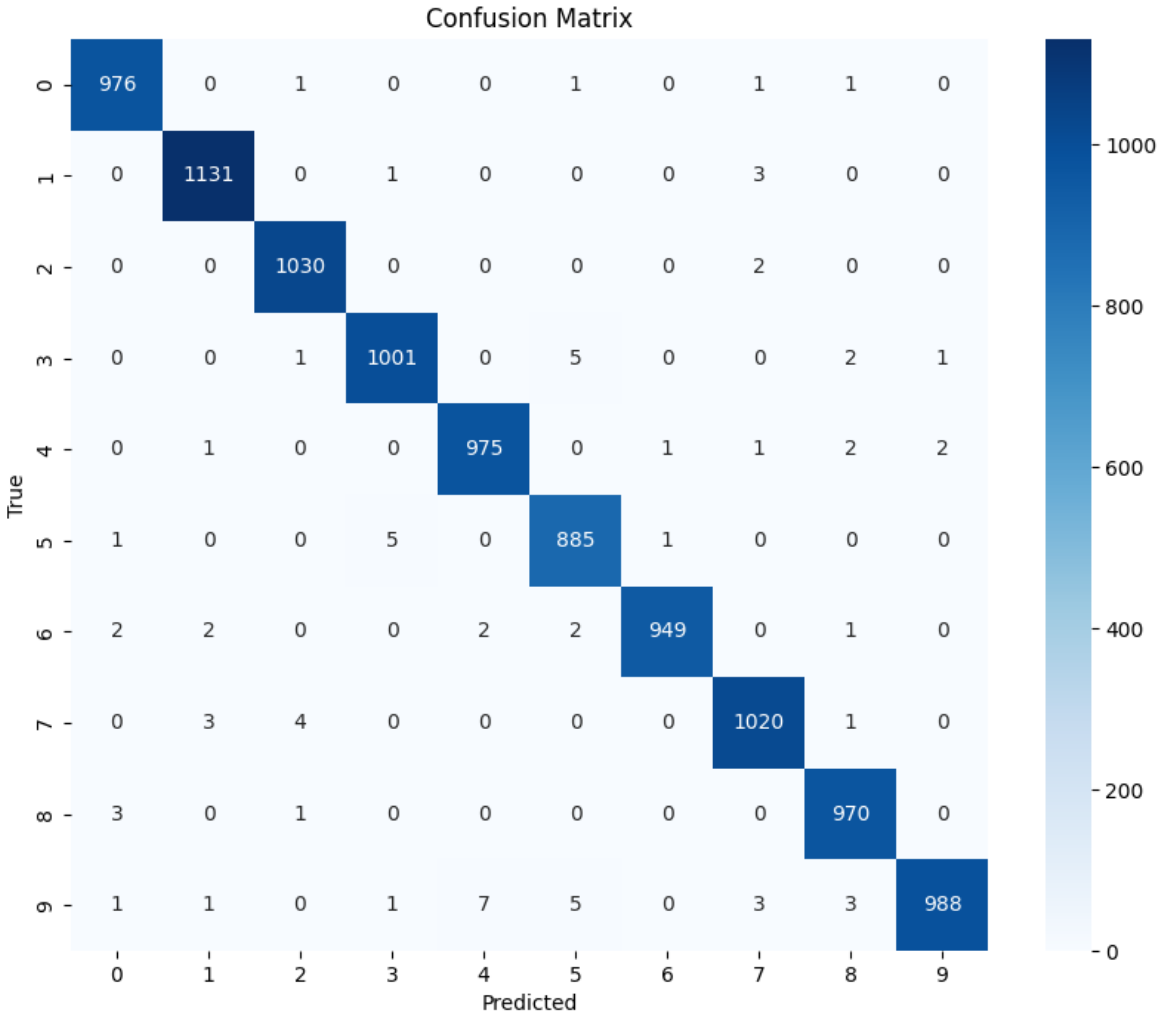

Figure 3.4: Confusion Matrix of the CNN Model.

3.5.10 Code Source

In this subsection, we present the source code for the central deep learning CNN model :

```
class CNN(nn.Module):

    def _init_(self):
        super(CNN, self)._init_()
        self.conv1 = nn.Conv2d(3, 20, 5, 1)
        self.conv2 = nn.Conv2d(20, 50, 5, 1)
        self.fc1 = nn.Linear(4*4*50, 500)
        self.fc2 = nn.Linear(500, 10)

    def forward(self, x):
        x = F.relu(self.conv1(x))
        x = F.max_pool2d(x, 2, 2)
        x = F.relu(self.conv2(x))
        x = F.max_pool2d(x, 2, 2)
        x = x.view(-1, 4*4*50)
        x = F.relu(self.fc1(x))
        x = self.fc2(x)
        return x

model = CNN()
```

Listing 3.1: Code Source of the CNN Model

3.6 Federated Learning Experimentation :

3.6.1 Data Partitioning and Distribution :

Data distribution and partitioning are key stages in building our project based on federated learning (FL) that guarantee the data is suitably distributed among several clients (nodes) for local model training. The effectiveness and efficiency of the federated learning system may be considerably impacted by this procedure. The following provides a thorough summary of data distribution and partitioning within the framework of federated learning :

3.6.1.1 Importance of Data Partitioning in Federated Learning :

1. **Preserving Data Ownership and Privacy:**
   - Federated learning allows each client to keep their data locally, ensuring that sensitive information never leaves the client's device. This setup inherently reduces the risk of data exposure and helps comply with stringent privacy regulations like GDPR and CCPA.
   - By partitioning data among clients, federated learning decentralizes the training process, meaning the model is trained across multiple clients without aggregating raw data on a central server.

2. **Compliance with Privacy Regulations:**
   - Data partitioning in federated learning helps organizations comply with privacy laws by ensuring that personal data remains within the confines of the data owner's device or local environment.
   - This approach minimizes the need for transferring large volumes of personal data across networks, thus reducing potential attack surfaces for data breaches.

In our project, we were more interested in studying and applying the division and symmetrical distribution of data to implement the federated learning project, and we will describe and Explanation to you the functions that we adopted in this regard:

3.6.1.2 iid_split Function :

The iid_split function is designed to partition a dataset into IID (Independent and Identically Distributed) subsets, which is a common requirement in federated learning to ensure each client has a representative sample of the entire dataset.

1. **Parameters:**
   - **dataset:** The dataset to be split.
   - **n_clients:** The number of clients among whom the dataset will be split.
   - **n_samples:** The number of samples each client will receive.
   - **batch_size:** The batch size for the data loaders created for each client.
   - **shuffle:** A boolean indicating whether to shuffle the dataset.
2. **Returns:** The function returns a list of **DataLoader** objects. Each **DataLoader** corresponds to a client's data and will be used in federated learning to train the model on the client's local data.

Here in the code is The iid_split Function :

```
def iid_split(dataset, n_clients, n_samples, batch_size, shuffle):
    data_loader = DataLoader(dataset, batch_size=n_samples,
   shuffle=shuffle)
    data_iter = iter(data_loader)
    partitions = [next(data_iter) for _ in range(n_clients)]
    loaders = [DataLoader(DatasetFromSubset(partition),
   batch_size=batch_size, shuffle=shuffle) for partition in
   partitions]
    return loaders
```

Listing 3.2: The iid_split Function

3.6.2 Defining a Client

In federated learning (FL), the primary objective is to empower individual clients, which could be devices or nodes, to conduct training on their respective datasets using their computa-

tional resources, such as CPUs or GPUs. Unlike traditional centralized approaches where data is collected and stored in a central server, FL allows the training process to take place locally on the clients' devices. This decentralized approach enhances privacy and security because sensitive data remains on the clients' devices and is not transmitted to a central location. By keeping data localized, FL mitigates the risks associated with storing sensitive information in a centralized manner, offering a more secure and privacy-preserving alternative for collaborative machine learning.

3.6.3 Federated Learning Strategy used

Choosing a strategy in federated learning involves selecting the appropriate algorithm and configuration to suit our federated learning task best. In our project, We used the FedAvg and FedProx strategy . Choosing between FedAvg and FedProx depends on the specific characteristics of our federated learning scenario. FedAvg is preferable for IID data due to its simplicity and efficiency. In contrast, FedProx is suitable for non-IID data, heterogeneous client capabilities, and situations where FedAvg faces instability or divergence. A careful evaluation of our data distribution, client capabilities, and observed training behavior will guide us to select the most appropriate strategy. In the second chapter, we talked in detail about these strategies.

3.7 Experimental Results

This work aims Federated Learning Experimentation and compare the performance of Federated Averaging (FedAvg) and Federated Proximal (FedProx) strategies on the MNIST dataset under Independent and Identically Distributed (IID) settings, considering different numbers of Clients (10, 15, and 20) and varying numbers of training rounds (10, 20, 30, and 40).

For each combination of the number of Clients and training rounds, we trained federated learning models using both FedAvg and FedProx strategies. The clients were randomly assigned to each training round. Model performance was evaluated on a separate test set after each training round to measure accuracy , precision, recall, and F1 score .

Therefore, we classified the results into 3 cases, that is, according to the number of clients. Then, in each case, we separated the presentation of the results according to the number of rounds in which we trained the model.

3.7.1 Results In the case of 10 clients :

The following tables presents the evaluation results of the federated learning server model trained across 10 clients with different training rounds (10, 20, 30, and 40) using the MNIST dataset under IID settings. The metrics include Accuracy, Precision, Recall, and F1 Score for two federated learning algorithms: FedAvg and FedProx.

1. **In 10 Rounds :**

Table 3.3: Results in of 10 Clients In 10 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	92.8%	92.67%	92.6%	92.56%
FedProx	92.7%	92.52%	92.48%	92.47%

2. **In 20 Rounds :**

Table 3.4: Results in of 10 Clients In 20 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	94.5%	94.4%	94.37%	94.36%
FedProx	94.5%	94.39%	94.37%	94.36%

3. **In 30 Rounds :**

Table 3.5: Results in of 10 Clients In 30 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	95.1%	95%	95%	94.98%
FedProx	95.19%	95.1%	95.09%	95.08%

4. **In 40 Rounds :**

Table 3.6: Results in of 10 Clients In 40 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	95.5%	95.41%	95.39%	95.39%
FedProx	95.5%	95.41%	95.39%	95.39%

Figure 3.5 shows the Accuracy, Precision, Recall, and F1 Score results for the federated learning model using the algorithms FedAvg and FedProx with 10 clients.

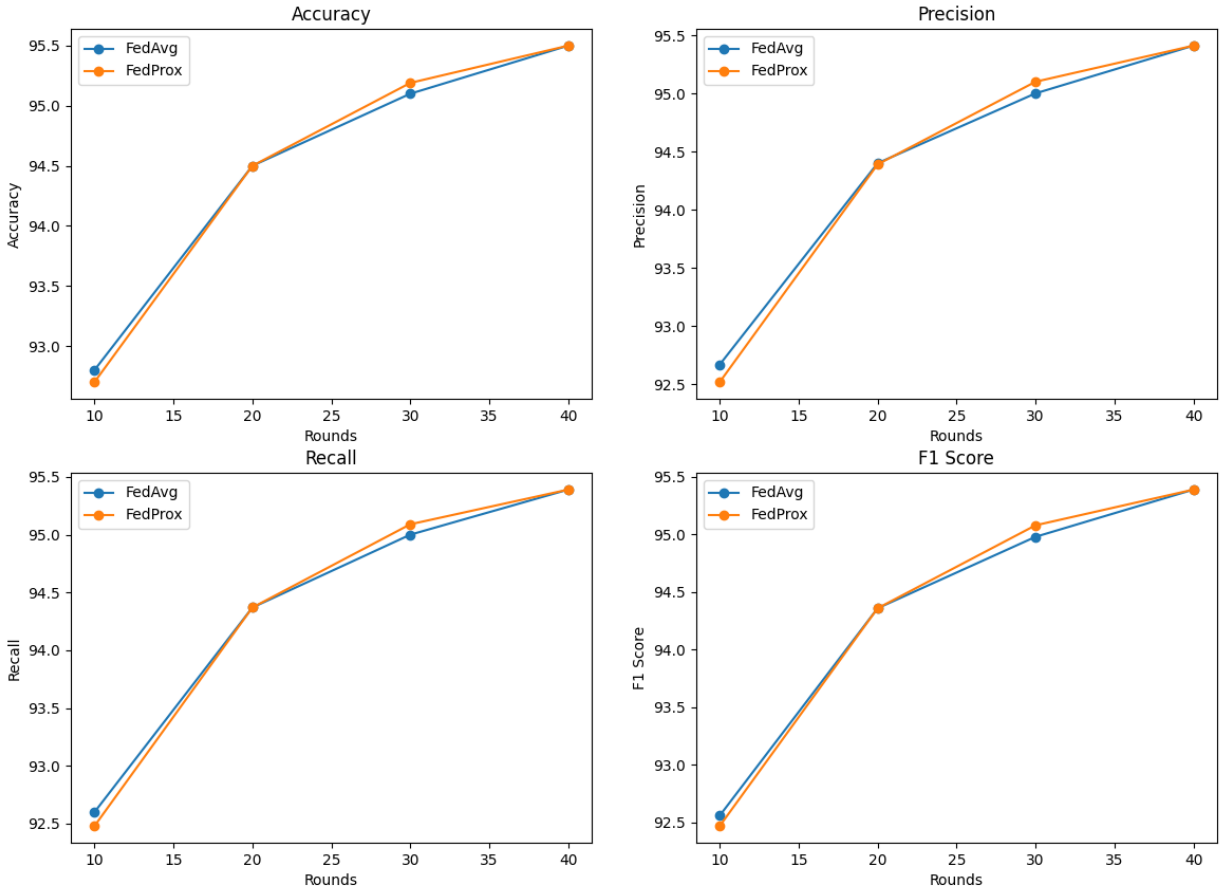

Figure 3.5: Plot Results in of 10 Clients.

The analysis shows that both FedAvg and FedProx are effective for federated learning on the MNIST dataset. Initially, after 10 rounds, FedAvg had a slight edge in accuracy (92.8% compared to FedProx's 92.7%). By 20 rounds, both algorithms achieved the same accuracy (94.5%). At 30 rounds, FedProx slightly outperformed FedAvg in accuracy (95.19% vs. 95.1%). By 40 rounds, both algorithms reached identical accuracy (95.5%).

In summary, both FedAvg and FedProx exhibit similar trends in Precision, Recall, and F1 Score metrics during early rounds of training, with FedAvg showing slightly better performance. However, by the 40th round, their performance metrics converge, indicating comparable effectiveness. This suggests that despite initial differences, both algorithms perform well in federated learning scenarios with small client sets and moderate training rounds, underscoring their robustness and effectiveness.

3.7.2 Results In the case of 15 Clients :

The following tables presents the evaluation results of the federated learning server model trained across 15 clients with different training rounds (10, 20, 30, and 40) using the MNIST dataset under IID settings. The metrics include Accuracy, Precision, Recall, and F1 Score for two federated learning algorithms: FedAvg and FedProx.

1. **In 10 Rounds :**

Table 3.7: Results in of 15 Clients In 10 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	95.26%	95.27%	95.18%	95.2%
FedProx	94.86%	94.83%	94.79%	94.79%

2. **In 20 Rounds :**

Table 3.8: Results in of 15 Clients In 20 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	96.6%	96.6%	96.58%	96.58%
FedProx	96.6%	96.61%	96.57%	96.57%

3. **In 30 Rounds :**

Table 3.9: Results in of 15 Clients In 30 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	96.8%	96.8%	96.77%	96.77%
FedProx	96.86%	96.86%	96.84%	96.84%

4. **In 40 Rounds :**

Table 3.10: Results in of 15 Clients In 40 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	97%	97%	97%	96.99%
FedProx	97.06%	97.06%	97.06%	97.05%

Figure 3.6 shows the Accuracy, Precision, Recall, and F1 Score results for the federated learning model using the algorithms FedAvg and FedProx with 15 clients.

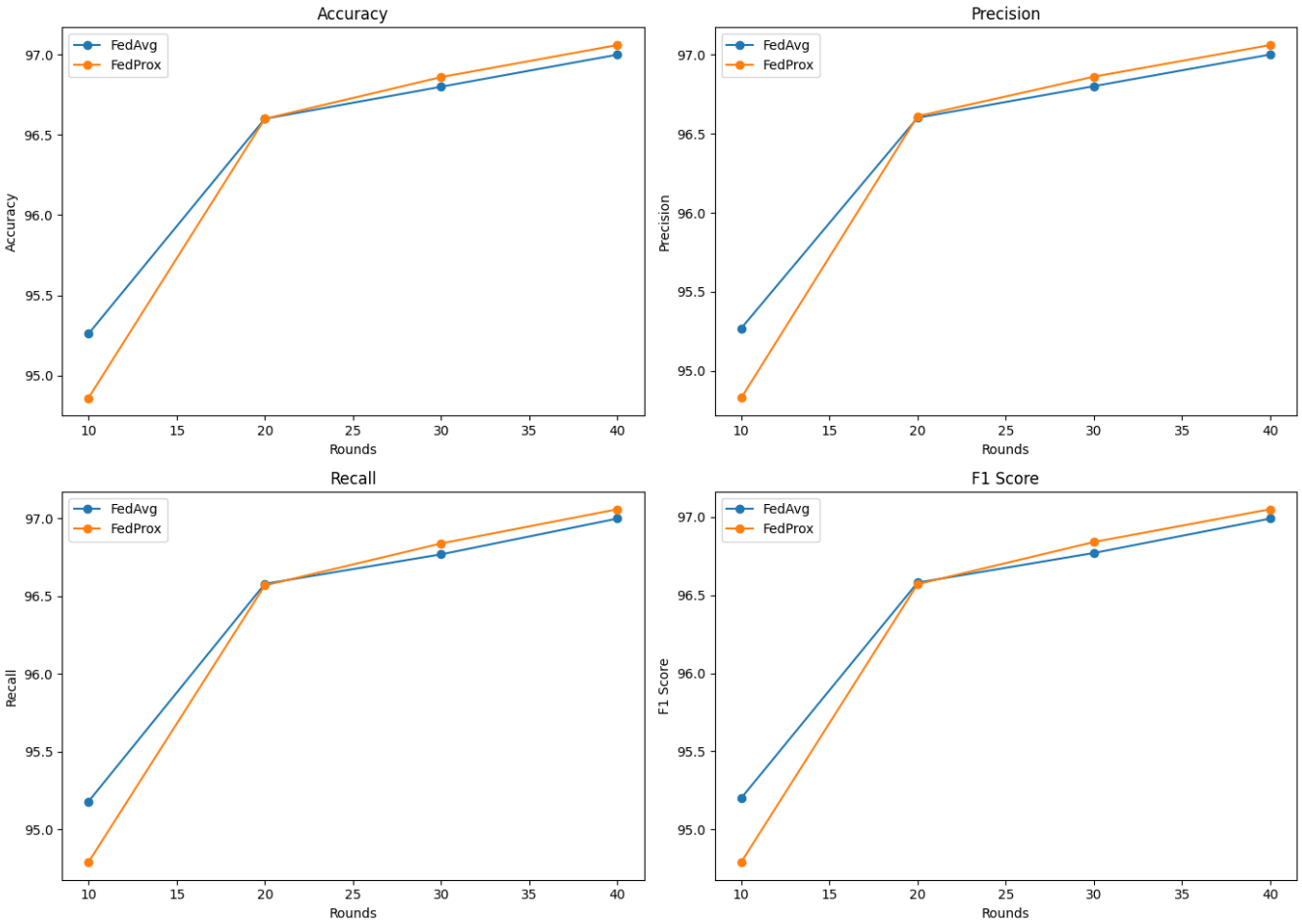

Figure 3.6: Plot Results in of 15 Clients.

The analysis indicates that both FedAvg and FedProx are highly effective for federated learning on the MNIST dataset. Initially, FedAvg had a slight advantage in accuracy at 10 rounds (95.26% compared to 94.86% for FedProx). By 20 rounds, both algorithms achieved the same accuracy (96.6%). At 30 rounds, FedProx slightly surpassed FedAvg in accuracy (96.86% vs. 96.8%). By 40 rounds, FedProx had a marginally higher accuracy (97.06% vs. 97% for FedAvg).

In summary, Precision, Recall, and F1 Score metrics exhibited consistent improvement with increasing rounds for both FedAvg and FedProx, with FedProx showing a slight advantage over FedAvg in later rounds. Overall, the results indicate that both algorithms perform exceptionally well in federated learning scenarios with a larger number of clients and extended training rounds. While FedProx holds a slight edge in later rounds, both algorithms demonstrate robust and reliable performance, affirming their suitability for federated learning applications.

3.7.3 Results In the case of 20 clients :

The following tables presents the evaluation results of the federated learning server model trained across 20 clients with different training rounds (10, 20, 30, and 40) using the MNIST dataset under IID settings. The metrics include Accuracy, Precision, Recall, and F1 Score for two federated learning algorithms: FedAvg and FedProx.

1. **In 10 Rounds :**

Table 3.11: Results in of 20 Clients In 10 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	94.1%	94.1%	94.05%	94.04%
FedProx	93.89%	93.9%	93.86%	93.84%

2. **In 20 Rounds :**

Table 3.12: Results in of 20 Clients In 20 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	95.89%	95.94%	95.88%	95.87%
FedProx	95.39%	95.48%	95.37%	95.39%

3. **In 30 Rounds :**

Table 3.13: Results in of 20 Clients In 30 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	96.6%	96.65%	96.58%	96.58%
FedProx	96.55%	96.61%	96.52%	96.54%

4. **In 40 Rounds :**

Table 3.14: Results in of 20 Clients In 40 Rounds

Metrics	Accuracy	Precision	Recall	F1score
FedAvg	96.89%	96.95%	96.9%	96.9%
FedProx	96.8%	96.86%	96.81%	96.8%

Figure 3.7 shows the Accuracy, Precision, Recall, and F1 Score results for the federated learning model using the algorithms FedAvg and FedProx with 20 clients.

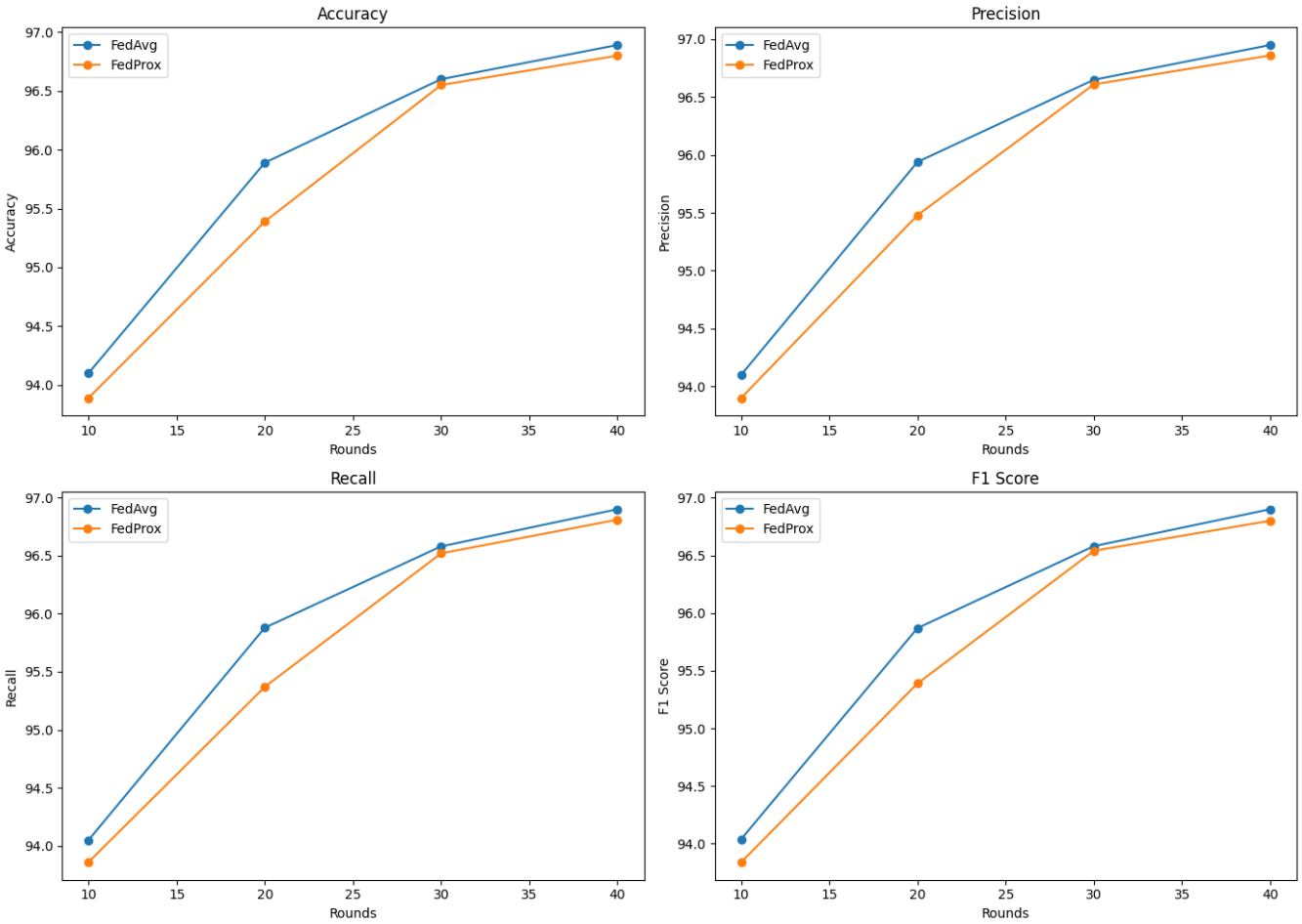

Figure 3.7: Plot Results in of 20 Clients.

The results indicate that both FedAvg and FedProx achieved high accuracy across different training rounds. FedAvg generally exhibited slightly higher accuracy compared to FedProx, especially in the later rounds. However, both algorithms showed consistent improvement in accuracy, precision, recall, and F1 score as the number of training rounds increased. Overall, both FedAvg and FedProx demonstrate effectiveness in federated learning with 20 clients on the MNIST dataset under IID settings, with FedAvg showing a slightly better performance across most metrics and rounds.

3.7.4 Comparing The Results of The Two Algorithms

In summary, our results indicate In the case of 10 clients, both FedAvg and FedProx demonstrated comparable performance across all metrics and rounds, with slight variations observed. As the number of clients increased to 15, FedAvg consistently showed slightly higher or comparable performance compared to FedProx across all metrics and rounds, indicating a slight advantage for FedAvg. Similarly, with 20 clients, FedAvg maintained its edge over FedProx, with both algorithms showing consistent improvement in performance metrics as the number of training rounds increased. Overall, our findings suggest that FedAvg tends to perform slightly better than FedProx across varying numbers of clients, demonstrating consistent and effective performance in federated learning scenarios with the MNIST dataset under IID settings.

3.8 Results Discussion

The results obtained in our work The results indicate that while centralized deep learning achieves the highest performance on the MNIST dataset, federated learning algorithms like FedAvg and FedProx still perform remarkably well. The centralized CNN model achieves superior metrics (Accuracy, Precision, Recall, and F1 Score), but both FedAvg and FedProx show strong performance with only a small reduction in accuracy and other metrics.

The MNIST dataset use case represents a relatively simple classification task and has been utilized in this study to demonstrate the feasibility and validate the effectiveness of the designed federated learning framework. The overarching goal of the study is to achieve an effective federated learning model while ensuring the security and privacy of data, particularly for applications in the Internet of Medical Things (IoMT) for healthcare.

Although centralized models achieve slightly higher performance, federated learning models like FedAvg and FedProx provide a viable alternative that balances performance with stringent privacy requirements. This makes federated learning particularly suitable for applications in the Internet of Medical Things, where secure and privacy-preserving data analytics are critical.

This analysis highlights the trade-offs between centralized and federated learning approaches and underscores the effectiveness of federated learning despite its inherent challenges.However, federated learning (FL) offers many advantages over centralized deep learning (CDL), especially in scenarios where data privacy, security, and decentralization are critical. The main advantages offered by federated learning are:

- **Privacy Preservation:** Federated learning ensures that sensitive data remains local to the devices or servers where it is stored, reducing the risk of data breaches and preserving user privacy. This is particularly important in healthcare applications where patient data confidentiality is paramount.
- **Data Security:** By keeping data decentralized and only sharing model updates rather than raw data, federated learning mitigates the risk of data exposure and unauthorized access. This enhances overall data security, which is crucial in environments like the Internet of Medical Things where security threats are prevalent.
- **Decentralization:** Federated learning allows for model training to occur on distributed devices or servers, avoiding the need for centralized data storage and processing. This decentralized approach enables scalability and resilience to system failures, improving overall system reliability.
- **Collaborative Learning:** Federated learning fosters collaboration among multiple stakeholders by enabling them to contribute to model training without sharing sensitive data. This collaborative approach promotes knowledge sharing and innovation while respecting data privacy and security.
- **Cost-Efficiency:** Federated learning reduces the need for large-scale data aggregation and centralized infrastructure, resulting in cost savings in terms of data transmission, storage, and processing. This makes it an attractive option for resource-constrained environments.

In the next image , a bar graph shows the results of our experiment and compares training the centralized deep learning model and the federated learning model.

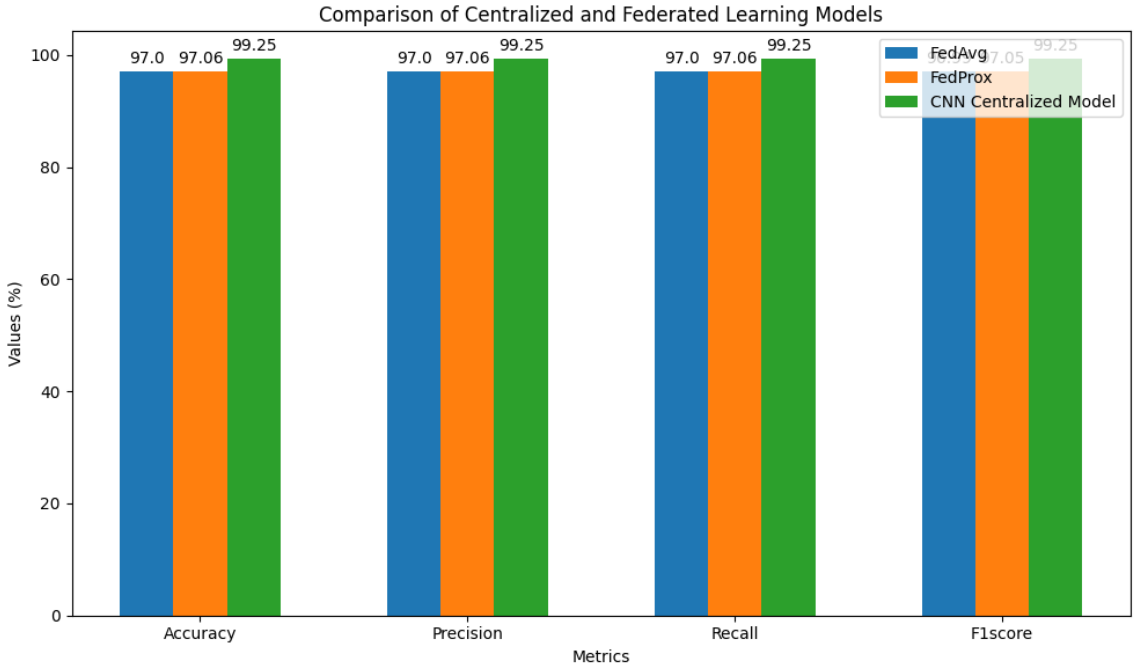

Figure 3.8: Compare Experiment Results.

In summary, while centralized deep learning may offer slightly higher performance in some cases, federated learning provides a compelling alternative that prioritizes privacy, security, and decentralization. These advantages make federated learning particularly well-suited for applications in the Internet of Medical Things and other domains where data privacy and security are critical concerns.

3.9 Conclusion

In conclusion,This chapter presents the implementation process of our project, dissecting foundational components for constructing centralized deep learning models and scrutinizing results through diverse evaluation factors. We shed light on the tools and techniques employed, emphasizing their pivotal role in our achievements. Our rigorous analysis uncovered valuable insights into model efficacy, strengths. While centralized deep learning showcases marginal superiority, federated learning emerges as a potent alternative prioritizing privacy and decentralization, particularly crucial in healthcare contexts like the Internet of Medical Things.

General Conclusion

This Thesis addressed the study of the impact of applying Federated Learning (FL) in Medical Internet of Things (IoMT) devices compared to centralized learning, such as traditional machine learning, with a focus on achieving data security. We have demonstrated how Federated Learning can enhance the security and efficiency of data processing in medical IoT systems without the need to transfer raw data to centralized data centers.

We successfully illustrated that Federated Learning excels in ensuring data security in medical IoT devices, unlike centralized learning. By training AI models across multiple devices without the need to transfer raw data, Federated Learning maintains the privacy and security of sensitive information, making it particularly suitable for applications requiring data protection, such as healthcare.

Despite the positive results, we encountered some challenges that deserve mention. These challenges include the need to improve security protocols to counter cyber-attacks, as well as the necessity for effective power management in connected devices. Implementing Federated Learning requires high computing resources and advanced coordination among different devices, which can be complex in certain environments.

In our future work on Federated Learning for Internet of Medical Things (IoMT) Healthcare Applications, we plan to explore the following areas:

- **Privacy-Preserving Techniques:** Investigating advanced privacy-preserving techniques within federated learning to enhance data security and confidentiality. This includes exploring techniques such as differential privacy, homomorphic encryption, and secure multi-party computation to further safeguard sensitive medical data.
- **Testing IoT Device Capabilities:** There is a critical need to assess the capabilities of IoT devices for running Federated Learning (FL) models, as there currently exists no standardized minimum requirement. Understanding the feasibility and limitations of running FL models on IoT devices is essential for practical deployment in healthcare settings.
- **Evaluation on Larger Datasets under non-IID settings :** We're focusing on evaluating models on larger datasets under non-IID (non-Independently and Identically Distributed) conditions to understand the robustness and generalization capabilities of federated learning models in diverse environments. This will help develop more adaptable models that handle medical data complexities while maintaining patient privacy.

In conclusion, this study has demonstrated that Federated Learning can provide an effective and secure solution for data processing in medical IoT devices, outperforming centralized learning in terms of data security. With continuous technological advancements, we look forward to seeing more innovations leveraging this technique to enhance data security and quality across different sectors. We recommend further research to develop solutions that enhance security and privacy in IoT and Federated Learning applications. We also advocate for regulatory poli-

cies and government support to promote the use of these technologies in the healthcare sector, especially in remote areas.

Bibliography

[1] "What is IoT (Internet of Things) and How Does it Work?|Definition from TechTarget," IoT Agenda. Accessed: Mar. 04, 2024. [Online]. Available: https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT

[2] S. Alam et al., "Blockchain-Based Solutions Supporting Reliable Healthcare for Fog Computing and Internet of Medical Things (IoMT) Integration," Sustainability, vol. 14, no. 22, Art. no. 22, Jan. 2022, doi: 10.3390/su142215312.

[3] R. Hireche, H. Mansouri, and A.-S. K. Pathan, "Security and Privacy Management in Internet of Medical Things (IoMT): A Synthesis," J. Cybersecurity Priv., vol. 2, no. 3, Art. no. 3, Sep. 2022, doi: 10.3390/jcp2030033.

[4] S. F. Ahmed, S. B. Alam, S. Afrin, S. J. Rafa, N. Rafa, and A. H. Gandomi, "Insights into Internet of Medical Things (IoMT): Data fusion, security issues and potential solutions," Inf. Fusion, 2024.

[5] S. Rani, A. Kataria, S. Kumar, and P. Tiwari, "Federated learning for secure IoMT-applications in smart healthcare systems: A comprehensive review," Knowl.-Based Syst., vol. 274, p. 110658, Aug. 2023, doi: 10.1016/j.knosys.2023.110658.

[6] Gupta, Nishu, and Sara Paiva, eds. IoT and ICT for healthcare applications. Cham, Switzerland: Springer, 2020.

[7] MAJEED, Fahad; NAZIR, Maria; SCHNEIDER, Jens. ISA: Internet of Medical Things (IoMT) in Smart Healthcare and its Applications: A Review. In: 2023 3rd International Conference on Artificial Intelligence (ICAI). IEEE, 2023. p. 129-135.

[8] Ghosh, U., Chakraborty, C., Garg, L., Srivastava, G. (Eds.). (2022). Intelligent Internet of Things for Healthcare and Industry. Springer International Publishing.

[9] R. Hireche, H. Mansouri, and A.-S. K. Pathan, "Security and Privacy Management in Internet of Medical Things (IoMT): A Synthesis," J. Cybersecurity Priv., vol. 2, no. 3, pp. 640–661, Aug. 2022, doi: 10.3390/jcp2030033.

[10] S. Alam et al., "Blockchain-Based Solutions Supporting Reliable Healthcare for Fog Computing and Internet of Medical Things (IoMT) Integration," Sustainability, vol. 14, no. 22, p. 15312, Nov. 2022, doi: 10.3390/su142215312.

[11] R. Dwivedi, D. Mehrotra, and S. Chandra, "Potential of Internet of Medical Things (IoMT) applications in building a smart healthcare system: A systematic review," J. Oral Biol. Craniofacial Res., vol. 12, no. 2, pp. 302–318, Mar. 2022, doi: 10.1016/j.jobcr.2021.11.010.

[12] leonid, "Internet of Medical Things: Challenges and Adoptions," CIFS Health. Accessed: Mar. 05, 2024. [Online]. Available: https://cifs.health/backgrounds/internet-of-medical-things-challenges-and-adoptions/

[13] "IoMT Security: Risks Best Practices to Secure IoMT Devices." Accessed: Mar. 06, 2024. [Online]. Available: https://binariks.com/blog/iomt-security-risks-best-practices/

[14] Harvey, P., Toutsop, O., Kornegay, K., Alale, E., Reaves, D. (2020, December). Security and privacy of medical internet of things devices for smart homes. In 2020 7th International Conference on Internet of Things: Systems, Management and Security (IOTSMS) (pp. 1-6). IEEE.

[15] Y. Sun, "Security and Privacy for the Internet of Medical Things Enabled Healthcare Systems: A Survey," vol. 7, 2019.

[16] Ameen, A. H., Mohammed, M. A., Rashid, A. N. (2023). Dimensions of artificial intelligence techniques, blockchain, and cyber security in the Internet of medical things: Opportunities, challenges, and future directions. Journal of Intelligent Systems, 32(1), 20220267.

[17] Sundaresan, S.; Doss, R.; Zhou, W. RFID in healthcare–current trends and the future. In Springer Series in Bio-/Neuroinformatics; Kasabov, N., Ed.; Springer: Berlin/Heidelberg, Germany, 2015; Volume 5, pp. 839–870.

[18] Sarigiannidis, P.; Karapistoli, E.; Economides, A.A. Detecting sybil attacks in wireless sensor networks using UWB ranging-based information. Expert Syst. Appl. 2015, 42, 7560–7572. [CrossRef]

[19] Peng, H. WIFI network information security analysis research. In Proceedings of the 2nd IEEE International Conference on Consumer Electronics, Communications and Networks, Yichang, China, 21–23 April 2012.

[20] Yang, X.; Karampatzakis, E.; Doerr, C.; Kuipers, F. Security vulnerabilities in LoRaWAN. In Proceedings of the IEEE/ACM 3rd International Conference on Internet-of-Things Design and Implementation, Orlando, FL, USA, 17–20 April 2018.

[21] Duggal, A. HL7 2. x security. In Proceedings of the 8th Annual HITB Security Conference, Amsterdam, The Netherlands, 10–14 April 2017.

[22] Flury, M.; Poturalski, M.; Papadimitratos, P.; Hubaux, J.P.; Le Boudec, J.Y. Effectiveness of distance-decreasing attacks against impulse radio ranging. In Proceedings of the 3rd ACM Conference on Wireless Network Security, Hoboken, NJ, USA, 22–24 March 2010.

[23] Yaacoub, J.P.A.; Noura, M.; Noura, H.N.; Salman, O.; Yaacoub, E.; Couturier, R.; Chehab, A. Securing internet of medical things systems: Limitations, issues and recommendations. Future Gener. Comput. Syst. 2020, 105, 581–606. [CrossRef]

[24] Kumar, R.; Tripathi, R. Towards design and implementation of security and privacy framework for internet of medical things (iomt) by leveraging blockchain and ipfs technology. J. Supercomput. 2021, 77, 7916–7955. [CrossRef]

[25] Saif, S.; Biswas, S.; Chattopadhyay, S. Intelligent, secure big health data management using deep learning and blockchain technology: An overview. In Deep Learning Techniques for Biomedical and Health Informatics; Dash, S., Acharya, B., Mittal, M., Abraham, A., Kelemen, A., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 68, pp. 187–209.

[26] Maji, S.; Banerjee, U.; Fuller, S.H.; Abdelhamid, M.R.; Nadeau, P.M.; Yazicigil, R.T.; Chandrakasan, A.P. A low-power dual-Factor authentication unit for secure implantable devices. In Proceedings of the IEEE Custom Integrated Circuits Conference, Newport Beach, CA, USA, 22–25 March 2020.

[27] Andrea, I.; Chrysostomou, C.; Hadjichristofi, G. Internet of things: Security vulnerabilities and challenges. In Proceedings of the IEEE Symposium on Computers and Communication, Larnaca, Cyprus, Greek, 6–9 July 2015.

[28] Abosata, N.; Al-Rubaye, S.; Inalhan, G.; Emmanouilidis, C. Internet of things for system integrity: A comprehensive survey on security, attacks and countermeasures for industrial applications. Sensors 2021, 21, 3654. [CrossRef]

[29] Jafarnia-Jahromi, A.; Broumandan, A.; Nielsen, J.; Lachapelle, G. GPS vulnerability to spoofing threats and a review of antispoofing techniques. Int. J. Navig. Obs. 2012, 2012, 127072.

[30] Kalyani, G.; Chaudhari, S. An efficient approach for enhancing security in Internet of Things using the optimum authentication key. Int. J. Comput. Appl. 2020, 42, 306–314. [CrossRef]

[31] Salem, O.; Alsubhi, K.; Shaafi, A.; Gheryani, M.; Mehaoua, A.; Boutaba, R. Man-in-the-Middle Attack Mitigation in Internet of Medical Things. IEEE Trans. Ind. Inform. 2022, 18, 2053–2062.

[32] Khader, R.; Eleyan, D. Survey of DoS/DDoS attacks in IoT. Sust. Eng. Innov. 2021, 3, 23–28. [CrossRef]

[33] Sharma, M.; Arora, B. Detection and prevention of DoS and DDoS in IoT. In Lecture Notes in Networks and Systems; Singh, P.K., Wierzcho ´n, S.T., Tanwar, S., Ganzha, M., Rodrigues, J.J.P.C., Eds.; Springer: Singapore, 2021; Volume 203, pp. 845–855.

[34] Sethuraman, S.C.; Vijayakumar, V.; Walczak, S. Cyber-attacks on healthcare devices using unmanned aerial vehicles. J. Med. Syst. 2020, 44, 29. [CrossRef]

[35] Pathan, A.-S.K.; Lee, H.-W.; Hong, C.S. Security in wireless sensor networks: Issues and challenges. In Proceedings of the 8th International Conference on Advanced Communication Technology (IEEE ICACT 2006), Gangwon, Korea, 20–22 February 2006; Volume II.

[36] Marin-Jiménez, M.J.; Castro, F.M.; Guil, N.; De la Torre, F.; Medina-Carnicer, R. Deep multi-task learning for gait-based biometrics.In Proceedings of the IEEE International Conference on Image Processing, Beijing, China, 17–20 September 2017.

[37] Schwartz, O.; Mathov, Y.; Bohadana, M.; Elovici, Y.; Oren, Y. Opening pandora's box: Effective techniques for reverse engineering IoT Devices. In Proceedings of the International Conference on Smart Card Research and Advanced Applications, Lugano,Switzerland, 13–15 November 2017.

[38] Pathan, A.-S.K.; Kindy, D.A. Lethality of SQL injection against current and future internet-technologies. Int. J. Comput. Sci. Eng.2014, 9, 386–394. [CrossRef]

[39] P. M. Mammen, "Federated Learning: Opportunities and Challenges." arXiv, Jan. 13, 2021. Accessed: May 06, 2024. [Online]. Available: http://arxiv.org/abs/2101.05428

[40] "Flower Framework main." Accessed: May 06, 2024. [Online]. Available: https://flower.ai/docs/framework/tutorial-series-what-is-federated-learning.html

[41] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and S. Avestimehr, "Federated Learning for Internet of Things: Applications, Challenges, and Opportunities." arXiv, Apr. 05, 2022. Accessed: May 06, 2024. [Online]. Available: http://arxiv.org/abs/2111.07494

[42] T. Sun, D. Li, and B. Wang, "Decentralized Federated Averaging." arXiv, Apr. 22, 2021. Accessed: May 06, 2024. [Online]. Available: http://arxiv.org/abs/2104.11375

[43] M. Dhada, A. K. Jain, and A. K. Parlikad, "Empirical Convergence Analysis of Federated Averaging for Failure Prognosis," IFAC-PapersOnLine, vol. 53, no. 3, pp. 360–365, 2020, doi: 10.1016/j.ifacol.2020.11.058.

[44] S. Bharati, M. R. H. Mondal, P. Podder, and V. B. S. Prasath, "Federated learning: Applications, challenges and future directions," HIS, vol. 18, no. 1–2, pp. 19–35, May 2022, doi: 10.3233/HIS-220006.

[45] T. O. Ayodele, "Machine Learning Overview," New Advances in Machine Learning.

[46] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review".

[47] S. Pouyanfar et al., "A Survey on Deep Learning: Algorithms, Techniques, and Applications," ACM Comput. Surv., vol. 51, no. 5, pp. 1–36, Sep. 2019, doi: 10.1145/3234150.

[48] V. Nasteski, "An overview of the supervised machine learning methods," HORIZONS.B, vol. 4, pp. 51–62, Dec. 2017, doi: 10.20544/HORIZONS.B.04.1.17.P05.

[49] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50–60, May 2020, doi: 10.1109/MSP.2020.2975749.

[50] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," Computers Industrial Engineering, vol. 149, p. 106854, Nov. 2020, doi: 10.1016/j.cie.2020.106854.

[51] X.-T. Yuan and P. Li, "On Convergence of FedProx: Local Dissimilarity Invariant Bounds, Non-smoothness and Beyond".

[52] A. Rauniyar et al., "Federated Learning for Medical Applications: A Taxonomy, Current Trends, Challenges, and Future Research Directions." arXiv, Oct. 29, 2023. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/2208.03392

[53] Supriya Shende and Government College of Engineering, Amravati, "Long Short-Term Memory (LSTM) Deep Learning Method for Intrusion Detection in Network Security," IJERT, vol. V9, no. 06, p. IJERTV9IS061016, Jul. 2020, doi: 10.17577/IJERTV9IS061016.

[54] CAO, Longbing. Beyond iid: Non-iid thinking, informatics, and learning. IEEE Intelligent Systems, 2022, 37.4: 5-17.

[55] H. Bodagala and P. H, "Security for IoT using Federated Learning," in 2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems (ICMACC), Hyderabad, India: IEEE, Dec. 2022, pp. 131–136. doi: 10.1109/ICMACC54824.2022.10093557.

[56] H. Elayan, M. Aloqaily, and M. Guizani, "Deep Federated Learning for IoT-based Decentralized Healthcare Systems," in 2021 International Wireless Communications and Mobile Computing (IWCMC), Harbin City, China: IEEE, Jun. 2021, pp. 105–109. doi: 10.1109/IWCMC51323.2021.9498820.

[57] T. Borja, D. Alamillo, A. Anhari, and I. Demirkol, "Scalable Federated Learning Simulations Using Virtual Client Engine in Flower," in 2023 31st Signal Processing and Communications Applications Conference (SIU), Istanbul, Turkiye: IEEE, Jul. 2023, pp. 1–4. doi: 10.1109/SIU59756.2023.10223791.

[58] K. Pereira, A. Parikh, P. Kumar, and K. Devadkar, "Healthcare Diagnostics Service Using Federated Learning," in 2023 International Conference for Advancement in Technology (ICONAT), Goa, India: IEEE, Jan. 2023, pp. 1–6. doi: 10.1109/ICONAT57137.2023.10080053.

[59] J. Peta and S. Koppu, "Enhancing Breast Cancer Classification in Histopathological Images through Federated Learning Framework," IEEE Access, vol. 11, pp. 61866–61880, 2023, doi: 10.1109/ACCESS.2023.3283930.

[60] Y. Otoum, Y. Wan, and A. Nayak, "Federated Transfer Learning-Based IDS for the Internet of Medical Things (IoMT)," in 2021 IEEE Globecom Workshops (GC Wkshps), Madrid, Spain: IEEE, Dec. 2021, pp. 1–6. doi: 10.1109/GCWkshps52748.2021.9682118.

[61] S. Rachakonda et al., "Privacy Enhancing and Scalable Federated Learning to Accelerate AI Implementation in Cross-Silo and IoMT Environments," IEEE J. Biomed. Health Inform., vol. 27, no. 2, pp. 744–755, Feb. 2023, doi: 10.1109/JBHI.2022.3185418.

[62] A. K. Nair, J. Sahoo, and E. D. Raj, "Privacy preserving Federated Learning framework for IoMT based big data analysis using edge computing," Comput. Stand. Interfaces, vol. 86, p. 103720, Aug. 2023, doi: 10.1016/j.csi.2023.103720.

[63] L. Zhao and J. Huang, "A distribution information sharing federated learning approach for medical image data," Complex Intell. Syst., vol. 9, no. 5, pp. 5625–5636, Oct. 2023, doi: 10.1007/s40747-023-01035-1.

[64] A. Makkar and K. Santosh, "SecureFed: federated learning empowered medical imaging technique to analyze lung abnormalities in chest X-rays," Int. J. Mach. Learn. Cybern., vol. 14, no. 8, pp. 2659–2670, Aug. 2023, doi: 10.1007/s13042-023-01789-7.

[65] H. AlSalman, M. S. Al-Rakhami, T. Alfakih, and M. M. Hassan, "Federated Learning Approach for Breast Cancer Detection Based on DCNN," IEEE Access, vol. 12, pp. 40114–40138, 2024, doi: 10.1109/ACCESS.2024.3374650.

[66] Banabilah S, Aloqaily M, Alsayed E, Malik N, Jarar-weh. Y (2022) Federated learning review: fundamentals, enabling technologies, and future applications. Inf Process Manage 59(6):103061

[67] Nguyen DC, Ding M, Pathirana PN, Seneviratne A, Li J, Vincent Poor H (2021) Federated learning for internet of things: a comprehensive survey. IEEE Commun Surv Tutori 23(3):1622–1658

[68] Alam T, Gupta R (2022) Federated learning and its role in the privacy preservation of IoT devices. Future Int 14(9):246

[69] Elayan, Haya, Moayad Aloqaily, and Mohsen Guizani. (2021): Deep federated learn- ing for IoT-based decentralized healthcare systems. In 2021 International Wireless Communications and Mobile Computing (IWCMC), 105–109. IEEE

[70] Elayan H, Aloqaily M, Guizani M (2021) Sustainability of healthcare data analysis IoT-based systems using deep federated learning. IEEE Int Things J 9(10):7338–7346

[71] Singh S, Rathore S, Alfarraj O, Tolba A, Yoon B (2022) A framework for privacy-preservation of IoT healthcare data using Federated Learning and blockchain technology. Future Gener Comput Syst 129:380–388

[72] Brisimi TS, Chen R, Mela T, Olshevsky A, Paschalidis IC, Shi W (2018) Federated learning of predictive models from federated electronic health records. Int J Med Inf 112:59–67

[73] Dai, W., Brisimi, T. S., Adams, W. G., Mela, T., Saligrama, V., Paschalidis, I. C. (2015). Prediction of hospitalization due to heart diseases by supervised learning methods. International journal of medical informatics, 84(3), 189-197.

[74] Blanquer I, Brasileiro F, Brito A, Calatrava A, Carvalho A, Fetzer C, Silva F (2020) Federated and secure cloud services for building medical image classifers on an intercontinental infrastructure. Future Gener Comput Syst 110:119–134

[75] Stephanie, Veronika, Ibrahim Khalil, Mohammed Atiquzzaman, and Xun Yi. (2022): Trustworthy privacy-preserving hierarchical ensemble and federated learning in healthcare 4.0 with blockchain. IEEE Transac Ind Inf

[76] Zhang, Li, Jianbo Xu, Pandi Vijayakumar, Pradip Kumar Sharma, and Uttam Ghosh. (2022): Homomorphic encryption-based privacy-preserving federated learning in IoT-enabled healthcare system." IEEE Transac Netw Sci Eng

[77] Khoa, Tran Anh, Do-Van Nguyen, Minh-Son Dao, and Koji Zettsu. (2021): Fed xData: a federated learning framework for enabling contextual health monitoring in a cloud-edge network. In 2021 IEEE International Conference on Big Data (Big Data), pp 4979–4988. IEEE

[78] Lian, Zhuotao, Qinglin Yang, Weizheng Wang, Qingkui Zeng, Mamoun Alazab, Hong Zhao, and Chunhua Su. (2022): DEEPFEL: decentralized, efcient and privacy- enhanced federated edge learning for healthcare cyber physical systems. IEEE Transac Netw Sci Eng

[79] Astillo PV, Duguma DG, Park H, Kim J, Kim B, You I (2022) Federated intelligence of anomaly detection agent in IoTMD- enabled diabetes management control system. Future Gener Comput Sys t 128:395–405

[80] M. A. Rahman, M. S. Hossain, M. S. Islam, N. A. Alrajeh, and G. Muhammad, "Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach," Ieee Access, 2020.

[81] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial iot," IEEE Transactions on Industrial Informatics, 2019.

[82] L. Huang and et. al., "Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records," Journal of biomedical informatics, 2019.

[83] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," International journal of medical informatics, 2018.

[84] H. Chen, H. Li, G. Xu, Y. Zhang, and X. Luo, "Achieving privacypreserving federated learning with irrelevant updates over e-health applications," in ICC 2020-2020 IEEE International Conference on Communications (ICC), 2020, pp. 1–6.

[85] O. Choudhury, A. Gkoulalas-Divanis, T. Salonidis, I. Sylla, Y. Park, G. Hsu, and A. Das, "Differential privacy-enabled federated learning for sensitive health data," arXiv preprint arXiv:1910.02578, 2019.

[86] P. Yu and Y. Liu, "Federated object detection: Optimizing object detection model with federated learning," in Proceedings of the 3rd Conference on Vision, Image and Signal Processing, 2019.

[87] H. Jiang, M. Liu, B. Yang, Q. Liu, J. Li, and X. Guo, "Customized federated learning for accelerated edge computing with heterogeneous task targets," Computer Networks, 2020.

[88] J. T. Raj, "Building decentralized image classifiers with federated learning," in 2020 IEEE Region 10 Symposium, 2020, pp. 489–494.

[89] J. Feng, Y. X. Xu, Y. G. Wang, and Y. Jiang, "Federated soft gradient boosting machine for streaming data," in Federated Learn., Cham, Switzerland: Springer, 2020, pp. 93–107

[90] A. Baldominos, Y. Saez, and P. Isasi, "A Survey of Handwritten Character Recognition with MNIST and EMNIST," Appl. Sci., vol. 9, no. 15, p. 3169, Aug. 2019, doi: 10.3390/app9153169.

[91] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," Insights Imaging, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.

[92] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning." arXiv, Nov. 08, 2018. Accessed: May 11, 2024. [Online]. Available: http://arxiv.org/abs/1811.03378

[93] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning." arXiv, Nov. 08, 2018. Accessed: May 11, 2024. [Online]. Available: http://arxiv.org/abs/1811.03378

[94] H. Elayan, M. Aloqaily, and M. Guizani, "Deep Federated Learning for IoT-based Decentralized Healthcare Systems," in 2021 International Wireless Communications and Mobile Computing (IWCMC), Harbin City, China: IEEE, Jun. 2021, pp. 105–109. doi: 10.1109/IWCMC51323.2021.9498820.

[95] M. Abadi et al., "TensorFlow: A system for large-scale machine learning." arXiv, May 31, 2016.

[96] "Welcome to Python.org," Python.org. Accessed: Jun. 02, 2024. [Online]. Available: https://www.python.org/

[97] "Project Jupyter." Accessed: Jun. 02, 2024. [Online]. Available: https://jupyter.org

[98] "Google Colab." Accessed: Jun. 02, 2024. [Online]. Available: https://colab.research.google.com/

[99] "Keras: Deep Learning for humans." Accessed: Jun. 02, 2024. [Online]. Available: https://keras.io/

[100] "NumPy -." Accessed: Jun. 02, 2024. [Online]. Available: https://numpy.org/

[101] "TensorFlow." Accessed: Jun. 02, 2024. [Online]. Available: https://www.tensorflow.org/?hl=ar

[102] “pandas - Python Data Analysis Library.” Accessed: Jun. 02, 2024. [Online]. Available: https://pandas.pydata.org/

[103] “Matplotlib — Visualization with Python.” Accessed: Jun. 02, 2024. [Online]. Available: https://matplotlib.org/

[104] “PyTorch.” Accessed: Jun. 02, 2024. [Online]. Available: https://pytorch.org/

[105] T. F. Authors, “Flower: A Friendly Federated Learning Framework.” Accessed: Jun. 02, 2024. [Online]. Available: https://flower.ai/

[106] Yeung, M., Sala, E., Schönlieb, C. B., Rundo, L. (2022). Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. Computerized Medical Imaging and Graphics, 95, 102026.

[107] Bottou, L. (1991). Stochastic gradient learning in neural networks. Proceedings of Neuro-Nımes, 91(8), 12.

[108] Bock, S., Weiß, M. (2019, July). A proof of local convergence for the Adam optimizer. In 2019 international joint conference on neural networks (IJCNN) (pp. 1-8). IEEE.