BLOCKCHAIN BASED FEDERATED LEARNING MODEL FOR PNUEMONIA DETECTION

A PROJECT REPORT

Submitted By

S. A. MADHULICA 195001094

VASUNDHHARA KATOCH 195001125

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Department of Computer Science and Engineering

Sri Sivasubramaniya Nadar College of Engineering
(An Autonomous Institution, Affiliated to Anna University)
Kalavakkam - 603110

May 2023

Sri Sivasubramaniya Nadar College of Engineering

(An Autonomous Institution, Affiliated to Anna University)

BONAFIDE CERTIFICATE

Certified that this project report titled **“BLOCKCHAIN BASED FEDERATED LEARNING MODEL FOR PNUEMONIA DETECTION”** is the *bonafide* work of “**SALIBINDLA AROGYA MADHULICA (195001094)**, **VASUNDHHARA SINGH KATOCH (195001125)** ” who carried out the project work under my supervision.

Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. T. T. MIRNALINEE
HEAD OF THE DEPARTMENT
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Dr. S. SARASWATHI
SUPERVISOR
Associate Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Place:
Date:

Submitted for the examination held on. . . . .

Internal Examiner **External Examiner**

ACKNOWLEDGEMENTS

I thank GOD, the almighty for giving me strength and knowledge to do this project.

I would like to thank and deep sense of gratitude to my guide **Dr. S. SARASWATHI**, Associate Professor, Department of Computer Science and Engineering, for her valuable advice and suggestions as well as her continued guidance, patience and support that helped me to shape and refine my work.

My sincere thanks to **Dr. T. T. MIRNALINEE**, Professor and Head of the Department of Computer Science and Engineering, for her words of advice and encouragement and I would like to thank our project Coordinator **Dr. B. BHARATHI**, Associate Professor, Department of Computer Science and Engineering for her valuable suggestions throughout this project.

I express my deep respect to the founder **Dr. SHIV NADAR**, Chairman, SSN Institutions. I also express my appreciation to our **Dr. V. E. ANNAMALAI**, Principal, for all the help he has rendered during this course of study.

I would like to extend my sincere thanks to all the teaching and non-teaching staffs of our department who have contributed directly and indirectly during the course of my project work. Finally, I would like to thank my parents and friends for their patience, cooperation and moral support throughout my life.

S. A. MADHULICA **VASUNDHHARA SINGH KATOCH**

ABSTRACT

We are aiming to achieve decentralization of traditional Federated Learning by integrating Blockchain into the system for pneumonia detection using Chest X-rays. Patient data is very sensitive and private and proper systems like Blockchain network is essential to ensure security in such applications.

A distributed Machine Learning framework called Federated Learning (FL) trains global models without utilizing local data and maintaining privacy. Federated Learning has emerged as an efficient technology in the field of healthcare. FL still has it's own shortcomings like Single Point of Failure, Malicious Clients and False Data. Blockchain has been used to address the issues that the traditional FL has. This paper examines the incorporation of Blockchain into Federated Learning systems and it's advantages by building a model for the same. Different aggregation techniques like FedAvg, FedBoosting and Geometric Mean are explored and implemented in a decentralised manner using public Blockchain network.The accuracy of the local models and the global model is studied and compared. Global model's accuracy is seen to be higher.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This chapter gives a detailed explanation of the motivation and background of this project. It discusses in detail, about each component that a blockchain based federated learning consists including Machine Learning(ML), Federated Learning(FL), Blockchain. It also discusses about types of each of these components.

1.1 Motivation

To implement a Blockchain Based Federated Learning(BCFL) system for Pneumonia Detection using Chest X-rays, we suggest a BCFL architecture, which improves the security and privacy of traditional FL systems [17]. A distributed Machine Learning technique called Federated Learning makes it possible to train global models without the need of sending data to a central server, but rather sending model parameters to a server where a global model is aggregated. Traditional FL approaches, however, are susceptible to intrusions like Single Point of Failure, Denial of Service [2] since it depends on a centralized server. We describe a novel strategy that uses Blockchain technology to build a secure and reliable environment for FL in order to overcome these problems. We outline the proposed system's architecture, which consists of a Federated Learning layer and a blockchain layer (which incorporates a smart contract). Using a real-world dataset, we assess the model's performance in terms of data privacy, security, and trust. Our proposed BCFL model provides a promising solution for decentralized and privacy-preserving system.

In this project, we are using a neural network to train the local model at each hospital with hospital's local data. Image recognition is used to classify the Chest X-rays. After the local model is built, the model parameters are sent to the smart contract inside the blockchain, which makes sure that the aggregation is happening in a decentralised manner. After the aggregation is done and the global model is in place, the global model parameters are sent back to the hospitals via the blockchain system for local model updation. The global model can be used by each Hospital client to predict and analyse any patient's chest x-ray.

1.2 Background

This section provides context and sets the stage for the research that will be presented in this thesis. It includes a brief overview of FL explaining it's process in brief and Blockchain.

1.2.1 Federated Learning

Federated learning is a machine learning strategy that enables many parties to jointly train a machine learning model while maintaining the decentralisation and privacy of their respective data. In federated learning [18], the learning model is delivered to the devices for local training rather than being centralised, so the data stays on the individual devices. Additionally, federated machine learning may

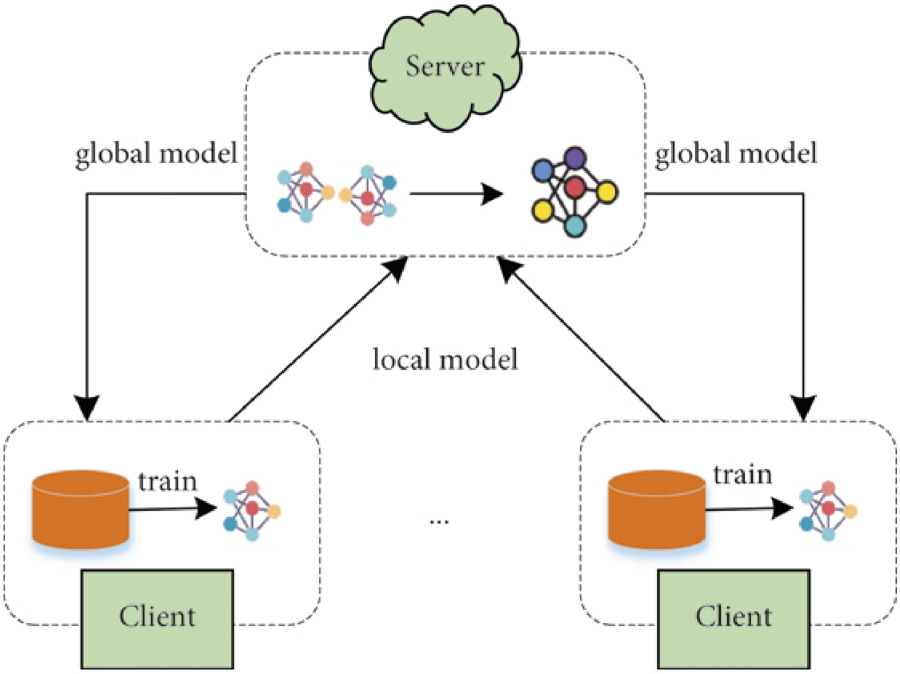


FIGURE 1.1: Traditional Federated Learning System

result in lower communication and computation costs than centralised machine learning. Data transport to a central location is necessary for centralised machine learning, which can be costly in terms of communication expenses. Only the updated model parameters are transferred to the central server when using federated learning because the rest of the data stays on the device. This can substantially lower communication costs and increase machine learning accessibility in resource-limited environments.

The functions performed by a typical Federated Learning model involve :

1. **Data Collection:** The data sources, which in this case are hospitals, collect the data from their patients. The data may not be consistent across all devices and hence must be preprocessed.

2. **Data Preprocessing:** The data sources preprocess the data to standardize them for consistency across different hospitals. This may involve resizing, cropping, or normalizing the images.

3. **Model Training:** On the basis of their own data, [30] the data sources develop a local machine learning model. The new model weights are shared across participants in the federated learning process, which trains the local model without disclosing the raw data.As seen in figure 1.1, for instance, each client trains its own local model using the data at hand.

4. **Global Model:** The server receives the local model parameters, and according to predetermined criteria, aggregation occurs. For updating, the global model parameters are provided back to the clients.

5. **Model Update:** Local Model parameters are updated using the received global models.

6. **Model Evaluation:** The data sources assess how well the machine learning model performs using a validation dataset. To make sure the model is precise and useful, this is crucial.

Based on the features of the data distribution [22], federated learning (FL) can be divided into three groups. These groups include Federated Transfer Learning, Horizontal FL, and Vertical FL.

1. When many clients have datasets with the same characteristics but different samples, this is known as a **Horizontal FL** [4] situation. Each client in this scenario trains a local model on a specific dataset before sending model

updates to the central server. These changes are combined by the server to create a global model, which is then returned to the clients for additional training.

2. On the other hand, datasets with the same sample space but various feature spaces are included in **Vertical FL** [4] . Without disclosing their data, the clients work together in this instance to train a collaborative model.

3. When there are only a few overlapping samples and features across the two datasets, **federated transfer learning** is employed. When there is a dearth of data or labels, transfer learning can be utilised to exploit knowledge from previously trained models without having to slice the data.

In our system, **Horizontal FL** is used because of the nature of our dataset.

1.2.2 Blockchain Network

A distributed and decentralised database or ledger known as a blockchain network enables several parties to share and maintain a single, impenetrable record of transactions or data. [8]. Because every member of the network has a copy of the ledger and can independently verify transactions.

The central server in the figure 1.1 can be replaced by a Blockchain network to overcome the isuues with traditional FL systems. Blockchain is an immutable ledger and hence the possibility of attacks or single point of failure is eliminated. It uses Consensus algorithms and smart contracts to achieve the same.

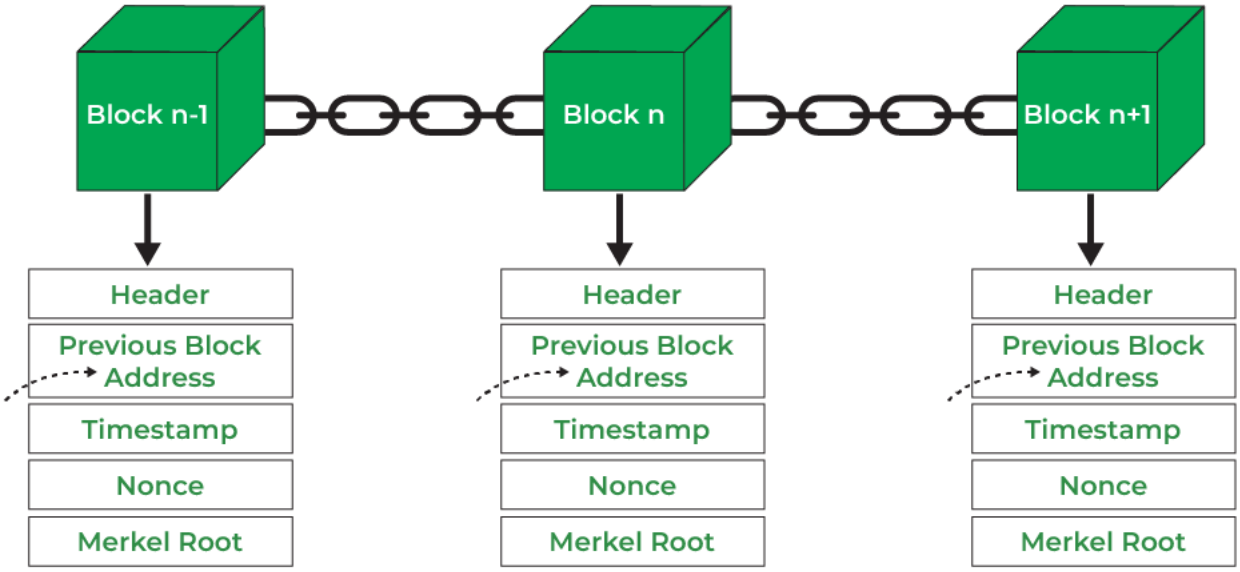


FIGURE 1.2: A basic Blockchain network

In the figure 1.2, there are n blocks in the network. The current block holds the previous blocks address and this network is immutable.

Here are the main parts of a blockchain:

1. **Blocks:** A block is a container that stores a group of transactions. Each block is identified by a unique hash code and contains a reference to the previous block in the chain.

2. **Transactions:** A transaction is a record of the exchange of value between two parties. Each transaction includes the amount transferred, the address of the receiver, the address of the sender, and a digital signature.

3. **Nodes:** A node is a computer that participates in the blockchain network. Nodes store copies of the blockchain and validate transactions by verifying

their digital signatures and checking their consistency with the rules of the blockchain protocol.

4. **Consensus mechanism:** A consesus mechanism specifies few rules that ensure that all the participating nodes agree to the content on the blockchain [6]. Some common consensus mechanisms are proof of work,proof of stake, delegated proof of stake and and proof of authority.

5. **Cryptography:** Cryptography uses mathematical algorithms to secure the blockchain. Cryptography ensures that only the owner of a private key can access their digital assets and that transactions are verified and recorded accurately.

6. **Smart contracts:** Smart contracts are basically a piece of self-executing code which runs on the blockchain when a specific condition is triggered [3].

Overall, these parts work together to create a decentralized, secure, and transparent system for recording transactions and managing digital assets. Blockchain technology is used in many applications for pure security reasons.

There are generally three categories of blockchain: private, consortium, and public [29] :

A **private blockchain** is a permissioned network where access is only allowed for approved users. [9] This indicates that only specific people or groups can sign up for and use the blockchain. Given that they give the user more network control and the ability to manage sensitive data, private blockchains are frequently utilised for internal purposes within a business or organisation [11].

A **consortium blockchain** is similar to a private blockchain, but it is governed by multiple organizations instead of just one. [9] Consortium blockchains are typically used for collaborations between companies, where each company can participate and validate transactions on the network. Consortium blockchains can be permissioned or permissionless, meaning they may or may not require permission to join [11].

A **public blockchain** is a permissionless network where anyone is allowed to participate in the network.This means that there is no centralized authority. [9] Public blockchains, such as Bitcoin and Ethereum, are entirely decentralized and allow for transparency and immutability of the ledger. However, they can also be slower due to the large number of participants and transactions on the network.We are using public blockchain for our model as it is accessible easily [11] .

1.3 Blockchain based Federated Learning(BCFL)

The aggregation of the local model parameters are handled by the blockchain nodes thus avoiding central point of failure [13] . Instead of the central server in figure 1.1, we use a blockchain network. The smart contract will receive the local model parameters and aggregation will take place. After the aggregation takes place in a decentralised manner, global model parameters are sent back to the clients.

In a BCFL system(figure 1.3), the training process is distributed among multiple devices, and the model updates are stored on a blockchain. The system consists of the following components:

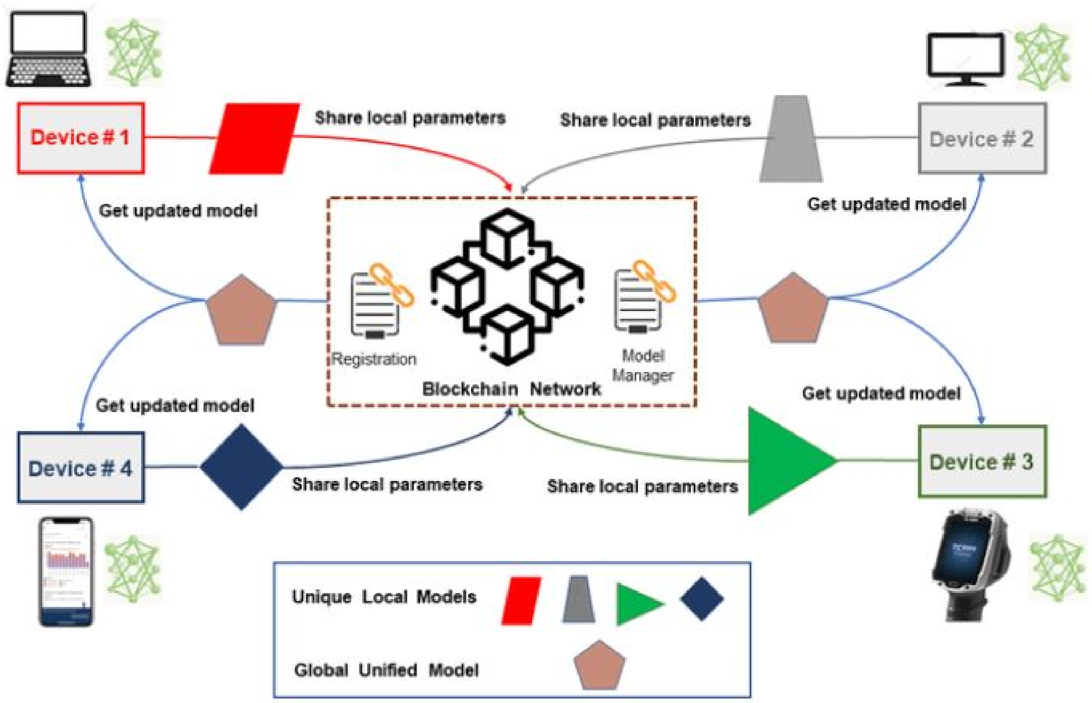


FIGURE 1.3: A simple BCFL system

1. **Data Owners:** These are the individuals or organizations that own the data. In a federated learning system, the data is not shared with a central server, but rather remains on the devices owned by the data owners.

2. **Federated Learning Algorithm:** The federated learning algorithm basically trains a ML model on the data available at the data owner's side, and the model's parameters are sent to be later on aggregated to get global model parameters. These global model parametrs are sent back to the client to update the local model and furthur usage.

3. **Blockchain:** The blockchain is used to store and aggregate the model updates securely and transparently. Each update is considered as a

transaction on the blockchain, and the blockchain ensures that the updates are tamper-proof and can be audited by anyone.

4. **Smart Contracts:** Smart contracts are used to manage the interactions between the data owners, the federated learning algorithm, and the blockchain. They automate the process of sending the model updates to the blockchain.

1.3.1 Blockchain Types in BCFL

We will discuss two kinds of blockchain systems that can be used in a BCFL model: public blockchain and permissioned blockchain. Figure 1.4 gives an accurate venn diagram for the same.

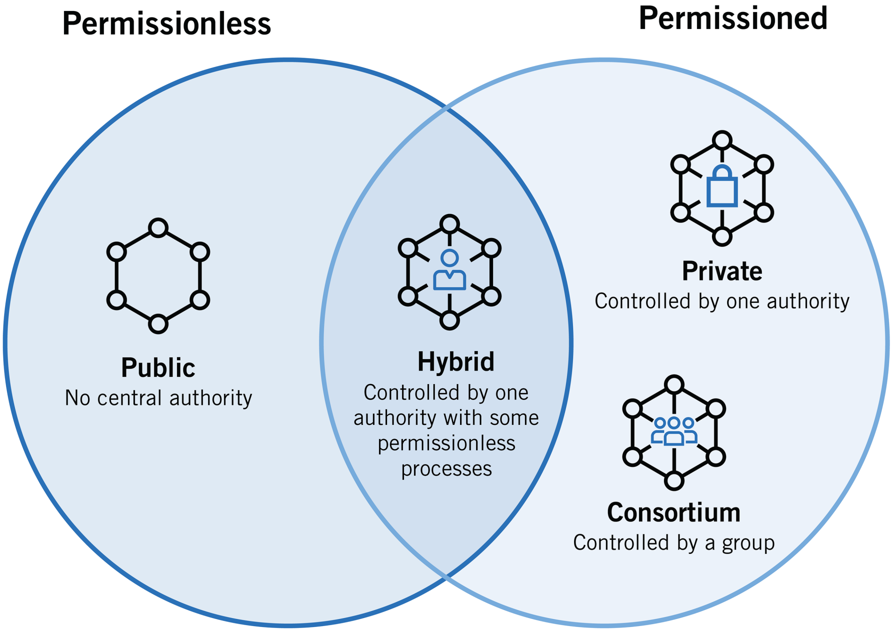


FIGURE 1.4: Types of blockchain networks

Public blockchain in BCFL :

Public blockchains can be used in BCFL to ensure that all participants can access and verify the accuracy of the data being used. Paper [5] talks about a proposed BCFL system which operates on a public blockchain network, allowing training nodes and miners to participate in the system without requiring permission. The nodes and miners can collaborate to train a global model.

Using a public blockchain in BCFL can provide several benefits. Firstly, it allows for transparent and auditable transactions. Because all transactions are recorded on the blockchain, participants can easily verify that the correct updates are being used to train the model. Additionally, the transparency of the blockchain can help prevent malicious actors from tampering with the data or the learning process.

Secondly, a public blockchain can provide a secure way to manage access and permissions. Access to the data and the model can be controlled using smart contracts. This can help prevent unauthorized access to the data or the model and ensure that participants only have access to the data they are allowed to use.

Permissioned blockchain in BCFL :

Unlike public blockchains, permissioned blockchains are restricted to authorized clients.. The permissioned nature of the blockchain ensures that only authorized participants can join the network and access the shared pool of data. The centralized control of the blockchain network allows for the management of access to the shared pool, as well as the distribution of rewards and incentives to the participants.

The permissioned aspect of this type of blockchain ensures that only authorized participants can participate in the blockchain and access the shared data. The centralized control of the blockchain network allows for the management of access to the shared pool, as well as the distribution of rewards and incentives to the participants.

But for the scope of our project, **Public Blockchain** is used because of it's accessibility.

1.3.2 Functions of BCFL

In this section, we examine the distinct operations performed by BCFL by looking at how it operates, such as verifying updates to the model, combining the global model, making use of the distributed ledger, and incentivizing participants.

Data Collection and Preprocessing :

The first step in building a Local model is to gather the data you need to train the model. This could involve collecting data from different sources, including databases, APIs, web scraping, or manual data entry.Once you've collected your data, the next step is to preprocess it so that it's ready for training your Local model. Preprocessing involves several steps like data cleaning, data transformation,feature engineering and data splitting. After preprocessing, local model training can take place.

Local Model Training :

Machine learning algorithms are used to perform the actual federated learning. These algorithms allow the different entities to share knowledge and collaborate on the development of machine learning models without exposing sensitive data. A suitable machine learning algorithm based on the data being used has to be chosen to build local models. Local models are the building blocks of our Federated learning system. The efficiency of the algorithm chosen plays a major role in the overall efficiency of the system.

The local model has to be trained on train data and later on test data. Based on the accuracy the weights and biases should be adjusted while keeping over-fitting in mind. After achieving the desired accuracy, the local model parameters have to be extracted so that they can be sent to the blockchain network where aggregation takes place. The parameters must be sent through a secure channel to the smart contract in the blockchain either by using http protocols or other secure communication methods.

After receiving the global model parameters, local model updation must take place and the updated model can later be used for prediction.

Role of the Smart Contract :

A smart contract is a self-executing piece of code that specifies few conditions to be triggered for some transaction to take place, it runs on the blockchain network.In the context of a BCFL system, a smart contract can play a crucial role in ensuring secure and transparent execution of the federated learning process.

Figure 1.5 shows the working of smart contracts. Federated learning is a distributed machine learning approach in which data is kept locally on different

SMART CONTRACT
PARTIES
SMART CONTRACT
EXECUTION

FIGURE 1.5: Working of a Smart Contract

devices or nodes, and the model is trained by aggregating updates from all these nodes [7]. In a BCFL system, smart contracts can be used to aggregate the local model updates and govern the interactions between the participating nodes.

A smart contract can do many things in the context of BCFL such as :

1. **Incentivization:** Smart contracts can be used to incentivize participation in the federated learning process. For example, a smart contract can offer tokens or other rewards to nodes that contribute their data or computation resources to the system so that the nodes refrain from malicious behaviour.

2. **Data privacy and security:** Smart contracts can be used to enforce data privacy and security. For instance, a smart contract can specify the terms under which nodes can access and use the data. This can include access

controls, data anonymization, and data deletion policies. It can also get permissions from all the clients before executing a piece of code.

3. **Model training and aggregation:** Smart contracts can be used to automate the model training and aggregation process. For instance, a smart contract can specify the training algorithm and parameters to be used, the number of iterations, and the aggregation method. This can ensure that the training process is standardized and transparent.
   The major purpose of a smart contract in our BCFL is to perform aggregation in a decentralised manner on the distributed ledger so that the need of a central server can be eliminated. The smart contract can use any one of the aggregation methods available, after receiving the local model parameters and after consensus is achieved the aggregation can take place. Once the global model parameters are ready, they can be sent back to the clients for local model updation.

4. **Consensus and validation:** Smart contracts can be used to ensure consensus and validation among the participating nodes. For example, a smart contract can require that a certain percentage of nodes agree on the model update before it is accepted. This can help prevent malicious nodes from manipulating the training process.

1.4 Problem Definition

The problem addressed by a blockchain based federated learning system are, first, the issues of data privacy in normal ML models and second, the issues created by the use of a central server in FL models. It addresses the problems by decentralizing the system by replacing the central server by a blockchain.

We have achieved decentralization of traditional FL by integrating blockchain into the system for pneumonia detection using Chest X-rays. Patient data is very sensitive and private, and hence proper systems like blockchain network is essential to ensure security in such applications.

Machine learning algorithms have become increasingly popular and have been successfully applied to various fields, such as image recognition, natural language processing, and predictive modeling. However, the traditional approach to machine learning requires large amounts of data to be stored and transferred to a central server, which can create privacy and security concerns, and can be inefficient in terms of computation and communication.

Federated learning is a promising approach that allows multiple devices or nodes to collaborate on training a machine learning model without sharing their data with each other, but rather by sharing the model parameters of each data. However, the current federated learning systems still have some limitations, such as the lack of a robust mechanism for securely exchanging data and model updates between nodes, and the need for a central authority to coordinate the training process.

The goal of this thesis is to propose a blockchain-based federated learning system that addresses these limitations and provides a secure, decentralized, and transparent approach to collaborative machine learning. The system will leverage blockchain technology to provide a secure and transparent mechanism for exchanging data and updates between nodes, while also ensuring data privacy and integrity.

1.5 Organisation of the Report

This report talks about what exactly Blockchain based federated learning consists of and how a model built on the same, performs in a hospital system. We start by introducing the individual components of the project - Federated learning and Blockchain network. We move forward with chapter 2, where we discuss extensively about Blockchain based federated learning, its types and functions. In chapter 3, we explore the papers currently present in this project domain, that are relevant to our topic. We identify gap in different publications and research objectives. We talk about different algorithms and techniques used to build our model in chapter 4. Further we discuss about the proposed system and the architectural design for the same. We explain each part of the system in detail in chapter 5. In chapter 6, we talk about the results achieved by our system and further analyse the performance of our system in chapter 7. Here, the report discusses about the complexity of the dataset, the hardware and software specifications and the experiments conducted.

In chapter 8, we discuss about the healthcare and security issues solved by our project and the social impact of the same. We conclude the report in chapter 9 by talking about the advantages and disadvantages of BCFL and how it can be incorporated in our life.We discussed different open problems that can be solved. We discuss the future work that is possible and future research directions.chapter 9 cites all the reference papers we used to make this project happen.

CHAPTER 2

LITERATURE SURVEY

In this section we talk about the extensive research work that we conducted. This section talks about related work, gaps identified and research objectives of this thesis.

2.1 Related Work

In the healthcare industry, traditional deep learning is employed to create precise models. But unlike traditional ML models, where data is gathered and processed in a single location, a federated learning system should be immune to errors and attacks while receiving the local model parameters. The centralised training data centre needs to receive data from many places. From the standpoint of securely storing all of these data, this is difficult [15] .

According to the study by McMahan [10], Google pioneered Federated Learning (FL) as a decentralized approach to machine learning that enables training models using local data from devices while preserving user privacy. The FL model is converged by aggregating data from numerous participating businesses and users. The remarkable use cases of FL in various industries, including healthcare, finance, transportation, and smart cities, have been extensively covered in numerous articles [28].

Toyoda [24] explains the need to incorporate blockchain in federated learning systems. He details that even though FL outperforms the performance of traditional ML, it has high risk of SPoF and DDoS attack. Consequently, a powerful decentralised system is needed to identify and stop rogue updates. Blockchain offers a tremendous deal of potential to draw model training participants by including a digital currency.

Patients and hospitals are reluctant to share their health data since it is a sensitive subject in the healthcare industry. While data leakage is the biggest challenge, FL can assist with distributive model training [26]. Patients or hospitals can use blockchain to exchange data without disclosing any personal information. Because the protection of patient privacy prevents researchers from analysing health data and because the available tools are insufficient to address the problem, Passerat-Palmbach et al. [12] propose the use of blockchain and FL for healthcare consortia. Data access, model integration, weight encryption, and learning process auditing are highlighted in their model. However, since consortia are the focus of this study, it is inappropriate for the majority of health issues and is unable to provide solid working techniques that can solve the issue.

Compared to [12], Kumar et al. [23] provides a specialised BCFL model-based method for the Detection of COVID-19 virus. Clients train the local model using their own private data and only share the weights and gradients. The learning process and accompanying data are recorded on the blockchain. Patients' privacy is a concern raised by researchers, and the BCFL architecture can safeguard it while the global model is being trained. In that study, a safe and decentralised framework for hospital data sharing is created, making it possible to detect COVID-19 automatically and securely. There aren't many studies using BCFL in healthcare today, but the direction of the study is encouraging because BCFL can provide secure learning settings and a lot of medical data has to be processed.

2.2 Gaps Identified

The blockchain-based federated learning techniques (2021) explored the BCFL design with regard to types,improvement, and incentive mechanisms. Several studies on federated learning are available in the literature (2020). In papers published in 2022 [1], various blockchain deployment frameworks, dependencies, and aggregation techniques are examined.

However, there is a lack of papers discussing the implementation of BCFL (Blockchain based federated learning) using optimized algorithms for Federated Averaging. We aim to implement a BCFL system including optimized algorithms for federated averaging.

There is also a lack of discussion of the drawbacks that are currently faced by the developer community of BCFL systems. Our paper has extensively discussed about the open issues and suggests possible solutions according to the current resources. We have discussed possible future directions that can be explored in terms of scalability.

2.3 Open Issues

Even though the system is highly secure and completely decentralised, there are a few drawbacks currently discussed in the developer community. This section includes a perspective on those problems and suggests possible solutions.

1. Immutability as a drawback :

Immutability in blockchain refers to the property that once data is recorded on the blockchain, it cannot be altered, deleted, or tampered with. This is achieved through the use of cryptographic techniques such as hashing, digital signatures, and consensus algorithms.

Once a transaction is recorded on the blockchain, it becomes a permanent part of the ledger and is replicated across all nodes in the network. This makes it practically impossible for anyone to alter or delete the transaction without being detected, as any attempt to do so would require changing the cryptographic hash of the block and the subsequent blocks, which would invalidate the entire chain.

The immutability of blockchain data is a key feature that ensures the integrity and security of the network, and is one of the reasons why blockchain technology is widely used in applications such as cryptocurrency, supply chain management, and digital identity verification.

But, immutability can be a drawback in some cases, as it can make it difficult to correct errors or make changes to data that has been recorded on the blockchain.

For example, if a mistake is made in a transaction and it is recorded on the blockchain, it cannot be corrected or reversed. This means that any errors or mistakes made on the blockchain can potentially have permanent and irreversible consequences.

In addition, the immutability of blockchain data can also pose challenges for compliance with regulations that require data to be modified or deleted under certain circumstances, such as data privacy laws.

Possible Solution :

However, there are ways to address these issues, such as implementing off-chain solutions to manage data updates or building in flexibility to allow for corrections and modifications in certain cases.

2. Vulnerabilities in blockchain frameworks:

For the implementation of BCFL syatems, a third party blockchain framework has to be used to replace the central server. However, different blockchains come with different vulnerabilities which may affect the performance or security of a BCFL system. For example, a popular blockchain is Ethereum. Transactions on the Ethereum network require gas fees. Ethereum gas fees are the fees paid by users for transactions to be processed on the Ethereum blockchain. Gas fees are denominated in Ether (ETH) and are paid to miners, who are responsible for processing transactions and adding them to the blockchain. However a common problem Ethereum developers face is the high gas fees which practically limits the feasible number of transactions that can be done on the Ethereum network. This vulnerability will affect the BCFL system if Ethereum framework is used.

Possible Solution :

Different blockchain frameworks have different vulnerabilities which may not affect any particular type of system implemented on it. Therefore, choosing a blockchain whose vulnerabilities will not affect the performance of your particular system is the current possible solution for this.

3. Authenticity of data provided by nodes:

Malicious end devices can pose a significant threat to a blockchain-based federated learning system in several ways:

- **Data poisoning**: Malicious end devices can intentionally provide false or incorrect data during the training process. This can mislead the model and negatively impact the accuracy of the final model.
- **Model poisoning**: Malicious end devices can also inject biased or inaccurate models into the federated learning system, causing the entire system to learn from faulty models.
- **Distributed denial-of-service (DDoS) attacks**: Malicious end devices can launch DDoS attacks on the blockchain network, causing delays and disruptions to the federated learning system.

Possible Solution:
The current possible solution for this problem is to allow only trusted end devices to provide their local models for global model aggregation. For example, in our system only big hospitals which can be trusted to provide untempered data could be allowed to participate in aggregation.

Another way to address this issue is to implement off-chain solutions to check the performance or authenticity of the local models of participating devices before allowing them to participate in the global aggregation.

2.4 Research Objectives

The main objectives of this research are as follows:

- Analysing different systems and models in place for BCFL that are relevant to our goal.

- Understanding the need of integrating blockchain into FL systems in terms of decentralisation and security, as well as potential performance and scalability issues.
- Analysing different aggregation methods that can be used for our dataset and their impact on system efficiency.
- Comparing blockchain-based federated learning with other machine learning techniques that preserve privacy, such as secure multi-party computation and differential privacy.
- Discussing the challenges and future research directions in blockchain-based federated learning, including ways to address the trade-offs between privacy, security, and efficiency, and how to design effective incentive mechanisms for participants.
- Exploring different blockchain platforms that are accessible as well have a solid consensus mechanism in place.

CHAPTER 3

PROPOSED METHODOLOGY

This chapter contains of two sections i.e. Algorithms Used and Proposed Methodology. In Algorithms Used, the algorithms used for data preprocessing, building the ML model, testing the ML model, evaluating the ML model as well as the algorithms used under smart contract for model aggregation, have been discussed. Further, the proposed methodology discusses the architecture of the proposed system. It includes the explanation of each component of the architecture in detail.

3.1 System Components

There are various algorithms used for Blockchain Based Federated Learning system depending on the layer of implementation. The different layers include :

3.1.1 Data Sources and Federated Learning

In our case the data sources are hospitals and the data is chest X-ray images as shown in figure 3.1. The ”Data Sources” component of the blockchain-based federated learning system architecture diagram represents the various hospitals that participate in the federated learning process by providing their chest X-ray images for model training. The chest X-ray images would need to be first pre-processed and annotated to ensure consistency in the image data across

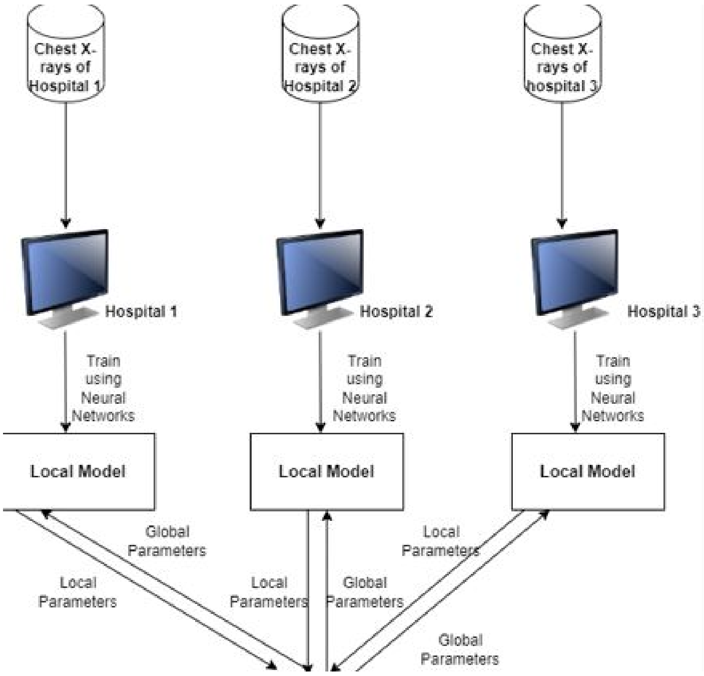


FIGURE 3.1: Data Sources

different hospitals. Once the images are ready, the machine learning model would be trained on the encrypted chest X-ray images contributed by the hospitals using a federated learning approach. In our case, at this level, each participating hospital would train a local model on their own data and generate the local parameters. The algorithms implemented at this level include:

1. **Data Collection**

The data sources, which in this case are hospitals, collect the chest X-ray images from their patients. The images may be in different formats, resolutions, and quality, and may require pre-processing before being used for model training. For our project, dataset is taken from Kaggle, the description of which is provided in the later sections of this thesis.

2. **Data Preprocessing**

Data preprocessing is an essential step in machine learning that involves transforming raw data into a format that is suitable for analysis and modeling. This step is crucial because the quality of the data used for machine learning models directly affects the accuracy of the model's predictions. The data sources pre process the chest X-ray images to standardize them for consistency across different hospitals. This involves resizing, cropping, or normalizing the images. The following data augmentation techniques were used according to the suitability of the model:

- **Horizontal Flip** : A data augmentation technique that takes both rows and columns of such a matrix and flips. them horizontally.
- **Brightness Range** : Increases the overall lightness of the image.
- **Height and Width Shift** : Change the height and width ratio of images according to the suitability of the model.
- **Rotation of the Image** : We rotated the images by 0 to 360 degrees clockwise.

Further, the following algorithms were developed to store the data as training, testing and validation data:

- **Traindata** : stored 1518 images belonging to 2 classes from the train directory.
- **Testdata** : stored 484 images belonging to 2 classes from the test directory.
- **Valdata** : stores 14 images belonging to 2 classes from the val directory.

3. **Model Building**

Tensorflow library is used which is basically a software library for numerical computation using data flow graphs where Tensor is the central unit of data. Keras is used for developing and training the deep learning model. Keras is built on top of lower-level deep learning libraries such as TensorFlow and Theano, which provide the backend computation engine. Keras provides a simplified interface for building neural networks, making it easier to prototype, experiment, and deploy deep learning models.

The model was built using a **Convolutional Neural Network** Technique. There are several layers used in building a CNN ML model, which include:

- **Convolutional layers** : The convolutional layer is the basic building block of a CNN. This layer applies filters to the input image to extract features and generate a feature map.
- **Pooling layers** : By downsampling the feature map, the pooling layer decreases its spatial size, resulting in a reduction of model parameters and computation. This is achieved to optimize the model's efficiency.
- **Activation functions** : Activation functions introduce non-linearity into the model and help the model to learn complex patterns. Some common activation functions used in CNNs include ReLU, sigmoid, and tanh.
- **Dropout** : Dropout is a regularization technique that randomly drops out some of the neurons during training to prevent overfitting.

- **Optimizers** : Optimizers are used to update the weights and biases of the model during training. Some common optimizers used in CNNs include stochastic gradient descent (SGD), Adam, and Adagrad.
- **Loss functions** : Loss functions are used to measure the difference between the predicted output and the actual output. The goal of the model is to minimize the loss function during training. Some common loss functions used in CNNs include categorical cross-entropy and mean squared error.

For this project, a 9 layer deep learning network was trained, the layers including Conv2D() as the convolutional layer, MaxPooling2D() as the pooling layer, Relu and Sigmoid for activation functions.

Conv2D is used in our convolutional neural networks (CNNs) for image processing tasks. It performs a two-dimensional convolution operation on the input data. In this layer, a set of filters are applied to the input image or feature map, with each filter being a small matrix of weights. The filters are convolved with the input to produce a set of feature maps that highlight different aspects of the input image. The output of this layer is typically passed through an activation function to introduce non-linearity into the network. This layer served as a fundamental building block of CNNs followed by other layers such as pooling and fully connected layers.

MaxPooling2D was used as pooling layer in the convolutional neural network (CNN) for image processing tasks. It performs downsampling which is the process of splitting each feature map into rectangular regions that do not overlap, and selecting the highest value within each region. In the MaxPooling2D layer, the size of the rectangular regions to be pooled and the stride of the pooling operation was specified. The output of a MaxPooling2D layer is a downsampled feature map with reduced spatial dimensions. The main benefit of MaxPooling2D is to reduce the spatial size of the input feature maps while retaining the most important information. This helps to reduce the computational burden of the network and prevent overfitting by introducing some degree of translation invariance. MaxPooling2D is often used in conjunction with Conv2D layers in CNNs and can be repeated multiple times to progressively reduce the spatial dimensions of the feature maps.

The activation function **ReLU**, short for **Rectified Linear Unit**, is an activation function used in neural networks. It is a mathematical function that takes an input value and returns the maximum of that value and zero. In other words, if the input value is positive, ReLU returns the input value, but if the input value is negative or zero, ReLU returns zero. Mathematically, the function can be defined as follows: f(x) = max(0, x) The main purpose of ReLU is to introduce non-linearity into a neural network. This is important because many real-world problems are inherently non-linear, and without non-linear activation functions like ReLU, neural networks would be limited to only learning linear relationships. ReLU has several advantages over other activation functions, including being computationally efficient and avoiding the vanishing gradient problem that can occur with other activation functions like sigmoid and tanh. ReLU has become a popular choice for activation functions in deep learning due to its simplicity and effectiveness in many applications.

In the next layer, **Sigmoid** is used as activation layer. Sigmoid is another commonly used activation function in neural networks. It is a mathematical function that maps any input value to a value between 0 and 1. The function has an S-shaped curve, and as the input value increases or decreases, the output value of the sigmoid function changes gradually and smoothly. Mathematically, equation 3.1 represents the sigmoid function.

$$f(x) = 1/(1+e^{-x}) \tag{3.1}$$

The main purpose of sigmoid is also to introduce non-linearity into a neural network.

4. **Model Training**

The data sources train a local machine learning model on their own data. The local model is trained using the federated learning approach, which involves sharing the updated model weights without revealing the actual data. The following configuration is used to train the model:

Adam optimizer : Adam (Adaptive Moment Estimation) is a stochastic gradient descent optimization algorithm that is commonly used for training deep neural networks. It is an adaptive learning rate optimization algorithm. Adam optimizer maintains a set of exponentially decaying average of past gradients and squared gradients, and uses them to compute adaptive learning rates for each weight. It also includes bias correction to ensure the estimates of the first and second moments are unbiased.

Binary crossentropy : It is a commonly used loss function in binary classification problems. It measures the difference between the predicted probability distribution and the true probability distribution for a binary classification problem. In binary classification, we have two classes (positive and negative) and we want to predict which class an input belongs to. Mathematically, equation 3.2 represents the binary crossentropy function.

$$L(y,\hat{y}) = -[y * log(\hat{y}) + (1-y) * log(1-\hat{y})] \tag{3.2}$$

where y is the true label (either 0 or 1), and ŷ is the predicted probability of the positive class. The binary crossentropy loss function heavily penalizes the model when it makes a high confidence prediction for an incorrect class. In contrast, when the model makes a high confidence prediction for the correct class, the loss function is nearly zero. Consequently, the loss function is very high when the model predicts the wrong class with high confidence. During training, the goal is to minimize the binary crossentropy loss function by adjusting the weights and biases of the model using backpropagation and gradient descent. The lower the binary crossentropy loss, the better the model's predictions on the binary classification task.

Finally before training the model, **ModelCheckpoint** callback function is utilized alongside model training using model.fit() to save a model or weights into a checkpoint file at a specific point. This allows the saved model or weights to be reloaded later on to resume training from the previously saved state. and **Early stopping** is utilized to determine the number of training epochs. It halts the training process when the model's performance no longer improves on a held-out validation dataset.

The model is finally trained in 10 epochs and by specifying the checkpoints, train data and validation data.

5. **Model Update**

The global model is updated through data sources sharing their local model parameters with the blockchain, which creates a link between hospitals and the blockchain network. The blockchain holds the global model weights, and a smart contract is in charge of model aggregation. In federated learning systems based on blockchain, diverse aggregation algorithms can be applied to combine the updates of local models from the participants. The algorithms used for aggregation in our project are:

- **Federated Averaging**: This is a simple and widely used aggregation algorithm where the server aggregates the participants' model updates by taking the average of the local models' weights [20]. The procedure is repeated numerous times until the model reaches an acceptable level of convergence. FedAvg has been widely adopted in federated learning because it is a simple and effective method to combine local models. FedAvg works well for evenly distributed data.

  The federated averaging algorithm has several advantages over traditional centralized learning approaches. First, it enables participants to collaboratively train a global model without sharing their raw data, thereby preserving data privacy. Second, it allows for a more diverse set of data to be used for training, which can lead to better performance on a broader range of tasks. Finally, it reduces the need for centralized data storage and processing, which can improve scalability and reduce costs.

- **Federated Boosting**: This is an aggregation algorithm that focuses on improving the performance of the local models by using boosting techniques. The server aggregates the models' updates by assigning higher weights to the models with better performance or higher priority.

In Federated Boosting, the aggregation method used to combine the models is based on the idea of weighted averaging. The weights are determined by the performance of each model on a validation set. Federated Boosting has shown promising results in various applications of ML. It has the potential to improve the accuracy of machine learning models in a wide range of scenarios, particularly in situations where data privacy and security are paramount.

- **Geometric Mean** : In the Geometric Mean Federated Learning Algorithm, each device trains a local model on its own dataset, and then the geometric mean of the models is calculated to produce a global model. This is done by taking the product of the weights of each model and then taking the nth root of the product, where n is the number of local models. It is less sensitive to outliers or extreme values in the local models, which can lead to more robust global models.

  The Geometric Mean Federated Learning Algorithm has several advantages over other aggregation methods. First, it is less sensitive to outliers or extreme values in the local models, which can lead to more robust global models. Second, it can handle non-i.i.d. data distributions, where the data on each device is not identically and independently distributed, which is common in many real-world scenarios. This method is not sensitive to outliers while FedAvg is.

For aggregation, the participants first download the current global model from the server. They then perform model training on their local data and produce a local model update. The local model update represents the difference between the local model's weights and the global model's weights.

The participants then send their local model update to the server, which aggregates the updates using an averaging method. Specifically, the server computes the weighted average of the local model updates received from the participants, where the weights correspond to the number of samples each participant has in their local dataset. The aggregated model update is then used to update the global model by adding the average model update to the current global model. The new global model is then sent back to the participants, and the process repeats for the next round of training.

Local models are essentially updated by receiving the updated model weights from the blockchain after establishing the connection between the individual systems and the blockchain network.

6. **Model Evaluation**

The data sources evaluate the performance of the machine learning model on a validation dataset. This is important to ensure that the model is accurate and effective in detecting chest X-ray abnormalities. The various techniques used for model evaluation in the system are:

Confusion Matrix : A confusion matrix shows a number of metrics, which are used for calculation of different metrics required to evaluate the performance of a machine learning model in a supervised learning task. It is also known as an error matrix or a contingency table. The confusion matrix provides a summary of the model's predicted classifications compared to the actual classifications in the dataset. It shows the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class in the dataset.

In the confusion matrix, the rows represent the number of predicted values of the model, and the columns represent the actual values in the dataset. The TP mean the predictions in which the model correctly predicted a positive class, and the true negatives TN are the predictions where the model correctly predicted a negative class. False positives FP are the cases where the model predicted a class to be positive, but the actual value was negative, and false negatives FN are the cases where the model predicted a class to be negative, but the actual value was positive.

The confusion matrix provides several metrics that can be used to evaluate the model's performance, such as accuracy, precision, recall, and F1 score. These metrics help assess the model's ability to correctly classify data points and identify any areas where the model may need improvement. There are certain metrics that are calculated from Confusion Matrix which help us to evaluate the quality of the ML model. In our system, accuracy is given importance in terms of evaluation. In machine learning, accuracy is a metric used to evaluate how well a classification model correctly predicts the class labels of the test data. The formula for accuracy is:

Accuracy = (Number of correct predictions) / (Total number of predictions)

Accuracy is typically expressed as a percentage, with values ranging from 0% to 100% A model with 100% accuracy means that it correctly predicted all the test data labels. Other metrics such as precision, recall, and F1 score are considered depending on the specific scenarios.

Feature map visualization : Feature map visualization is a technique used to understand and analyze the features learned by a deep neural network in a visual and interpretable way. Deep neural networks, such as Convolutional Neural Networks (CNNs) learn a hierarchy of increasingly complex and abstract features from the input data in each layer of the network, which are represented by the feature maps.

Feature map visualization involves extracting the feature maps from the intermediate layers of the network and visualizing them as images. This allows us to see what the network has learned in each layer and how it represents the input data. Feature map visualization can also help identify any patterns or structures that the network has learned, which can provide insights into the underlying data distribution and the network's decision-making process. There are several methods for visualizing feature maps in deep neural networks, including:

- **Activation maximization**: This method involves maximizing the activation of a specific neuron or feature map in the network by generating an input image that produces a high response from that neuron. This results in an image that highlights the features learned by that neuron or feature map.

- **Gradient-based visualization**: This method involves using the gradients of the output with respect to the input image to visualize the regions of the image that contribute most to the output. This can help identify the important regions of the input image that the network is focusing on.

- **Deconvolutional networks**: This method involves using a deconvolutional network to reconstruct the input image from the feature maps in the intermediate layers of the network. This provides a visual representation of the features learned by the network and how they are combined to produce the final output.

In our system, we extract the top 15 feratures learnt from the last second layer using the Deconvolutional networks and then Evaluate the updated local model after receiving the global weights.

Evaluating the model on validation dataset : In machine learning, after training a model on a dataset, it is essential to evaluate its performance on a separate set of data that it has not seen before. This is done to assess how well the model generalizes to new, unseen data. The dataset used for this purpose is called the validation dataset.

To evaluate the model on the validation dataset, the following steps were taken:

1. The model was run on the validation dataset, and its predictions were compared with the actual labels in the validation dataset.

2. Metrics such as accuracy, precision, recall, and F1 score were calculated to assess the model's performance.

3. If the model's performance was not found to be satisfactory, it was retrained with different hyperparameters.

4. The process was repeated until the model's performance on the validation dataset was found to be satisfactory.

3.1.2 Blockchain Network

In a blockchain-based federated learning system, the model aggregation happens at the blockchain level. The blockchain network plays a critical role in enabling secure and transparent model updates. The participating hospitals train local machine learning models on their own chest X-ray images and share the updated model weights instead of the actual images with the blockchain network. The blockchain network aggregates the model updates from all participating hospitals and generates a global model. Figure 3.2 gives an outline of the blockchain part of our system.

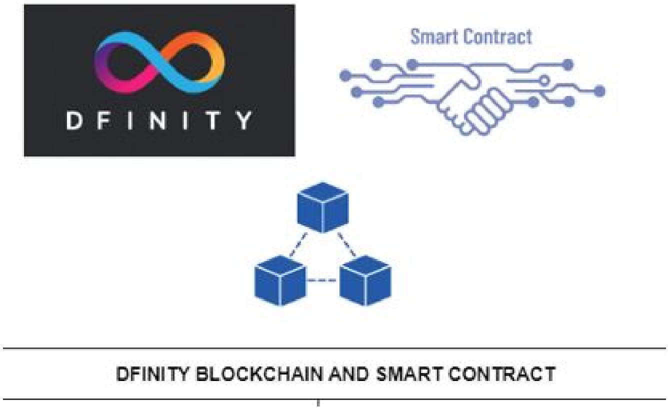


FIGURE 3.2: Internet Computer Blockchain

The algorithms used at this level implement the following key features in the architecture:

1. **Decentralization**

The blockchain network is decentralized, meaning that it does not have a central authority that controls the data. Instead, each node in the network maintains a copy of the ledger, which enables the hospitals to share the updated model weights securely and without a central point of failure.

The Internet Computer Blockchain that we used achieves decentralization through a combination of several innovative technologies and design choices.

First, the Internet Computer network is based on a decentralized peer-to-peer architecture, where multiple independent nodes in the network collectively maintain and validate the blockchain ledger. This means that no single entity has control over the network, and all participating nodes have equal voting rights in the consensus process.

Second, Internet Computer has a governance system that allows stakeholders to vote on proposals and changes to the network. This ensures that the network is controlled by its community of users and not by any single entity.

Finally, Internet Computer is designed to be highly resistant to censorship and tampering. The platform uses cryptography to ensure that all data and transactions are secure and tamper-proof, while also allowing users to remain anonymous if they choose to do so. Together, these technologies and design choices enable Internet Computer to achieve a high degree of decentralization, making it a robust and resilient platform for building decentralized applications.

2. **Consensus Mechanism**

DFINITY uses it's own consensus mechanism to ensure that all nodes in the network agree on the state of the ledger. This mechanism ensures that the data shared among the hospitals is valid and that no node can modify the data without the agreement of other nodes.

Internet Computer uses a unique consensus mechanism called Chain Key Technology (CKT), which allows the network to scale out horizontally across a network of independent nodes. CKT enables multiple subnets to operate in parallel, each with their own set of validators, allowing the network to process large numbers of transactions in a highly parallelized manner.

3. **Smart Contract**

The blockchain network includes a smart contract, which defines the rules for participating in the federated learning process. The smart contract includes model aggregation algorithm which means that the blockchain network is responsible for aggregating the updated model weights from all participating hospitals and generating a global model. The aggregation process can be performed using different techniques, such as Federated Averaging or Secure Aggregation, which enable the network to generate an accurate global model without compromising the privacy of the local data. The Smart Contract in our system is responsible for defining the algorithm for global model aggregation.

4. **Data Sharing**

The blockchain network enables the hospitals to share their local model parameters and to collaborate on model training without compromising the privacy of their patients. The hospitals can access the global model stored on the ledger, but they cannot see the actual data which is contributed by other hospitals to train their respective local models. The hospitals are enabled to share the model parameters with the blockchain using Candid API and http requests.

The model parameters were extracted and then encoded into a binary format using the in-built functions in candid library. This binary format is then transmitted over the network or stored in a canister (the deployed canister) using the ‘requests’ library by specifying the canister id of the deployed canister. When data is received from the network or a canister, it is decoded by converting the binary format back into a high-level and passed to the FedAvg algorithm for parameter aggregation.

3.1.3 End Users

The model is deployed on the blockchain using different APIs provided by blockchain foundations as depicted by figure 3.3. The clients can be the data providing hospitals as well as other hospitals who can access the model through the blockchain. The model deployment component would deploy the trained machine learning model in a production environment, where it could be used to assist radiologists in diagnosing chest X-ray abnormalities.

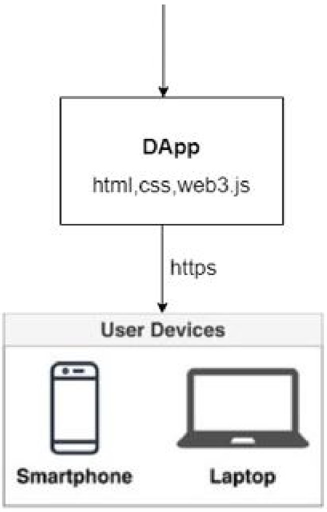


FIGURE 3.3: Interface for End Users

3.2 Proposed System

We propose a system shown in figure 3.4 that consists 3 hospital clients with its own Chest X-ray data. Each hospital will train it's local model using Convolutional Neural Network algorithm for image recognition. The local parameters are sent to the hyperledger[14](blockchain) which aggregates the local parameters without any centralisation. The global parameters are sent back to the hospitals which are later used for disease prediction. The workflow is as follows :

- Local models are trained using local data of each individual system.
- Local models are sent to blockchain for aggregation where the parameters are aggregated according to the algorithm used in smart contract.
- Updated model parameters are sent back to clients and then the local models are updated.
- New block which stores the updated model updates is added into the distributed ledger.

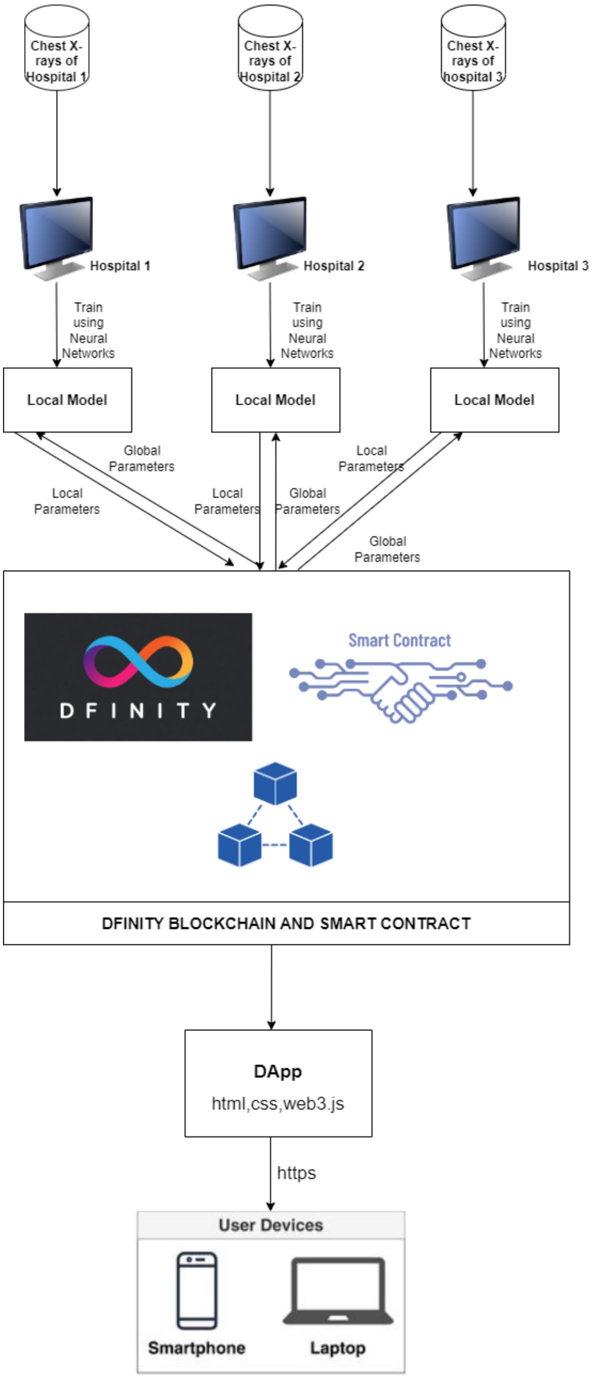


FIGURE 3.4: Architecture Diagram for proposed system

3.2.1 CNN (Convolutional Neural Network) for Local Model training

A CNN is a deep learning algorithm. The key idea behind a CNN is to extract features from an input image using convolutional layers. Convolution is a mathematical operation that involves sliding a filter (also known as a kernel) over an image and computing the dot product of the filter with the corresponding pixels of the image. This produces a feature map that highlights the presence of certain patterns or features in the input image. The output of the convolutional layers is then passed through one or more fully connected layers as shown in figure 3.5, which perform classification or regression on the extracted features. Python Libraries such Tensorflow, Keras, Conv2D etc... are used to achieve the above.

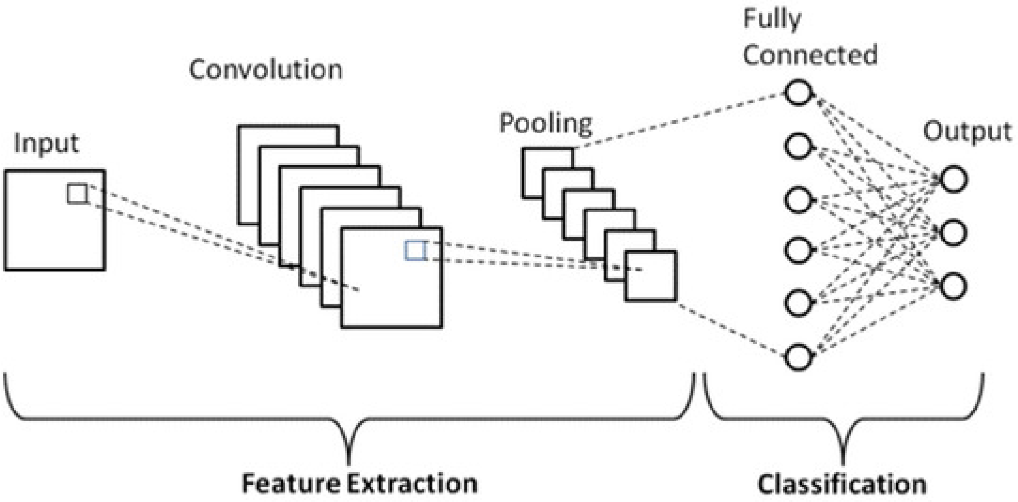


FIGURE 3.5: Convolutional Neural Network

3.2.2 Internet Computer Blockchain for Decentralisation

The Internet Computer blockchain which was created by the DFINITY Foundation, is designed to be a scalable, efficient, and secure blockchain platform that can host a wide range of decentralized applications. Unlike other blockchain platforms, it is capable of running smart contracts directly on the internet, without the need for a separate blockchain network. The Internet Computer blockchain uses a unique consensus mechanism called the ”Chain Key Technology” to achieve high-speed transaction processing and to ensure the security of the network. It also supports interoperability with other blockchain networks, enabling developers to build cross-chain applications.

3.2.3 Motoko Smart Contract – Canister

Internet Computer supports smart contracts in Rust, Motoko etc. A smart contract is a self-executing digital contract that is programmed to automatically execute the terms of an agreement between two or more parties. Smart contracts are typically deployed on a blockchain network, where they can be securely and transparently executed without the need for intermediaries. In the proposed system, the smart contract is written using Motoko language. The smart contract registers new clients and accepts model parameters from them. The smart contract is responsible for aggregating the local model parameters and generation of new global model parameters. After the code is compiled, it can be deployed to the Internet Computer blockchain as a canister, which is a secure container that holds the smart contract code, data, and resources necessary to run the application. Once deployed, the smart contract can be accessed and executed by

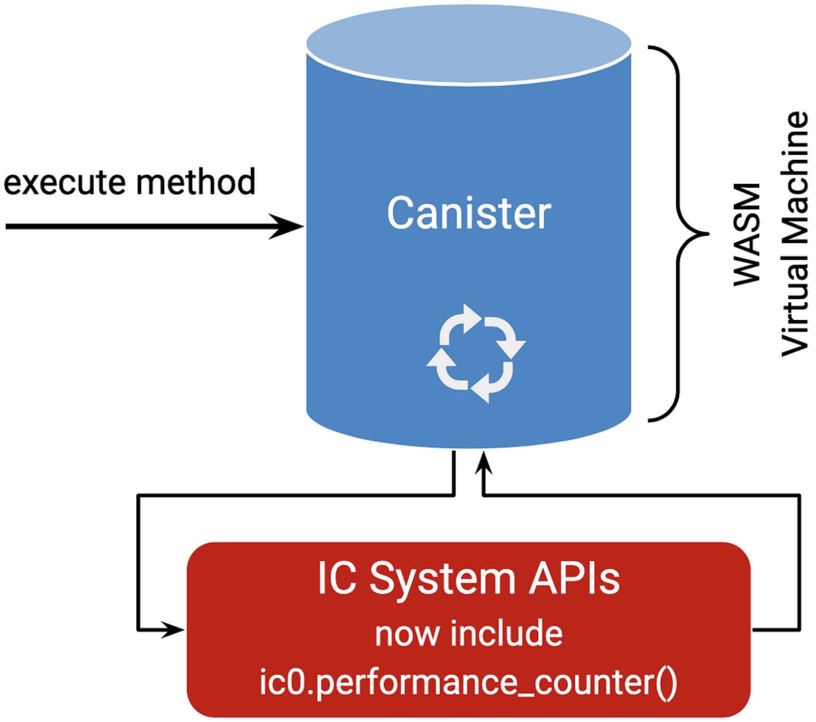


FIGURE 3.6: Internet Computer Canister

other canisters on the Internet Computer, or by external clients through a defined interface which enables the clients to access the global parameters from the Internet Computer Blockchain. Figure 3.6 shows how canisters are deployed in Internet Computer.

3.2.4 CANDID API - The Connection

In the context of the Internet Computer, Candid describes the public interface of a program running as a deployed canister smart contract. Candid enables the interoperation between front-ends and services that are written in different programming languages, such as Motoko, Rust, and JavaScript. In our system, candid API is used to establish communication between the client python codes and the canister deployed on internet computer blockchain. Candid is an Interface

Description Language (IDL) and is preferred over languages like JSON, XML which are Data Description Languages (DDL) because of the following features:

- Candid describes the services as a whole instead of only mapping individual values to bytes/characters.
- The focus lies on the methods that use the data types instead of focusing on the data types.
- The Candid value is mapped directly to the host language's types and values.

With the understanding of the detailed working of the proposed system explained in this chapter, upcoming chapters analyse the experimental results obtained and draws meaningful conclusions.

CHAPTER 4

EXPERIMENTAL RESULTS

The system was tested using three systems as three hospital nodes. This chapter discusses the description of dataset used for the local model training of each node. It further discusses the hardware and software ecosystem under which the experiments were conducted. The procedure of the experiments conducted is explained in detail.

4.1 Dataset Description

We are utilizing an up-to-date collection of X-ray images of Pneumonia sourced from Kaggle. The images were taken from pediatric patients between the ages of one and five, from Guangzhou Women and Children's Medical Center in Guangzhou, and were captured during their routine clinical care. To ensure accuracy, all chest X-rays were initially assessed for quality before being included in the dataset for the AI system's training. Furthermore, two experienced doctors evaluated the images before they could be used for training the AI system, and a third expert reviewed the assessment set to verify that no grading errors had occurred.

The system is designed using the dataset which is organized into 3 folders (train, test, val) and contains subfolders for each image category (Pneumonia/Normal). There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal). The dataset is further divided into 3 parts for our three hospital clients.

4.2 Ecosystem

The federated learning system operates in a decentralized environment [27], consisting of a network of 3 nodes. Each node is a system owned by a volunteer participant who has opted into the federated learning process. The devices have varying computational resources and are connected to the blockchain network through a secure communication protocol.

The system uses medical data from a local hospital to train a machine learning model for predicting patient outcomes. The data is pre-processed to suit the model architecture. The nodes train their respective machine learning models, with each node contributing its local data and computational power.

The canister(smart contract) is deployed on Motoko Playground. It consists of the consensus conditions and the federated aggregation algorithms. The federated aggregation algorithms used in addition to the vanilla aggregation algorithm, use a modified version of the federated averaging algorithm to improve the accuracy and convergence rate of the machine learning model. The algorithm incorporates a privacy-preserving mechanism that prevents any individual node from accessing or manipulating the data of other nodes.

The blockchain network provides a secure and transparent environment for the federated learning process. Transactions and activities are recorded on an immutable ledger, providing a transparent record of the training process.

Experimental results demonstrate that the machine learning model achieved a high level of accuracy, outperforming traditional centralized machine learning methods.

The federated learning process also proved to be robust, with the model retaining its accuracy even when individual nodes failed or left the network.

Overall, the ecosystem of the blockchain-based federated learning system creates a secure, private, environment which can also be incentivized in the future, that encourages collaboration among participants and leads to more accurate machine learning models.

4.3 Experiments Conducted

The system was tested on three systems as three hospitals. Each system was trained on it's local data and the model parameter extraction and encoding of the parameters happens at this level. The individual systems then sent the encoded parameters to the canister deployed on 'Motoko Playground'. Motoko Playground is an online platform that provides a web-based interface for developers to write, test, and execute code in the Motoko programming language. It provides 'cycles' of 20 minutes after which the canister can be re-deployed with a new canister id. The encoded parameters of each local model were sent to the canisters where the aggregation takes place. The aggregated parameters are sent back to the local systems and the local models were updated by the global model.

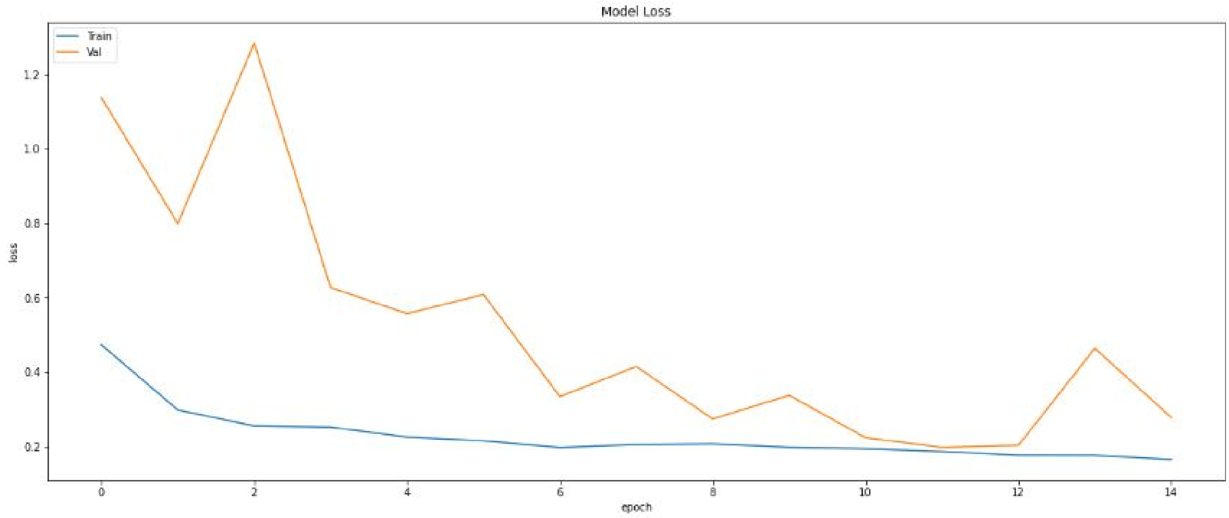


FIGURE 4.1: Loss Curve from training and validation data

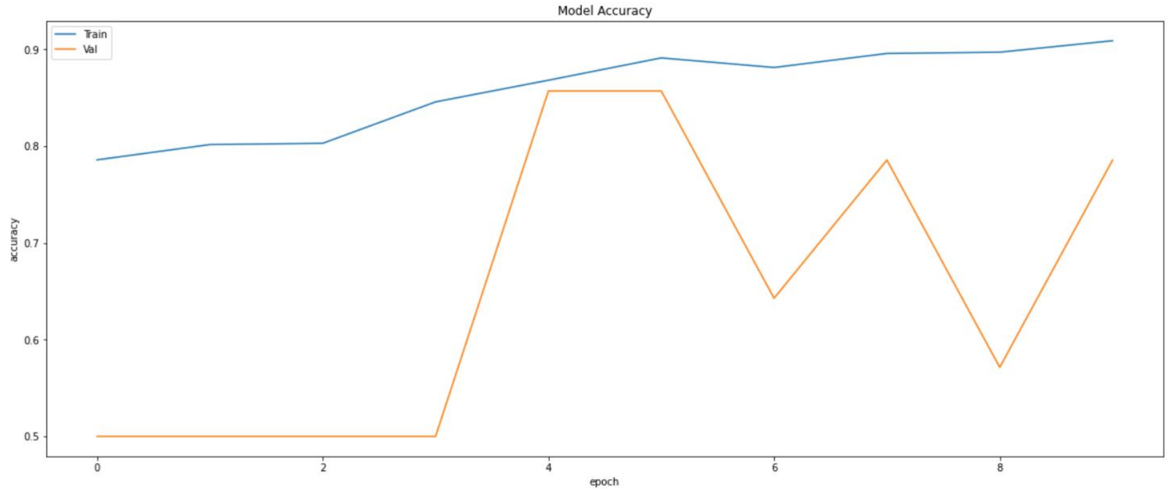
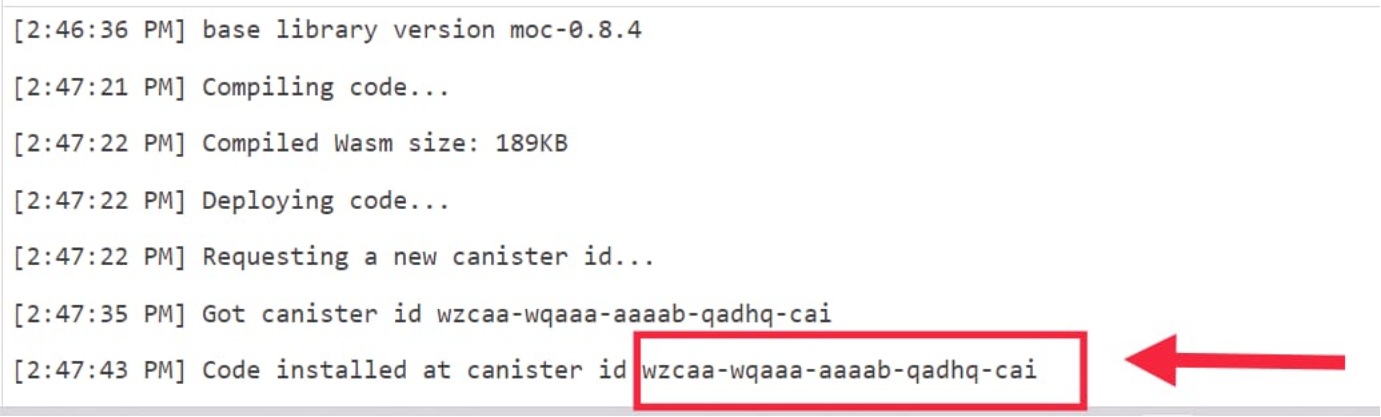


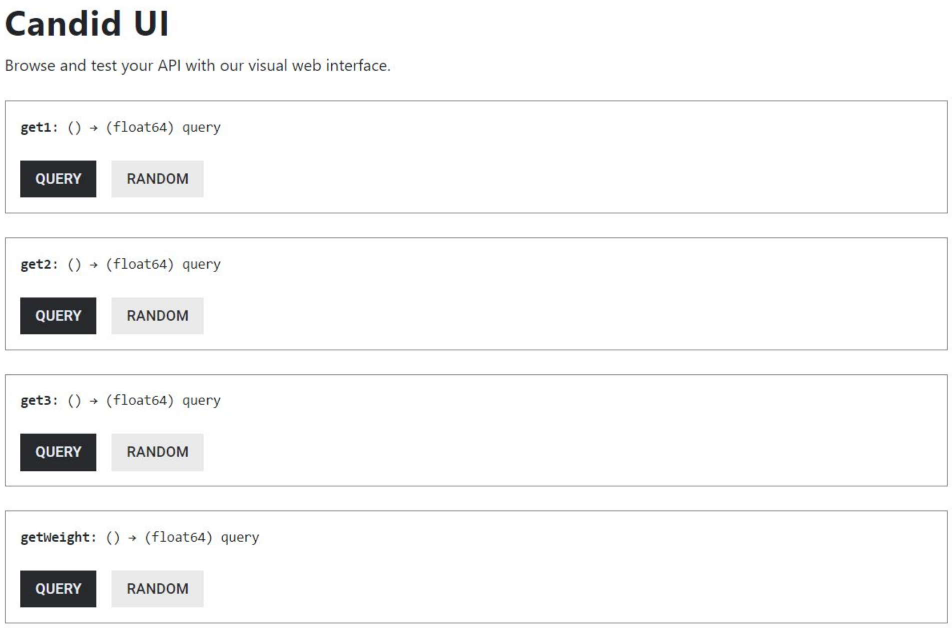
FIGURE 4.2: Loss Curve from training and validation data

The above figures, figure 4.1 and 4.2, are the loss curve and accuracy curve from training and validation data of one of the local models. Similar experiments were conducted for all the local models . The system was tested using three different aggregation algorithms and the three global models generated by them were tested for accuracy. The results obtained by these experiments are discussed in the next chapter.



```
[2:46:36 PM] base library version moc-0.8.4
[2:47:21 PM] Compiling code...
[2:47:22 PM] Compiled Wasm size: 189KB
[2:47:22 PM] Deploying code...
[2:47:22 PM] Requesting a new canister id...
[2:47:35 PM] Got canister id wzcaa-wqaaa-aaaab-qadhq-cai
[2:47:43 PM] Code installed at canister id wzcaa-wqaaa-aaaab-qadhq-cai
```

FIGURE 4.3: Canister ID of a canister deployed on Motoko Playground



Candid UI

Browse and test your API with our visual web interface.

```
get1: () → (float64) query
get2: () → (float64) query
get3: () → (float64) query
getWeight: () → (float64) query
```

FIGURE 4.4: Candid UI showing different functions in Smart Contract

Figure 4.3 represents the canister ID of a canister that is deployed on Motoko Playground. Further, Candid UI is an API in Motoko, which automatically builds a basic UI to represent the working of functions writtenn in the smart contract. Figure 4.4 shows the candid interface of our smart contract and depicts functions in smart contract which include recieving data from nodes, aggregating the model parameters and sending back the updated model parameters to each node.

CHAPTER 5

PERFORMANCE ANALYSIS

The performance analysis of the system is done based on the experiments conducted that are discusses in the previous chapter.

Performance analysis in terms of security depends highly on the Blockchain framework we use. Hosting it on Internet Computer blockchain and building the canister on Motoko playground increases the security by many folds as it is completely decentralised and there is no central server as there is in the case of traditional FL. The removal of central server ensures that attacks like Denial of service are prevented. The possibility of single point of failure is also avoided as blockchain makes the whole system decentralised and hence non dependent on a central server [12]. The problem of malicious clients is also prevented as blockchain ensures consensus between all clients. Internet Computer has its own consensus algorithm in place which is by default used by our project. Moreover, sensitive patient data is not transferred to a central entity which ensures the privacy of the data.

In the experiments conducted, the local models were found to have accuracies of 76%, 82% and 84% (Table 5.1). After the parameter aggregation using different aggregation methods and updation of local model parameters with the new global model parameters at the blockchain level, the new model accuracy was improved. Below are the tables depicting the accuracy analysis of the local models and of the global models built using different aggregation algorithms after each iteration (Table 5.2 and 5.3) .

TABLE 5.1: Local Models

Models	Accuracy
Local Model 1	76%
Local Model 2	82%
Local Model 3	84%

TABLE 5.2: Global Model - Iteration 1

Global Models	Accuracy
Global Model (FedAvg)	85%
Global Model(Geometric Mean)	85%
Global Model(FedBoosting)	87%

TABLE 5.3: Global Model - Iteration 2

Global Models	Accuracy
Global Model (FedAvg)	88%
Global Model(Geometric Mean)	86%
Global Model(FedBoosting)	89%

CHAPTER 6

SOCIAL IMPACT AND SUSTAINABILITY

The use of blockchain technology in a federated learning system for pneumonia detection can have a significant impact on society and sustainability. In terms of social impact, BCFL can help small scale or isolated hospitals to leverage the data of other big hospitals in terms of taking advantage of their contribution in the global model. The use of a federated learning system means that the models are trained using data from multiple sources, rather than from a single centralized source. This reduces the risk of bias and ensures that the models are more accurate and representative of the general population. A blockchain-based federated learning system for pneumonia detection has the potential to improve healthcare outcomes and reduce healthcare costs. By using machine learning to detect pneumonia at an early stage, patients can receive treatment before their condition worsens, leading to better outcomes and lower costs. It can also contribute towards sustainability because it uses less computation power for data transportation by sharing model parameters instead. Use of less computation power will eventually contribute towards sustainability [21].

6.1 Healthcare and security issues addressed by this project

Blockchain-based federated learning has the potential to solve various healthcare issues in hospital systems.

Some of the healthcare issues that can be addressed using this technology are:

- **Data Privacy and Security:** One of the most significant issues in healthcare is data privacy and security. Federated learning allows hospitals to collaborate and share parametric data while ensuring that sensitive information remains protected. Blockchain provides a decentralised environment for aggregation of local models generated by each client without the need to share sensitive patient data.

- **Disease Diagnosis and Treatment:** Federated learning enables hospitals to collaborate and share patient data to improve disease diagnosis and treatment. By pooling data from multiple hospitals, doctors can identify patterns and gain insights that can be used to develop more effective treatment plans.

- **Clinical Trials:** Clinical trials are essential for developing new treatments and medicines. However, recruiting patients for clinical trials can be challenging. Federated learning allows hospitals to collaborate and share data to identify potential candidates for clinical trials, making the process more efficient.

- **Medical Research:** Medical research requires vast amounts of data. Federated learning allows hospitals to collaborate and share data to accelerate medical research. By pooling data from multiple hospitals, researchers can gain insights that would be impossible to achieve with a single hospital's data.

- **Patient Engagement:** Federated learning can also be used to improve patient engagement. It enables the hospitals to leverage each other's data without the need of actually sharing the data through communication channels, doctors can provide patients with more personalized care, leading to better patient outcomes.

Overall, while blockchain-based federated learning has the potential to improve healthcare outcomes, it is crucial to address these issues and challenges to ensure patient health and safety. Hospitals must prioritize data quality, cyber security, regulatory compliance, patient consent, and avoid bias and discrimination when implementing federated learning in their systems.

6.2 Sustainability

Healthcare systems can use BCFL systems in a way that assures long-term sustainability and advantages for both patients and healthcare professionals by taking these elements into consideration.

- **Interoperability:** For sustainability, it's crucial to make sure the blockchain-based federated learning system can interface with current healthcare systems. This can lessen the need for redundant systems and guarantee that data can be shared and accessed with ease.
- **Regulatory Compliance:** Because the healthcare sector is one that is heavily regulated, it is crucial to make sure that the installation of blockchain-based federated learning complies with all relevant laws. Regulation non-compliance may have serious legal and financial ramifications.

In conclusion, a comprehensive approach that takes into account energy efficiency, data privacy and security, interoperability, and regulatory compliance is needed to achieve sustainability in the context of BCFL in a healthcare system.

With the discussion of security issues addressed by this project and the sustainability factors, the next chapter concludes the research done by this paper. It further gives an outline of possible future directions that can be explored in the context of making BCFL systems more reliable and efficient for industry use.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This chapter concludes the research work done in this thesis. It also discusses some possible scope of future directions of the system in terms of scalability, incentivization and more.

7.1 Conclusion

As we saw above, the usage of blockchain based federated learning model in hospital systems solves all the security issues and provides better healthcare facilities [25]. Small scale hospitals can leverage the data and the efficiency available at big scale hospitals without transferring of the actual data. There are still some disadvantages like computational power requirements, legal and regulatory hurdles and blockchain complexity [16]. Some of the open problems are discussed below.

7.2 Future work

With the advancement of technology in coming days, the following are some possible future directions developers can look into for better BCFL systems in terms for scalability and efficiency :

1. **Automating the process**:

In a federated learning system, automation can be used to manage the process of selecting and aggregating models from multiple participants, as well as to monitor the performance of the models and adjust the training process as needed. One approach to automating the process of a blockchain-based federated learning system is to use smart contracts, which are self-executing contracts with the terms of the agreement between parties written into code. Smart contracts can be used to automate many aspects of the federated learning process, such as selecting and aggregating models, monitoring performance, and distributing incentives [19].

2. **Incentivization**:

Incentivization can be a crucial aspect of a blockchain-based federated learning system, as it encourages participants to contribute their data and models to the training process [2]. There are several ways that incentivization can be included in a federated learning system, some of which are outlined below:

- **Reward-based system:** One approach to incentivization is to offer rewards to participants who contribute their data and models to the training process. The rewards can take the form of tokens or other cryptocurrencies, which can be distributed to participants based on their contribution to the training process. This incentivizes participants to contribute high-quality data and models, as they stand to benefit financially from doing so [19].

- **Reputation-based system:** Another approach to incentivization is to use a reputation-based system, where participants are rewarded based on their reputation within the community. Participants who contribute high-quality data and models are given a higher reputation score, which in turn entitles them to a greater share of the rewards. This incentivizes participants to contribute high-quality data and models, as they stand to benefit from an increased reputation within the community [19].

- **Penalty-based system:** A penalty-based system can be used to incentivize participants to contribute high-quality data and models by penalizing them for low-quality contributions. Participants who contribute low-quality data or models may be subject to a penalty, such as a reduction in their reputation score or a reduction in their share of the rewards. This incentivizes participants to contribute high-quality data and models, as they stand to lose out financially if they do not [19].

3. **Storing the ML model on chain**:

Storing the model on the blockchain has several advantages, including increased security, transparency, and immutability. In the Internet Computer blockchain, smart contracts can be used to store and manage the machine learning models. The smart contract can be used to store the model, as well as to manage the process of training and updating the model. Storing the machine learning model on the blockchain also enables the creation of a **decentralized marketplace** for machine learning models, where users can purchase and deploy pre-trained models directly from the blockchain. This would enable developers to access a wide range of machine learning models without the need for a centralized repository, and would provide a transparent and auditable record of the training and validation process.

4. **Using better aggregation techniques**:

The choice of aggregation technique in a blockchain-based federated learning system depends on the specific application requirements, the available resources, and the privacy and security concerns. The current system has been experimented using mathematical aggregation algorithms like FedAvg, FedBoosting (weighted averaging) and Geometric Mean. Other aggregation techniques such as differential privacy, FedMA and other algorithms as they are developed, can be used to ensure better performance of the aggregation global model.

5. **Making use of Kybra to write smart contract in Python**:

Kybra is a Python CDK for the Internet Computer developed by Dfinity Foundation. It can be used for canister development in Python on the Internet Computer (IC).As of May 2023, Kybra is in it's beta version and it does not yet have many live, successful, continuously operating applications deployed to the IC. Kybra is currently developed by Demergent Labs, a for-profit company with a grant from DFINITY.

Use of kybra will enable the use of Python at the blockchain level and hence will enable us to exploit the libraries provided by Python for Federated Learning (mainly Federated Aggregation) in the smart contract which can aid in better performance of the model.

Overall, while blockchain-based federated learning has the potential to improve healthcare outcomes, it is crucial to address these issues and challenges to ensure patient health and safety. Hospitals must prioritize data quality, cybersecurity, regulatory compliance, patient consent, and avoid bias and discrimination when implementing federated learning in their systems.

REFERENCES

1. Attia Qammar, Ahmad Karim2, Huansheng Ning1, Jianguo Ding3 (2022) ’Securing federated learning with blockchain: a systematic literature review’ , Artif Intell Rev, pp. 3951–3985.

2. Batool Z, Zhang K, Toews M (2022) ’Fl-mab: client selection and monetization for blockchain-based federated learning’, 37th ACM/SIGAPP symposium on applied computing, pp 299–307.

3. Chai H, Leng S, Chen Y, Zhang K (2021) ’A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles’, IEEE Trans Intell Transp Syst Vol. 22,No. 7,pp. 3975–3986.

4. Cheng Y, Liu Y, Chen T, Yang Q (2020) ’Federated learning for privacy-preserving AI’, Commun ACM ,Vol.63, No. 12,pp:33–36.

5. Chuan Ma, Jun Li, Ming Ding, Long Shi, Taotao Wang, Zhu Han, and H. Vincent Poor(2020)’When Federated Learning Meets Blockchain: A New Distributed Learning Paradigm’,arxiv,eprints,pp. 1–8.

6. Dai H-N, Zheng Z, Zhang Y (2019)’ Blockchain for internet of things: a survey ’, IEEE Internet Things, Vol. 6,No. 5, pp. 8076–8094.

7. Dawid Połap a, Gautam Srivastava b, Keping Yu (2021) ’Agent architecture of an intelligent medical system based on federated learning and blockchain technology’,Journal of Information Security and Applications, Vol. 58, No. 102748, pp. 2126-2214.

8. Eyal I, Sirer EG (2014) ’Majority is not enough: bitcoin mining is vulnerable’, Financial cryptography and data security, Springer, Berlin, pp. 436–454.

9. Gemeliarana IG AK, Sari RF (2018) ’Evaluation of proof of work (pow) blockchains security network on selfsh mining’, International Seminar on Research of Information Technology and Intelligent systems (ISRITI), pp. 126–130.

10. H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, Blaise Agüera y Arcas (2017) ’Communication-Efficient Learning of Deep Networks from Decentralized Data’, International Conference on Artificial Intelligence and Statistics, Vol. 4, pp. 3-8 .

11. Huang H, Li K-C, Chen X (2018) ’Blockchain-based fair three-party contract signing protocol for fog computing’, Concurr Comput, Vol. 31, No. 22,pp. 4469.

12. Jonathan Passerat-Palmbach, Tyler Farnan, Robert Miller, Marielle S Gross, Heather Leigh Flannery, and Bill Gleim(2019) ’A blockchain orchestrated federated learning architecture for healthcare consortia’ , IEEE International Conference on Blockchain, Vol. 1, pp. 8-11

13. Kim H, Park J, Bennis M, Kim S-L (2020) ’Blockchained on-device federated learning’, IEEE Commun Lett, Vol.24, No. 6, pp. 1279-1283.

14. Kang J, Xiong Z, Jiang C, Liu Y, Guo S, Zhang Y, Niyato D, Leung C, Miao C (2020) ’Scalable and communication-efficient decentralized federated edge learning with multi-blockchain framework’, International Conference on Blockchain and Trustworthy Systems, Vol. 1267, pp 152–165.

15. Kitchenham B (2004) ’Procedures for performing systematic reviews’,EBSE Technical Report of Keele University, Vol. 33, pp 1–26.

16. Li T, Sahu AK, Talwalkar A, Smith V (2020) ’Federated learning: challenges, methods, and future directions’, IEEE Signal Process Mag, Vol. 37,No. 3,pp. 50–60.

17. Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V (2020) 'Federated optimization in heterogeneous networks' , Dhillon I, Papailiopoulos D, Sze V (eds) Proceedings of Machine learning and systems, vol 2, pp 429–450.

18. Long G, Tan Y, Jiang J, Zhang C (2020) 'Federated learning for open banking', Springer International Publishing, Vol. 12500, pp 240–254.

19. Mansoor Ali a , Hadis Karimipour b, Muhammad Tariqa (2021) 'Integration of blockchain and federated learning for Internet of Things: Recent advances and future challenges', Computers and Security, Vol. 108, pp. 3-13

20. Nitin Nikamanth Appiah Balaji, Naveen Narayanan, Kevin J Thelly and Chitra Babu(2021)'Investigating Federated Learning strategies for Pneumonia Image Classification', IEEE MIT Undergraduate Research Technology Conference (URTC), pp. 1-5.

21. Omar El Rifai, Maelle Biotteau, Xavier de Boissezon, Imen Megdiche, Franck Ravat, et al (2020) 'BlockchainBased Federated Learning in Medicine', International Conference on Artificial Intelligence in Medicine, pp. 9-34

22. Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong (2019) 'Federated machine learning: Concept and applications', ACM Transactions on Intelligent Systems and Technology,Vol. 10, No. 2: 1–19.

23. Rajesh Kumar, Abdullah Aman Khan, Sinmin Zhang, WenYong Wang, Yousif Abuidris, Waqas Amin, and Jay Kumar(2020) 'Blockchain-Federated Learning and Deep Learning Models for COVID-19 detection using CT Imaging' ,Journal of Latex Class Files, Vol. 14, No. 8, pp. 1–12.

24. Toyoda K, Zhang A. N (2019) 'Mechanism design for an incentive-aware blockchain-enabled federated learning platform' ,IEEE international conference on big data, pp. 234-296.

25. Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, Ronald M. Summers. ChestX-ray8 (2022) ’Hospital-scale Chest X-ray Database and Benchmarks on Weakly- Supervised Classification and Localization of Common Thorax Diseases’, IEEE CVPR, pp. 3462-3471.

26. Yiqiang Chen, Xin Qin, Jindong Wang, Chaohui Yu, and Wen Gao (2020) ’Fedhealth: A federated transfer learning framework for wearable healthcare’, IEEE Intelligent Systems, pp. 2-6.

27. Yunlong Lu, Xiaohong Huang , Yueyue Dai , Sabita Maharjan , and Yan Zhang (2019) ’Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT’, IEEE Transactions on Industrial Informatics, Vol. 16, No. 6, pp. 4177-4186.

28. Zhang C, Xie Y, Bai H, Yu B, Li W, Gao Y (2021) ’A survey on federated learning’, Knowledge-Based Systems, Vol. 216, pp. 4563-4721.

29. Zhilin Wang, Qin Hu (2021) ’ Blockchain-based Federated Learning: A Comprehensive Survey’, arXiv e-prints, pp. 2110-2182.

30. Zhou Z, Liu P, Feng J, Zhang Y, Mumtaz S, Rodriguez J (2019)’Computation resource allocation and task assignment optimization in vehicular fog computing: a contract-matching approach’ IEEE Trans Veh Technol, Vol. 68, No. 4,pp. 3113–3125.