ABSTRACT

Cardiovascular diseases (CVDs) are one of the leading causes of global mortality, with early prediction playing a critical role in patient outcomes. Traditional machine learning models struggle with the heterogeneous, non-IID nature of healthcare data and often compromise privacy due to centralized data handling. This research will be proposing a framework that integrates blockchain storage with Clustered Federated Learning (CFL) technology to ensure data privacy during model communications and model robustness while enhancing prediction of non-IID data such as heart disease prediction.

This proposed system will group clients based on data similarities, mitigating the impacts of data heterogeneity, allowing federated, customized training within each group. Blockchain storage acts as a secure decentralized communication layer to prevent inference attacks and to eliminate a single point of failure by recording model updates in an immutable ledger.

Experimental results using a heart disease dataset demonstrate that the CFL model outperforms both traditional machine learning models and standard federated learning models in prediction metrics like accuracy, precision, etc., especially for both minority clusters while Blockchain storage assists the models in preserving privacy. This framework lays the groundwork for secure, scalable and personalized healthcare analytics in Internet of Medical Things (IoMT) environments, contributing to more resilient and privacy-respecting AI in medicine.

Keywords:

Clustered Federated Learning
non-IID data
Heart Disease
Blockchain

Chapter 1

Introduction

1.1 Background

Cardiovascular diseases (CVDs), particularly heart failure, continue to represent a significant global health burden. According to the World Health Organization, approximately **17.9 million deaths** worldwide in 2019 were attributed to CVDs [1], highlighting the urgent need for improved strategies in prediction and early intervention.

Accurate and early detection of heart failure can substantially improve clinical outcomes by enabling timely therapeutic interventions. However, traditional machine learning models often fall short when applied to healthcare data due to their inherent assumption that data is independent and identically distributed (IID). In real-world medical environments, patient data is highly heterogeneous, influenced by demographic diversity, evolving clinical practices, and technological advancements.

Moreover, the proliferation of the **Internet of Medical Things (IoMT)** — an interconnected ecosystem of health monitoring devices—has further amplified data diversity. Data collected from different hospitals, devices, and regions exhibits significant variability [2], challenging the effectiveness of centralized learning models and increasing the importance of decentralized, privacy-preserving learning frameworks.

1.2 Motivation

Federated Learning (FL) emerged as a promising solution: a collaborative training method where multiple devices or institutions contribute to model learning without ever sharing their sensitive raw data. By decentralizing training, FL promises both privacy preservation and large-scale collaboration.

However, traditional FL struggles under the weight of highly — exactly the kind of data common in healthcare. Standard federated models assume that every participant's data distribution is roughly the same. When this assumption is violated, as it inevitably is in practice, model accuracy and reliability suffer, especially for minority groups or outlier patients.

There is a clear need for an evolved approach: one that respects the diversity of real-world data while still benefiting from federated collaboration.

Federated Learning (FL) offers a promising decentralized approach by enabling multiple entities to collaboratively train machine learning models without exchanging raw data. In healthcare, this approach is particularly valuable for protecting patient privacy and complying with stringent data protection regulations.

However, FL faces significant challenges when applied to highly **heterogeneous (non-IID) data** [3]. Traditional FL models assume that data distributions across clients are similar, which is rarely the case in healthcare scenarios. As a result, models often exhibit reduced performance, especially for minority groups or underrepresented clusters within the data.

Thus, there is a critical need for approaches that can accommodate data heterogeneity while maintaining privacy and model performance.

1.3 Proposal

To address these limitations, this research proposes an enhanced framework that combines **Clustered Federated Learning (CFL)** with **blockchain integration**.

CFL improves upon traditional FL by grouping clients with similar data characteristics into clusters prior to training. Within each cluster, models are trained collaboratively, thereby better adapting to the specific data distributions of different client groups. This stratified training significantly reduces the biases introduced by non-IID data and improves overall model robustness.

Additionally, blockchain is incorporated as a secure, decentralized communication layer between clients and aggregators. Blockchain technology ensures that model updates are recorded immutably and tamper-resistently, mitigating risks such as inference attacks and single points of failure that are inherent in centralized systems.

Through a series of experimental evaluations, the effectiveness of this blockchain-assisted CFL framework is demonstrated in the context of heart disease prediction.

1.4 Contributions

The major contributions of this research are:

- *Patient Data Clustering:* Patient datasets are clustered based on feature similarities using **agglomerative clustering**, allowing for more balanced and representative model training.

- *Clustered Federated Learning with Logistic Regression:* A clustered federated learning approach using logistic regression is employed to enhance predictive performance and fairness across heterogeneous data clusters.

- *Blockchain-Based Model Communication:* Blockchain is utilized as a secure, decentralized medium for transmitting model updates between nodes, thereby enhancing the overall security and integrity of the learning process.

Chapter 2

Review of Literature

This chapter reviews significant prior work related to federated learning in healthcare, heart disease prediction, strategies to address non-IID data, and the integration of blockchain technology into decentralized learning systems. The objective is to contextualize the current research within the existing body of knowledge and identify gaps that this thesis aims to address.

2.1 Federated Learning in Healthcare

The sensitivity of healthcare data necessitates the development of privacy-preserving machine learning methodologies. Federated Learning (FL) provides a compelling solution by enabling collaborative model training across multiple institutions without transferring raw patient data.

Early works demonstrated the potential of FL to revolutionize healthcare analytics. Chang et al. [4] proposed the integration of blockchain into FL to enhance security and eliminate reliance on a centralized server, thereby addressing vulnerabilities associated with centralized data aggregation.

Additionally, Sheller et al. [5] demonstrated that FL can achieve near-centralized model performance for brain tumor segmentation tasks across multiple hospitals. Their results validated the practical feasibility of FL for large-scale, multi-institutional healthcare collaborations.

Recent systematic reviews such as by Xu et al. [6] further highlight the transformative potential of federated learning in healthcare while acknowledging ongoing challenges related to data heterogeneity, privacy, and interoperability.

However, conventional FL approaches often perform suboptimally when faced with non-IID data distributions, which are prevalent in medical datasets. To overcome this, Shoham et al. [7] introduced Federated Curvature (FedCurv), a technique that incorporates second-order information to enhance model convergence under heterogeneous conditions.

These advancements collectively establish FL as a promising methodology for secure and efficient healthcare data utilization, while also emphasizing the need for further improvements in handling non-IID scenarios.

2.2 Heart Disease Prediction

Heart disease prediction has been a core research domain in medical machine learning. Traditional methods have largely employed centralized datasets, which are very privacy-invasive and might not generalize to heterogeneous populations.

Much of the recent research focused on decentralized architecture implementation. Hasanova et al. [8] designed a heart disease prediction system with blockchain-based machine learning, utilizing a Sine Cosine Weighted K-Nearest Neighbor (SCA WKNN) classifier for prediction accuracy improvement as well as ensuring confidentiality in data.

Other studies have also explored the application of deep learning models, such as convolutional neural networks, for the diagnosis of cardiovascular disease. However, centralization of these models poses critical concerns about data protection and respect for ethical norms.

The integration of federated learning and blockchain technologies in the prediction of heart disease is one possible avenue to solve these problems; however, scalability and system performance problems still linger [9].

2.3 Clustered Federated Learning

Clustered Federated Learning (CFL) has emerged as an effective solution for addressing data heterogeneity in decentralized learning environments. Rather than training a single global model across all clients, CFL organizes clients into clusters based on similarities in their data distributions, enabling more tailored and effective model updates within each cluster.

Brisimi et al. [10] applied CFL to healthcare data, demonstrating improved predictive accuracy and faster convergence compared to traditional FL approaches. Yoo et al. [11] further validated CFL's effectiveness in heart rate variability prediction, where models trained on clustered data outperformed those trained using standard federated approaches.

Hierarchical clustering approaches, such as the method proposed by Briggs et al. [12], allow for dynamic determination of cluster counts, enhancing scalability and adaptability in practical deployments.

These studies suggest that CFL is particularly well-suited for real-world healthcare applications characterized by significant inter-client data diversity.

2.4 Non-IID Data Challenges

Handling non-IID data remains a fundamental challenge in federated learning. In healthcare, data collected across different institutions and devices often varies substantially due to differences in patient demographics, clinical practices, and equipment standards.

Zhao et al. [2] highlighted the detrimental effects of non-IID data on FL performance, proposing the introduction of a small shared dataset to mitigate model divergence. Alternatively, Li et al. [13] proposed FedProx, which modifies the federated objective function with a proximal term to enhance convergence stability under non-IID conditions.

Although these techniques represent important advances, they do not fully resolve the complex issues introduced by severe data heterogeneity. As a result, developing methods that effectively address non-IID challenges remains a critical research priority.

2.5 Integration with Blockchain

Blockchain technology offers unique advantages for enhancing the security, transparency, and robustness of federated learning systems. Its decentralized, immutable ledger structure can serve as a reliable medium for storing model updates and coordinating communication among distributed nodes.

Chang et al. [4] demonstrated that blockchain can be successfully integrated into federated learning workflows to eliminate single points of failure and secure model updates against tampering. Hasanova et al. [8] and Shynu et al. [14] extended this integration to healthcare predictive applications, validating blockchain's utility for sensitive medical data. Additionally, secure aggregation techniques, such as those proposed by Bonawitz et al. [15], further strengthen privacy in federated learning by ensuring that model updates are aggregated without exposing individual contributions. Nevertheless, blockchain's resource-intensive nature — particularly in terms of storage and transaction validation — poses challenges for large-scale deployment. Emerging solutions, such as lightweight consensus mechanisms and consortium blockchains, offer promising pathways to overcome these limitations.

2.6 Emerging Research Gaps and Solutions

Despite significant progress, several critical challenges persist:

- **Scalability:** Efficiently managing communication and computation in large-scale federated learning systems remains an open problem.
- **Bias and Fairness:** Ensuring equitable model performance across diverse patient groups requires further research, particularly in heterogeneous data environments.
- **Dynamic Adaptability:** Current FL systems struggle to adapt to evolving data distributions over time, limiting their long-term effectiveness.
- **Communication Security:** Although blockchain integration enhances security, optimizing performance without compromising security remains an active area of investigation.

The present work addresses these challenges by proposing a blockchain-assisted clustered federated learning framework that prioritizes secure communication, personalized model training, and robustness in heterogeneous healthcare environments.

Chapter 3

Preliminaries

This chapter presents the foundational concepts underlying the research conducted in this thesis. It covers the nature of non-independent and identically distributed (non-IID) data, the principles of machine learning, the fundamentals of federated learning, and the core features of blockchain technology. A clear understanding of these topics is essential to comprehend the motivations and methodologies employed in the proposed framework.

3.1 Non-IID Data

In traditional machine learning, it is often assumed that data samples are independent and identically distributed (IID). This assumption simplifies both the theoretical analysis and practical application of learning algorithms. However, in many real-world scenarios, particularly in healthcare, this assumption does not hold true. Data is frequently non-IID, meaning that it is either dependent across samples or derived from different underlying distributions.

Non-IID data can manifest in two principal ways:

- **Non-Independence:** Data points exhibit correlations or dependencies. Examples include time-series medical records where a patient's current health condition depends on prior states.

- **Non-Identical Distribution**: Data collected from different sources or populations follows different statistical patterns. For instance, patient demographics, medical practices, and disease prevalence rates vary significantly across hospitals and regions.

In the healthcare sector, non-IID data is particularly prevalent. For example, datasets collected at a specialized cardiac hospital may differ considerably from those gathered at a general hospital due to differences in patient populations, diagnostic protocols, and treatment procedures. Similarly, data from IoMT devices vary based on device calibration, patient behavior, and environmental conditions.

Non-IID data introduces substantial challenges for machine learning:

- **Model Bias:** Models trained on non-representative subsets may generalize poorly to the broader population.
- **Data Heterogeneity:** Variability between client datasets complicates aggregation and model convergence.
- **Training Instability:** Standard optimization algorithms designed for IID settings may struggle to achieve convergence or lead to suboptimal solutions.

In federated learning contexts, non-IID data is a critical issue. Since each client operates on its own local dataset, which often significantly differs from others, achieving effective global model updates becomes non-trivial. Techniques such as clustering, personalized model learning, and advanced aggregation methods have been proposed to mitigate the impact of non-IID data, but the problem remains an active area of research.

3.2 Machine Learning

Machine learning (ML) is a group of computational algorithms that enable systems to learn patterns from data and make predictions or decisions without being programmed explicitly for a specific task. In contrast to traditional programming approaches, which rely on direct rule-based coding, ML models learn patterns from the training data received.

The machine learning process generally includes a number of main stages:

- **Data Preparation:** Data collection and preprocessing are critical initial steps. Raw data must be cleaned, normalized, and converted into a suitable format for modeling. Preprocessing may involve missing value imputation, encoding categorical variables, and scaling numerical features to ensure consistent ranges.

- **Model Selection and Training:** The appropriate model is chosen depending on the type of problem and the characteristics of the data. The model is trained on a portion of the data (training set) by reducing the loss function that is used to measure prediction errors. Model parameters are repeatedly adjusted in an attempt to converge to optimal performance.
- **Model Evaluation and Testing:** Once the training process is complete, the model is tested on a validation set to tune hyperparameter values. Then, a distinct test set—never seen during training—is employed to test the model's generalization ability. Standard evaluation metrics employed are accuracy, precision, recall, and F1-score.
- **Model Deployment and Monitoring:** Once a model demonstrates satisfactory performance, it is deployed into production environments. Monitoring is required periodically to detect data drift, concept drift, and performance degradation with time. Retraining is required to maintain predictive effectiveness.

Machine learning has been applied extensively in healthcare for tasks such as disease diagnosis, treatment recommendation, and patient monitoring. However, centralized machine learning models are often constrained by privacy concerns and the need for large, integrated datasets, motivating the exploration of decentralized learning approaches.

3.3 Federated Learning

Federated Learning (FL) is a machine learning paradigm that allows training of models across different decentralized clients without the transfer of local data to a central server. Instead, the clients keep their data private and share model updates, hence raw data is not compromised.

Key characteristics of federated learning include:

- **Decentralized Data Processing:** It implies that the data is stored on local devices with only the model parameters or gradients transmitted. The model minimizes privacy threats and meets data protection regulation.

- **Privacy Preservation:** Mechanisms like differential privacy and secure aggregation protocols are implemented in FL to avoid sensitive data leakage via model updates.

- **Communication Efficiency:** Given the iterative nature of FL, reducing communication overhead is critical. Methods such as model compression, update sparsification, and client sampling are used to optimize communication between clients and the server.

- **Scalability:** FL systems are designed to support thousands or even millions of devices, each contributing to the global model updates asynchronously or synchronously.

- **System Heterogeneity and Client Adaptation:** Devices participating in FL often vary significantly in computational capabilities, network connectivity, and data availability. FL frameworks implement strategies such as adaptive learning rates and selective participation to accommodate such heterogeneity.

Despite these advantages, federated learning faces challenges. In centralized federated setups, a global aggregator may become a single point of failure, raising security concerns. Decentralized federated learning architectures, supported by blockchain or peer-to-peer networks, have been proposed to mitigate this risk. Furthermore, the effectiveness of FL diminishes in the presence of non-IID data, necessitating advanced techniques such as clustered federated learning and personalized federated learning [16].

Overall, FL offers a promising solution for privacy-preserving, scalable, and collaborative machine learning, particularly in sensitive domains such as healthcare.

3.4 Blockchain

Blockchain technology provides a distributed, decentralized ledger that records transactions securely, transparently, and immutably across nodes. Blockchain was originally created for use with cryptocurrencies, but it has been used in many fields that require tamper-proof and trustless systems.

Core features of blockchain include:

- **Decentralization:** Nobody has control over the system in a blockchain network. Consensus algorithms guarantee that everyone is in agreement with the state of the ledger, making it more resilient and reducing single points of failure.

- **Transparency and Immutability:** It is extremely difficult to modify a transaction after it is written into the blockchain. Transactions can be checked by every participant, thus allowing for transparency and accountability.

- **Security through Cryptography:** Advanced cryptographic techniques like hashing and public-private key systems are employed by the blockchain for data validation and transaction authentication.

- **Consensus Mechanisms:** A collection of consensus algorithms such as Proof of Work (PoW), Proof of Stake (PoS), and Byzantine Fault Tolerance (BFT) is used to ensure agreement on the network's state in a trustless environment.

- **Pseudonymity and Privacy:** Although transactions are open, identities of users are kept safe using pseudonymous cryptographic addresses, thus a degree of privacy is maintained while ensuring accountability.

- **Traceability and Auditability:** Each transaction in the blockchain is stamped with a timestamp and is referenced to earlier transactions to enable end-to-end audit trails and verifiable records of data ownership or model updates.

Blockchain in federated learning creates a secure setting for logging model updates and agreeing on distributed training protocols without having to rely on a trusted central authority. Blockchain systems must however be carefully engineered to balance computational costs against security requirements, especially in resource-constrained settings.

Chapter 4

Proposed Work

In this chapter, we will begin by examining the dataset used in our research, followed by a discussion of the preprocessing steps, including feature selection and data clustering methods. We will then provide a detailed overview of the training procedure, describing the complete algorithm and architectures utilized. Finally, we will outline the testing procedure applied in this research.

4.1 Dataset

The data set used in this investigation is obtained from Kaggle under the title ’Heart Disease Dataset’ [17]. It is a consolidated and cleaned version of heart disease records originally compiled from multiple reputable sources, including the countries namely Cleveland, Hungary, Switzerland etc. The dataset is structured for binary classification tasks and is widely used in medical AI applications for evaluating cardiovascular health risks.

The original dataset consists of 1,026 patient records, each described by 13 clinical attributes and 1 binary target label indicating whether the heart disease is present (1) or not (0). The features are as shown in description represent standard diagnostic metrics such as age, resting blood pressure (trestbps), serum cholesterol (chol) etc.

To enable batch-wise training and improve model generalization in a federated learning context, the dataset was expanded to 9,390 records. This augmentation was performed through a combination of oversampling (to replicate underrepresented data) and data synthesization (to introduce controlled noise and variability). The synthesis of new data samples helped simulate more realistic, diverse patient populations, while preserving the overall distributional characteristics of the original dataset.

This dataset served to be the foundation to train and evaluate traditional machine learning models, standard federated learning frameworks, and the proposed blockchain-assisted clustered federated learning model, providing a robust benchmark for heart disease prediction in a decentralized and privacy-aware setting. The attributes used in this dataset are:

- **Age :** The greatest cardiovascular disease risk factor is age. With increasing age, the arteries get hardened and become narrowed, which results in an increased chance of disease conditions such as hypertension and heart attack. This aspect characterizes trends relating to heart disease prevalence with increasing age.

- **Sex :** Biological differences that distinguish men and women shape risk for heart disease. Men typically have a greater risk at a younger age, and women can develop heart disease post-menopause. This dichotomous variable (1 = man, 0 = woman) does identify those trends.

- **Chest Pain Type (cp) :** Different types of chest pain may represent different underlying cardiac conditions. Classic angina (cp = 1) is a characteristic presentation of coronary artery disease, while atypical (cp = 2) and non-anginal (cp = 3) chest pain can be indicative of other conditions. Asymptomatic (cp = 4) individuals can have ”silent” heart disease, which is a dangerous condition in terms of the heightened difficulty of diagnosis without diagnostic examination.

- **Resting Blood Pressure (trestbps) :** Increased resting blood pressure (hypertension) makes the heart pump harder and may hurt arteries over time. It is a well-known reason for having a stroke or coronary thrombosis. This test assists in assessing baseline cardiovascular stress.

- **Serum Cholesterol (chol) :** This is measured in mg/dl. High cholesterol contributes to the formation of plaque in arteries (atherosclerosis), reducing blood flow and raising the risk of heart disease. This is relevant in evaluating lipid-related risk.

- **Fasting Blood Sugar (fbs) :** High fasting blood glucose (if greater than 120 mg/dl, fbs = 1. Else, fbs = 0) may be an indicator of diabetes, which significantly increases the risk of heart disease. Diabetes speeds up atherosclerosis and may harm blood vessels. This dual marker helps to identify metabolic issues.

- **Resting ECG (restecg) :** Electrocardiogram results are used to detect electrical anomalies inside of the heart. It is 0 if normal. ST-T wave abnormalities (restecg = 1) or evidence of left ventricular hypertrophy (restecg = 2) may reflect ischemia or chronic pressure overload. This examination is helpful to identify patients requiring additional cardiac evaluation.

- **Maximum Heart Rate (thalach) :** The capacity of the heart to reach higher rates under exercise can measure cardiovascular fitness. Low peak rate may signify impaired cardiac function or exercise intolerance — both with greater mortality.

- **Exercise-Induced Angina (exang) :** Exercise-induced angina or chest pain (exang = 1 if exists. If not, exang = 0) typically foretells that during exercise, the heart is not supplied with sufficient blood. This is a symptom of narrowed vessels or compromised cardiac function. Its detection allows early medical intervention.

- **Oldpeak :** This is how much the ST segment of the ECG is depressed below baseline during exercise — a key sign of myocardial ischemia. The more depressed, the worse the ischemia, and thus this is a good marker to diagnose it.

- **Slope of Peak Exercise (slope) :** The orientation of the ST segment during peak physical activity provides hints regarding cardiac health. A downsloping ST segment (slope = 3) or a flat ST segment (slope = 2) is especially worrisome, usually a sign of myocardial ischemia. Upsloping (slope = 1) is typically normal.

- **Number of Major Vessels (ca) :** Fluoroscopy identifies the degree of obstruction inside of the coronary arteries. Having an increased number of large vessels with demonstrable abnormalities (0–3) signifies more extensive disease. It serves as one of the most effective indicators of cardiovascular impairment.

- **Thallium (thal) :** This nuclear imaging scan shows how well the flow of blood inside the heart is (thal = 0 if normal). A ”reversible defect” (thal = 2) is areas of no blood during stress but normal at rest, very suggestive of coronary artery disease. A ”fixed defect” (thal = 1) is irreversible damage (e.g., previous heart attack).

- **Heart Disease (target) :** This is the label variable. Based on all other independent attributes the model explains whether the patient either is suffering from a heart disease (1 = presence) or not (0 = absence). All other independent attributes are used to explain this resulting variable. It's needed for training and testing prediction models.

4.2 Preprocessing

This step is an important task to convert raw data into an organized form suitable for modeling and analysis. Null values handling, data standardization or normalization for consistency, and categorical variable encoding are done in this sage. In this project, we undertake feature selection to determine the most influential variables, making the model more efficient and interpretable. Data clustering is also employed to cluster similar observations, revealing underlying patterns and relationships. These steps of preprocessing together facilitate meaningful and useful analysis of intricate data.

4.2.1 Feature Selection

Feature selection is the art of pinpointing the most influential variables in a dataset, revealing which features have the strongest relationship with the target. By focusing on these high-impact attributes, we simplify the model, boosting performance while minimizing irrelevant data and computational demands. Beyond technical efficiency, feature selection sharpens insights, illuminating which factors drive the predictions. Whether using correlation analysis, recursive elimination, or statistical scoring, this process refines the dataset, creating a streamlined path to deeper, more actionable discoveries.

4.2.2 Normalization

Normalization is a technique to transform features of records into a consistent scale, while preserving relative relationships among values. By smoothing out large disparities in feature ranges, normalization allows algorithms to interpret each variable's impact more fairly, especially when data spans different units or magnitudes. This process enhances model performance, preventing variables with larger scales from disproportionately influencing outcomes. In essence, normalization refines the dataset, ensuring that every feature contributes meaningfully and comparably to the analysis or prediction.

We have chosen min-max normalization to scale our data, transforming values to a common range (0 to 1) for non-categorical features while preserving the relationships between data points. This method enhances comparability across features and improves performance for models sensitive to data scaling. By opting for min-max normalization, we ensure a balanced and standardized dataset that supports more accurate and efficient analysis.

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{4.1}$$

where,

- $X \rightarrow$ original value,
- $X_{\text{min}} \rightarrow$ minimum of all Xs,
- $X_{\text{max}} \rightarrow$ maximum of all Xs,
- $X_{\text{norm}} \rightarrow$ normalized value of X scaling from 0 to 1.

4.2.3 Data Clustering

By clustering clients, federated learning systems can divide clients into groups with similar data distributions. Rather than applying a single global model that tries to generalize across all clients, the server can train more specialized models for each cluster. This approach enables the creation of models that are more tailored to the specific needs of different groups of clients, improving both accuracy and personalization. For example, in a healthcare setting, a model trained on data from one hospital might be very different from one trained at another hospital, due to differences in patient demographics, health conditions, or testing protocols. Clustering can mitigate these disparities and produce models that are more effective and relevant to the specific data characteristics of each cluster. Moreover, clustering addresses one of the key challenges in federated learning: reducing bias. When different data distributions are contributed to a shared global model, the model may become skewed or biased toward certain client populations. By grouping clients based on similar data patterns, clustering ensures that the model updates contributed by each cluster are more homogeneous, leading to a fairer and less biased model. This helps in training models that better represent the underlying patterns in the data rather than being disproportionately influenced by one subset of clients.

4.3 Training Procedure

In this research, we will be using Logistic Regression to perform Heart Disease prediction. We will be performing single model modeling CFL by following these steps:

- **Initialize a Global Model:** Create a starting global model with initial parameters.

- **Local Training on Nodes:** Each node uses the global model and its own data to train a local version of the model.

- **Cluster-Level Aggregation:** Each cluster's aggregator combines the local models within its cluster using a specified aggregation method to produce a cluster model.

- **Global Aggregation of Cluster Models:** The global aggregator aggregates all cluster models, producing an updated global model.

- **Iteration of Training and Aggregation:** The updated global model is distributed to nodes, repeating the process to refine the model iteratively.

The first step in this CFL architecture is to initialize a global model, which serves as the starting point for all training processes across nodes. This global model could be a basic model with initial parameters set randomly or based on prior knowledge if available. The global model acts as the central point of reference, ensuring that

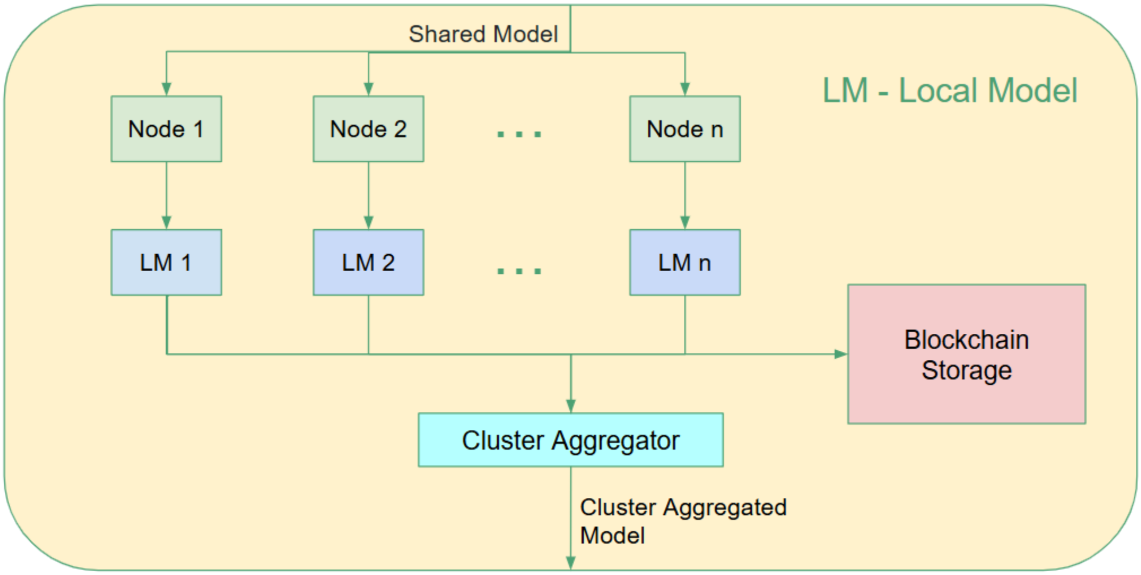

Figure 4.1: Cluster Architecture

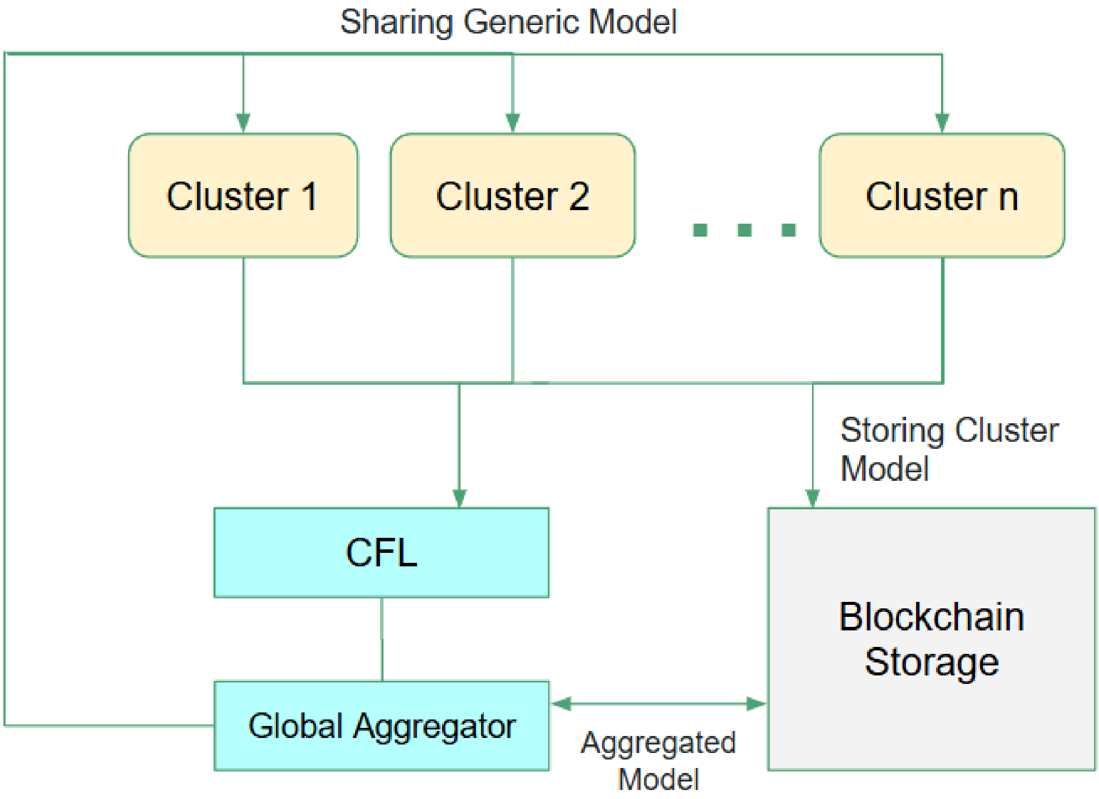

Figure 4.2: Global Architecture

all clients start with the same initial parameters. This initialization aligns the clients toward a shared objective while allowing them to integrate personalized insights from their unique datasets in subsequent steps.

Each node, representing a client, receives the global model and adapts it using its own data. Since data across nodes can vary significantly, each node's local training step is crucial for the model to capture individual nuances. This local training process involves loading the global model parameters and updating them based on the client's data, which could include specific features or unique patterns. Through this step, each node personalizes the global model, embedding localized insights that reflect its data distribution. As each node trains its local model, it adjusts parameters such as the coefficients and intercepts to minimize the loss specific to its dataset, preparing these parameters for the next aggregation phase.

Cluster aggregators then collect the updated local models from the nodes within their cluster. These aggregators apply an aggregation method to consolidate the parameters, creating a single model for each cluster that represents the knowledge accumulated within that group. For aggregation, methods like FedAvg, FedProx, or FedCurv can be used:

- **FedAvg (Federated Averaging)**: Simply averages the coefficients and intercepts of all local models, producing a unified cluster model that reflects an average representation of the nodes within the cluster.

  $$\theta_{t+1} \leftarrow \theta_t - \eta_t \nabla F(\theta_t)$$

  where,

  - ▷ θ_t: Current model parameters at time t
  - ▷ η_t: Learning rate at time t
  - ▷ $\nabla F(\theta_t)$: Gradient of the loss function at θ_t

Algorithm 1 FedAvg Aggregation

```
Initialize aggregated_coef ← None
Initialize aggregated_intercept ← None
Initialize total_clients ← 0
Get addresses of nodes whose variables are aggregated
for each address do
    (coef, intercept) ← get_parameters(address)
    if total_clients == 0 then
        aggregated_coef ← coef
        aggregated_intercept ← intercept
    else
        aggregated_coef ← aggregated_coef + coef
        aggregated_intercept ← aggregated_intercept + intercept
    end if
    total_clients ← total_clients + 1
end for
aggregated_coef ← aggregated_coef / total_clients
aggregated_intercept ← aggregated_intercept / total_clients
Create dummy data (dummy_X, dummy_y)
Fit aggregated_model on dummy_X, dummy_y
Update aggregated_model.coef_ with aggregated_coef
Update aggregated_model.intercept_ with aggregated_intercept
return aggregated_model
```

- **FedProx (Federated Proximal)**: This method also averages, but it includes a regularization term that penalizes deviations from the aggregate, reducing the impact of large variations in any one node's data.

  $$\theta_{t+1} \leftarrow \theta_t - \eta_t \left(\nabla F(\theta_t) + \mu(\theta_t - \theta^*)\right)$$

  where,

  ▷ θ_t: Current model parameters at time t

  ▷ η_t: Learning rate at time t

▷ $\nabla F(\theta_t)$: Gradient of the loss function at θ_t

▷ μ: Proximal term weight

▷ θ^*: Reference model parameters

Algorithm 2 FedProx Aggregation

```
Initialize aggregated_coef ← None
Initialize aggregated_intercept ← None
Initialize total_clients ← 0
Get addresses of nodes whose variables are aggregated
for each address do
    (coef, intercept) ← get_parameters(address)
    if total_clients == 0 then
        aggregated_coef ← coef
        aggregated_intercept ← intercept
    else
        aggregated_coef ← aggregated_coef + coef
        aggregated_intercept ← aggregated_intercept + intercept
    end if
    total_clients ← total_clients + 1
    regularization ← $\mu$ × (coef − aggregated_coef)
    aggregated_coef ← aggregated_coef − regularization
end for
aggregated_coef ← aggregated_coef / total_clients
aggregated_intercept ← aggregated_intercept / total_clients
Create dummy data (dummy_X, dummy_y)
Fit aggregated_model on dummy_X, dummy_y
Update aggregated_model.coef_ with aggregated_coef
Update aggregated_model.intercept_ with aggregated_intercept
return aggregated_model
```

- **FedCurv (Federated Curvature)**: This method uses both averaging and curvature-based adjustments to account for second-order changes across local models, yielding a cluster model that balances smoother adjustments across client data.

$$\theta_{t+1} \leftarrow \theta_t - \eta_t \left(\nabla F(\theta_t) + \lambda H \cdot (\theta_t - \theta^*)\right)$$

where,

- ▷ θ_t: Current model parameters at time t
- ▷ η_t: Learning rate at time t
- ▷ $\nabla F(\theta_t)$: Gradient of the loss function at θ_t
- ▷ λ: Curvature regularization weight
- ▷ H: Hessian matrix of the loss function with respect to θ_t
- ▷ θ^*: Reference model parameters

Algorithm 3 FedCurv Aggregation

```
Initialize aggregated_coef ← None
Initialize aggregated_intercept ← None
Initialize prev_coef ← None
Initialize total_clients ← 0
Get addresses of nodes whose variables are aggregated
for each address do
    (coef, intercept) ← get_parameters(address)
    if total_clients == 0 then
        aggregated_coef ← coef
        aggregated_intercept ← intercept
    else
        aggregated_coef ← aggregated_coef + coef
        aggregated_intercept ← aggregated_intercept + intercept
    end if
    total_clients ← total_clients + 1
    prev_coef ← aggregated_coef.copy()
    regularization ← $\alpha \times$ (coef − prev_coef)
    aggregated_coef ← aggregated_coef − regularization
end for
aggregated_coef ← aggregated_coef / total_clients
aggregated_intercept ← aggregated_intercept / total_clients
Create dummy data (dummy_X, dummy_y)
Fit aggregated_model on dummy_X, dummy_y
Update aggregated_model.coef_ with aggregated_coef
Update aggregated_model.intercept_ with aggregated_intercept
return aggregated_model
```

The result is a cluster model that encapsulates insights common to that cluster's data while mitigating overfitting to any single node's data distribution.

Once each cluster has its aggregated model, the global aggregator performs a higher-level aggregation, merging these cluster models into a comprehensive global model. The global aggregator uses the same aggregation techniques as the cluster aggregator, but here it operates on cluster models rather than individual client models. This step enables the system to capture broader patterns that span across clusters, consolidating diverse data distributions while retaining each cluster's specific contributions. The resulting global model now integrates a refined understanding of the entire system, having distilled knowledge first at the local node level and then at the cluster level.

The global model, now updated with insights from the latest aggregation, is sent back to all nodes, beginning a new iteration of training. Each node receives this refined global model and repeats the local training step, further adapting the model to its data. Through multiple iterations, the model continues to improve, progressively integrating local, cluster-level, and global insights with each cycle. This iterative process allows the model to converge, gradually achieving an optimal balance between global generalization and local personalization.

Through this structured process, the CFL framework effectively builds a robust global model while maintaining privacy and efficiency, capitalizing on the hierarchical structure of clustered federated learning. Each stage—local training, cluster aggregation, and global aggregation—adds another layer of insight, ensuring that the final model is both comprehensive and capable of nuanced generalization across diverse data environments. This cycle repeats, progressively refining the global model and driving it towards optimal performance across all nodes.

4.4 Testing Procedure

In our approach, we opted to cluster the data into four distinct groups, using the features **fbs, trestbps, chol and restecg**. This strategic clustering was driven by the goal of minimizing bias in the target value. By focusing on these specific features, we ensured that each cluster represents a more balanced and coherent subset of data, allowing the model to learn with greater fairness and accuracy. The result is a set of clusters that aligns closely with the underlying patterns in the data, offering a more reliable foundation for the subsequent learning process.

Our research journey began with an in-depth examination of the dataset, setting the foundation for effective preprocessing through feature selection and normalization techniques. We then immersed ourselves in the training phase, providing a detailed account of the algorithms and architectures employed. Lastly, we conducted a thorough analysis of the testing process, evaluating the model's performance and ensuring

its robustness and applicability to real-world predictions.
After the model training phase was completed, we evaluated the effectiveness of the generated global model by testing it on previously unseen data. Specifically, each of the four clusters, which had earlier been partitioned into training and testing subsets, provided its own reserved testing dataset. The generic model produced through Clustered Federated Learning (CFL) was then used to predict the outcomes for each of these four testing sets individually. By comparing the predicted results to the actual labels, we assessed the model's generalization capability across the distinct data clusters. This evaluation process enabled us to validate the robustness of our approach, ensuring that the model not only learned effectively within clustered environments but also maintained high predictive performance when faced with diverse unseen data.

4.5 Smart Contract and Blockchain Storage

In the proposed work, blockchain technology is integrated to securely store the parameters of the trained logistic regression model. Logistic regression estimates the probability P of heart disease using the equation:

$$P = \frac{1}{1 + e^{-t}} \tag{4.2}$$

where t is a linear combination of the dataset attributes:

$$t = \sum_{i=1}^{n} w_i x_i + b \tag{4.3}$$

Here, w_i represents the learned coefficient for each feature x_i, and b denotes the intercept term. Since the dataset consists of 13 attributes, the resulting model contains 13 coefficients along with 1 intercept.

To securely handle these parameters, a smart contract was developed in Solidity and is intended to be deployed on the Ethereum-based consortium blockchain connecting nodes and aggregators. The smart contract structure includes:

- An array of 13 integers representing the model's coefficients,
- A single integer representing the model's intercept.

The contract provides two key functionalities:

- `setModelParameters(int256[13], int256)`: Updates the coefficients and intercept.
- `getModelParameters()`: Retrieves the currently stored model parameters.

Managing Floating-Point Precision:

Solidity, the programming language used for smart contracts, does not natively support floating-point numbers. This design choice is intended to avoid non-deterministic behavior across different nodes and to ensure predictable gas consumption [18]. To overcome this limitation and accurately store real-valued coefficients:

- Each floating-point coefficient and intercept is multiplied by 10^8, preserving up to eight decimal places.
- The integral part of these results are stored on the blockchain.
- When retrieving the values, they are divided by 10^8 to recover the original floating-point values.

This method preserves the precision of the model parameters while maintaining compatibility with blockchain storage constraints. It provides a tamper-proof, decentralized, and verifiable mechanism for sharing model updates, thereby enhancing the security and reliability of the federated learning system.

Chapter 5

Experimental Results and Discussion

This chapter outlines the experimental details, beginning with a description of the experimental setup, followed by an analysis of the results obtained. It concludes with a discussion on the importance of minimizing false negatives and prioritizing patient safety, highlighting the impact of these factors on model effectiveness and clinical applicability.

5.1 Experiment Setting

For experimental purposes we have used three frameworks, namely traditional machine learning, federated learning, and clustered federated learning and compare the metric results within each other. We will be using **Logistic Regression** with $lbfgs$ solver in all the cases.

For CFL, the dataset is clustered into four main groups, with each cluster further divided into 2, 4, 1, and 4 sub-datasets, respectively. Each sub-dataset, containing between 680 and 800 records, is allocated to individual nodes, where local models are trained. These local models are then aggregated within their clusters to form the corresponding cluster models, which are ultimately combined to create a global model. In FL, these 11 local models are sent directly for global model aggregation.

In all experiments, **Python 3** was used to develop the training and aggregation algorithms. For blockchain integration, we utilized the **Truffle Suite** framework to compile, deploy, and manage smart contracts within the Ethereum environment. To simulate blockchain networks for testing purposes, **Ganache** was employed as a personal local Ethereum blockchain. Furthermore, the **Web3** python package was integrated into our Python programs to enable seamless interaction with deployed smart contracts, allowing efficient communication between the machine learning components and the blockchain network.

5.2 Experimental Results and Analysis

In this experiment, we analysed the performance of four different heart disease prediction models:

- Centralized Logistic Regression (LR) model,
- Federated Learning (FL) LR global model,
- Clustered Federated Learning (CFL) LR global model,
- Personalized CFL model.

To systematically assess model performance, confusion matrices were generated for the test datasets of four distinct clusters under each approach.

5.2.1 Confusion Matrix and Performance Metrics

The confusion matrix provides a detailed summary of prediction outcomes:

- **True Positive (TP)**: Heart disease is present and predicted correctly.
- **False Positive (FP)**: Heart disease is absent but predicted incorrectly.
- **True Negative (TN)**: Heart disease is absent and predicted correctly.
- **False Negative (FN)**: Heart disease is present but predicted incorrectly.

According to the confusion matrices, certain performance measures were calculated:

- **Accuracy**: Probability of correct predictions. Calculated by rating the correct predictions over all predictions.

- **Precision**: Calculated by rating true positives among all positive predictions, it reflects the reliability of positive predictions.
- **Recall**: Calculated by rating true positives among all actual positive cases, indicating the sensitivity of the model.
- **F1 Score**: The harmonic mean of precision and recall which offer a balanced evaluation of both metrics.

The traditional Logistic Regression model performed well under all the criteria measured. Nevertheless, despite being robust in prediction, the requirement for centralized data collection poses serious security issues, exposing sensitive patient data to interception during transmission and storage.

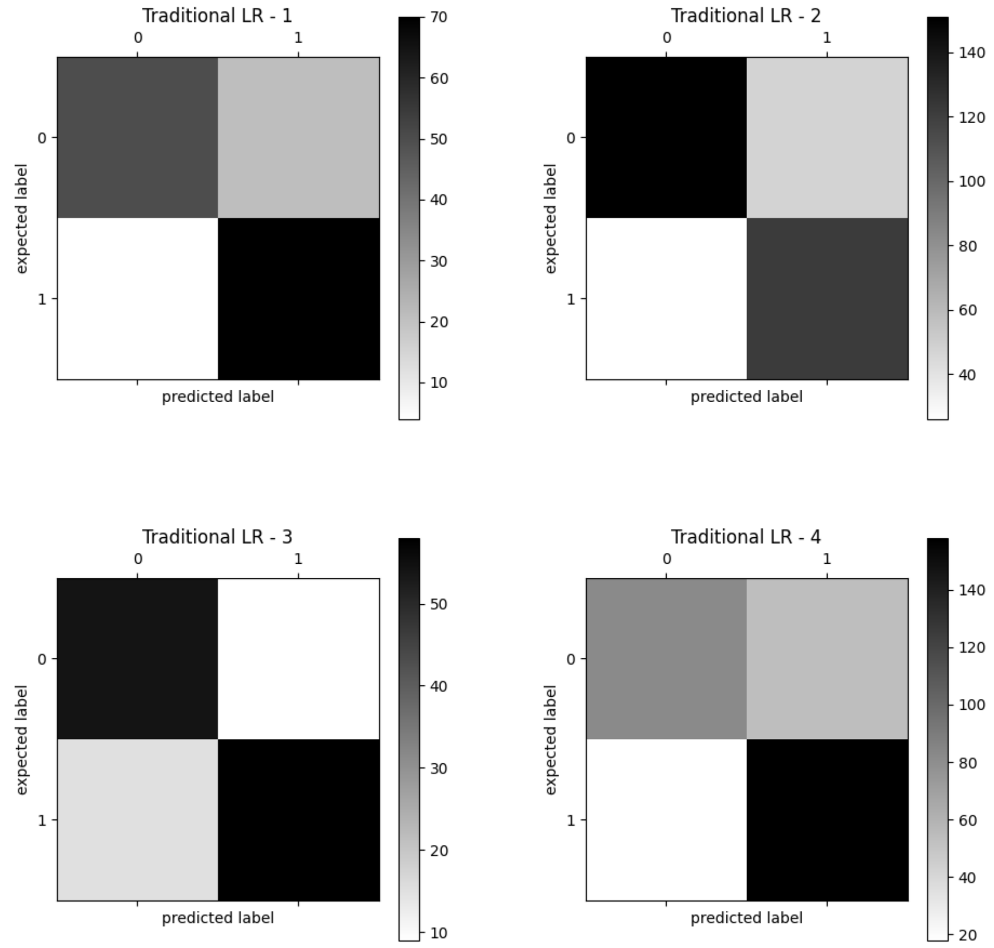

Figure 5.1: Confusion matrices for Traditional Logistic Regression across 4 clusters.

	Test 1	**Test 2**	**Test 3**	**Test 4**
Accuracy	82.758620%	78.901734%	82.352941%	76.923076%
Precision	84.595736%	79.698118%	82.719395%	77.785195%
Recall	82.758620%	78.901734%	82.352941%	76.923076%
F1 Score	82.474399%	79.009821%	82.375859%	76.233551%

Table 5.1: Performance metrics for Traditional Logistic Regression across test datasets of 4 clusters.

Conversely, Federated Learning design, where data has been stored on-site, exhibited a significant reduction in its performance. This fall can be explained by the Non-IID data available on nodes, a point confirmed by Li et al. [13], which created bias that reduced the model's capacity to generalize well.

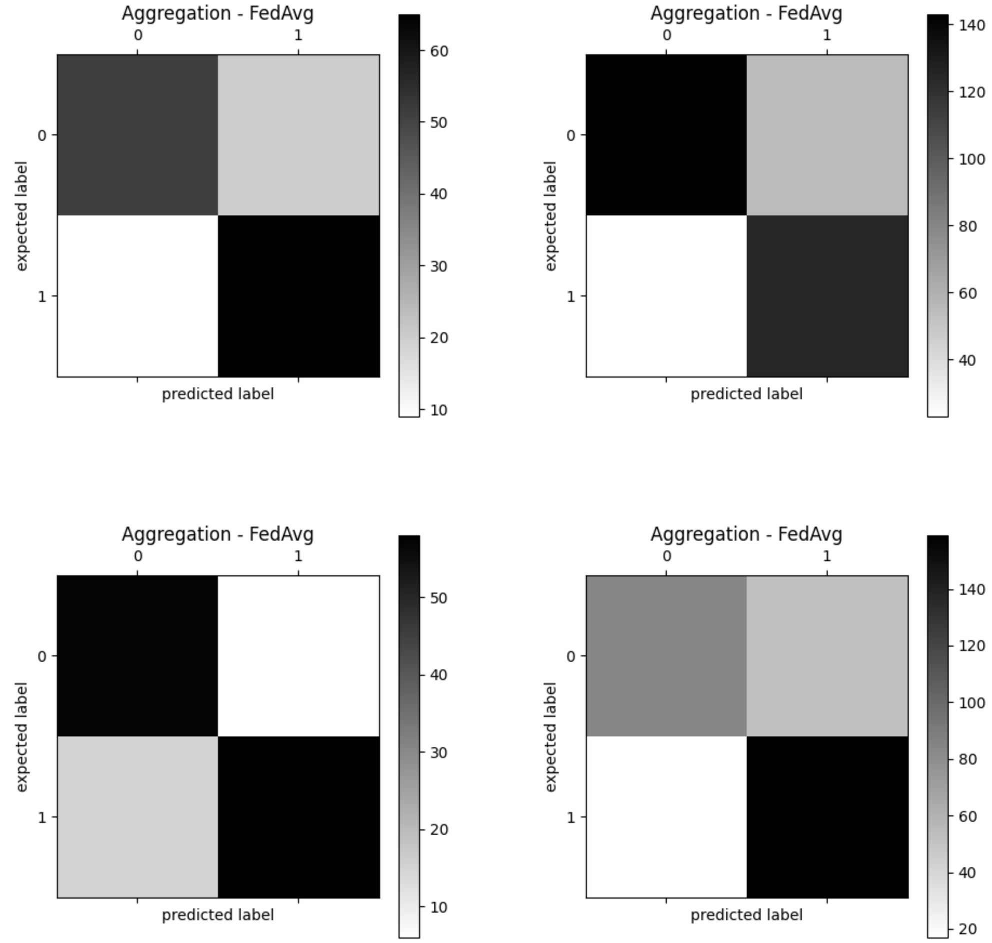

Figure 5.2: Confusion matrices for Logistic Regression global model using Traditional FL across 4 clusters.

	Test 1	**Test 2**	**Test 3**	**Test 4**
Accuracy	80.000000%	77.456647%	84.558823%	77.884615%
Precision	80.647058%	79.001168%	85.317095%	78.761058%
Recall	80.000000%	77.456647%	84.558823%	77.884615%
F1 Score	79.852129%	77.565406%	84.566337%	77.251672%

Table 5.2: Performance Metrics of global model of Traditional FL predicting the test data of 4 clusters

The use of Clustered Federated Learning improved the traditional Federated Learning approach by clustering data of similar characteristics prior to federated aggregation. While large clusters registered slight decreases in performance, small clusters registered notable increases in performance. This shows that the clustering phase effectively reduced bias and enhanced the model's ability to represent the underlying data structure effectively.

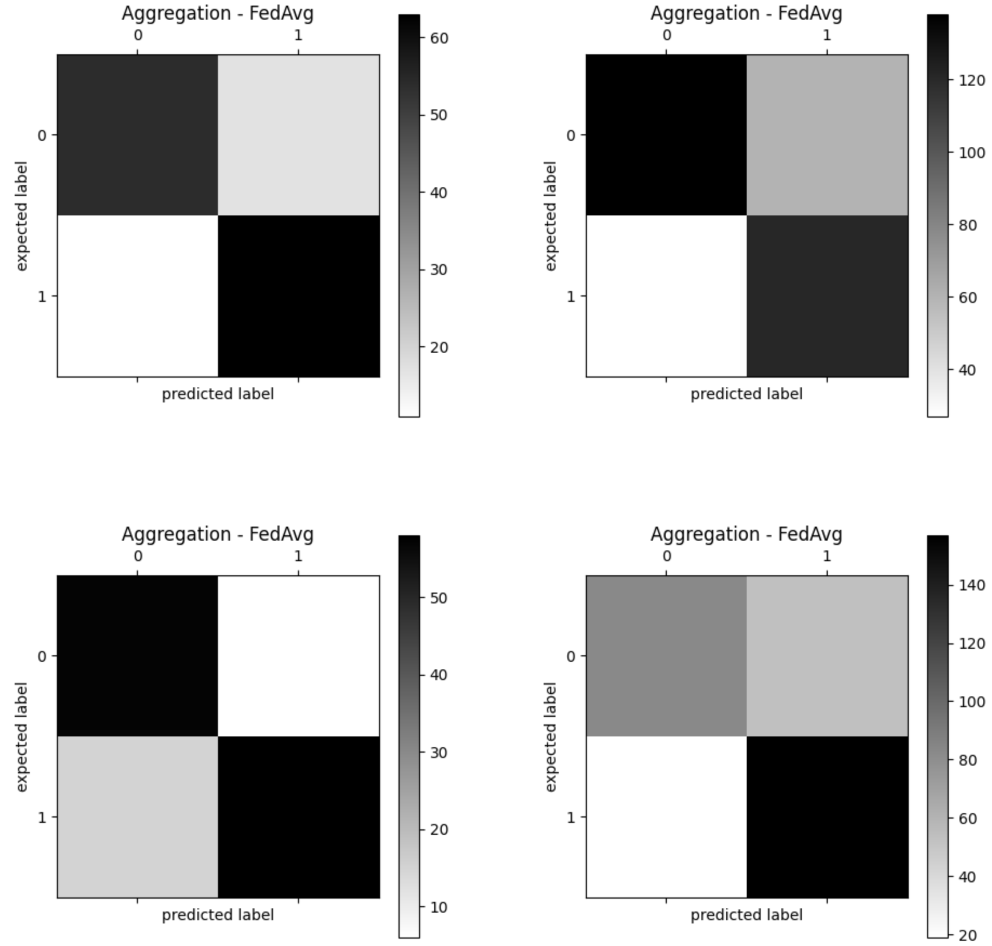

Figure 5.3: Confusion matrices for Logistic Regression global model using Clustered FL across 4 clusters.

	Test 1	**Test 2**	**Test 3**	**Test 4**
Accuracy	80.689655%	74.855491%	84.558823%	76.923076%
Precision	80.868700%	76.456423%	85.317095%	77.643467%
Recall	80.689655%	74.855491%	84.558823%	76.923076%
F1 Score	80.639867%	74.973605%	84.566337%	76.291066%

Table 5.3: Performance Metrics of global model of Clustered FL predicting the test data of 4 clusters

This can be demonstrated by applying personalization to the CFL model, where fine-tuning the generic cluster models locally on individual client datasets led to significant improvements in performance metrics, thereby validating the effectiveness of the clustering approach in addressing data heterogeneity.

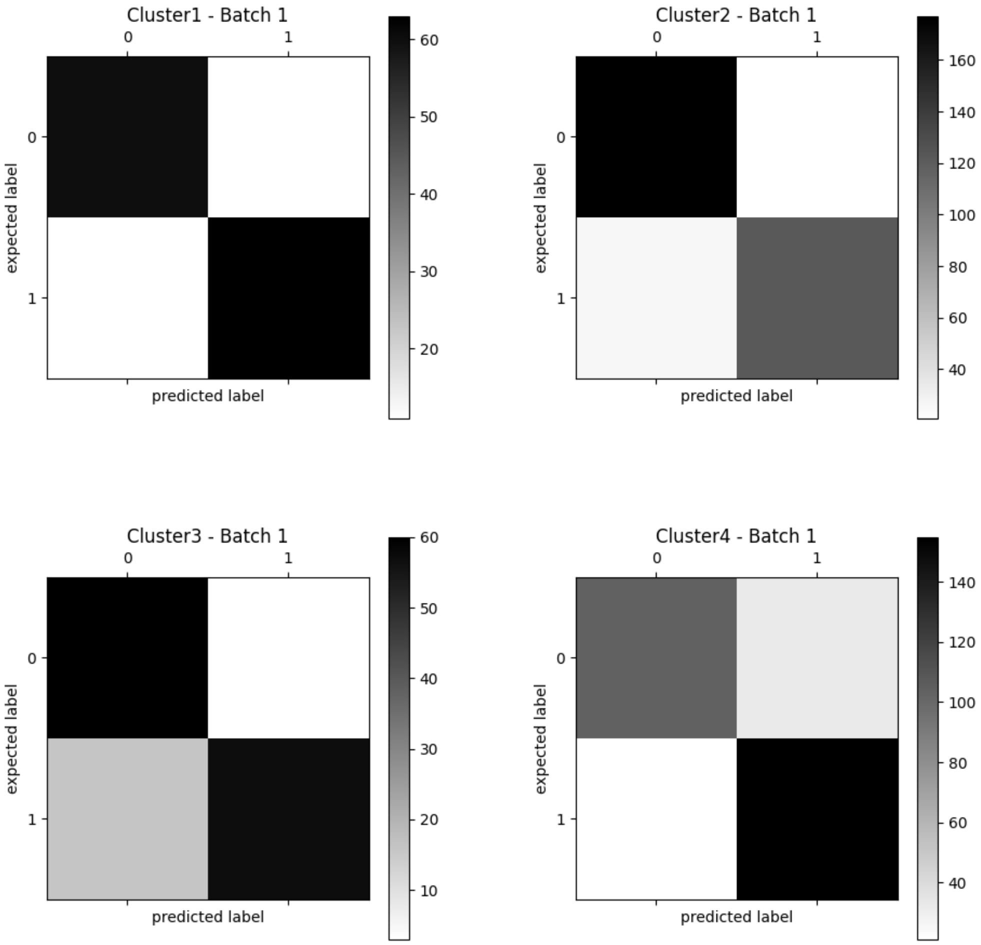

Figure 5.4: Confusion matrices for Logistic Regression personalized model using Clustered FL across 4 clusters.

	Test 1	**Test 2**	**Test 3**	**Test 4**
Accuracy	84.827586%	86.461849%	86.029411%	83.012820%
Precision	84.827586%	86.389054%	87.563854%	83.023831%
Recall	84.827586%	86.461849%	86.029411%	83.012820%
F1 Score	84.827586%	86.384172%	85.999939%	82.912234%

Table 5.4: Performance Metrics of personalized model of Clustered FL predicting the test data of 4 clusters

5.2.2 Security and Robustness using Blockchain Integration

In addition to improving predictive performance, system security and robustness were also emphasized. Blockchain technology was incorporated into the system design to protect model communication and storage. Blockchain's decentralization removed single points of failure, and model composite aggregations were conducted on the latest models stored on the blockchain ledger. This design allowed for continuous operation even on the failure of individual nodes, thus improving the security and integrity of the federated learning system.

Chapter 6

Limitations and Future Work

One of the significant drawbacks of this experiment is the diversity and quantity of the data. The dataset utilized was primarily oversampled and synthesized from a single small dataset. Consequently, the clustering was performed on a generic dataset where the differences between clusters were minimal. As a result, the analysis exhibited only small variations in performance across clusters, rather than significant and pronounced differences that would have been more indicative of the clustering strategy's effectiveness.

If this project were to be implemented on a global scale, where each country possesses distinct data characteristics influenced by factors such as food habits, lifestyle, climate, and other socio-economic conditions, the differences between clusters would be much more significant. In such a scenario, the proposed architecture would likely demonstrate even greater success in handling data heterogeneity and improving personalized model performance.

Furthermore, the system architecture introduced in this study can be extended beyond heart disease prediction. It can be applied to other fields involving non-IID data distributions, especially where security and privacy measures are critical, such as in finance, legal records, and personalized healthcare systems. Investigating these potential applications may provide further evidence to the scalability, flexibility, and fault-tolerance of the proposed framework integrating blockchain technology and clustered federated learning.

Chapter 7

Conclusion

In this paper, we presented a privacy-aware, scalable, and secure heart disease prediction system using the integration of Clustered Federated Learning and blockchain. The suggested method has been proved to be effective in processing non-IID data by clustering the clients with identical data distribution before the federated aggregation phase. The optimization of the cluster models also enabled the advantages related to the clustering mechanism, achieving significant improvements in prediction accuracy on several datasets.

With those objectives in view, the use of blockchain technology created a trusted decentralized platform to store data and communicate, lower security risks such as inference attacks, and eliminate single points of failure.

Despite the limitations experienced in the present study because of the unavailability of diverse real-world datasets, the resulting architecture and methodology provide a solid basis for possible large-scale implementation, where differences in regional data attributes may further enhance the efficacy of the proposed method. Furthermore, the methodology is also potentially applicable to other areas other than healthcare, especially to those industries where confidentiality and protection of sensitive information are of the utmost importance.

This research establishes a foundation for the creation of robust, reliable, and extremely flexible federated learning systems that are required for future use in the Internet of Medical Things (IoMT) and other decentralized settings.

Bibliography

[1] World Health Organization. Cardiovascular diseases (cvds) fact sheet, 2021. URL https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds). Accessed: 2025-04-27.

[2] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, David Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

[3] Nicola Rieke, Jonah Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, et al. The future of digital health with federated learning. *npj Digital Medicine*, 3:119, 2020.

[4] Haoyue Chang, Zeyu Xiong, Dusit Niyato, Junshan Kang, Ning Zhao, Yu Zhang, and Zhu Han. A blockchain-based federated learning method for smart healthcare. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2020.

[5] Micah J Sheller, Guido A Reina, Brandon Edwards, Jacob Martin, and Spyridon Bakas. Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10(1):12598, 2020.

[6] Junyi Xu, Benjamin S Glicksberg, Chang Su, Patrick Walker, and Jiang Bian. Federated learning in healthcare: a systematic review. *Computers in Biology and Medicine*, 138:104928, 2021.

[7] Nadav Shoham, Ofer Dekel, and Naftali Tishby. Overcoming forgetting in federated learning on non-iid data. *arXiv preprint arXiv:1910.07796*, 2019.

[8] Hatam Hasanova, Young-Sik Baek, and Kwangjo Cho. A novel blockchain-enabled heart disease prediction mechanism using machine learning. *Sensors*, 19(24):5429, 2019.

[9] Sandeep Angraal, Harlan M Krumholz, and Wade L Schulz. Blockchain technology: applications in health care. *Circulation: Cardiovascular Quality and Outcomes*, 10(9):e003800, 2017.

[10] Theodora S Brisimi, Ruijia Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International Journal of Medical Informatics*, 112:59–67, 2018.

[11] Daesik Yoo, Sung-Hyun Lee, and Seungmoon Moon. Personalized federated learning with clustering: Non-iid heart rate variability data application. *IEEE Access*, 10:16092–16103, 2022.

[12] Connor Briggs, Zhao Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6147–6154, 2020.

[13] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Robust and communication-efficient federated learning from non-iid data. *arXiv preprint arXiv:2003.02133*, 2020.

[14] P G Shynu, Fahad Alharbi, and Hongyu Wang. Blockchain-based secure healthcare application for diabetic-cardio disease prediction in fog computing. *Computers, Materials & Continua*, 72(1):1499–1516, 2022.

[15] Keith Bonawitz, Vladimir Ivanov, Benjamin Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191. ACM, 2017.

[16] Peter Kairouz, H Brendan McMahan, Brendan Avent, et al. Advances and open

problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

[17] Matthias Pfisterer Robert Detrano Dheeru Dua Andras Janosi, William Steinbrunn and Casey Graff. Heart disease dataset. https://archive.ics.uci.edu/ml/datasets/Heart+Disease, https://www.kaggle.com/datasets/sintariosatya/heart-disease-dataset, 1988. Original data contributors: Janosi, Steinbrunn, Pfisterer, Detrano. Curated at the UCI Machine Learning Repository. Accessed: 2025-04-27.

[18] Solidity documentation: Types - value types. https://docs.soliditylang.org/en/latest/types.html#value-types. Accessed: April 26, 2025.