Fed-MStacking: Heterogeneous Federated Learning With Stacking Misaligned Labels for Abnormal Heart Sound Detection

Temidayo Tom Afelumo
Registration: 2411655
School of Computer Science & Electronic Engineering

University of Essex

July 2, 2025

Abstract

Cardiovascular diseases (CVDs) are the world's leading cause of death, with early diagnosis often limited in low-resource settings by inconsistent manual interpretation and privacy concerns around centralized AI solutions. This project proposes **MStacking**, a novel federated learning (FL) framework designed to detect abnormal heart sounds while safeguarding patient data. In MStacking, each healthcare institution trains its own binary classifier using only local data—typically covering the normal class and one abnormal class—reflecting real-world disadvantages such as label imbalance and varied computing resources.

Instead of sharing sensitive data or model parameters, clients send prediction scores and data density estimates to a central server. A meta-learner then combines these outputs using a stacking ensemble approach to build a comprehensive global classifier. The development supports multimodal inputs, including 1D acoustic signals and 2D spectrograms of phonocardiogram (PCG) recordings.

Experiments using publicly available datasets, like PhysioNet/CinC 2016, simulate federated conditions to benchmark MStacking against traditional centralized and FL models. Key evaluation metrics include accuracy, communication efficiency, and robustness to label noise and data imbalance.

By enabling collaborative model training without exposing patient data, MStacking offers a practical, scalable, and privacy-preserving solution for AI-powered auscultation—especially in decentralized and diverse healthcare environments.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Cardiovascular diseases (CVDs) remain the leading cause of illness and death globally, with the burden falling especially hard on people in low- and middle-income countries. In many of these settings, access to advanced diagnostic tools like ECGs and cardiac imaging is limited [18]. In contrast, listening to heart sounds—a method known as auscultation—offers a non-invasive, affordable, and widely accessible way to detect heart conditions such as valve disorders, coronary artery disease, and arrhythmias [7, 19].

Thanks to recent advances in artificial intelligence (AI) and the growing ecosystem of the Internet of Health Things (IoHT), it's now possible to automate and remotely monitor these heart sounds. This development opens the door to early diagnosis and continuous care, even in remote or resource-limited clinics [25, 14]. However, most existing AI models rely on centralized data collection, which poses serious privacy risks and faces ethical and legal constraints, especially under regulations like HIPAA [21]. This makes many hospitals hesitant to share sensitive patient data.

Federated Learning (FL) offers a promising alternative. It allows institutions to collaboratively train models without ever exchanging raw data [26]. Still, traditional FL methods often assume all participants use the same model types and share the same label distributions—an assumption that doesn't reflect reality. Clinics may only have access to a narrow slice of heart sound data and might use different machine learning tools based on their resources [15, 13].

To tackle these limitations, we introduce **MStacking**—a federated learning framework that supports heterogeneous models and learns from partially labeled datasets. It uses a stacking ensemble method to merge insights from each client's locally trained model, combining both raw acoustic features and time–frequency images like PCG spectrograms. MStacking aims to make AI-assisted auscultation truly practical and inclusive, supporting scalable and privacy-preserving diagnostics across diverse medical environments [26].

Chapter 2

Literature Review

2.1 Abnormal Heart Sound Detection

Cardiovascular diseases (CVDs) remain the leading cause of mortality globally, emphasizing the need for early and accurate diagnosis to improve patient outcomes and quality of care. While diagnostic tools such as echocardiograms and electrocardiograms (ECGs) are standard in high-income countries, many low- and middle-income regions face significant challenges in accessing these technologies. In such settings, heart auscultation—the practice of listening to heart sounds using a stethoscope—serves as a practical, non-invasive, and affordable screening method for detecting abnormalities like murmurs, valve disorders, and arrhythmias. However, the effectiveness of auscultation heavily relies on the clinician's expertise, which can result in inconsistent interpretations and diagnostic errors.

To overcome these limitations, significant progress has been made in computer-aided auscultation (CAA) systems. These systems combine digital stethoscopes with advanced signal processing and machine learning algorithms to enable automated and standardized analysis of heart sounds [21]. CAA systems extract relevant features from phonocardiogram (PCG) signals—such as energy, entropy, time-frequency characteristics, and wavelet transforms—to detect key components like the first (S1) and second (S2) heart sounds, murmurs, and abnormal extra sounds [7, 14]. Traditional machine learning methods, including support vector machines (SVMs), decision trees, and random forests, have demonstrated strong performance when applied to well-annotated datasets.

The advent of deep learning has further enhanced these capabilities, with models such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) being utilized to learn complex patterns from raw or preprocessed PCG recordings [25, 3]. Modern approaches often integrate both 1D audio waveforms and 2D visual features—like spectrograms—to capture complementary information and improve classification accuracy. The availability of publicly accessible datasets, such as those from the PhysioNet/CinC 2016 Challenge, has greatly facilitated the development and benchmarking

of these systems by providing a wide array of expert-labeled heart sound recordings [3, 9].

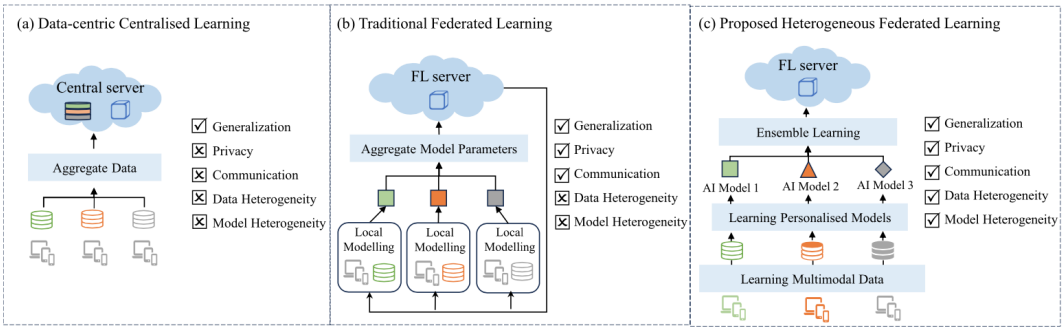

Figure 2.1: Research background. (a) Data-centric centralised learning, which pools data together to train a central ML model. (b) In traditional FL, the global model is trained under the coordination of the central server while data resides in different data silos. (c) The proposed heterogeneous FL, which addresses the limitations of FL through ensemble personalised models learning.

Despite major strides in heart sound analysis, several challenges remain. One persistent issue is the variability between patients, which can affect how heart sounds are interpreted. Recordings often contain background noise or artifacts, and some critical heart conditions—like rare anomalies—are underrepresented in datasets, making it harder for models to learn from them. For AI systems to be trusted and adopted in clinical settings, they also need to generalize well across populations, be interpretable by healthcare professionals, and run efficiently on edge devices such as digital stethoscopes or wearables.

To tackle these hurdles, researchers are turning to methods like domain adaptation to help models adapt across different patient groups and clinical conditions. Explainable AI (XAI) techniques are being developed to make model decisions more transparent, helping clinicians understand why a certain prediction was made. Meanwhile, privacy-preserving approaches like federated learning are gaining traction, enabling collaborative model training without sharing sensitive patient data. Together, these innovations are paving the way toward a scalable and equitable future for AI-powered heart sound diagnostics—one that could benefit patients around the world regardless of where they live.

2.2 AI and IoHT in Remote Healthcare

The Internet of Health Things (IoHT) brings together wearable devices, mobile health applications, and smart sensors to enable continuous patient monitoring and real-time data collection. This integrated ecosystem allows healthcare providers to remotely track key physiological signals such as heart rate, respiration, and body temperature—an espe-

cially valuable tool for managing chronic conditions and supporting elderly populations [1, 15].

Artificial intelligence (AI) takes this a step further by analyzing the large volumes of multimodal data generated by these devices. With AI, systems can detect early warning signs, issue timely alerts, and offer tailored health recommendations—all while reducing the workload on clinicians and giving patients more control over their care [13].

However, traditional AI systems typically rely on centralized data collection, where patient information is uploaded to remote servers for model training. While effective, this approach poses significant privacy and security risks. Healthcare data is not only sensitive but also highly regulated, and centralization increases the chance of breaches and biases—especially when the training data doesn't reflect diverse populations [26].

To overcome these limitations, federated learning has emerged as a powerful alternative. Rather than sharing raw data, federated learning allows models to be trained locally on each device or institution's data and then aggregated into a shared global model. This preserves privacy, reduces data exposure, and fosters secure collaboration across different healthcare stakeholders—all while maintaining the performance benefits of modern AI.

2.3 Federated Learning in Healthcare

Federated learning (FL) offers a privacy-conscious way for healthcare institutions to collaboratively train AI models without sharing sensitive patient data [**?**]. Instead of moving data, each institution trains models locally and only shares model updates with a central server. Using algorithms like FedAvg, these updates are combined to build a stronger global model. As shown in Figure 2.2, this process supports local training, centralized aggregation, and privacy by design. FL is already proving effective in medical use cases like disease prediction [26], mortality risk estimation [4], and secure electronic health record (EHR) sharing [2, 23].

Despite its promise, real-world FL systems face three major challenges, as illustrated in Figure 2.2: (1) non-identical data distributions (non-IID) across institutions [12], (2) differences in local model architectures [28], and (3) incomplete class labels within client datasets [30]. While earlier studies [**?** **?**] have shown FL's potential for heart sound classification, they often assumed ideal scenarios—uniform models and fully aligned label spaces. For FL to succeed in clinical settings, its architecture must adapt to these real-world constraints.

2.4 Personalization and Heterogeneity in FL

To better reflect the diversity across clients in federated learning, researchers have explored several personalization strategies. These include clustered FL [28], multi-branch

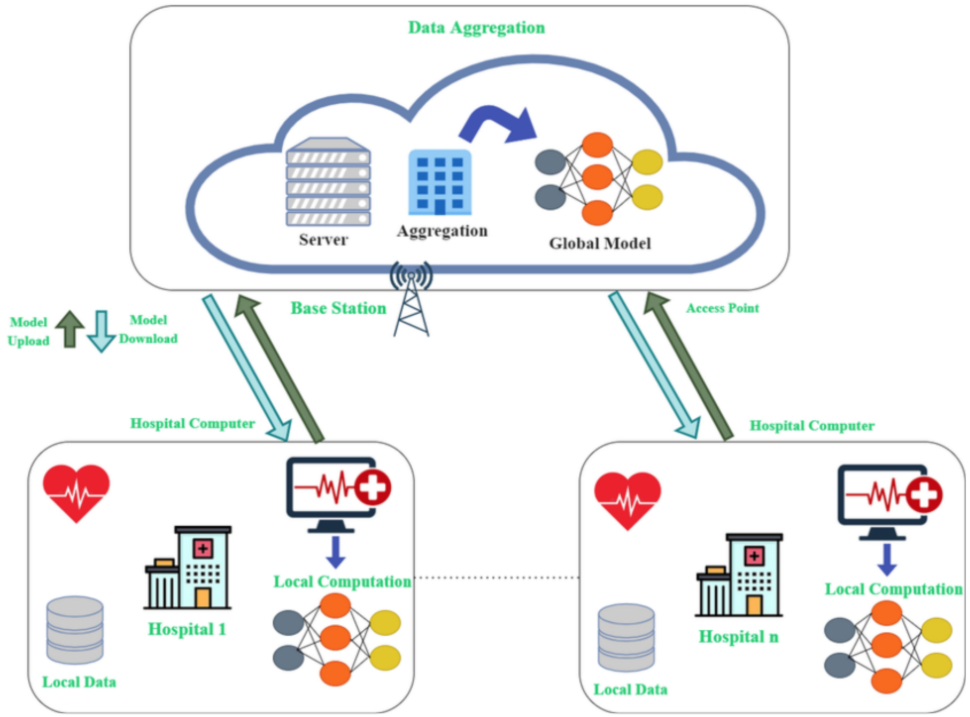

Figure 2.2: Federated learning workflow with local model training, secure aggregation, and global model distribution.

architectures [11], and client-specific training techniques [22] [5]. Ensemble and semi-supervised approaches have also shown promise in managing non-IID settings, as seen in studies like [6, 20]. For instance, FedStack [20] used a stacking ensemble for personalized activity monitoring, while Liu et al. [27] tackled inconsistent label distributions in multi-organ segmentation using federated ensembles. Collectively, these efforts highlight the potential of ensemble learning—especially model stacking—as a practical solution to heterogeneity and label misalignment in FL.

2.5 Ensemble Learning and Model Stacking

Ensemble learning techniques like bagging, boosting, and stacking are well-known for enhancing model generalization and robustness [**?** 17]. Among these, stacking stands out for its ability to combine diverse model architectures, making it particularly valuable in settings like network intrusion detection[29**?**] and time-series forecasting [17]. In federated learning scenarios, stacking is especially effective because client models often differ in structure due to varying computational capacities or domain-specific needs [8]. By training a meta-model on the predictions from these heterogeneous clients, stacking offers a practical way to integrate their outputs—an approach particularly beneficial for complex tasks like multi-class heart sound classification, where clients may have incomplete label sets [**?**].

2.6 Summary and Gaps

While previous research has applied federated learning to medical diagnostics and explored ensemble strategies for model fusion, few have tackled the combined challenges of label misalignment, client model heterogeneity, and multimodal auscultation data. Our proposed framework, **MStacking**, addresses this gap by using a stacking ensemble approach within a federated learning setup. This allows integration of insights from clients with varying model types and incomplete label spaces—conditions common in real-world healthcare. By doing so, MStacking offers a practical and scalable path toward deploying AI-assisted auscultation across diverse, decentralized medical institutions.

Chapter 3

Goals

3.1 Goals

The primary goal of this project is to develop **MStacking**, a heterogeneous federated learning framework tailored for classifying abnormal heart sounds in a privacy-preserving manner. The system is designed to function effectively across decentralized healthcare settings, where institutions vary in both computational capabilities and the types of heart sound data they can access. By using a stacking ensemble strategy, MStacking enables collaborative learning without requiring sensitive patient data to be shared, aiming to deliver a more accurate, robust, and adaptable AI-driven auscultation tool.

3.2 Objectives

This project is guided by the following objectives:

- **Support model heterogeneity:** Build a federated system that allows each client to choose its own model architecture based on local data volume and hardware constraints.
- **Stacking ensemble integration:** Use meta-learning to combine predictions from various client models into a unified global classifier, even when local label distributions differ.
- **Leverage multimodal data:** Enhance classification by incorporating both 1D acoustic features and 2D spectrogram representations of phonocardiogram (PCG) signals.
- **Preserve data privacy:** Avoid direct data sharing by transmitting only high-level model outputs and statistical metadata between clients and the central server.

- **Benchmark performance:** Evaluate MStacking against centralized and traditional federated learning methods using public datasets, with a focus on accuracy, generalization, and resilience to real-world data challenges.

3.3 Scope and Limitations

Scope

This project centers on building a proof-of-concept federated learning framework—MStacking—for classifying abnormal heart sounds using heterogeneous client models and misaligned class labels. The system is tested in a simulated federated environment using public datasets like the PhysioNet/CinC 2016 Challenge and other open-access PCG sources. It supports multimodal input types, such as 1D audio signals and 2D spectrograms, and accommodates various model architectures (e.g., CNNs, random forests, FNNs) chosen by individual clients. Development is based on open-source libraries (e.g., PyTorch, Flower/FedML), with training designed to mimic privacy-preserving, decentralized healthcare scenarios.

Limitations

- The prototype is simulation-only; it won't be deployed in real hospitals or connected to physical stethoscope devices.
- Experiments use static, publicly available datasets and do not involve real-time or clinical data.
- Due to time and resource constraints, extensive hyperparameter tuning and in-depth ablation analysis are outside the project scope.
- The focus is solely on classification tasks—regression or diagnostic recommendation features are not included.
- Evaluation relies on offline performance metrics (e.g., accuracy, F1-score), without clinical trials or human-in-the-loop validation.

Chapter 4

Methodology

4.1 Overview of the Proposed System

This project introduces **MStacking**, a novel heterogeneous federated learning (FL) framework for classifying abnormal heart sounds from decentralized, privacy-sensitive data. The key challenge addressed is that hospitals often use different machine learning models suited to their data and computing capabilities, and typically only have access to a subset of heart sound classes—leading to label misalignment.

To solve this, MStacking employs a stacking ensemble strategy. Each hospital (client) trains a binary classifier using its local data—usually covering one abnormal class and the normal class—and sends prediction scores and density estimations (not raw data) to a central server. There, a meta-learner integrates these diverse outputs into a unified, global multi-class model, enabling collaborative learning without compromising patient privacy.

A summary of prominent studies utilizing the PhysioNet/CinC 2016 dataset is presented in Table 4.1, highlighting the evolution of machine learning applications in heart sound classification.

4.2 Local Model Architecture

4.2.1 Local Model Architecture

In the MStacking setup, each hospital independently chooses and trains a binary classifier tailored to its dataset size, signal type, and computational capacity (see Figure 4.1). Three model types are supported:

- **Random Forest (RF)**: Ideal for low-resource clients with structured feature data. RF ensembles decision trees built on bootstrap samples and uses the Gini index for splits.

Table 4.1: Summary of Studies Using the PhysioNet/CinC Challenge 2016

Study	**Year**	**Task**	**Model(s) Used**	**Contribution**
[3]	2016	Normal/Abnormal Classification	Ensemble SVMs	Introduced the PhysioNet Challenge dataset
[10]	2016	Classifier Evaluation	Open baseline models	Created open-access benchmark for PCG classifiers
[24]	2023	Explainability in PCG Detection	SHAP + FNN	Applied feature attribution methods to enhance transparency
[16]	2022	FL for Binary Heart Sound Detection	CNN (FedAvg)	Early federated prototype for heart sound classification
Qiu et al. (This Work)	2024	Multi-Class FL for PCG Analysis	RF, FNN, CNN (Hetero)	Introduces MStacking with misaligned labels and heterogeneity

- **Feedforward Neural Network (FNN)**: Suitable for clients with moderate resources and low-dimensional data. FNNs learn non-linear patterns through layers of neurons trained via backpropagation.
- **Convolutional Neural Network (CNN)**: Best for institutions handling spectrograms (e.g., from Continuous Wavelet Transform). CNNs use convolutional layers, global adaptive pooling (GAP), and fully connected layers to classify input efficiently.

All models are trained using cross-entropy loss and the Adam optimizer. Clients follow a "star-structured" label setup—each has the normal class ($y = 0$) and one abnormal class ($y = m$), mimicking real-world healthcare data limitations.

4.2.2 Stacking Ensemble Strategy

Conventional federated learning (FL) methods like FedAvg assume that all clients use the same model architecture and share the same label space. However, this is rarely the case in real-world healthcare environments. To overcome these limitations, the proposed MStacking framework adopts a stacking ensemble strategy, which aggregates predictions instead of model parameters—making it flexible to both label and architectural heterogeneity.

The ensemble works in two levels (see Figure 4.2):

- **Level-0 (Base Learners)**: Each client trains a binary classifier—such as a Random Forest (RF), Feedforward Neural Network (FNN), or Convolutional Neural

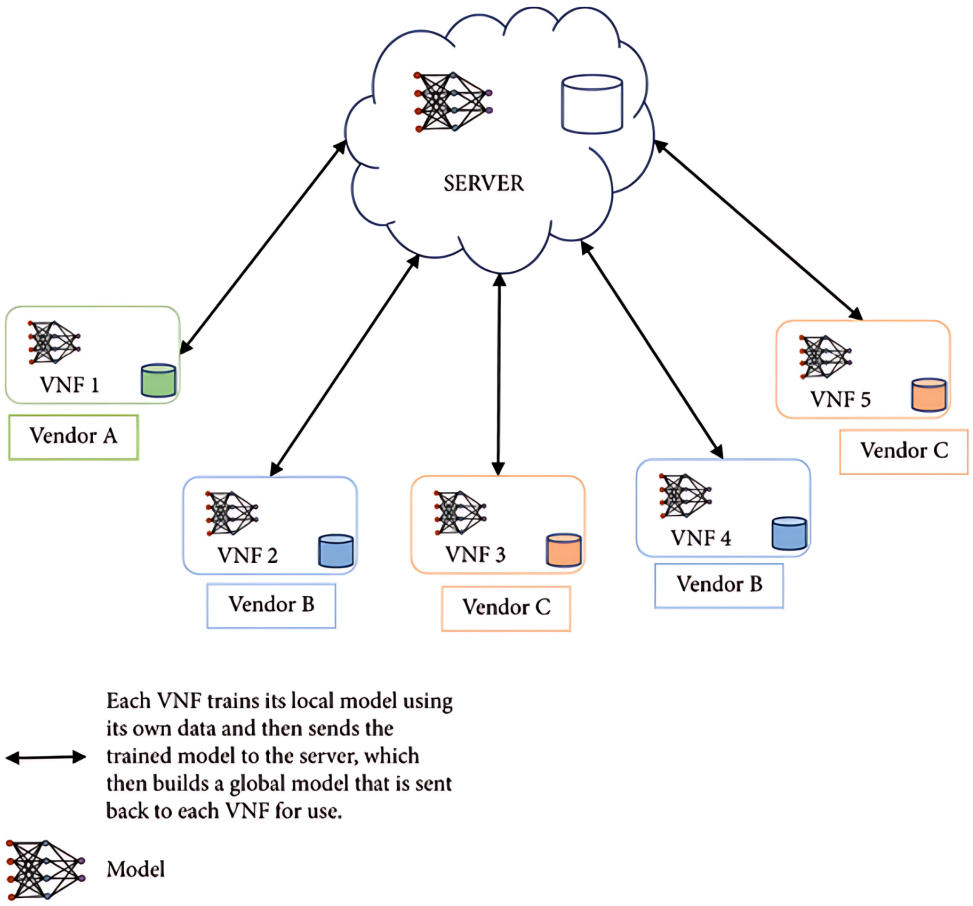

Figure 4.1: Figure 3 illustrates the architecture diversity across clients, showing how each local model (Virtual Network Function) feeds into the broader federated learning system while maintaining autonomy over its training process.

Network (CNN)—on its local dataset. These models output probabilistic predictions for the normal class and one abnormal class.

- **Level-1 (Meta-Learner)**: A central model on the server aggregates predictions and density estimates (from Gaussian Mixture Models) sent by each client. It learns how to combine these varied outputs into a global multi-class classifier.

To enhance the quality of aggregation, each client also estimates the probability distribution of its input features and flags the classes it can recognize. These values are used to compute ensemble weights, which determine how much each client contributes to the final model.

As shown in Figure 4.2, this two-level stacking framework enables robust, privacy-preserving learning—even when clients have partial data or different computational capabilities.

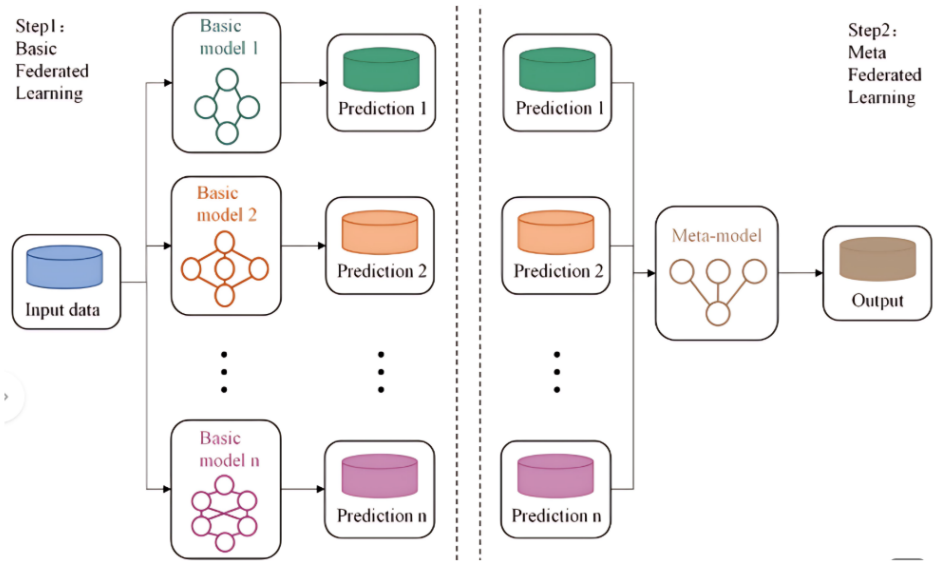

Figure 4.2: MStacking Two-Level Ensemble Architecture: Client-level models generate predictions and densities, aggregated by a central meta-learner into a global classifier.

4.2.3 Federated Aggregation Process

Unlike traditional FL methods like FedAvg that rely on aggregating model parameters from identical architectures, MStacking takes a different route. It performs meta-level fusion, where predictions and feature distributions from each client's model are combined at the server level.

Each participating client sends the following metadata to the central server:

- Prediction probabilities from its trained binary classifier (e.g., RF, FNN, CNN).
- Density estimates of its local input data, modeled using Gaussian Mixture Models (GMMs).
- Class availability indicators that specify which classes the client has in its local dataset.

Using this information, the server constructs a global multi-class classifier $c_m(\theta)$ through a weighted integration of the client models. The weight $\omega_{j,m}$ assigned to client j for class m depends on:

- Whether the class m is present in client j's data.
- The estimated quality of the local model, which factors in data distribution f_{x_j} and the proportion of training samples from that client.

This aggregation strategy enables:

- **Missing label handling**: Clients that lack some class labels can still contribute to the global model through inferred relationships.

- **Model flexibility**: Clients can use any architecture; uniformity is not required.
- **Privacy preservation**: Only high-level summaries (e.g., predictions and densities) are shared—never raw data or internal parameters.

4.2.4 Evaluation Strategy

To assess the performance and applicability of the MStacking framework, a structured evaluation plan will be followed using real-world and simulated data.

Datasets:

- *PhysioNet/CinC Challenge 2016* [3]: A large, annotated dataset of heart sounds, covering both normal and abnormal cases.
- *Additional open-access PCG datasets* [10]: Used to simulate client-specific data distributions and class imbalances.

Simulation Setup:

- Experiments will use Flower or FedML to simulate a federated environment.
- Clients will receive non-IID partitions, each containing the normal class and one abnormal class to replicate real-world heterogeneity.

Metrics for Evaluation:

- Accuracy, Precision, Recall, F1-score (macro & class-wise)
- AUROC for threshold robustness
- Confusion matrices to analyze prediction patterns
- Training time and communication cost for efficiency
- SHAP or similar methods for interpretability (if feasible)

Baselines for Comparison:

- Centralized CNN trained on full data
- Traditional FedAvg with homogeneous models and labels
- Personalized FL with clustering or fine-tuning

4.2.5 Implementation Details

Development Environment:

- *Language*: Python 3.x
- *Federated Simulation*: Flower or FedML
- *Deep Learning*: PyTorch
- *Signal Processing*: OpenSMILE, LibROSA, and Continuous Wavelet Transform (CWT)

Model Training:

- *Optimizer*: Adam
- *Monitoring*: Early stopping and validation tracking
- *Architectures*: RF, FNN, CNN—based on local data modalities

Execution Platform:

- *Hardware*: Lab PCs or cloud-based GPUs (e.g., Google Colab, Kaggle)
- *Version Control*: Git and GitHub
- *Visualization & Reporting*: Matplotlib, Seaborn, and LaTeX (Overleaf)

This implementation plan ensures that MStacking remains reproducible, scalable, and practical within an academic setting.

Chapter 5

Evaluation

5.1 Evaluation Objectives

The evaluation phase aims to rigorously assess how well the proposed MStacking framework performs under realistic, decentralized healthcare scenarios. Key goals include:

- Measuring the classification accuracy of the global multi-class model built via stacking.
- Comparing MStacking's performance to traditional centralized learning and standard FL methods like FedAvg.
- Testing robustness against client heterogeneity, label misalignment, and imbalanced data distributions.
- Assessing system efficiency by tracking communication overhead and computational cost.
- Exploring the interpretability of predictions using feature attribution tools such as SHAP.

5.2 Evaluation Metrics

To ensure a comprehensive analysis, the following metrics will be employed:

- **Accuracy**: Overall rate of correct predictions.
- **Precision, Recall, F1-Score**: Evaluated per class to reflect detection sensitivity and specificity. Macro-averaged versions will be emphasized to address class imbalance.

- **AUROC**: Measures the model's ability to distinguish between classes under varying thresholds.
- **Confusion Matrix**: Highlights where and how often misclassifications occur.
- **Model Communication Cost**: Total size of metadata exchanged between clients and the server.
- **Training Time per Round**: Gauges the runtime efficiency of global model updates.
- **Model Interpretability**: Investigated via SHAP or similar techniques to understand decision rationale.

5.3 Experimental Setup

Datasets:

- *PhysioNet/CinC Challenge 2016* [3]: Main benchmark dataset, with 3,000+ annotated heart sound recordings.
- *Additional PCG datasets* [10]: Used to simulate diverse client data conditions and class variety.

Client Configuration:

- 5 to 10 virtual clients will be created.
- Each client will have a non-IID data partition containing the normal class and a distinct abnormal class (star-structure).
- Clients will use varying model types (e.g., RF, FNN, CNN) to simulate hardware and algorithmic diversity.

Baselines for Comparison:

- Centralized CNN trained on full data.
- Traditional FL (FedAvg) with uniform models and aligned labels.
- Personalized FL using clustered or fine-tuned models.
- Ablation Study to evaluate the role of metadata: predictions only vs. predictions + densities.

5.4 Evaluation Procedure

1. Training Phase:

- Clients train local binary classifiers.
- Output includes prediction probabilities and Gaussian Mixture-based density estimates.
- These are sent to the server and integrated into a global model using stacking.

2. Validation Phase:

- A held-out multi-class dataset is used to evaluate the global model.
- Metrics are updated after each training round.

3. Stress Testing:

- Client dropouts to test resilience.
- Label noise to assess robustness.
- Imbalanced class samples to evaluate fairness.

4. Result Analysis:

- Results will be visualized using ROC curves, training/validation loss plots, and confusion matrices.
- Comparative tables will summarize performance against baselines.
- A detailed discussion will highlight the strengths and trade-offs of MStacking.

Table 5.1: Performance Comparison with Existing Methods

Method	**Model Type**	**Setting**	**Data Mode**	**Accuracy**	**UF1 Score**	**UAR Score**	**Notes**
Centralized CNN	CNN	Centralized	Balanced	Moderate	Moderate	Moderate	Traditional learning with full label sharing
Fed-BIDS [**?**]	RF + NN Meta	Federated	Balanced	High	Moderate	Moderate	Blends metadata for intrusion detection
Fed-MStacking (Homog.)	RF / FNN / CNN	Federated	Balanced	Higher	Higher	Higher	Uses identical architectures across clients
Fed-MStacking (Hetero)	RF + FNN + CNN	Federated	Balanced	Highest	Highest	Highest	Supports diverse models & misaligned class labels (This work)

Chapter 6

Project Plan

6.1 Work Breakdown Structure (WBS)

To ensure smooth progress and timely delivery, the development of the MStacking framework is organized into seven structured phases. Each phase includes specific tasks that align with the overall project objectives:

- **Timeline:** Each phase is allocated 2-3 weeks with overlapping activities where possible
- **Dependencies:** Sequential flow with iterative refinement loops between phases
- **Deliverables:** Technical reports, prototype implementations, and evaluation results at each phase

6.2 Timeline and Gantt Chart

The project will be completed over a four-month period, with major deliverables planned at each stage. Below is the proposed timeline in a Gantt chart format:

6.3 Resources and Tools

To develop, test, and evaluate the MStacking framework efficiently, the following tools and platforms will be utilized:

- **Software Tools:** Python 3.x, PyTorch, OpenSMILE, LibROSA, Flower, FedML, and Jupyter Notebooks.
- **Hardware:** University-provided GPU-enabled machines and cloud services like Google Colab and Kaggle for additional compute support.
- **Version Control:** Git and GitHub for collaborative development and version tracking.

Table 6.1: Project Workflow Phases

Phase	Description
1. Research & Literature Review	Study recent work in federated learning, stacking, heart sound classification, and AI in healthcare.
2. System Design	• Define the MStacking architecture. • Plan client-server interactions and simulation topology. • Outline metadata and communication protocol.
3. Data Preparation	• Preprocess PCG signals using MFCC and CWT. • Partition datasets into star-structured splits for FL simulation. • Validate multimodal input formats.
4. Local Model Implementation	• Implement and train Random Forest, FNN, and CNN models. • Optimize each for local client data characteristics.
5. Stacking & Federated Aggregation	• Design metadata schema (predictions, densities). • Build meta-learner. • Integrate client outputs into a unified global model.
6. Evaluation & Validation	• Run experiments under non-IID, imbalanced, and noisy data scenarios. • Compare with centralized and traditional FL models. • Use metrics like accuracy, F1-score, AUROC, SHAP.
7. Documentation & Presentation	• Prepare final report, figures, and codebase documentation. • Present results and insights clearly for academic submission.

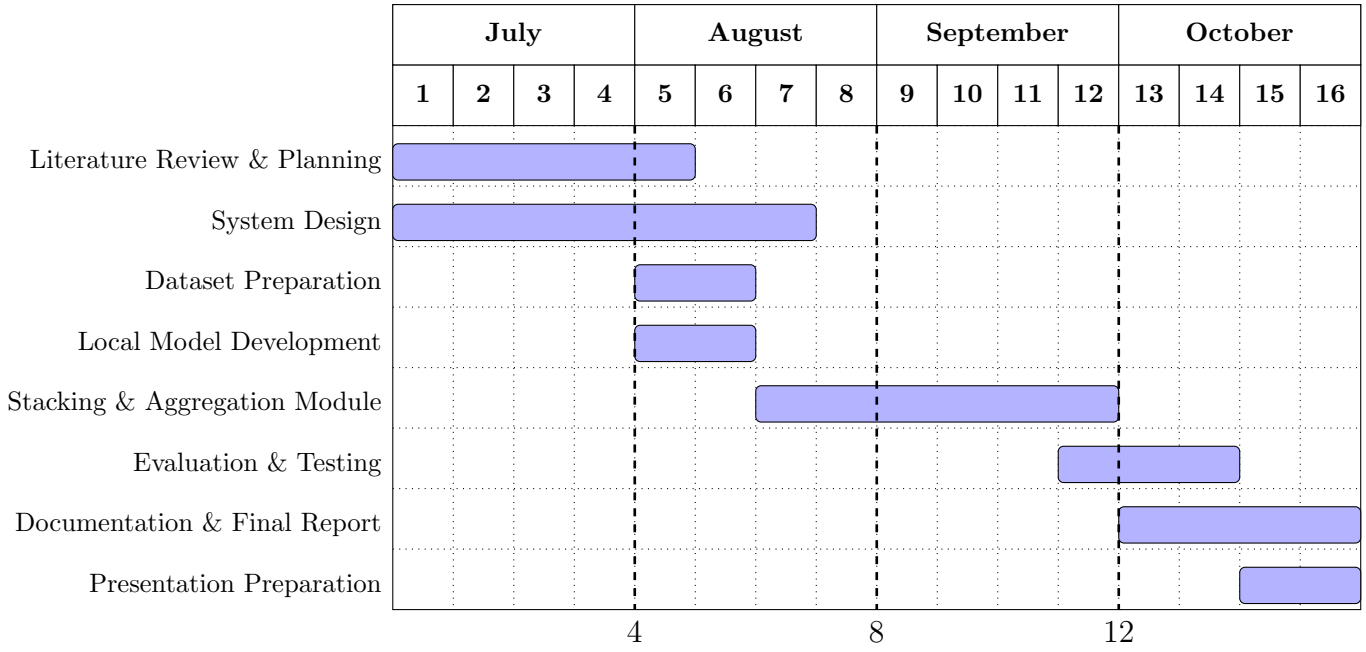

Figure 6.1: Project Timeline Gantt Chart

- **Documentation** Overleaf (LaTeX) for professional academic writing and formatting.
- **Visualization Monitoring:** Matplotlib, Seaborn, and TensorBoard for experiment tracking and performance visualization.

6.4 Risk Management

Identified risks and corresponding mitigation strategies are outlined in the table below:

Table 6.2: Risk and Mitigation Strategy

Risk	**Mitigation Strategy**
1. Insufficient compute resources	Leverage cloud-based platforms (Google Colab, Kaggle) for GPU access.
2. Time constraints during academic semester	Apply agile development cycles; prioritize the core MStacking framework before add-ons.
3. Lack of access to real-time medical data	Use realistic, publicly available datasets like PhysioNet to simulate clinical settings.
4. Debugging federated heterogeneity issues	Conduct isolated tests for each local model type (RF, FNN, CNN) before integration.

Bibliography

[1] M. Alshamrani. Iot and artificial intelligence implementations for remote healthcare monitoring systems: A survey. *J. King Saud Univ.- Comput. Inf. Sci.*, 34(8):4687–4701, 2022.

[2] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi. Federated learning of predictive models from federated electronic health records. *Int. J. Med. Inform.*, 112:59–67, 2018.

[3] G. D. Clifford et al. Classification of normal/abnormal heart sound recordings: The physionet/computing in cardiology challenge 2016. In *Proc. Comput. Cardiol.*, pages 609–612, 2016.

[4] L. Huang, A. L. Shea, H. Qian, A. Masurkar, H. Deng, and D. Liu. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. *J. Biomed. Inform.*, 99, 2019. Art. no. 103291.

[5] Wenke Huang, Mao Ye, and Bo Du. Learn from others and be yourself in heterogeneous federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10143–10153. IEEE, June 2022.

[6] Sohei Itahara, Takayuki Nishio, Yusuke Koda, Masahiro Morikura, and Koji Yamamoto. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data. *IEEE Transactions on Wireless Communications*, 22(1):152–165, 2023.

[7] S. Li, F. Li, S. Tang, and W. Xiong. A review of computer-aided heart sound detection techniques. *Biomed Res. Int.*, 2020:5846191, 2020.

[8] T. Lin, L. Kong, S. U. Stich, and M. Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 2351–2363, 2020.

[9] Chengyu Liu, David Springer, Qiao Li, Benjamin Moody, Ricardo Abad Juan, Francisco J Chorro, Francisco Castells, José Millet Roig, Ikaro Silva, Alistair EW Johnson, et al. An open access database for the evaluation of heart sound algorithms. *Physiological Measurement*, 37(12):2181, 2016.

[10] Chengyu Liu, David B Springer, Qiao Li, Benjamin Moody, Rafael Juan, Francisco J Chorro, Francisco Castells, Jose M Roig, Ikaro Silva, Alistair E W Johnson, and Gari D Clifford. An open access database for the evaluation of heart sound algorithms. *Physiological Measurement*, 37(12):2181–2213, 2016.

[11] J. Mori, T. Yoshiyama, R. Furukawa, and I. Teranishi. Personalized federated learning with multi-branch architecture. In *Proc. IEEE Int. Joint Conf. Neural Netw.*, pages 1–8, 2023.

[12] D. C. Nguyen et al. Federated learning for smart healthcare: A survey. *ACM Comput. Surv.*, 55(3):1–37, 2022.

[13] K. Qian, T. Koike, K. Yoshiuchi, B. W. Schuller, and Y. Yamamoto. Can appliances understand the behavior of elderly via machine learning? a feasibility study. *IEEE Internet Things J.*, 8(10):8343–8355, 2021.

[14] K. Qian, Z. Ren, F. Dong, W.-H. Lai, B. W. Schuller, and Y. Yamamoto. Deep wavelets for heart sound classification. In *2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 1–2. IEEE, 2019.

[15] K. Qian, Z. Zhang, Y. Yamamoto, and B. W. Schuller. Artificial intelligence internet of things for the elderly. *IEEE Signal Process. Mag.*, 38(4):78–88, 2021.

[16] Siyu Qiu, Yilin Wang, Chengyu Liu, and Gari D Clifford. Federated learning for heart sound classification. *IEEE Transactions on Biomedical Engineering*, 69(8):2562–2572, 2022.

[17] Mateus Henrique Dal Molin Ribeiro and Leandro dos Santos Coelho. Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series. *Applied Soft Computing*, 86:105837, 2020.

[18] G. A. Roth et al. Global burden of cardiovascular diseases and risk factors, 1990–2019: Update from the gbd 2019 study. *J. Amer. College Cardiol.*, 76(25):2982–3021, 2020.

[19] J. K. Roy, T. S. Roy, and S. C. Mukhopadhyay. Heart sound: Detection and analytical approach towards diseases. In *Modern Sensing Technologies*, pages 103–145. Springer, 2019.

[20] Thanveer Shaik, Xiaohui Tao, Niall Higgins, Raj Gururajan, Xujuan Zhou, and U Rajendra Acharya. Fedstack: Personalized activity monitoring using stacked federated learning. *Knowledge-Based Systems*, 257:109929, 2022.

[21] S. Swarup and A. N. Makaryus. Digital stethoscope: Technology update. *Medical Devices: Evidence and Research*, 11:29–36, 2018.

[22] A. Z. Tan, H. Yu, L. Cui, and Q. Yang. Towards personalized federated learning. *IEEE Trans. Neural Netw. Learn. Syst.*, 34(12):9587–9603, Dec. 2023.

[23] E. Tramel. Federated learning: Rewards & challenges of distributed private ml. Accessed May, 2019. Art. no. 2019.

[24] Li Wang, Hao Zhang, Yutong Chen, and Gari D Clifford. Explainable ai for phonocardiogram classification. *IEEE Journal of Biomedical and Health Informatics*, 27(4):1872–1881, 2023.

[25] S. Winther et al. Advanced heart sound analysis as a new prognostic marker in stable coronary artery disease. *European Heart Journal-Digital Health*, 2(2):279–289, 2021.

[26] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang. Federated learning for healthcare informatics. *J. Healthcare Inform. Res.*, 5(1):1–19, 2021.

[27] Xiaoxiao Xu, Howard H Deng, Jaime Gateno, and Pingkun Yan. Federated multi-organ segmentation with inconsistent labels. *IEEE Transactions on Medical Imaging*, 42(10):2948–2960, 2023.

[28] J. H. Yoo et al. Personalized federated learning with clustering: Non-iid heart rate variability data application. In *Proc. IEEE Int. Conf. Inf. Commun. Technol. Convergence*, pages 1046–1051, 2021.

[29] Hongliang Zhang, Jianguo Li, Xiaoming Liu, and Chen Dong. Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection. *Future Generation Computer Systems*, 122:130–143, 2021.

[30] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr. Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet Things Mag.*, 5(1):24–29, 2022.