Federated Learning-Based Edge Device Framework for Heart Anomaly Detection in IoT Systems

Alen Scaria

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Federated Learning-Based Edge Device Framework for Heart Anomaly Detection in IoT Systems

A report submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Alen Scaria

(Roll Number: 121CS0237)

based on research carried out

under the supervision of

Prof. Judhistir Mahapatro

March, 2025

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Prof. Judhistir Mahapatro

March 06, 2025

Supervisor's Certificate

This is to certify that the work presented in the project report entitled *Federated Learning-Based Edge Device Framework for Heart Anomaly Detection in IoT Systems* submitted by *Alen Scaria*, Roll Number 121CS0237, is a record of original research carried out by him under my supervision and guidance in partial fulfillment of the requirements of the degree of *Bachelor of Technology* in *Computer Science and Engineering*. Neither this project report nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

Judhistir Mahapatro

Abstract

Traditional heart monitoring systems face challenges related to centralized data processing, including concerns about data privacy, communication overhead, and real-time abnormality detection. This project focuses on heart abnormality detection using **Federated Learning**, leveraging IoT-based health monitoring. It explores a decentralized approach to heart abnormality detection by implementing a federated learning-based model. The system utilizes multiple sensor nodes (clients) in conjunction with a Raspberry Pi as server to ensure secure, privacy-preserving data analysis. By using federated learning, the model is trained across decentralized devices, reducing the need for centralized data storage. This ensures real-time, privacy-preserving heart monitoring with improved anomaly detection. The system provides a low-power, scalable solution for intelligent healthcare, with potential future improvements in sensor integration and advanced anomaly detection algorithms.

Contents

Chapter 1

Introduction

The healthcare sector is witnessing a significant transformation with the integration of Internet of Things (IoT) and artificial intelligence (AI) technologies. Wearable devices and IoT-based systems are increasingly used for remote monitoring of patients, especially those with heart conditions, allowing for early detection and timely intervention. [1]

However, traditional centralized data processing systems pose challenges such as data privacy concerns, high communication overhead, and difficulties in real-time anomaly detection. The shift towards decentralized monitoring is becoming more practical due to advancements in remote monitoring techniques, which reduce the need for on-site visits by capturing vital signs remotely.[2]

1.1 Problem Definition

Traditional heart monitoring systems face several challenges.They typically rely on centralized data processing, which raises significant concerns about data privacy, communication overhead, and the ability to detect abnormalities in real-time. Single-device monitoring often provides limited insights due to insufficient local data, and real-time queries are challenging in areas with limited communication bandwidth. Remote monitoring techniques, while beneficial, require robust solutions to address these issues effectively.

1.2 Objectives

This project aims to implement a federated learning-based approach for healthcare monitoring, focusing on decentralized heart abnormality detection using multiple sensor nodes (clients) and a Raspberry Pi (server). By utilizing federated learning, the system maintains data privacy while collaboratively enhancing model accuracy across multiple edge devices, enabling real-time queries in areas with limited communication bandwidth.Further it aims at evaluating the model's accuracy, reliability and efficiency in detecting heart abnormalities.

Chapter 2

Project Work

This section provides an overview of the current progress in understanding the topic, along with the proposed approach for addressing the problem.

2.1 Literature Review

Several studies have explored the use of federated learning in health care monitoring and IoT based systems.

In 2023, a paper by Srinivasa Raju Rudraraju et al.[3] explored the application of federated learning (FL) to process heterogeneous sensor data in a fog computing-based smart home environment. Their system utilized edge nodes to train machine learning (ML) models using FL, eliminating the need to send data to a central server. Instead, the results from training on multiple edge nodes were aggregated at a central node to create the final ML model. This approach is ideal for environments with limited communication bandwidth or where data privacy is a concern.

This work puts forward a promising approach for using FL in remote healthcare monitoring systems, leveraging IoT systems to improve data privacy and enable real-time abnormality detection.

2.2 Federated Learning

Federated Learning (FL) is an innovative machine learning approach that enables decentralized training of models across multiple devices or systems while preserving data privacy. Below is a brief overview of its key advantages and features:

- FL eliminates the need to centralize data, reducing costs associated with data aggregation and storage. This makes it particularly effective for organizations handling large-scale, sensitive datasets

- FL has been shown to perform well even when client devices have uneven amounts of data or when the data is non-independent and identically distributed.
- FL ensures that raw data remains on local devices, significantly reducing the risk of data breaches and ensuring compliance with privacy regulations
- This framework is transforming industries by enabling scalable, privacy-preserving AI solutions while maintaining high levels of model performance and adaptability.

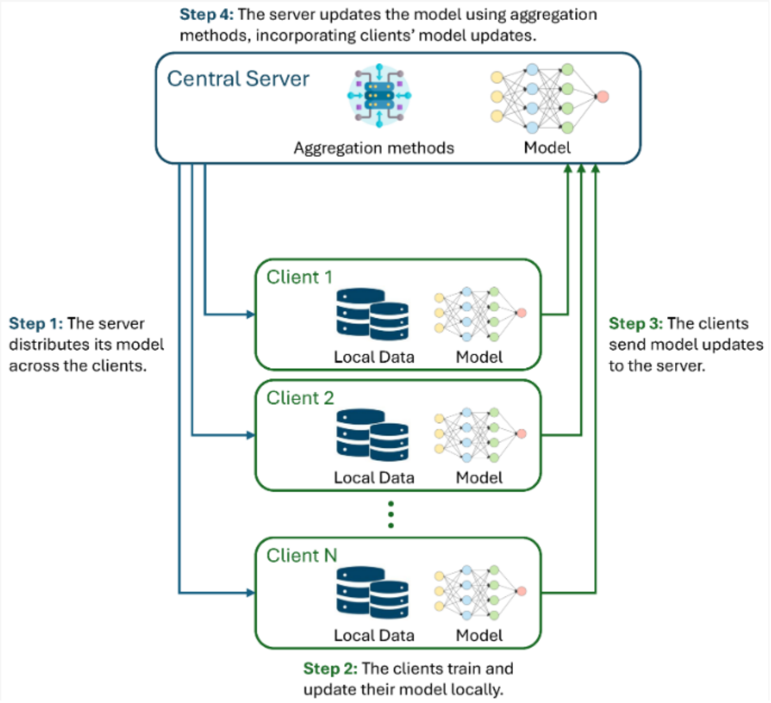

Step 4: The server updates the model using aggregation methods, incorporating clients' model updates.
Central Server
Aggregation methods
Model
Step 1: The server distributes its model across the clients.
Client 1
Local Data
Model
Client 2
Local Data
Model
Client N
Local Data
Model
Step 3: The clients send model updates to the server.
Step 2: The clients train and update their model locally.

Figure 2.1: Architecture of Federated Learning

2.2.1 Heart Rate Abnormality Detection

Heart rate abnormalities, including arhythmias and variations in heart rate variability (HRV), pose significant health risks and can indicate underlying cardiovascular conditions. Recent advancements in machine learning (ML) have enabled the development of robust techniques for detecting these abnormalities. Various ML models, including Support Vector Machines (SVM), Random Forests, and Neural Networks, have been employed to analyze physiological signals such as electrocardiograms (ECGs) and photoplethysmography (PPG). These models are trained on features extracted from time-domain, frequency-domain, and non-linear analyses of heart rate data. Research indicates that deep learning approaches, particularly Convolutional Neural Networks (CNNs), demonstrate superior performance in automated detection tasks (Ravi et al., 2017; Yildirim et al., 2020).[4] [5] Additionally, ensemble methods combining multiple algorithms have shown promising results in enhancing predictive accuracy (Bourke et al., 2019).[6] The integration of wearable technology further facilitates continuous monitoring, allowing for real-time detection of abnormalities and timely medical intervention. This review summarizes the current landscape of ML techniques for heart rate abnormality detection, highlighting their potential and the challenges faced in clinical implementation.

2.3 Hardware Description

Our project utilizes a combination of IoT hardware components to collect real-time heart rate data and implement federated learning for anomaly detection. Below is a detailed description of each component

2.3.1 Raspberry Pi 3 Model B

The Raspberry Pi 3 Model B is a compact, low-power single-board computer powered by a 1.2 GHz quad-core ARM Cortex-A53 processor with 1GB RAM. It features Wi-Fi, Bluetooth, Ethernet, USB ports, and 40 GPIO pins, making it ideal for IoT and edge computing applications. In this project, it acts as the edge device that receives BPM data from Arduino via serial communication, processes and stores the data, and trains a federated learning model for heart abnormality detection while ensuring data privacy.

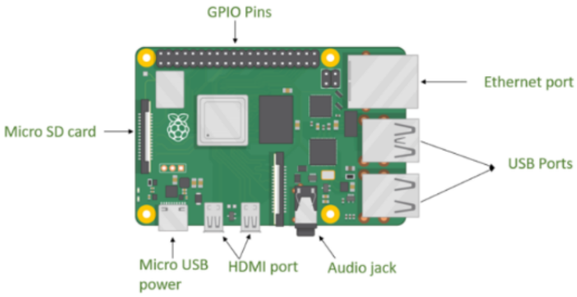


Figure 2.2: Raspberry Pi 3

2.3.2 Ardinuo Uno

The Arduino Uno is a microcontroller board based on the ATmega328P chip, featuring 14 digital I/O pins, 6 analog inputs, and a 16 MHz clock speed. It is widely used for embedded systems and IoT applications due to its simplicity and efficiency. In this project, the Arduino Uno acts as a sensor node, collecting real-time BPM data from the heart rate sensor and transmitting it to the Raspberry Pi via serial communication for further processing and federated learning.

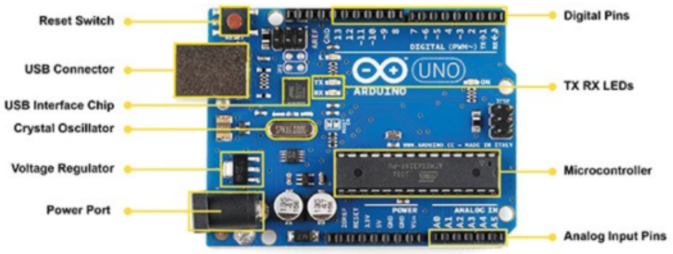


Figure 2.3: Ardinuo Uno

2.3.3 Pulse Sensor

The Pulse Sensor is an optical heart rate sensor that measures BPM (Beats Per Minute) by detecting blood flow changes using a built-in LED and photodiode. It is compact, energy-efficient, and designed for easy integration with Arduino and Raspberry Pi via analog input. In this project, it collects real-time heart rate data and transmits it to the Arduino for further processing and anomaly detection.

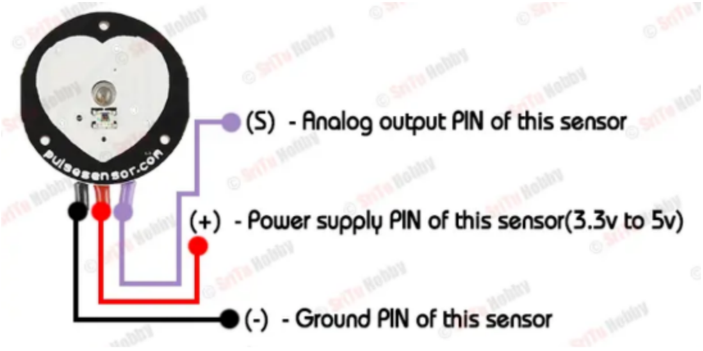

(S) - Analog output PIN of this sensor
(+) - Power supply PIN of this sensor(3.3v to 5v)
(-) - Ground PIN of this sensor
pulsesensor.com

Figure 2.4: Sensor

2.4 Environment Setup

To implement our heart abnormality detection system using federated learning, we set up both the hardware and software environments as follows:

2.4.1 Hardware setup

The Arduino Uno is connected to a Pulse Sensor, which continuously reads BPM (Beats Per Minute) data through its analog input. This data is transmitted to the Raspberry Pi 3 Model B, which serves as the primary computing unit for data processing and federated learning. Serial communication between the Arduino and Raspberry Pi is established using UART, enabling real-time data transfer. The Raspberry Pi is powered through a 5V/2.5A micro-USB adapter, while the Arduino receives power via a USB connection to the Raspberry Pi or an external 9V battery, if necessary.

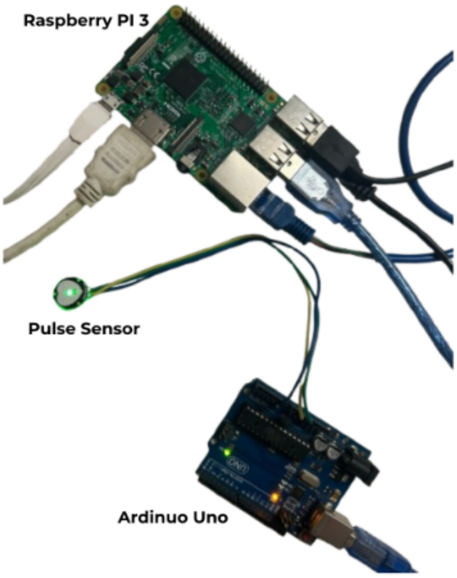
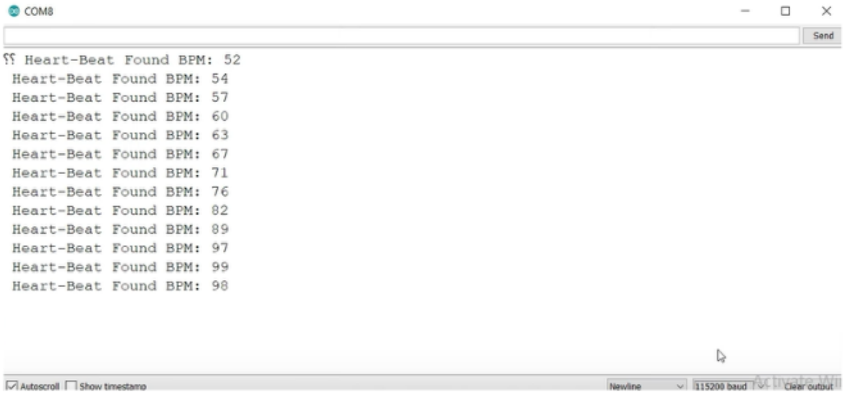


Figure 2.5: Setup

2.4.2 Software setup

For the software setup, Raspberry Pi OS (32-bit) is installed on the Raspberry Pi to provide a stable environment for processing and computation. The Arduino IDE is used to program the Arduino Uno boards and read BPM data from the pulse sensor. On the Raspberry Pi, Python 3 is installed along with necessary libraries for data handling, Serial for communication with Arduino, and Matplotlib for visualizing BPM trends.

The serial communication between two Arduino Uno boards and the Raspberry Pi allows for synchronized data collection, ensuring reliable and accurate readings. This environment setup enables the system to function efficiently as a real-time, low-power IoT-based anomaly detection system while maintaining data privacy through federated learning.



```
⸮⸮ Heart-Beat Found BPM: 52
 Heart-Beat Found BPM: 54
 Heart-Beat Found BPM: 57
 Heart-Beat Found BPM: 60
 Heart-Beat Found BPM: 63
 Heart-Beat Found BPM: 67
 Heart-Beat Found BPM: 71
 Heart-Beat Found BPM: 76
 Heart-Beat Found BPM: 82
 Heart-Beat Found BPM: 89
 Heart-Beat Found BPM: 97
 Heart-Beat Found BPM: 99
 Heart-Beat Found BPM: 98
```

Figure 2.6: BPM

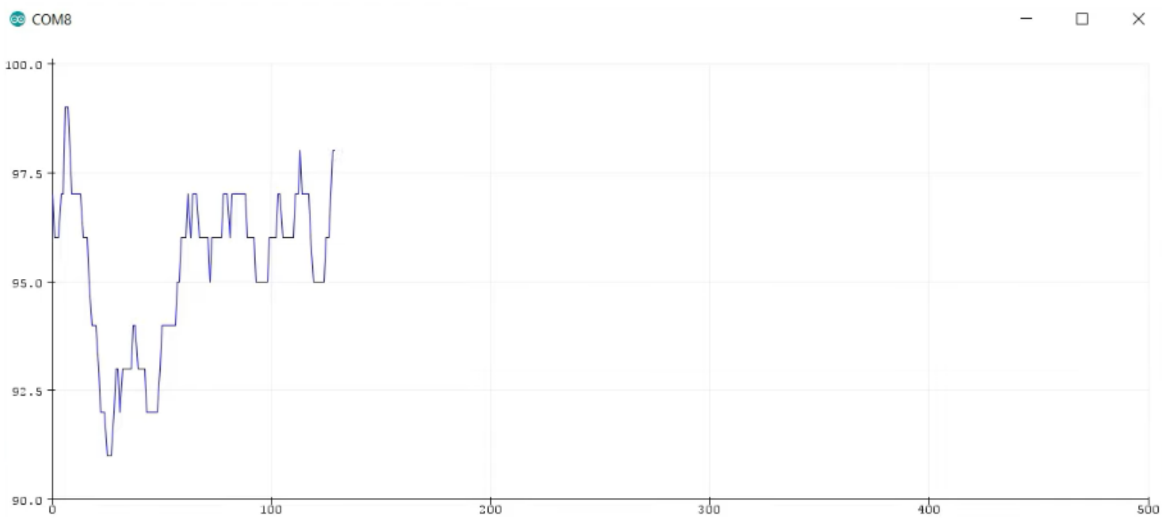


Figure 2.7: Graph

2.5 Algorithm

We are considering a server node where the model will be trained initially and client nodes will clone this server model and where this model is trained and tested using client's local data and then just the updated weights are passed on to the Server where the weights are aggregated to update the individual weights.

Algorithm 1 Federated Learning

```
Server-Side:
Initialize global model weight $w_0$
for each round $t = 0, 1, \dots$ do
    for each client $i = 0, \dots, n-1$ in parallel do
        Send $w_t$ to client $i$
        Receive updated weight $w_{t+1}^{(i)}$ from client $i$
    end for
    $$w_{t+1} \leftarrow \frac{1}{n}\sum_{i=1}^{n} w_{t+1}^{(i)}$$
end for

Client-Side:
Split local dataset into batches of size $B$
for each local epoch $e = 0, \dots, E-1$ do
    for each mini-batch $b$ of size $B$ do
        Update local model:
        $$w_t = w_t - \eta \cdot \nabla L(w, b)$$
    end for
end for
Return updated weight $w_{t+1}$ to the server
```

Chapter 3

Conclusion

3.1 Future Work

- Broader Architecture: using Raspberry Pi as client nodes instead of Arduinos.
- Inspect the possiblity of using Wireless Edge Nodes.
- Training the Federated Learning Model.
- Evaluation of the model on the setup.

3.2 Conclusion

In conclusion, this report has presented the progress in the research work done in the possibility of using Federated Learning Method for Heart Rate Abnormality using IoT System.

References

[1] Sana, F., Isselbacher, E. M., Singh, J. P., Heist, E. K., Pathik, B., and Armoundas, A. A., 2020. “Wearable devices for ambulatory cardiac monitoring: Jacc state-of-the-art review”. *Journal of the American College of Cardiology,* **75**(13), pp. 1582–1592.

[2] Siwicki, B., 2024. “How iot is transforming remote patient monitoring”. *Healthcare IT News*, Nov.

[3] Rudraraju, S. R., Suryadevara, N. K., and Negi, A., 2023. “Heterogeneous sensor data acquisition and federated learning for resource constrained iot devices—a validation”. *IEEE Sensors Journal,* **23**(15), Jun., pp. 17602–17610.

[4] Ravi, D., Wong, D. W. K., Lo, B. P. L., and Yang, G. Z., 2017. “A survey on deep learning in medical image analysis”. *Medical Image Analysis,* **42**, pp. 60–88.

[5] Yıldırım, □., Kucuk, F., and Aydin, M., 2020. “Heart rate variability analysis using deep learning techniques”. *Journal of Biomedical Informatics,* **112**, p. 103614.

[6] Bourke, A. K., and et al., 2019. “Evaluation of machine learning techniques for heart rate variability analysis”. *Journal of Healthcare Engineering*, pp. 1–12.