Human Activity Recognition using Federated Learning

Bachelor of Technology
in
Computer Science and Engineering

by

Gokuleshwaran N (20BCI0031)

Solomon Abhilash Martin (20BCE0638)

Pradyumn Tendulkar (20BCE0762)

Under the guidance of

Prof. Raja S. P.

Assistant Professor Grade 2

School of Computer Science and Engineering
VIT Vellore.

May 2024.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION

I hereby declare that the thesis entitled **“Human Activity Recognition using Federated Learning”** submitted by me, for the award of the degree of Bachelor of Technology in Programme to VIT is a record of bonafide work carried out by me under the supervision of **Prof. Raja S. P.**

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore

Date: 09/05/2024

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled **“Human Activity Recognition using Federated Learning”** submitted by **Gokuleshwaran N (20BCI0031)** School of Computer Science and Engineering, VIT, for the award of the degree of Bachelor of Technology in Programme, is a record of bonafide work carried out by him under my supervision during the period, 15.12.2023 to 08.05.2024, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place: Vellore

Date: 09/05/2024

Signature of the Guide

Internal Examiner **External Examiner**

Head of the Department of Programme

Acknowledgement

We express our deepest gratitude to our respected **Founder Chancellor and President Prof. Dr. G. VISWANATHAN.**

We are very much grateful to our beloved **Vice Chancellor DR. V. S. Kanchana Bhaaskaran**, for providing us with an environment to complete our project successfully.

We are obligated to our beloved Registrar **DR. Jayabarathi T**, for providing immense support in all our endeavors.

We also take this opportunity to express a deep sense of gratitude to Our Internal Supervisor **Prof Raja S. P.** For his cordial support, valuable information and guidance, he helped us in completing this project through various stages. We thank our department faculty, supporting staff and friends for their help and guidance to complete this project. Finally, we would like to thank our ever-loving parents for their blessings and our friends for their timely help and support.

Gokuleshwaran N	**(20BCI0031)**
Solomon Martin	**(20BCE0638)**
Pradyumn Tendulkar	**(20BCE0762)**

Executive Summary

Human activity recognition has broad applications in medical research and human survey systems. In this project, we designed a smartphone-based recognition system that recognizes human activities such as walking, limping, jogging, going upstairs, and going downstairs. The system collected time series signals using a built-in accelerometer, generated 31 features in both time and frequency domains, and then reduced the feature dimensionality to improve performance. To address the challenges of distributed data privacy and accessibility, the activity data were trained using a federated learning framework across three client devices, which then contributed to a global model.

The best classification rate in our experiment was 91.35%, achieved by the global model aggregating insights from the federated nodes. Federated learning proved effective in utilizing decentralized data while maintaining data privacy and reducing the need for data centralization. This approach also demonstrated robustness to the orientation and position of smartphones, essential for real-world applications.

Future work may consider including more diverse activities and implementing a real-time system on smartphones. Further exploration of federated learning optimization techniques, such as advanced aggregation algorithms and client selection strategies, may enhance the system's performance and efficiency.

1) INTRODUCTION:

A) OBJECTIVES

The primary aim of this project is to develop a federated learning (FL)-based HAR system that leverages the computational resources of edge devices (smartphones and wearables) to perform local data processing. By doing so, the project seeks to preserve the privacy of the users' sensor data and reduce the dependency on centralized server infrastructure. This approach aims to achieve comparable or superior accuracy in activity recognition tasks without compromising on privacy and scalability.

B) MOTIVATION

As the proliferation of digital sensors in everyday devices continues, an increasing amount of sensor data is being generated and utilized in various domains. Organizing this data, particularly by recognizing human actions based on sensor readings, is crucial for enhancing the functionality and user interaction of numerous systems. Human Action Recognition (HAR) using sensor data is a burgeoning field of research with potential applications across several areas including health monitoring, smart home automation, human-computer interaction, and even in sports training. For instance, in health monitoring systems, successful HAR can automatically detect abnormal patient movements, triggering alerts and facilitating timely interventions. In smart homes, sensor-based HAR can enhance automation by adjusting environments to suit the activities of residents, such as altering lighting and temperature based on their actions. Moreover, in human-computer interaction, understanding and interpreting human movements through sensors can lead to more intuitive and engaging interfaces, significantly improving user experience. This emphasis on sensor-based action recognition offers a robust alternative to video-based systems, avoiding privacy concerns associated with continuous video recording and enabling more scalable and versatile implementations.

C) BACKGROUND

Although the field of sensor-based human action recognition (HAR) is rich with potential, it presents distinct challenges that complicate research and development. First, there is significant variability in how actions are performed, both within and between individuals. For example, the same activity like walking can vary immensely in pace and stride length across different people and even in the same person under different conditions. These variations can make it difficult for algorithms to consistently recognize actions within the same class and distinguish between different classes.

Second, the quality and type of sensor data can vary widely depending on the device's specifications and the environment in which it is used. Factors such as sensor placement, sensitivity, and the presence of noise can all influence the data accuracy. For instance, a sensor worn on the wrist may provide different data from one attached to the foot or torso, impacting the system’s ability to accurately interpret the activity.

Third, the dynamic nature of human activities and the continuous stream of sensor data necessitate algorithms that can effectively process and analyze data in real time, often with limited computational resources. This is particularly challenging when the system must operate on low-power devices like smartphones or wearable fitness trackers, where processing capabilities and battery life are limited.

Moreover, integrating sensor data from multiple sources to improve accuracy and reliability adds another layer of complexity. This necessitates advanced data fusion techniques that can handle discrepancies and gaps in the data collected from different sensors.

2) PROJECT DESCRIPTION AND GOALS:

A) SURVEY ON EXISTING SYSTEM

Chen, H. et al. introduce the AttCLHAR model, a novel approach to human activity recognition (HAR) in smart homes, particularly designed for scenarios with sparse or absent annotations. This model ingeniously combines the SimCLR self-supervised learning framework with a self-attention mechanism and sharpness-aware minimization (SAM) to improve the learning of representative features from unlabeled ambient sensor data. The architecture comprises a shared encoder with convolutional layers and an LSTM layer, enhanced by a self-attention layer to focus on different segments of input sequences more effectively. AttCLHAR stands out for its ability to pre-train in an unsupervised manner on extensive unlabeled data, subsequently fine-tuned to achieve superior performance in semi-supervised and transfer learning contexts compared to SimCLR and its variations. This approach not only mitigates the challenge of collecting and annotating massive volumes of sensor data but also significantly advances the state-of-the-art in HAR within smart home environments, as demonstrated in extensive evaluations using three CASAS smart home datasets. However, the model's effectiveness could be influenced by the imbalance of daily activities in datasets and its primary application to ambient sensors, potentially limiting its adaptability to diverse environments or sensor configurations.

Müller, P.N. et al. explore the effectiveness of Convolutional Neural Networks (CNNs) in recognizing fitness activities using data from inertial measurement units (IMUs) found in mobile devices. This research is pivotal for mobile fitness applications that aim to provide real- time feedback on users' activities by accurately identifying different fitness activities through sensor data. Despite CNNs' success in various time series classification tasks, the recognition of fitness activities poses unique challenges due to similarities between activities and limited data availability for model training. The study evaluates three existing CNN architectures adapted for fitness activity recognition (FAR) and introduces a novel architecture, the Scaling Fully Convolutional Network (Scaling-FCN), designed specifically for this task. A preprocessing pipeline was established, and a dataset involving running exercises from 20 participants was compiled to assess the networks' performance. Traditional machine learning methods commonly employed in human activity recognition (HAR) were also compared to these CNN architectures. The findings reveal that while CNN architectures consistently achieve high test accuracy (at least 94%), traditional machine learning methods, particularly support vector machines (SVMs), show superior performance in certain scenarios. Interestingly, reducing the sensor count to just one foot sensor decreased traditional methods' effectiveness but enhanced CNN performances, with the Scaling-FCN achieving the highest accuracy (99.86%). This study contributes significantly to the field by introducing the Scaling- FCN architecture tailored for sensor-based FAR, offering a publicly available dataset for running exercises, and providing a comprehensive analysis of CNN performance compared to traditional machine learning approaches under various input data conditions. The results suggest CNNs are generally well-suited for fitness activity recognition, with potential performance improvements when sensors are selectively dropped. However, traditional machine learning architectures can still compete or even surpass CNNs given favorable input data conditions.

Subburam, R. et al. developed a fall detection system for elderly care using machine learning techniques, leveraging a dataset of activities performed by 11 individuals. The study preprocesses data by handling missing values and encoding categorical variables, training various classifiers like Support Vector Machine (SVM), Logistic Regression, Random Forest, AdaBoost, and Gradient Boosting (GB) to evaluate their performance. The Enhanced Random Forest (ERF) model was highlighted for its superior accuracy in fall detection. The research aimed to improve elderly well-being by precisely detecting falls through features extracted from tri-axial accelerometer and gyroscope data. The dataset included both fall and non-fall activities, balanced to ensure equal representation in training and testing. Preprocessing involved dealing with missing data and outliers, enhancing data quality for model training. The models were evaluated based on metrics such as precision, recall, accuracy, F1 score, and a confusion matrix. The ERF model outperformed others, demonstrating the effectiveness of machine learning in fall detection. Visualizations like bar charts and heatmaps were used to interpret model performance and feature importance. This study underscores the potential of machine learning, especially the Enhanced Random Forest algorithm, in fall detection systems, suggesting future research could integrate IoT techniques and clinical support for real-world applications.

Hassan, N. et al. developed a cutting-edge method for recognizing human activities dynamically, a significant step forward in the domains of computer vision and pattern recognition. Their approach integrates the MobileNetV2 Convolutional Neural Network for extracting features from video frames and a Deep Bidirectional Long Short-Term Memory

(Deep BiLSTM) network to analyze and leverage temporal dependencies, facilitating accurate predictions of human activities. This innovative combination aims to tackle the prevalent issues of lower accuracy and computational complexity in current human activity recognition (HAR) systems, which struggle with large video datasets and activities that closely resemble each other. By conducting rigorous evaluations on three benchmark datasets—UCF11, UCF Sport, and JHMDB—the team achieved remarkable accuracies of 99.20%, 93.3%, and 76.30%, respectively. These outcomes not only demonstrate the effectiveness of merging CNN with Deep BiLSTM for HAR but also signify a considerable advancement in the field, offering a robust solution to the challenges faced by existing systems. Through this research, Hassan, N. et al. contribute significantly to the ongoing development of more adaptive and efficient human activity recognition technologies.

Bouazizi et al. introduce a novel method for detecting activities within indoor environments by utilizing multiple 2D Lidar sensors. This approach addresses the limitations of current non- contact sensor-based activity detection systems hindered by environmental obstacles. By positioning multiple 2D Lidars around an indoor space, the system creates a comprehensive representation of activities by transforming Lidar data into image-like formats. These images are then processed using a convolutional Long Short-Term Memory (LSTM) Neural Network to classify different activities, achieving high accuracy rates for activity detection (96.10%), fall detection (99.13%), and unsteady gait detection (93.13%). This method promises to enhance the monitoring and identification of potentially hazardous events for the elderly, leveraging non-intrusive 2D Lidars. The research delves into various aspects, including the optimal settings for Lidar height and grid size for data processing, and compares the proposed method against conventional ones, showing superior performance. The study highlights the potential of 2D Lidar technology in healthcare and assisted living applications, offering a privacy-preserving, efficient solution for real-time activity monitoring. Future work aims to address technical challenges such as interference between Lidars and inaccuracies in distance estimation to further improve the method's robustness and reliability.

Ordóñez, F.J., and Roggen, D. introduce DeepConvLSTM, a deep learning framework for human activity recognition (HAR) from wearable sensors, combining convolutional and LSTM recurrent units. This model automatically learns features from raw sensor data, performs sensor fusion, and models temporal dynamics without requiring expert knowledge. Evaluated on two datasets, including the OPPORTUNITY challenge dataset, DeepConvLSTM outperforms deep non-recurrent networks and previous methods, improving recognition accuracy significantly. It works with multimodal wearable sensors, directly processes raw sensor data, and its performance benefits from multimodal sensor fusion. The framework's effectiveness is demonstrated across various activities, showing that LSTM units are crucial for distinguishing similar gestures by learning temporal feature activation dynamics. The architecture is suitable for online HAR and suggests potential for open-ended learning scenarios and transfer learning approaches. The study provides insights into optimizing key architectural hyperparameters and indicates that deep learning can improve HAR performance, even with relatively small datasets.

Kalabakov et al. explore the integration of Federated Learning (FL) for Human Activity Recognition (HAR) to address privacy and cost issues associated with centralized data collection. Their study contrasts the performance and behavior of FL against traditional deep learning (DL) models under real-world conditions such as varying sensor placements, bandwidth efficiency, and label accuracy. Utilizing the JSI-FOS and PAMAP datasets, they demonstrate FL's competitive performance with enhanced bandwidth efficiency despite a performance deficit compared to DL models. FL models also showed robustness to incorrectly labeled data and adapted well to heterogeneous sensor data. This work aims to offer a comprehensive system-level analysis of FL in HAR, providing valuable insights for designing FL systems under practical deployment scenarios.

Gad et al. propose a novel Federated Learning via Augmented Knowledge Distillation (FedAKD) framework for distributed training of heterogeneous deep learning models in Human Activity Recognition (HAR) systems. This approach is designed to operate efficiently on wearable devices for health monitoring and activity tracking, addressing the challenge of leveraging large, distributed datasets while maintaining data privacy on users' devices. FedAKD distinguishes itself from standard federated learning by enabling the collaborative training of models with different architectures and learning capacities, significantly reducing communication overhead by 200 times compared to traditional FL methods. Evaluated on two distinct HAR datasets—one waist-mounted tabular and one wrist-mounted time-series—the FedAKD method demonstrates superior performance and flexibility. It achieves up to 20% performance gains for clients and shows greater robustness in scenarios with statistical heterogeneity. This advancement suggests a promising direction for developing privacy- preserving, efficient, and adaptable HAR systems in the realm of Internet of Things (IoT), potentially revolutionizing remote health monitoring and other applications requiring sensitive data handling.

Shen, Q. et al. address significant challenges in deploying real-world human activity recognition (HAR) systems due to privacy concerns and the diversity of behavioral patterns among individuals. They introduce DivAR, a federated meta-learning framework designed to recognize human activities by leveraging a centralized embedding network for extracting shared sensory features and decentralized attention modules for individual-specific features. This approach clusters individuals based on behavioral patterns and social factors, applies meta-learning within a federated learning architecture to learn common and cluster-specific features, and uses CNN-based attention modules to capture individual discrepancies. Two multi-individual heterogeneous datasets were constructed to evaluate the model, showing DivAR's competitive performance in multi-individual activity recognition tasks. The model ensures privacy by training locally on user devices and transferring only parameter updates, addressing heterogeneity and 'cold-start' problems in context-aware applications.

Khan et al. introduce a novel hybrid spiking-LSTM (S-LSTM) model within a federated learning framework for privacy-preserving and energy-efficient human activity recognition (HAR) using wearable sensors. This model ingeniously merges the computational efficiency of spiking neural networks (SNNs) with the sequence modeling prowess of long short-term memory (LSTM) networks, optimized through surrogate gradient learning and backpropagation through time. The S-LSTM model's superiority is demonstrated through extensive evaluations on two public datasets, showing remarkable performance improvements in accuracy and energy efficiency over conventional LSTM, CNN, and S-CNN models. For example, the S-LSTM model achieved accuracies of 97.36% and 89.69% for indoor and outdoor scenarios, respectively, and showed a 32.30% enhancement in energy efficiency compared to a simple LSTM model. Moreover, the study emphasizes the importance of personalization in HAR, where model accuracy improved by up to 9% through fine-tuning with local data for individual users.

B) RESEARCH GAP

Human activity recognition (HAR) has been extensively explored, with various methodologies proposed to enhance its accuracy and applicability. Traditional approaches often utilize a combination of vision sensors, inertial sensors like accelerometers and gyroscopes, or both. While machine learning algorithms have become popular due to their robustness and accuracy, threshold-based algorithms continue to be valued for their simplicity and speed. In terms of hardware, setups frequently involve multiple cameras or a variety of sensors positioned across different parts of the body to capture comprehensive movement data.

Recent advancements have also seen the integration of vision and inertial sensors to provide richer datasets, although each method has its limitations. The effectiveness of these systems heavily relies on the quality of data processing and feature extraction techniques. Significant efforts have been directed toward optimizing feature generation from time series data, analyzing signals in both time and frequency domains to maximize the informativeness of the data. However, despite these advancements, there remains a critical need for more adaptive and efficient algorithms capable of handling diverse and noisy datasets typically encountered in real-world settings. This includes improving the interpretability of sensor data, enhancing the energy efficiency of data processing, and developing more sophisticated models that can dynamically adapt to new user behaviors and environmental contexts without extensive retraining.

C) PROBLEM STATEMENT

Human Activity Recognition (HAR) is the problem of identifying a physical activity carried out by an individual dependent on a trace of movement within a certain environment. Activities such as WALKING, RUNNING, SITTING, STANDING, JOGGING, BIKING, WALKING_UPSTAIRS, and WALKING_DOWNSTAIRS are classified as regular physical movements and form our class of activity which is to be recognized. To record movement or change in movement, sensors such as triaxial accelerometer and gyroscopes, capture data while the activity is being performed. A triaxial accelerometer data detects acceleration or movement along the three axes and a gyroscope measures rotation along the three axes to determine direction. Data recorded is along three dimensions of the X, Y and Z axis at the specified frequency. For example, a frequency of 20Hz would indicate that 20 data points are recorded each second of the action. Various other physiological signals such as heartbeat, respiration, etc. and environmental signals such as temperature, time, humidity, etc. can further augment the recognition process. Activity recognition can be achieved by exploiting the information retrieved from these sensors.

3) PROJECT DESCRIPTION:

This section outlines the core components of the Human Activity Recognition (HAR) project using Federated Learning (FL), providing insights into the requirements, a comprehensive feasibility study, and detailed system specifications.

A) Requirements:

i) Functional:

1. Data Collection: The system must continuously collect accelerometer data from smartphones in real-time. It should support data collection at a specified frequency (e.g., 20Hz) to ensure consistent and useful data input.

2. Activity Classification: Must classify a set of predefined activities (walking, running, lying down, sitting, standing) accurately. Should process data and perform classification directly on the device to support decentralized model operation.

3. Model Training and Updating: Should implement federated learning mechanisms to train the model with data from multiple devices without requiring data centralization. Must periodically receive model updates from a central server while ensuring that individual data contributions remain anonymous and secure.

4. User Interaction: User interface on the smartphone to show real-time classification results. Option for users to provide feedback on the accuracy of activity recognition to improve model training.

5. Data Integration: Ability to integrate additional sensor data in future to enhance activity recognition accuracy (e.g., gyroscope, GPS).

ii) Non-Functional

1. Privacy: Ensure data privacy by implementing data processing locally on user devices and only sharing model updates. Adhere to relevant data protection regulations (e.g., GDPR, HIPAA).

2. Scalability: The system should be scalable to handle data from an increasing number of users without degradation in performance. Efficient management of bandwidth and computational resources to accommodate growing user base.

3. Performance: Achieve high accuracy and low latency in activity classification to ensure user satisfaction and system reliability. Optimize power consumption to minimize the impact on device battery life.

4. Usability: User-friendly interface that requires minimal interaction for setup and daily use. Provide clear instructions and support for troubleshooting.

5. Reliability: The system must operate robustly under various conditions, including different smartphone models and operating systems. Handle errors gracefully, with mechanisms to recover from failures without losing significant functionality.

B) Feasibility Study

i) Economic Feasibility: The economic feasibility of implementing a federated learning system for HAR is promising. By utilizing existing hardware (smartphones and wearables) and open-source software frameworks for FL, the upfront costs are significantly reduced. Additionally, the distributed nature of FL reduces the need for high-end server infrastructure, lowering operational costs related to data storage and processing.

ii) Technical Feasibility: From a technical standpoint, the proposed system is feasible with current technology. Smartphones and wearable devices possess the necessary computational power to run local machine learning models for HAR. Federated learning frameworks, such as TensorFlow Federated, provide the tools needed to implement and manage the FL process. Challenges related to model synchronization, data heterogeneity, and communication efficiency are being actively addressed in current research, making the technical implementation increasingly viable.

iii) Social Feasibility: The social feasibility of the project is high, given the growing awareness and concern over data privacy. By ensuring that data remains on the user's device, the proposed system addresses these privacy concerns, making it more acceptable to users. Additionally, the potential for improved activity recognition can lead to better personalized services, further increasing user acceptance.

C) System Specification

i) Hardware Specification:

- **User Devices**: Smartphones and wearable devices with motion sensors (accelerometers, gyroscopes) capable of running local FL algorithms.
- **Federated Server**: A server with sufficient processing power to aggregate model updates from multiple devices. This does not need to be as powerful as a central server in traditional systems, as heavy computation is distributed among user devices.

ii) Software Specification:

- **Federated Learning Framework**: TensorFlow Federated (TFF) or PySyft for managing FL processes.
- **Machine Learning Library**: TensorFlow or PyTorch for designing and training HAR models.

- **Operating Systems**: Android, iOS, and other wearable OS for user devices; server- side software compatible with Linux/Unix systems.

iii) Standards and Policies:

- **Data Privacy Regulations**: Compliance with GDPR, CCPA, and other relevant privacy regulations.
- **Data Security Standards**: Implementation of secure communication protocols (such as TLS) for transferring model updates.
- **Model Sharing and Aggregation Policies**: Policies to ensure that model updates are properly aggregated and shared among participating devices while maintaining user privacy.

4) DESIGN APPROACH AND DETAILS:

A) System Architecture

Our proposed system architecture for Human Activity Recognition (HAR) using federated machine learning is designed to address the challenges of traditional HAR systems while preserving data privacy, ensuring data security, and enhancing scalability. The architecture consists of the following key components:

i) Client Devices

- Client devices are distributed sensors or smartphones worn by users to collect activity data locally. Each client device independently processes and analyzes raw sensor data to identify human activities. Data remains localized on the client devices, ensuring user privacy and minimizing the risk of data exposure.

ii) Central Server

- The central server acts as the orchestrator of the federated learning process. It coordinates communication between client devices and facilitates model training and aggregation. The central server aggregates model updates from participating client devices to update the global model iteratively.

iii) Communication Protocol

- Secure and efficient communication protocols are implemented to facilitate communication between client devices and the central server. These protocols ensure that data transmission is encrypted to prevent unauthorized access or tampering. Communication overhead is minimized to optimize the efficiency of the federated learning process.

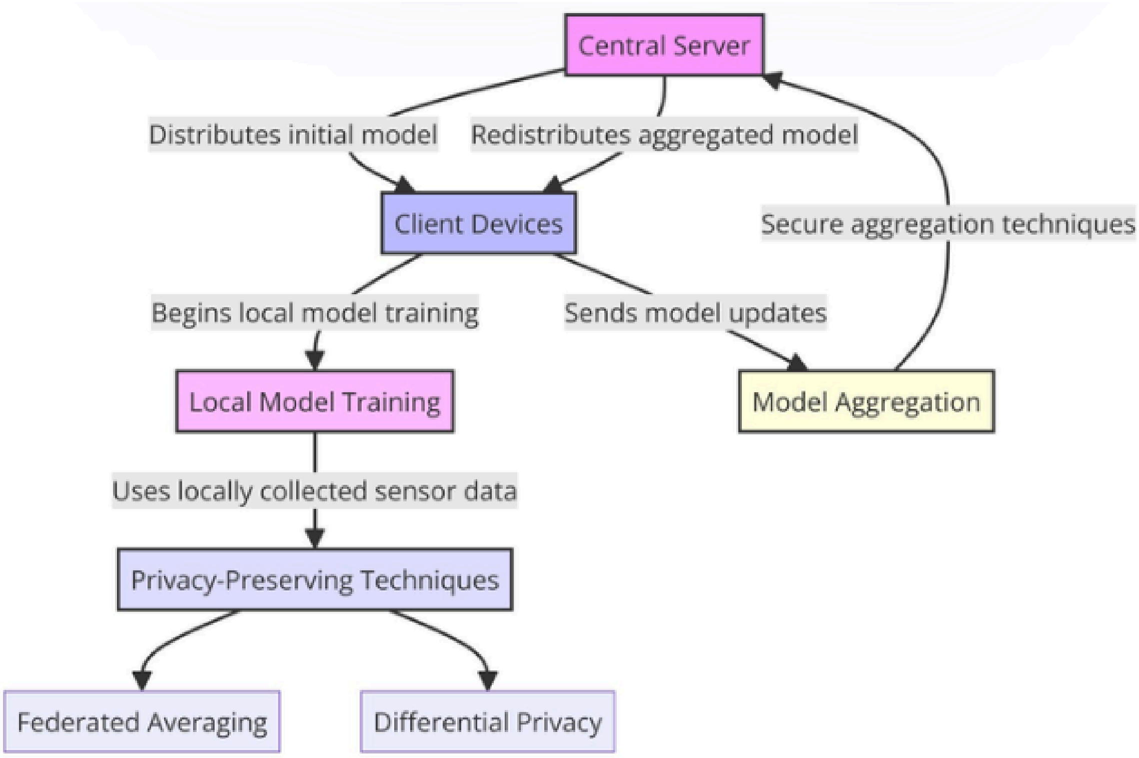

Central Server
Distributes initial model
Redistributes aggregated model
Client Devices
Secure aggregation techniques
Begins local model training
Sends model updates
Local Model Training
Model Aggregation
Uses locally collected sensor data
Privacy-Preserving Techniques
Federated Averaging
Differential Privacy

Figure 1: System Architecture

B) System Workflow:

i) Initialization:

- The central server distributes an initial model to client devices to initiate the federated learning process. Client devices receive the initial model and begin local model training using their respective datasets.

ii) Local Model Training:

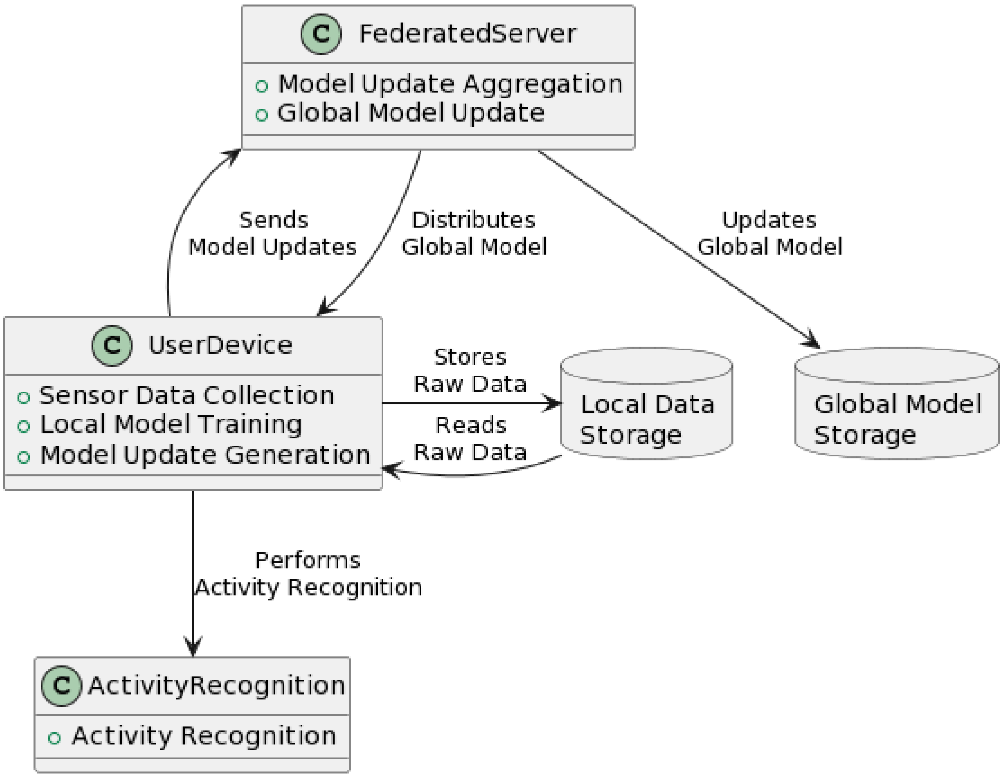
- Client devices independently train the model using locally collected sensor data. Privacy-preserving techniques such as federated averaging and differential privacy are employed to ensure that sensitive user data remains private during model training.

iii) Model Aggregation:

- After completing local model training, client devices send model updates to the central server. The central server aggregates these updates using secure aggregation techniques to update the global model. Aggregated model updates are then redistributed to client devices for the next iteration of training.

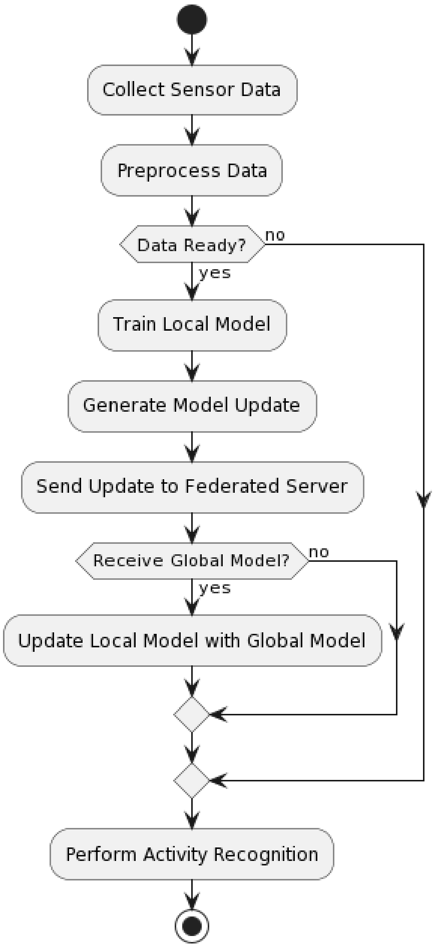
C) Design

i) Data flow diagram



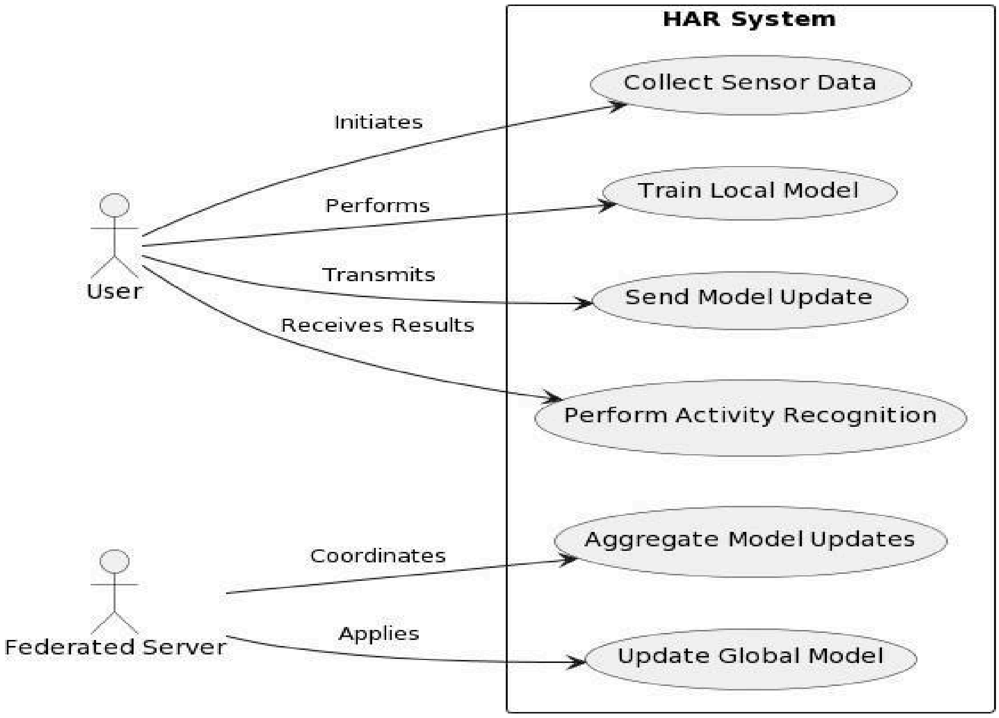
This diagram illustrates the flow of data within the proposed Human Activity Recognition system using Federated Learning. It showcases the process starting from data collection at user devices, local model training, model update transmission to the federated server, and finally, the aggregation of these updates to improve the global model. This representation emphasizes privacy by keeping raw data on the device and only sharing model updates, highlighting the system's decentralized nature.

ii) UML Diagram



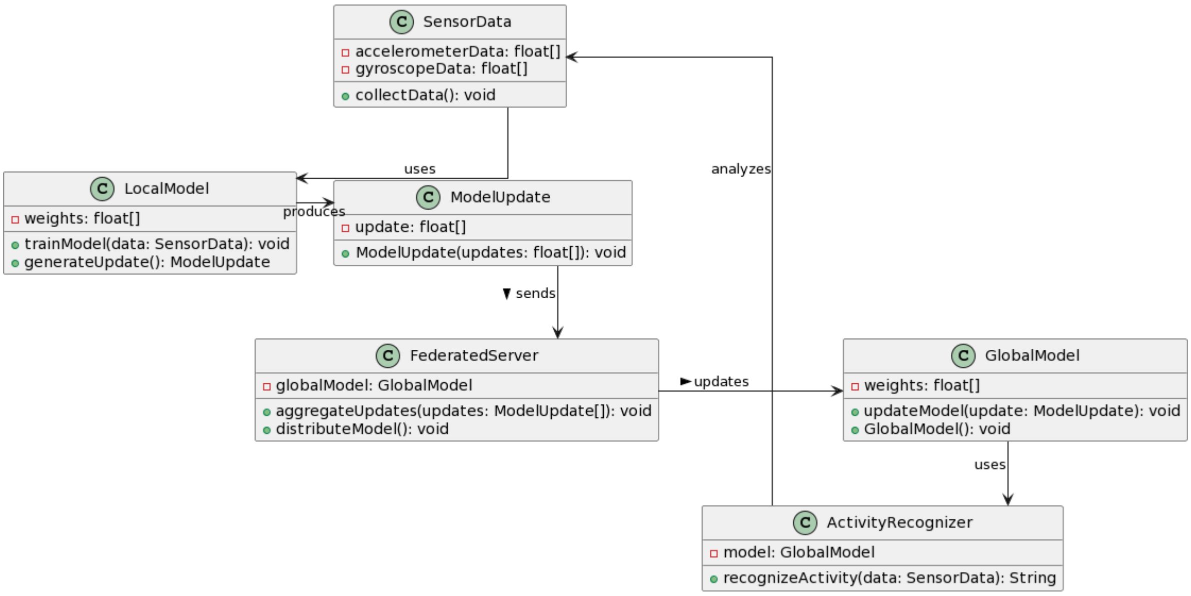
The UML (Unified Modeling Language) diagram presents the system's architecture and the relationships between its various components. It outlines the classes, objects, and interactions within the federated learning-based HAR system, providing a blueprint of the system's structure. This diagram is crucial for understanding the system's design and the interaction between its modular components, aiding in the development process.

iii) Use Case Diagram



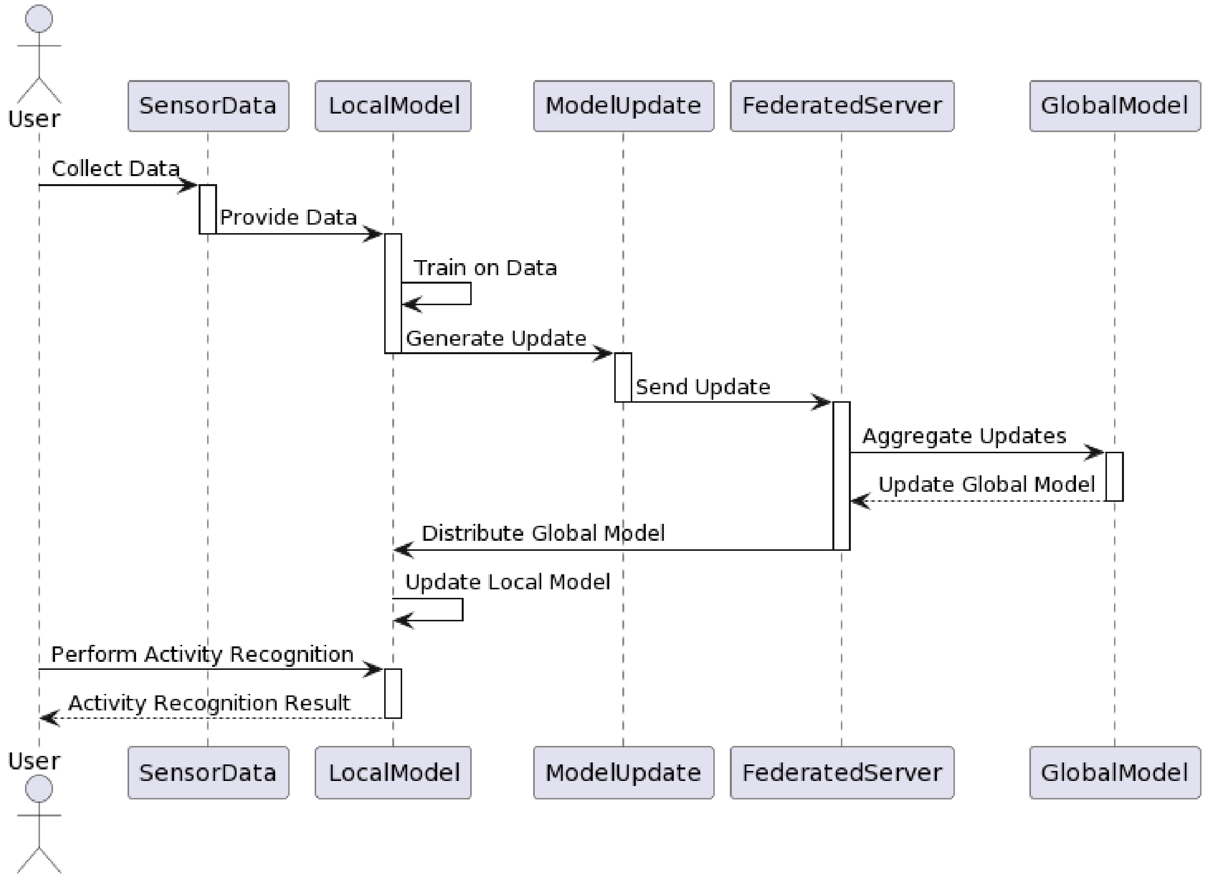
This diagram identifies the system's key actors and the interactions they have with the various use cases within the system. It illustrates how users (with their devices) and the federated server engage with the system, including activities such as initiating model training, sending updates, and receiving aggregated models. This visualization helps in understanding the functionalities available to different actors and their roles in the federated learning process.

iv) Class Diagram



The class diagram details the object-oriented structure of the system, specifying the classes, their attributes, methods, and the relationships among them. This diagram is fundamental for developers to understand how the system is organized and how its various parts are expected to interact, laying the groundwork for the system's coding phase.

v) Sequence Diagram



This diagram depicts the sequence of operations that occur between objects in the system over time, focusing on the order of messages exchanged for a particular scenario. It visualizes the flow of control among the system's components, from the initiation of local model training on user devices to the aggregation of model updates on the federated server, showcasing the dynamic behavior of the system during the federated learning process.

D) Constraints, Alternatives, and Trade-offs

Developing a Human Activity Recognition system using Federated Learning involves navigating various technical and practical constraints. This section outlines the key constraints encountered, explores alternatives, and discusses the trade-offs that were made to balance system performance with practical feasibility.

i) Constraints

The project faced several constraints that shaped the final design:

Privacy Concerns: The need to ensure user data privacy significantly constrained the data handling and processing methods employed. Compliance with data protection regulations (e.g., GDPR) was mandatory.

Resource Limitations: Limited computational resources on client devices restricted the complexity of the local models that could be deployed.

Network Variability: Variations in network connectivity among users affected the reliability of data transmission to the federated server.

Scalability: The system needed to be scalable to handle potentially thousands of users without a degradation in performance.

ii) Alternatives

For each constraint, several alternatives were considered:

Privacy-Preserving Techniques: Instead of traditional data aggregation, techniques like Differential Privacy and Secure Multi-party Computation were evaluated to enhance privacy.

Model Complexity: Lightweight machine learning models were considered to accommodate the limited processing power of mobile devices.

Data Transmission: Different strategies for data synchronization, such as on-demand or scheduled updates, were explored to mitigate network issues.

System Architecture: Both centralized and decentralized architectures were considered to improve scalability and fault tolerance.

iii) Trade-offs

Choosing between alternatives often involved trade-offs, which included:

Privacy vs. Accuracy: Implementing stricter privacy controls through techniques like Differential Privacy often reduced the accuracy of the activity recognition models. A balance was struck by tuning the privacy parameters to optimize both privacy and performance.

Complexity vs. Resource Efficiency: More complex models could potentially improve accuracy but would drain battery life and processing power on client devices. A compromise was found in using simplified models that were periodically updated to enhance accuracy without overly taxing client resources.

Real-time vs. Batch Processing: Real-time data processing offers immediate insights but is resource-intensive. Batch processing, while less demanding, introduces delays. The decision was made to use a hybrid approach, processing data in near-real-time during periods of low network traffic.

Centralized vs. Decentralized Learning: While centralized learning simplifies model management, it poses privacy and scalability issues. Decentralized learning enhances privacy and distributes the computational load but complicates data synchronization and model aggregation. The project adopted a federated learning approach, which is inherently decentralized but coordinates centrally for model aggregation.

5) MODULE DESCRIPTION:

A) Local Model Training

Purpose: This module handles the training of local models on client devices. It uses locally collected sensor data to train models that recognize human activities.

Components:

- **Data Collection**: Utilizes smartphone sensors (accelerometer, gyroscope) to gather motion data.
- **Feature Extraction**: Converts raw sensor data into meaningful features suitable for machine learning models.
- **Model Training**: Employs machine learning algorithms to train models directly on the device, ensuring data privacy.
- **Technologies Used**: TensorFlow, Keras for model building and training.

B) Global Model Aggregation

Purpose: This module is responsible for the aggregation of local model updates to update and enhance the global model in a privacy-preserving manner.

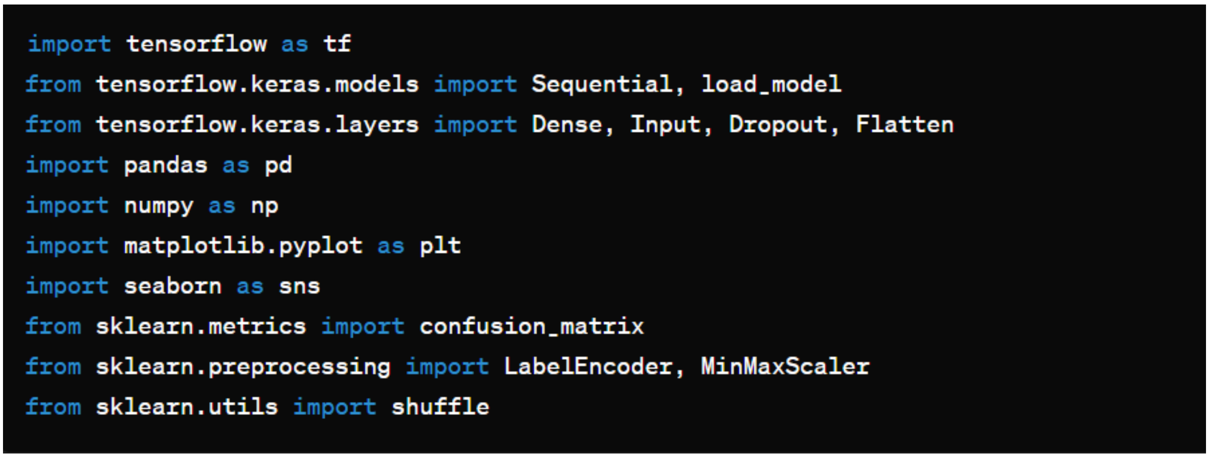
Components:

- **Model Update Collection**: Receives trained model parameters from client devices.
- **Aggregation Algorithm**: Applies federated averaging or other aggregation techniques to combine updates.
- **Global Model Update**: Updates the central global model based on aggregated data, improving overall model accuracy.
- **Technologies Used**: TensorFlow Federated, PySyft for federated learning processes.

6) PROJECT DEMONSTRATION

This section outlines the implementation details of the Human Activity Recognition (HAR) project, focusing on the setup, data preprocessing, and model building phases.

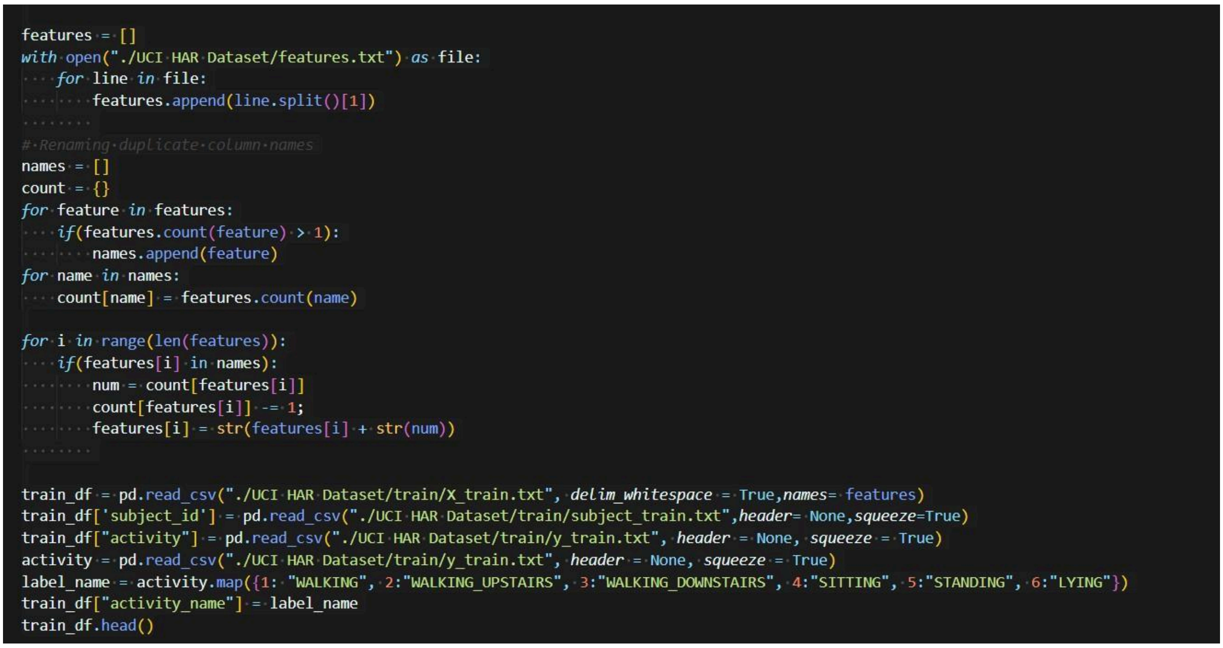
1. Initial Setup and Library Imports



```
import tensorflow as tf
from tensorflow.keras.models import Sequential, load_model
from tensorflow.keras.layers import Dense, Input, Dropout, Flatten
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
from sklearn.preprocessing import LabelEncoder, MinMaxScaler
from sklearn.utils import shuffle
```

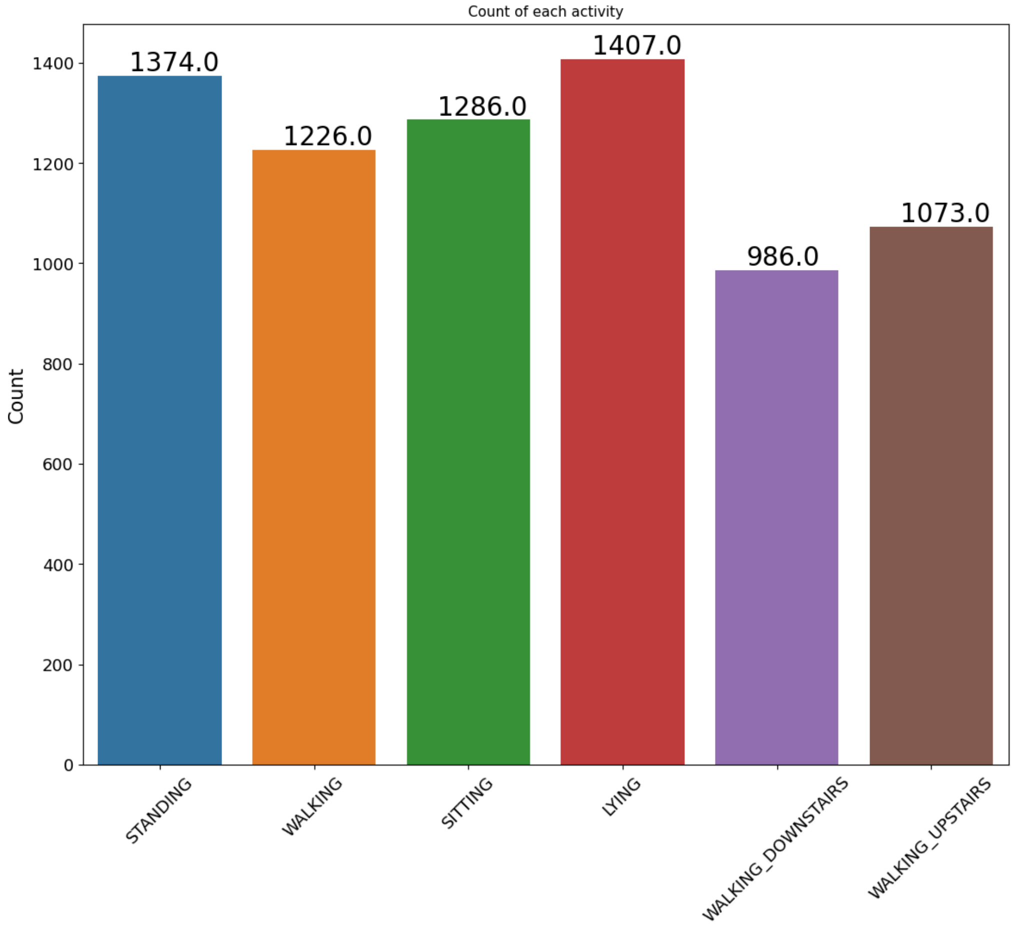
This code snippet imports all the necessary libraries required for data manipulation, visualization, model building, and evaluation.

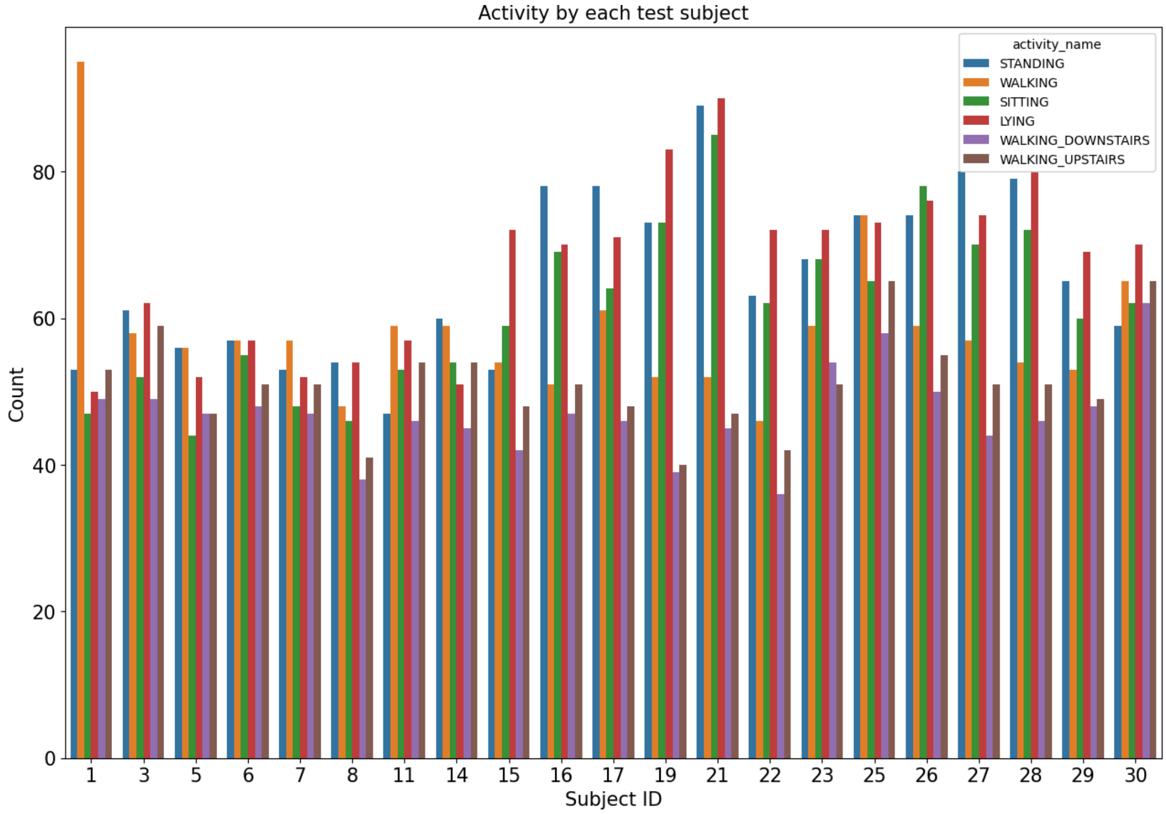
2. Data Loading and Preprocessing



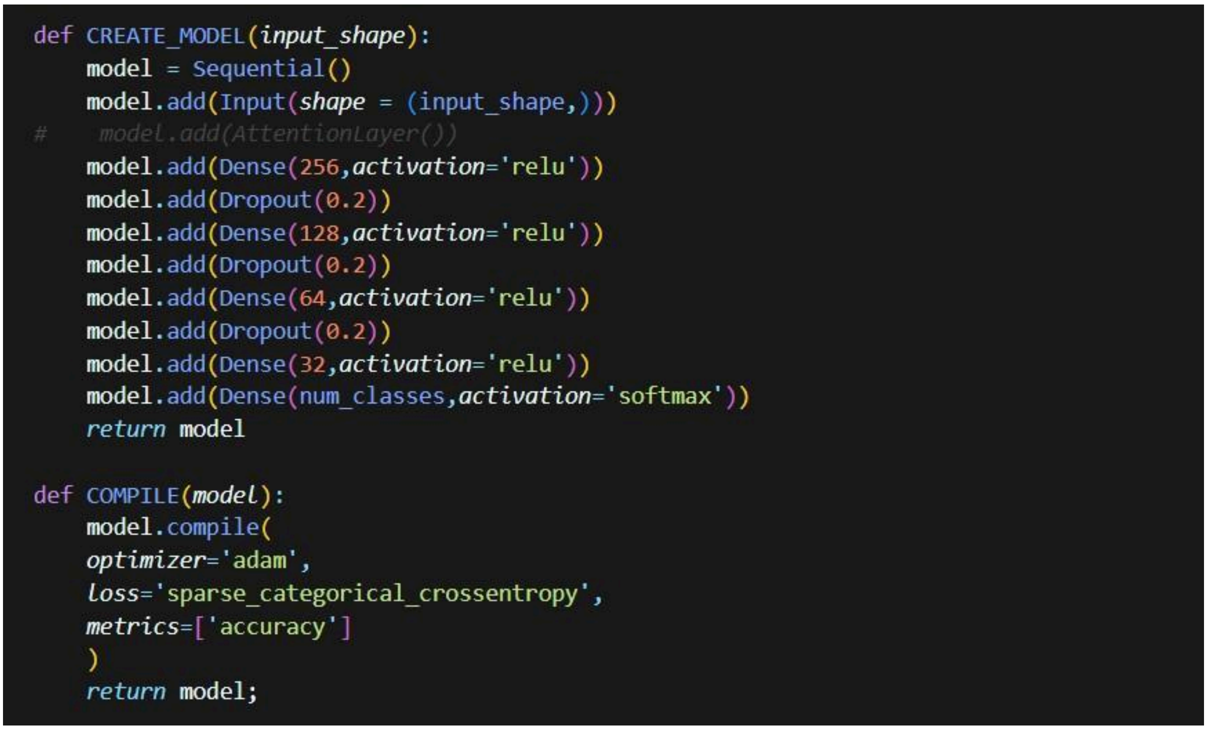
```
features = []
with open("./UCI HAR Dataset/features.txt") as file:
    for line in file:
        features.append(line.split()[1])
        
#·Renaming·duplicate·column·names
names = []
count = {}
for feature in features:
    if(features.count(feature) > 1):
        names.append(feature)
for name in names:
    count[name] = features.count(name)

for i in range(len(features)):
    if(features[i] in names):
        num = count[features[i]]
        count[features[i]] -= 1;
        features[i] = str(features[i] + str(num))
        

train_df = pd.read_csv("./UCI HAR Dataset/train/X_train.txt", delim_whitespace = True,names= features)
train_df['subject_id'] = pd.read_csv("./UCI HAR Dataset/train/subject_train.txt",header= None,squeeze=True)
train_df["activity"] = pd.read_csv("./UCI HAR Dataset/train/y_train.txt", header = None, squeeze = True)
activity = pd.read_csv("./UCI HAR Dataset/train/y_train.txt", header = None, squeeze = True)
label_name = activity.map({1: "WALKING", 2:"WALKING_UPSTAIRS", 3:"WALKING_DOWNSTAIRS", 4:"SITTING", 5:"STANDING", 6:"LYING"})
train_df["activity_name"] = label_name
train_df.head()
```

This snippet demonstrates how the training data is loaded from text files, how features are extracted and duplicates are handled, and how the data is structured into a pandas DataFrame with clear labeling for activities.


Count of each activity
Count
0
200
400
600
800
1000
1200
1400
1374.0
1226.0
1286.0
1407.0
986.0
1073.0
STANDING
WALKING
SITTING
LYING
WALKING_DOWNSTAIRS
WALKING_UPSTAIRS

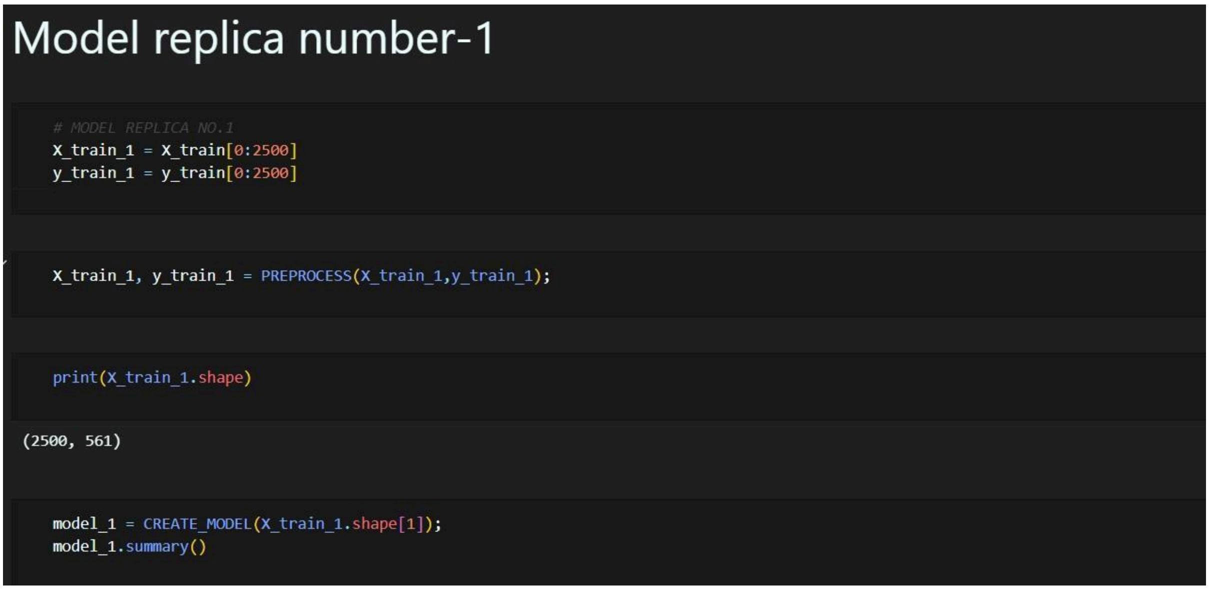

Activity by each test subject
activity_name
STANDING
WALKING
SITTING
LYING
WALKING_DOWNSTAIRS
WALKING_UPSTAIRS
Count
0
20
40
60
80
1
3
5
6
7
8
11
14
15
16
17
19
21
22
23
25
26
27
28
29
30
Subject ID

3. Model Building



```
def CREATE_MODEL(input_shape):
    model = Sequential()
    model.add(Input(shape = (input_shape,)))
#    model.add(AttentionLayer())
    model.add(Dense(256,activation='relu'))
    model.add(Dropout(0.2))
    model.add(Dense(128,activation='relu'))
    model.add(Dropout(0.2))
    model.add(Dense(64,activation='relu'))
    model.add(Dropout(0.2))
    model.add(Dense(32,activation='relu'))
    model.add(Dense(num_classes,activation='softmax'))
    return model

def COMPILE(model):
    model.compile(
    optimizer='adam',
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy']
    )
    return model;
```

4. Training and Evaluating 3 client-side models



Model replica number-1

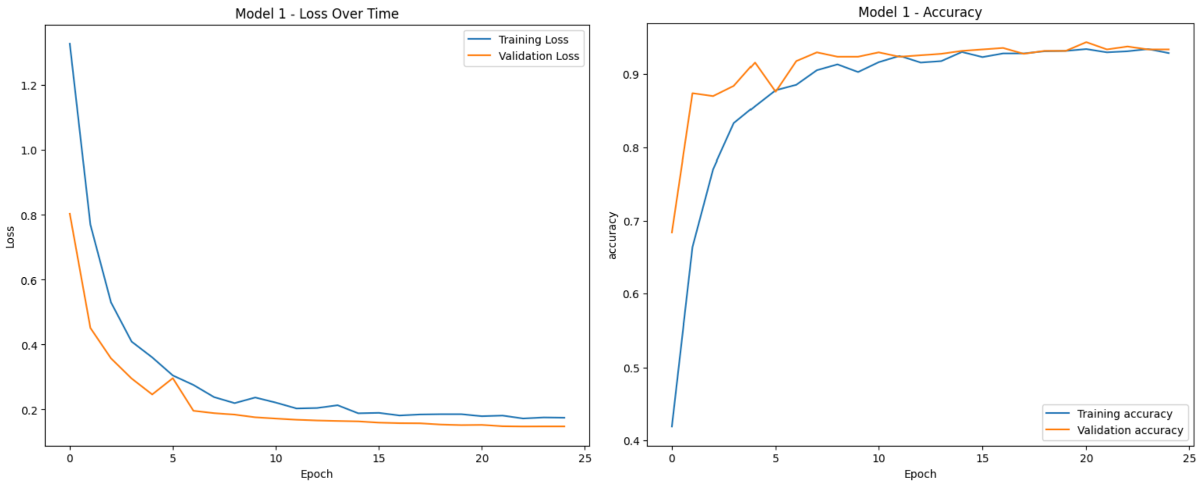
```
# MODEL REPLICA NO.1
X_train_1 = X_train[0:2500]
y_train_1 = y_train[0:2500]
```

```
X_train_1, y_train_1 = PREPROCESS(X_train_1,y_train_1);
```

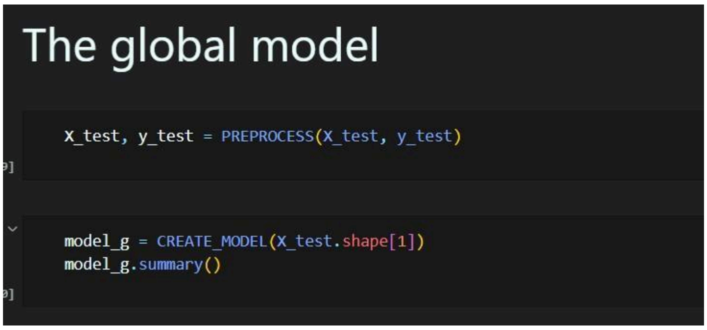
```
print(X_train_1.shape)
```

```
(2500, 561)
```

```
model_1 = CREATE_MODEL(X_train_1.shape[1]);
model_1.summary()
```



5. Global Model

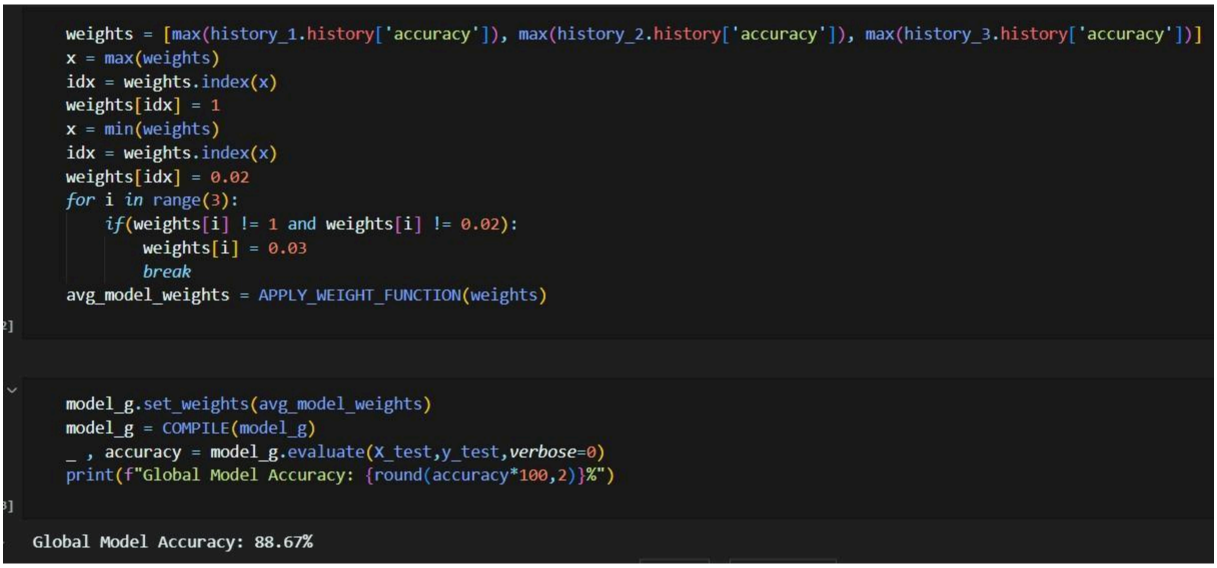


The global model

```
x_test, y_test = PREPROCESS(x_test, y_test)
```

```
model_g = CREATE_MODEL(x_test.shape[1])
model_g.summary()
```

6. Updating global model weights and calculating accuracy

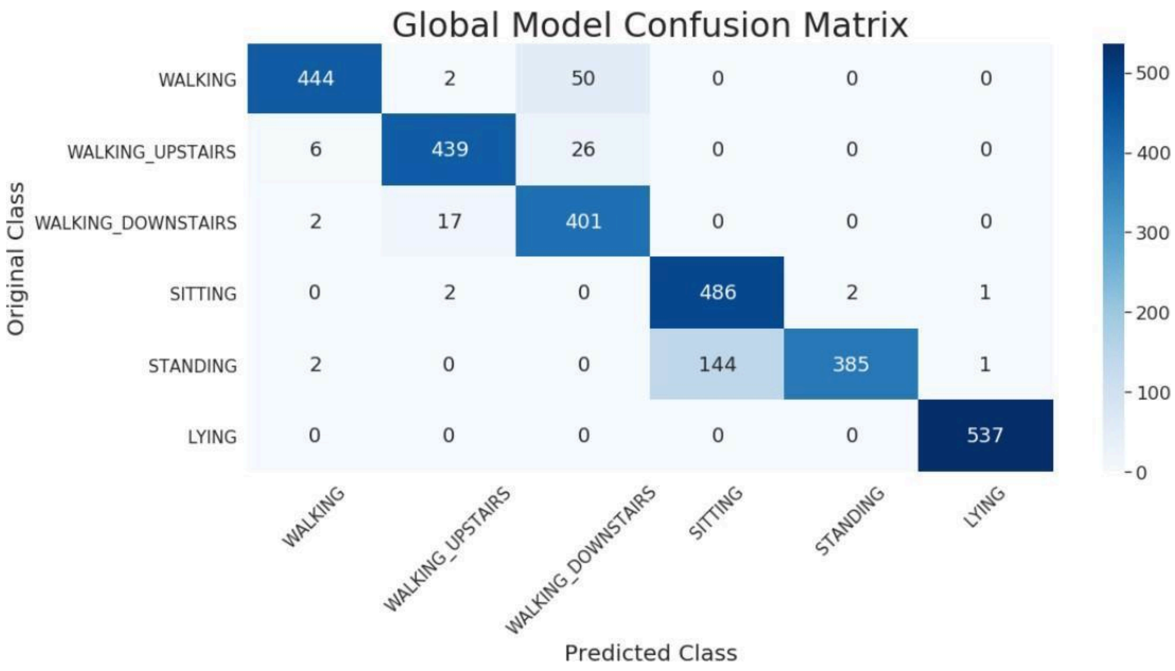


```
weights = [max(history_1.history['accuracy']), max(history_2.history['accuracy']), max(history_3.history['accuracy'])]
x = max(weights)
idx = weights.index(x)
weights[idx] = 1
x = min(weights)
idx = weights.index(x)
weights[idx] = 0.02
for i in range(3):
    if(weights[i] != 1 and weights[i] != 0.02):
        weights[i] = 0.03
        break
avg_model_weights = APPLY_WEIGHT_FUNCTION(weights)
```

```
model_g.set_weights(avg_model_weights)
model_g = COMPILE(model_g)
_ , accuracy = model_g.evaluate(x_test,y_test,verbose=0)
print(f"Global Model Accuracy: {round(accuracy*100,2)}%")
```

```
Global Model Accuracy: 88.67%
```

7. Confusion Matrix



8. Application Overview

The Android app serves as a user-friendly interface that allows users to monitor their activity in real-time. The app receives sensor data from the device, processes it through the TensorFlow Lite model, and displays the predicted activities along with their corresponding probabilities.

9. Model Integration

The machine learning model, initially developed in TensorFlow, was converted to a TensorFlow Lite format to ensure compatibility and efficiency on mobile devices. This conversion optimizes the model for the low-power and resource-constrained environments typical of mobile devices.

Code Snippet: Converting TensorFlow Model to TensorFlow Lite

```
import tensorflow as tf

# Loading the existing TensorFlow model
model = tf.keras.models.load_model('model_g’’)

# Converting the model to TensorFlow Lite
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()

# Saving the converted model
with open('model.tflite', 'wb') as f:
        f.write(tflite_model)
```

10. Loading the Model in the App

The TensorFlow Lite model is loaded into the Android application to perform real-time predictions. The app utilizes the sensor data from the device's accelerometer and gyroscope to feed into the model.

Code Snippet: Loading TensorFlow Lite Model in Android

```
import org.tensorflow.lite.Interpreter
import java.io.File

// Load the TFLite model
val tfliteModel = File(context.filesDir, "model.tflite").readBytes()
val interpreter = Interpreter(tfliteModel)
```

11. Real-Time Activity Prediction

The application processes the sensor data, applies the TensorFlow Lite model, and updates the activity probabilities in real-time. This provides immediate feedback to the user about their current activity.

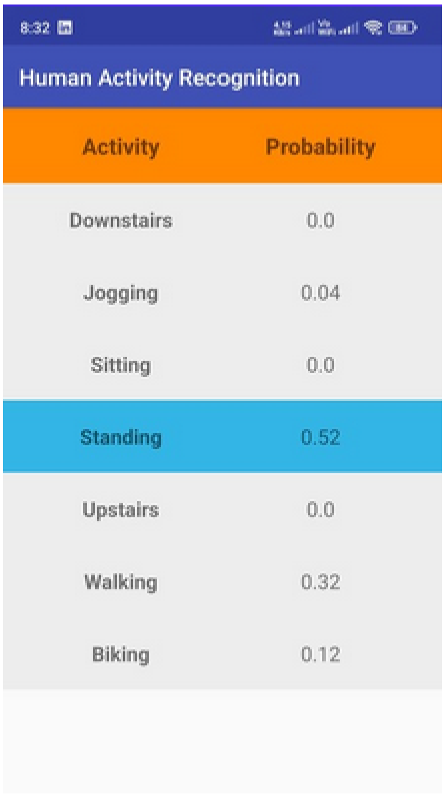
Code Snippet: Predicting Activity in Android

```
// Function to predict activity from sensor data
fun predictActivity(data: FloatArray): Map<String, Float> {
        val inputData = arrayOf(data)
        val outputData = Array(1) { FloatArray(num_activities) }

        // Run the model
        interpreter.run(inputData, outputData)

        // Convert output to readable format
        return activityLabels.zip(outputData[0]).toMap()
}
```

12. User Interface

The displayed screenshot represents an implementation of a Human Activity Recognition (HAR) system, showcasing our mobile application interface that predicts various activities based on sensor data. The app displays a list of activities such as Standing, Walking, Jogging, and Biking, along with their respective probabilities. For instance, the app currently highlights 'Standing' as the most probable activity with a 52% likelihood, followed by 'Walking' at 32%, indicating real-time analysis and user interaction facilitation within the HAR framework. This application is a practical example of utilizing machine learning to interpret sensor data for enhancing user engagement and activity tracking.

Interface Screenshot:



8:32

Human Activity Recognition

Activity	Probability
Downstairs	0.0
Jogging	0.04
Sitting	0.0
Standing	0.52
Upstairs	0.0
Walking	0.32
Biking	0.12

7) Conclusion

In conclusion, our proposed federated machine learning architecture for Human Activity Recognition (HAR) presents a comprehensive solution to the challenges inherent in traditional HAR systems. By decentralizing data processing to client devices while maintaining robust privacy and security measures, our system not only preserves user privacy but also enhances scalability and efficiency. Through the orchestration of model training and aggregation by a central server, our architecture ensures seamless collaboration between client devices while minimizing communication overhead. With the successful implementation of our system workflow, we demonstrate the viability and effectiveness of federated learning in HAR, paving the way for innovative solutions in activity recognition while safeguarding user data privacy.

8) References

Journal:

1. Chen, H.; Gouin-Vallerand, C.; Bouchard, K.; Gaboury, S.; Couture, M.; Bier, N.;Giroux, S. Enhancing Human Activity Recognition in Smart Homes with Self- Supervised Learning and Self-Attention. Sensors 2024, 24, 884. https://doi.org/10.3390/s24030884
2. Müller, P.N.; Müller, A.J.; Achenbach, P.; Göbel, S. IMU-Based Fitness ActivityRecognition Using CNNs for Time Series Classification. Sensors 2024, 24, 742. https://doi.org/10.3390/s24030742
3. Subburam, R.; Chandralekha, E.; Kandasamy, V. An Elderly Fall Detection SystemUsing Enhanced Random Forest in Machine Learning. Eng. Proc. 2023, 59, 172. https://doi.org/10.3390/engproc2023059172
4. Hassan, N.; Miah, A.S.M.; Shin, J. A Deep Bidirectional LSTM Model Enhancedby Transfer-Learning-Based Feature Extraction for Dynamic Human Activity Recognition. Appl. Sci. 2024, 14, 603. https://doi.org/10.3390/app14020603
5. Bouazizi, M.; Mora, A.L.; Feghoul, K.; Ohtsuki, T. Activity Detection in IndoorEnvironments Using Multiple 2D Lidars. Sensors 2024, 24, 626. https://doi.org/10.3390/s24020626
6. Ordóñez, F.J.; Roggen, D. Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. Sensors 2016, 16, 115.https://doi.org/10.3390/s16010115
7. S. Kalabakov et al., "Federated Learning for Activity Recognition: A System LevelPerspective," in IEEE Access, vol. 11, pp. 64442-64457, 2023, doi: 10.1109/ACCESS.2023.3289220.
8. Gad, G.; Fadlullah, Z. Federated Learning via Augmented Knowledge Distillation for Heterogenous Deep Human Activity Recognition Systems. Sensors 2023, 23, 6.https://doi.org/10.3390/s23010006
9. Shen, Q.; Feng, H.; Song, R.; Song, D.; Xu, H. Federated Meta- Learning with Attention for Diversity-Aware Human Activity Recognition. Sensors 2023, 23,1083. https://doi.org/10.3390/s23031083
10. Khan, A.R.; Manzoor, H.U.; Ayaz, F.; Imran, M.A.; Zoha, A. A Privacy and Energy-Aware Federated Framework for Human Activity Recognition. Sensors2023, 23, 9339. https://doi.org/10.3390/s23239339