Federated Learning With IoT Devices

A Project Work Synopsis

Submitted in partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE and ENGINEERING - INTERNET of THINGS

Submitted by :

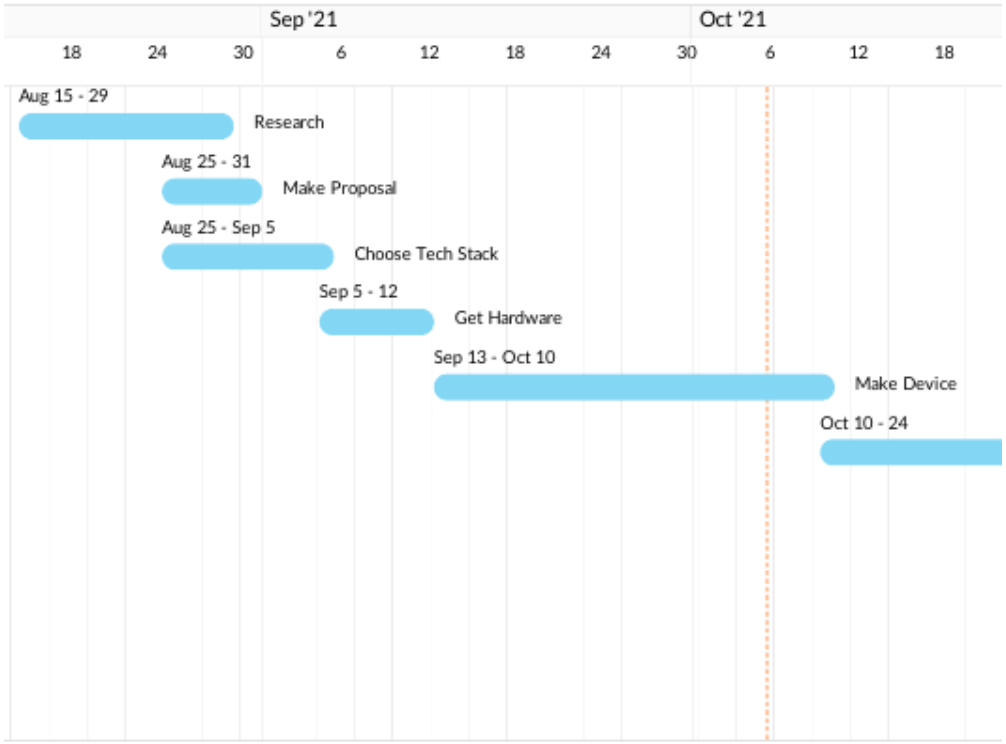
Rishabh Anand	Udita Mitra	Vandana Chauhan	Khushwant Rathore	Abhishek Gupta
19BCS4525	19BCS4662	19BCS4532	19BCS4644	19BCS4579

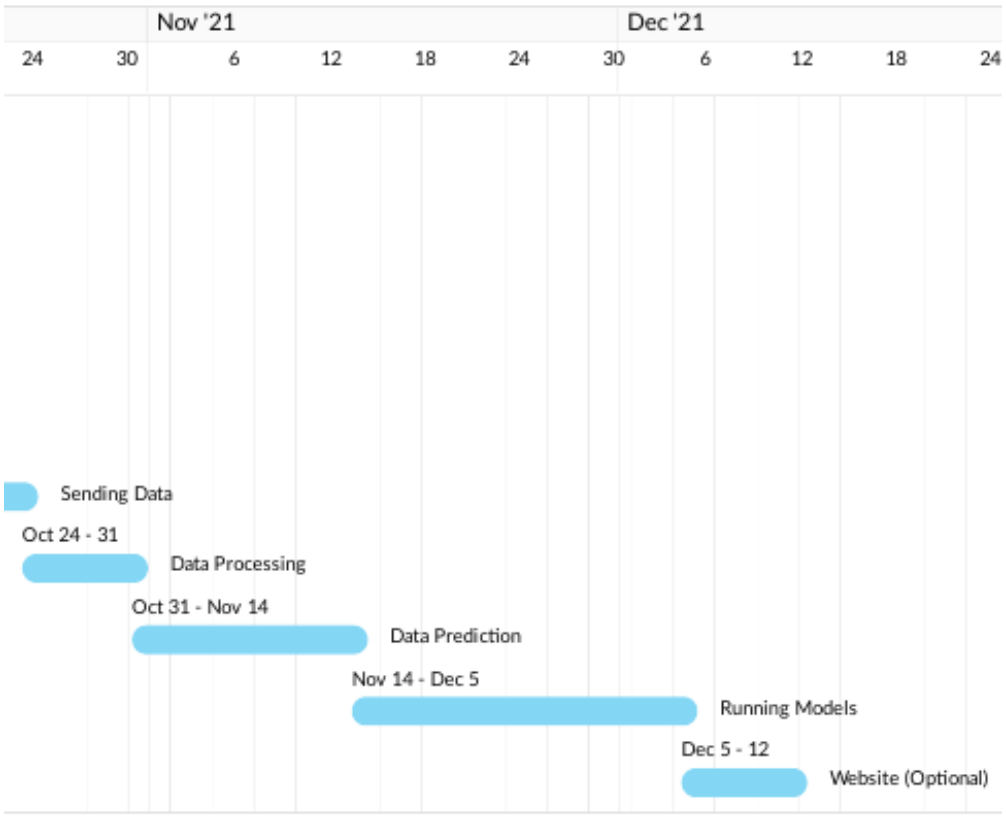
Under the Supervision of :
Piyush Samanth

CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413
Punjab

August, 2021

Timeline


Sep '21
Oct '21
18
24
30
6
12
18
24
30
6
12
18
Aug 15 - 29
Research
Aug 25 - 31
Make Proposal
Aug 25 - Sep 5
Choose Tech Stack
Sep 5 - 12
Get Hardware
Sep 13 - Oct 10
Make Device
Oct 10 - 24


Nov '21
Dec '21
24
30
6
12
18
24
30
6
12
18
24
Sending Data
Oct 24 - 31
Data Processing
Oct 31 - Nov 14
Data Prediction
Nov 14 - Dec 5
Running Models
Dec 5 - 12
Website (Optional)

List of Figures

List of Tables

Contents

INTRODUCTION

The rapid development of Internet of Things (IoT) systems has led to the problem ofmanaging and analyzing the large volumes of data that they generate. Traditional approaches that involve collection of data from IoT devices into one centralized repository for further analysis are notalways applicable due to the large amount of collected data, the use of communication channels with limited bandwidth, security and privacy requirements, etc.

Federated learning (FL) is an emerging approach that allows one to analyze data directly on data sources and to federate the results of eachanalysis to yield a result as traditional centralized data processing. FL is being actively developed,and currently, there are several open-source frameworks that implement it.

1.1 Project Definition

This project presents a demonstration of Federated Learning after an extensive and comparative review and analysis of the existing open-source FL frameworks, including their applicability in IoT systems.

We evaluated the following features of the frameworks:

- Ease of use and deployment
- Development
- Analysis capabilities
- Accuracy
- Performance

The data set was prepared in-house using MAX30100/02/05 sensors. To model low-power IoT devices, computing nodes with small resources were defined inthe test bed. The research results revealed FL frameworks that could be applied in the IoT systems now, but with certain restrictions on their use.

1.2 Project Overview

In the world of all-things-smart, everything is being run on data. Anything that we see is data and everything that we use either generates or uses data. Data can comprise of anything, ranging from the weather details of your city to your personal health details. The data generated, may contain sensitive information about an individual or even an organization. If the owner has to share there data with various other groups of people for various reasons like analysis. A link of the data is then made available to the person of choice in encrpyted form.

Looking at this from a developers perspective, we may need data from different sources to fully complete our analysis and give meaningful results. But since we don't have the data at one place, it becomes a really difficult task to use any particular mode of training. Well any mode other than FEDERATED LEARNING. Federated Learning is a very good way to use sensitive data from different parties who are not willing to disclose their exact data.

In our case, we are using a custom built oximeter to generate data and then analysis the Heart Rate and SPO2 of different individuals and then predict things like who has a higher chance of getting a heart attack and who is running low on SPO2. Now, people may not want to share their heart rate information with general public, so instead they hash their reading before passing the information. We then aggregate that data onto our process and then, predict the desired information and keep the private information private.

1.3 Hardware Specification

1.3.1 ESP32

ESP32 is a series of low-cost, low-power system on a chip microcontrollers with integrated Wi-Fi and dual-mode Bluetooth. The ESP32 series employs either a Tensilica Xtensa LX6 microprocessor in both dual-core and single-core variations, Xtensa LX7 dual-core microprocessor or a single-core RISC-V microprocessor and includes built-in antenna switches, RF balun, power amplifier, low-noise receive amplifier, filters, and power-management modules. ESP32 is created and developed by Espressif Systems, a Shanghai-based Chinese company, and is manufactured by TSMC using their 40 nm process.

The esp32 chip used in our expriments and project had :

- 2 CPU core(s)
- WiFi/BT/BLE
- Silicon revision 1
- 2MB external flash
- Minimum free heap size: 294448 bytes

The ESP32 Board can be used in conjuction of Arduino or as a stand- alone board. It can be programmed by using Arduino-IDE or by using Espressif-IDF given by it's makers.

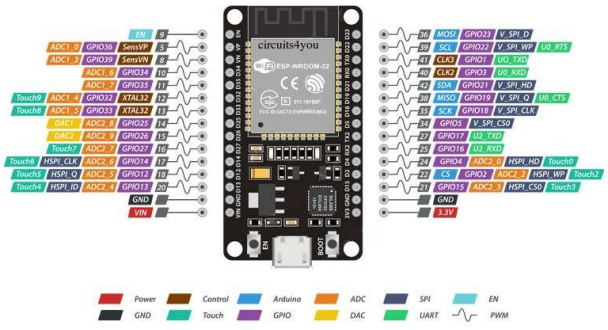


Figure 1: ESP32

1.3.2 MAX30100/02/05

The MAX30102 is an integrated pulse oximetry and heart-rate monitor biosensor module based on PPG ((PhotoPlethysmoGraphy). It is so small that you can just wear it on your finger or wrist for data collecting. Internally integrated 18bit ADC, the sensor supports I2C data output, which could be compatible for most controllers.

Features :

- Extremely low standby current
- High sampling rate
- High SNR

Application :

- Heart-rate Measurement
- SPO2 Detection

Specification :

- Power Supply: 3.3V 5V
- Working Current: <5mA
- RED/IR LED Driving Current: 0-50mA
- Communication: I2C
- I2C Address: 0x57
- Operating Temperature: -40℃ 85℃
- Dimension: 18×14mm/0.71×0.55”

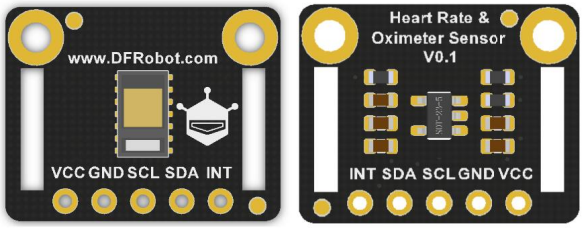


Figure 2: MAX30100/02/05

1.4 Software Specification

Tools needed to work with ESP Boards :

- Toolchain to compile code for ESP32 (gcc git make flex bison gperf python-pip)
- Build tools to build full application (cmake ninja ccache dfu-util libusb)
- ESP-IDF

1.4.1 Espressif IDF

ESP-IDF contains the API, software libraries and source code, for ESP32 and scripts to operate the Toolchain.

This can be obtained from espressif's repositories on git. We also need some tools that the idf needs to work. There is an installation script that does all that for us. We then need to set all the installed tools in PATH env variable.

With proper permission to access the PORTS, we can also run the build, flash, monitor and see the results, all straight from the terminal.

LITERATURE SURVEY

2.1 Existing System

Title	Comparision	Source	Findings
Distributed Learning	Different Architechture More Secure	Online Papers Custom Tests	Privacy can be increased Data Security can be better

Table 1: Literature Review summary

2.2 Proposed System

FL as a distributed machine learning paradigm that supports data analysis, such neural network (NN) training, directly on the data storage,only the results of such processing. There are three major components in an FL system:

1. Server (e.g., manager).
2. Communication–computation framework.
3. Clients (e.g., parties, data sources).

FL can treat IID data as non - IID data because training can be performed on sources connected with each other, i.e., sources that store different types of data about the same artifacts or events, or independent data sources. FL uses nodes as data sources and performs calculations as close to the data as possible.

The FLDP framework is a very simple open-source framework for FL, which is distributed under the Apache 2.0 license. The framework is developed by Sherpa. This framework implements several aggregation algorithms for different models:

- A FedAvg aggregator for NN and LR.
- A weighted FedAverage aggregator and IOWA FedAverage aggregator for LR.
- A cluster FedAverage aggregator for centroid cluster models.

The framework implements these privacy mechanisms to protect personal data:

- A simple mechanism adding random noise to binary data.
- An adaptive differential privacy mechanism based on Privacy Filters.

2.2.1 Comparision of Various Framworks

Features	TFF 0.17.0	FATE 1.4.4/1.5	PySyft 0.2.8	PFL 1.1.0	FL&DP 0.1.0
OS	Mac Linux	Mac Linux	Mac Linux Win iOS Android	Mac Linux Win	Linux Win
Settings	Cross-silo	Cross-silo	Cross-silo Cross-devices	Cross-silo Cross-devices (in future)	Cross-silo
Data Partitioning	Horizontal	Horizontal Vertical	Horizontal Vertical	Horizontal Vertical Transfer	Horizontal
Data type	Time series Images	Time series	Images	Time series Images	Time series Images
Mode	Simulation	Simulation Federated	Simulation Federated	Simulation Federated	Simulation
Clustering model	No	No	No	No	Yes (Kmeans SciKitLearn)
ML Model	No	Yes (very slow)	No	Yes	Yes (SciKitLearn)
Decision Tree Model	No	Yes (very slow)	No	No	No
Protocol	gRPC/proto (in future)	gRPC/proto	Doesn't use	ZeroMQ	Doesn't use

Figure 3: Comparision of Frameworks

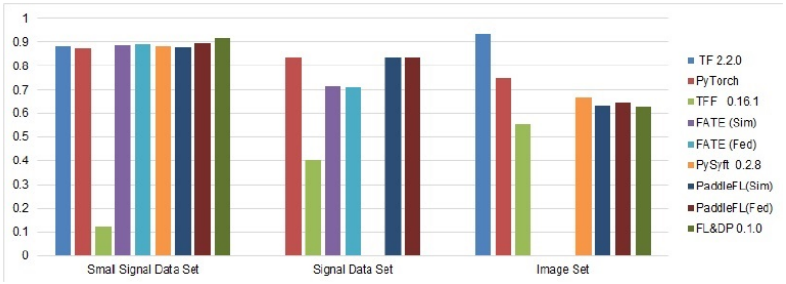


Figure 4: Accuracy

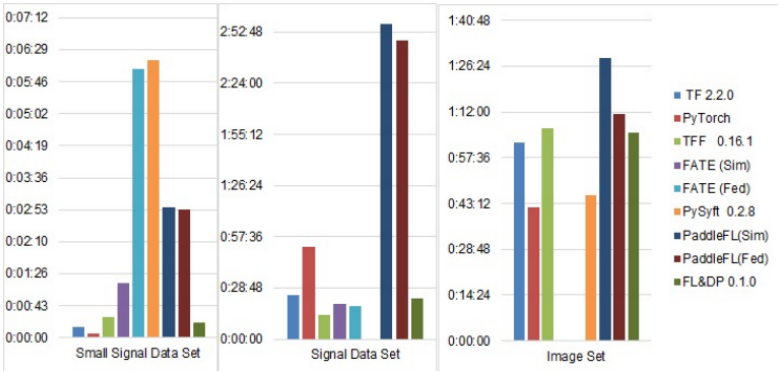


Figure 5: Training Time

PROBLEM FORMULATION

The main objective of this paper is to design not only a heart rate monitoring system but also to made a Federated System. Federated learning (FL) is a feasible solution to solve the problems of data islands, break data barriers, and protect data security and privacy, especially in the context of the Internet of Things , and big data.

Distributed IOT and big data users need to collaboratively train a classification or regression model to implement perfect data prediction results without compromising privacy. Unlike privacy-preserving outsourced training, rather than submitting data to the centralized cloud server, users train data locally in FL. The federated center is only responsible for aggregating the gradient information (or model parameters) uploaded by users and distributing the global training model.

RESEARCH OBJECTIVES

Firstly, we have to understand how federated learning (FD) is different from distributed learning.

- The goal of distributed learning is to scale the parallel processing of a large amount of data, while the purpose of FL is to process data, i.e., train a model, directly on the data sources.
- Distributed learning works with identically and independently distributed (IID) data, which are collected in a single repository, from which they are extracted for further training. FL can treat IID data as non-IID data because training can be performed on sources connected with each other, i.e., sources that store different types of data about the same artifacts or events, or independent data sources.
- Another difference is the usage of network nodes. Distributed learning uses network nodes as computing resources for scaling, while FL uses nodes as data sources and performs calculations as close to the data as possible.

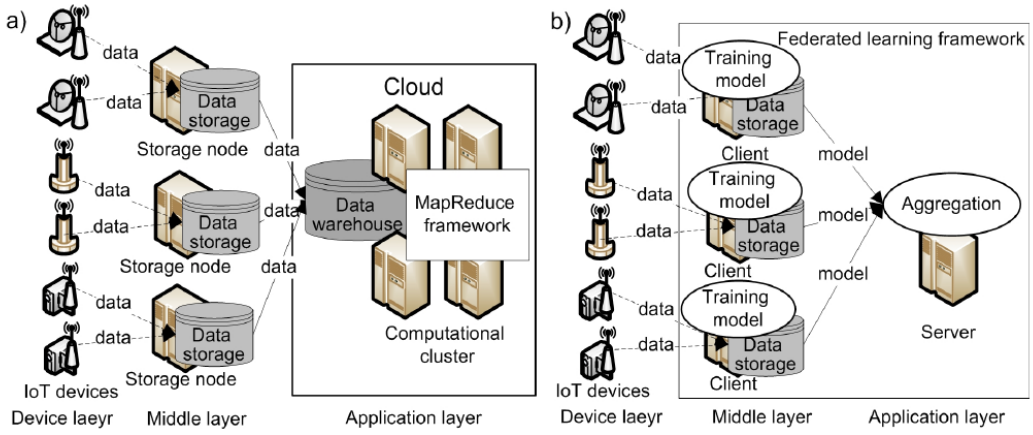


Figure 6: Distribued Learning vs Federated Learning

So, by this we can conclude that distributed learning focuses on scalable parallelized big data processing whereas, federated learning (FD) mainly focuses on processing distributed data on heterogeneous data sources.

4.1 The Steps in Federated Learning

- Identifying a problem to be solved
- Modifying the client's application (optional)
- Simulating prototyping (optional)
- Training the federated model
- Evaluating the federated model
- Deploying FL at the server and clients. Thus, FL allows decreasing of:
  - The risk of unauthorized data access, since data are not transmitted over the network
  - Network traffic because the training results are usually much smaller in volume than the data themselves
  - Time and cost of information transfer by reducing the amount of data transmitted
- Requirements to the central computational cluster and the central storage, as there is no need to store all data in one place. At the same time, to implement FL, the following challenges must be solved:
  - Processing IID data as non-IID data, which can have different data partitions;
  - Working with clients with different computing and storage capacity, as well as scale and stability;
  - Implementation of different communication schemes: centralized and decentralized;
  - Protection of transmitted analysis results from various types of attacks;
  - Aggregation of the results obtained from data sources to calculate inequality.

4.2 Challanges of Federated Learning:

Data Partitioning:

There are two different cases of how data are distributed in the IoT system.

- Vertical partitioning: In this each storage node collects and stores data about different features of all objects.
- Horizontal partitioning: In this each storage node collects and stores data about all features of different objects.

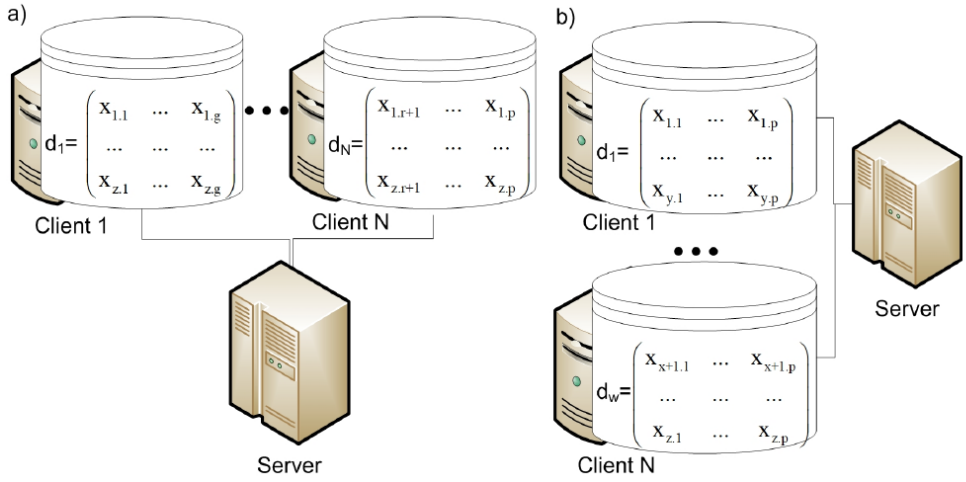


Figure 7: Horizontal vs Vertical

Clients' Settings:

There are two different types of FL systems depending on the scale of federation.

- Cross-silo systems have low scalable federation. They include organizations or data centers. Their numbers are small and rarely change.
- Cross-device systems have a scalable number of clients. They can be added and disabled at any moment of time. These are usually mobile devices and IoT devices.

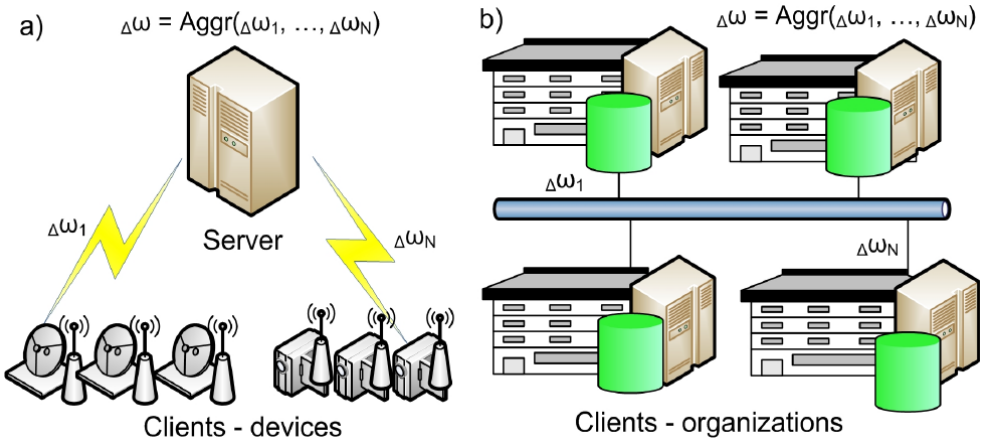


Figure 8: Cross-Silo vs Cross-Device

Communication Schemes:

FL systems can implement two communication schemes between their components centralized and decentralized.

- The centralized scheme includes a central server. It is used to orchestrate different steps of the FL process and coordinate all clients therein. This scheme is typical for cross device systems. Since all selected nodes must send updates to a single entity, the server may become a bottleneck of the system.
- In the decentralized scheme, all clients can coordinate themselves to obtain the global model. This scheme is often used in cross-silo systems where clients have high-performance resources. Nevertheless, the specific network topology may affect the performance of the learning process.

Data Privacy and Security Mechanisms:

Data privacy and security are essential properties of FL. There is a need to secure the models and the analysis process to provide meaningful privacy guarantees. The attacks can be performed at all stages of the FL process and can target all FL elements.

- Trained model
- Client
- Server

It is possible to outline two main types of FL-specific attacks—poisoning and inference attacks. The first type of attack aims to modify either the input data set or the parameters of the trained model in order to bias it in a way that is preferable to the adversary. The goal of inference attacks is to get access to personal or confidential data. Depending on the attack implementation mechanisms, it is possible to derive information about the properties of training data or the labels of training samples, or to determine if the sample was used in the training process. To protect FL against these attacks, the following security mechanisms are suggested:

Secure multi-party computation (MPC) is a family of cryptographic protocols that allow a set of users to perform computations that use some private inputs without revealing them to other participants of the protocol. The most widely used implementations of MPC are the ABY3 and SecureML protocols. These protocols implement a server–aided computational model in which data owners send their data in encrypted format to a number of servers, usually two or three, which perform model training or apply a pre-trained model to analyze input data.

Aggregation Algorithms

One of the key issues of FL is aggregating model changes made by clients into a single model, as the aggregation function should not impair the accuracy of the model. The aggregation function depends on the model built in the FL process. For example, for centroid clusters constructed by the K-means algorithm, a prefix frequency filtering (PFF) method for data aggregation is suggested.

To support energy and computationally efficient data aggregation of similar data sets, the authors applied the K-means clustering algorithm on the data and then aggregated the generated clusters using PFF. The ordered weighted averaging (OWA) algorithm was one of the first proposed for aggregating the weight coefficients of NNs.

The aggregation of the parameters is based on the amount of data in every node. When calculating a regular average, each data point has an equal "weight", i.e., it contributes equally to the final value. Weighted averages, on the other hand, weight each data point differently. Yager and Filev suggested a generalization of the OWA operator called the induced ordered weighted averaging (IOWA) operator. This operator firstly induces the ordering of arguments before their aggregation.

METHODOLOGY

The following methodology will be followed to achieve the objectives defined for the proposed research work:

1. Detailed study of Federated Learning frameworks for IoT will be done.
2. Data will be generated in-house using various sensors and custom built devices for the training and simulation of FL.
3. An in-depth study and Hands-on experience on existing approaches of Federated Learning will be done, to identify the Relative pros and cons for developing an efficient system
4. Various Hardware and software-related parameters will be identified to evaluate the proposed system.
5. Data security and user privacy will be maintained.
6. Comparison of our newly implemented approach with exsiting approaches will be done.
7. Testing of the project by applying different conditions on it, so remove the remaining minorities in the project and to distinguish it in a better way.

TENTATIVE CHAPTER PLAN

CHAPTER 1: INTRODUCTION

This chapter introduces the reader to Federated learning and the basics of the project.

CHAPTER 2: LITERATURE SURVEY

This chapter includes the research already available for Applying Federated learning. The findings of the researchers will be highlighted which will become the basis of current implementation. And the existing and current approaches are compared.

CHAPTER 3: PROBLEM FORULATION

This chapter covers the basic problem formulations and defines the solution as provided by the paper.

CHAPTER 4: RESEARCH OBJECTIVES

This chapter covers the main differences between distributed learning and federated learning. Also, it includes the steps and the challenges of federated learning, how we can use it and how we can overcome the challenges.

CHAPTER 5: METHODOLOGY

This chapter covers the technical details of the proposed approach.

REFERENCES

This Section contains the references to other documents that are either useful or are directly reffered in this article.

REFERENCES

1. Santucci, G. From internet of data to internet of things. In Proceedings of the International Conference on Future Trends of the Internet, Luxembourg, 28 January 2009.

2. Tsai, C.W.; Lai, C.F.; Vasilakos, A.V. Future Internet of Things: Open Issues and Challenges. Wirel. Netw. 2014, 20, 2201–2217.

3. Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. Comput. Netw. 2010, 54, 2787–2805.

4. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Gener. Comput. Syst. 2013, 29, 1645–1660. [CrossRef]

5. Voigt, P.; Von dem Bussche, A. The EU general data protection regulation (GDPR). In A Practical Guide, 1st ed.; Springer International Publishing: Cham, Switzerland, 2017.

6. California Consumer Privacy Act Home Page.

7. Personal Data Protection Act 2012.

8. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; d'Oliveira, R.G. Advances and open problems in federated learning. arXiv 2019, arXiv:1912.04977.

9. Li, Q.; Wen, Z.; Wu, Z.; Hu, S.; Wang, N.; He, B. A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. arXiv 2019, arXiv:1907.09693.

10. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership Inference Attacks Against Machine Learning Models. In Proceedings of the IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–24 May 2017; pp. 3–18. [CrossRef]

11. Sun, G.; Cong, Y.; Dong, J.; Wang, Q.; Liu, J. Data Poisoning Attacks on Federated Machine Learning. arXiv 2020, arXiv:2004.10020.

12. TensorFlow Federated: Machine Learning on Decentralized Data.

13. Laulkar R,Daimiwal N2015 Application of Finger Photoplethysmography International Journal of Engineering Research and application volume 2., Issue 1,Jan-Feb 2015,pp.887-880

14. Wei M,Chang R,Wang C, Lin C, Chen H 2016 Design of a Flexible PPG signal Process Wireless Device International Conference on Consumer Electronics-Taiwan

15. Mohan P,Nagarjan V, Nisha A 2017 A frame work to estimate Heart Rate and Arterial Oxygen Saturation (Spo2) International Conference on Communication and Signal Processing April 6-8 2017

16. Xie Y,Gao Y,Lu W,Li W 2017 Development of Wearable Pulse Oximeter Based on Internet of Things and Signal Processing Techniques European Modelling Symposium 2017