TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

A

PROJECT PROPOSAL

ON

PEER-AI : A DISTRIBUTED DATA TRAINING FRAMEWORK

SUBMITTED BY:

AAVASH CHHETRI (PUL077BCT004)

KUSHAL PAUDEL (PUL077BCT039)

MUKTI SUBEDI (PUL077BCT048)

SUBMITTED TO:

DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING

December, 2023

Acknowledgments

We would like to express our heartfelt gratitude to the **Department of Electronics and Computer Engineering** at Pulchowk Campus for providing us with the invaluable opportunity to undertake our minor project. The unwavering support, guidance, and encouragement from the department have been instrumental in shaping the trajectory of our project.

Special appreciation goes to **Assistant Prof. Dr. Babu R Dawadi** for his inspiring words, encouragement and guidance. His belief in our potential has been a significant motivator, and we are truly thankful for his guidance throughout this endeavor.

The department's thorough review of our concept note has provided us with invaluable insights, significantly enhancing our perspective and refining our approach to the project. We are confident that with the continued support from the Department of Electronics and Computer Engineering and the guidance of Assistant Prof. Dr. Babu R Dawadi, our project will reach new heights of success.

Thank you to the department and Assistant Prof. Dr. Babu R Dawadi for their pivotal roles in the early stages of our project. We eagerly look forward to the continued collaboration and the discoveries that lie ahead.

Abstract

The traditional process of training Machine Learning(ML) models involves a centralized approach where all the data is collected, stored and processed in a central location. While this approach has been effective to some extent, the data privacy issues and the computational load limitations poses some serious challenges. New approaches like Federated Learning(FL) has been proposed to address the above concern but they fail to incorporate the rising 5G and 5GB technologies. Therefore, we propose Peer-AI, a purely distributed training Framework for ML models, specializing in edge devices and harnessing the power of peer-to-peer network based on Device-to-Device (D2D) communications. The system is a peer-to-peer distributed training framework based on the Ultra Reliable Low Latency Communication (URLLC) 5G Network defined by the 3GPP standards. The system harnesses the power of Radio Access Technology (RAT) for ML training purpose.

Keywords: Distributed Machine Learning, Blockchain, 5G

Contents

List of Figures

List of Abbreviations

P2P	Peer to Peer
ML	Machine Learning
FL	Federated Learning
GPU	Graphics Processing Unit
DNN	Deep Neural Network
SGD	Stochastic Gradient Descent
FedSGD	Federated Stochastic Gradient Descent
AdaGrad	Adaptive Gradient Algorithm
FedAvg	Federated Averaging
SSS	Shamir's Secret Sharing
PoF	Proof of Federation
4G	4^{th} Generation
5G	5^{th} Generation
5GB	Beyong 5G
D2D	Device-to-Device
URLLC	Ultra Reliable Low Latency Communications
RAT	Radio Access Technology
eNB	Evolved Node B
CC	Cellular Communications
3GPP	3^{rd} Generation Partnership Project
eMBB	Enhanced Mobile Broadband
RAN	Radio Access Network
BS	Base Station

1. Introduction

Peer-AI is a Distributed Training Framework for ML models. This framework leverages the power of peer-to-peer (P2P) networks to distribute the computational load, enabling efficient, secure, and collaborative model training across a distributed network of participants in the P2P Network. Our primary focus is training ML models in the P2P network based on the D2D communications based on URLLC and Enhanced Mobile Broadband(eMBB) 5G Networks specializing in edge devices.

1.1 Background

Distributed ML involves training machine learning models across multiple devices or servers rather than on a single centralized system. This can be achieved through various techniques like federated learning, ensemble learning, and parameter server architectures. It allows for the parallel processing of data, enabling the handling of large datasets and complex models, training models without exchanging raw data, preserving privacy, and are often more robust to failures in individual components.

5G is the fifth generation of mobile networks, providing faster data transfer speeds, lower latency, increased device connectivity, and improved network reliability compared to its predecessors (3G and 4G). 5G networks offer significantly higher data rates, enabling faster communication between devices, reduced latency allowing for near real-time communication. This technology provides massive device connectivity facilitating the Internet of Things(IoT) and smartphones. One of the essential part of 5G is D2D communication, which we aim to harness for establishing the P2P Network.

1.2 Problem statements

Some of the problems with the traditional approach:

1. Computation Power Bottlenecks: In a centralized setup, the processing of vast amounts of data on a single server or data center can lead to performance bottlenecks and longer training times.

2. Data Privacy Concerns: Centralizing data from various sources raises privacy concerns, as sensitive information is concentrated in one location.

3. Limited Data Diversity: Centralized training relies heavily on the data available within the central repository. This can result in biased or incomplete models if the dataset

doesn't adequately represent the full range of scenarios or demographics.

4. Access to Diverse Datasets: Smaller organizations or individuals might not have access to large and diverse datasets due to limitations in data acquisition or data-sharing agreements.

1.3 Objectives

Our objective is to try to overcome the drawbacks of the traditional system of model training by building a peer-to-peer network solution comprising of the following main features:

1. Peer-to-Peer Distributed Training: The framework facilitates distributed training by allowing participants to contribute their computational resources for model training.

2. Private Data Training: Leakage of private data will be prevented using aggregation and noise addition to modal updates.

3. Reduced Latency: The use of 5G technology in the project will ensure low latency and high bandwidth, increasing the throughput of the system.

1.4 Scope

The scope of leveraging 5G D2D communication for P2P networks in Distributed ML training is broad and encompasses diverse aspects related to applications, network architectures, device characteristics, security, scalability, integration, and adaptability to changing technological landscapes. It aims to provide a comprehensive and efficient solution for collaborative ML in decentralized environments. Some of the possible scopes of the project include:

1. Government and Public Services: Government agencies often deal with sensitive data for decision making, policy formulation, and disaster response. A decentralized framework ensures that critical data remains localized, reducing the risk of centralized data breaches. Different departments can collaborate while maintaining data security and sovereignty with the ease of their edge devices.

2. Healthcare and Medical Research: In healthcare, patient data privacy is paramount. The increasing use of personal health monitoring device enables the collection of critical data. A decentralized approach to ML training in 5G based network allows medical institutions and researchers to harness the critical medical information without the fear of data leakage and privacy breach. This fosters collaboration in disease analysis, drug discovery, and personalized medicine.

3. Military and Defense: The military sector involves classified and sensitive information. A decentralized system ensures that data is shared only among authorized participants, enhancing security and preventing unauthorized access to critical military data.

4. Financial Services: The financial industry handles vast amounts of personal and transaction data. Decentralization can enhance security, reduce the risk of data breaches, and enable efficient fraud detection and risk assessment.

5. Research and Academia: Research institutions can collaborate on complex problems without sharing raw data. A decentralized framework encourages cross-institutional research, enabling diverse datasets to be used for training models.

2. Literature Review

A number of research has been conducted relating to Multi Party Machine Learning (ML). Distributed learning involves training machine learning models across multiple devices or nodes, allowing for parallel processing and collaboration. Instead of centralizing all data and computation in a single location, distributed learning distributes the workload across a network of devices, which can include servers, edge devices, and even individual user devices. By distributing the training process, organizations can achieve better scalability, faster model training, and more comprehensive insights from diverse data sources.

2.1 Related theory

1. Muhammad Shayan et al. in their paper *Biscotti: A Blockchain System for Private and Secure Federated Learning*[1] have proposed a fully decentralized peer-to-peer(P2P) approach to multiparty ML using blockchain and cryptographic primitives. It is claimed to be scalable, fault tolerant and can perform even with 30% adversaries in the system. But the system is vulnerable to attacks that exploit privacy leakage from the model and has stake limitations, as it is dependent on the higher stake parties as noiser, verifier and aggregator.

2. *BrainTorrent: A Peer-to-Peer Environment for Decentralized Federated Learning* [2] by Abhijit Guha Roy et al. introduces BrainTorrent, a new Federated Learning(FL) framework without a central server offering a highly dynamic P2P environment and demonstrated that the proposed system outperformed the traditional server-based ones. It is particularly targeted towards medical applications.

3. Tian Wang et al.'s *Edge-Based Communication Optimization for Distributed Federated Learning* [3] provides an study of optimization framework to reduce the number of end devices directly connected to the parameter server while avoiding unnecessary local updates. Latency associated with high server concurrency is avoided by clustering of devices in same network location and deployment of mobile edge nodes in different network location. Experimental results show 60% reduction in number of local updates and 10.3% increase in convergence speed of the evaluated model.

4. An indepth and inbreadth investigation on FL can be found in *A Survey on Federated Learning: The Journey from Centralized to Distributed On-Site Learning and Beyond*

[4] by Sawsan AbdulRahman et al. The paper examines and compares different ML-based deployment architectures and provides a new classification of FL topics and research fields based on through analysis of main technical challenges and current related work.

5. P2P applications can be incorporated in D2D communication which has been considered as potential technology in 4G(4^{th} Generation) and 5G(5^{th} Generation) cellular network. In D2D communications, two user's equipments directly communicate with each other without the involvement of eNB(Evolved Node B). This can reduce the burden of eNB and offload the workload. This is talked about by Mahendra Singh et al. in their paper *P2P Applications in 4G/5G Networks using D2D Communication Based on Social Attributes of Users* [5].

6. Iunius is a P2P system which optimises D2D communications for P2P local file sharing proposed by Yue Wu et al. in *Iunius: A Cross Layer Peer-to-peer System with Device-to-device Communications*[6]. They propose a 4 cross layered system for P2P and D2D communications with interference cancellation technique and radio resource management schemes for devices with Cellular Communication(CC) capabilities. Their simulation results show Iunius can increase the cellular efficiency, reduce traffic loads of Base Stations and improve data rate and energy saving for mobile users.

2.2 Related work

1. peer-nnet from github user *tdarnett* is an implementation of the BrainTorrent framework. The project establishes a P2P network in golang and uses pytorch to create ML training model. The model uses MNIST image data as training set.

2. Biscotti is a ledger for private and secure P2P ML. It uses a blockchain based ledger for storing the model weights after each update. Centralized anomaly detection, differential privacy and secure aggregation is used to prevent poisoning and information leakage attacks. Multi Krum is used for verification of modal updates which is a Byzantine-tolerant aggregation scheme. A Proof of Federation(PoF) scheme is used to ensure that malicious clients cannot take part in the process without substantial contribution to the model.

3. Proposed Methodology

The project is a peer-to-peer (P2P) network based distributed system specializing for distributed ML Training. Several considerations should be done to ensure a proper fault tolerant, consistent and available system while maintaining the privacy of data and accuracy of model. Here's a proposed methodology for building the P2P ML system:

3.1 Problem Definition

The challenge is to optimize the training time of a neural network, considering the varying computational resources of training machines. The training time is influenced by factors such as the presence of GPUs, the availability of multi-core processors, the size of the dataset, and the parallelizability of the algorithm. The goal is to develop strategies or algorithms that can adapt to different computational environments, making neural network training more efficient and scalable across diverse hardware configurations. One of the most promising architecture seems to be FL but it is has privacy and security concerns, as FL models are prone to attacks introduced by compromised central server or compromised clients.[7] The project aims to reduce the communication bottleneck in the current system while considering the security and privacy challenges. We aim to develop a framework while using MNIST dataset for the proof of concept of the project.

3.2 Network Architecture

A P2P network with Low Latency and High Bandwidth Communication is to be established as the network infrastructure of the project. The P2P network is supposed to distribute the computational load across multiple nodes enabling more scalable and distributed machine learning. The establishment of P2P network ensures fault tolerance, proper resource utilization, scalibility and reduced communication overhead. The P2P network will be based on the D2D communication Network of the 3GPP 5G Standards.

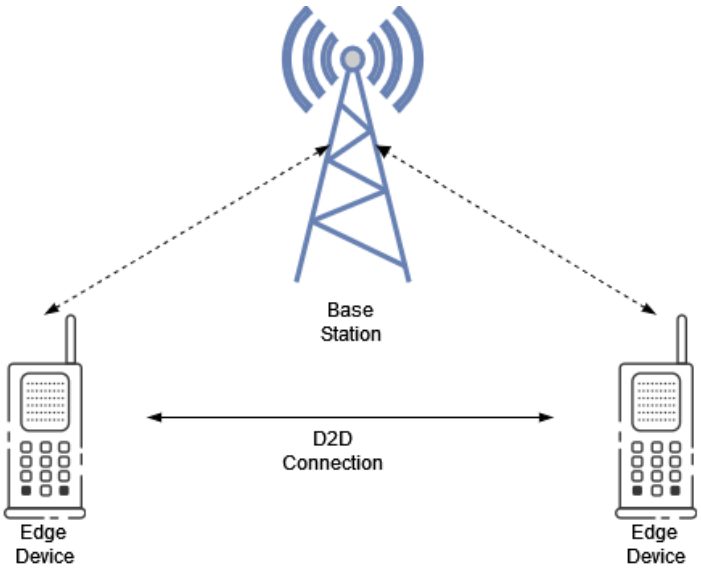

Figure 3.1: D2D Network Architecture

D2D communication allows devices to communicate directly with each other when they are in close proximity, without routing the traffic through a centralized Base Station (BS). This feature of D2D communication favourable for P2P network establishment. Thus leveraging on this feature of 3GPP standard 5G Network's D2D communication, we aim to establish a P2P network based on the same. This will ensure the reliability, low latency and high bandwidth of the P2P network, with peers in close range of proximity.

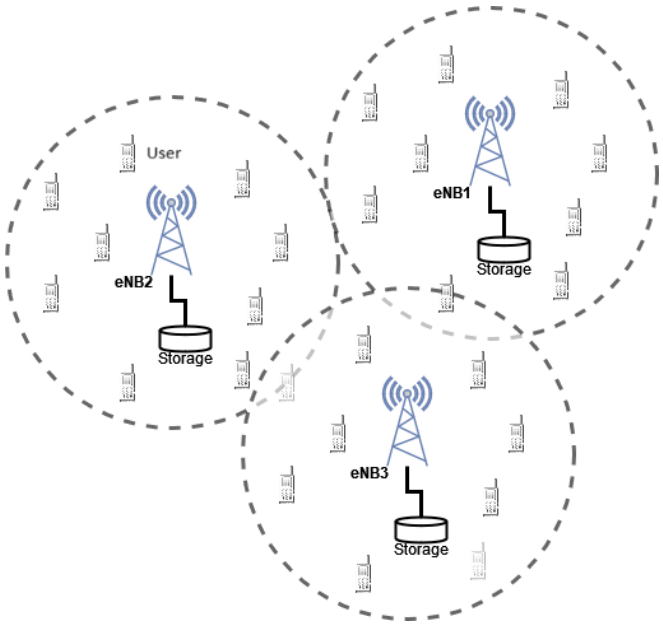

Figure 3.2: 5G Network for multiple Base Stations

For peers not within close range of proximity, the Enhanced Mobile Broadband (eMBB)

feature of 5G Radio Access Network(RAN) helps the peers in the P2P network for seamless communication.

3.3 Training Algorithm

We plan to use Deep Neural Network(DNN) as the model for distributed training. For the Optimization Algorithm, we plan to implement Stochastic Gradient Descent(SGD), Federated SGD(FedSGD), FedSGD + Adam and FedSGD + Adaptive Gradient Algorithm(AdaGrad), and benchmark the above optimization algorithms for distributed ML. For Aggregation of the local models, we plan to implement Secure Aggregation, Federated Averaging(FedAvg) and Federated Matched Averaging. Also, the above algorithms are to be bench marked.

For Privacy Preservation of the private data of each node, privacy preserving technique like Differential Privacy and for protection against Byzantine Attack algorithm like Multi-Krum Defence is planned to be implemented for verification process.(See Appendix A for details)

For sharing the optimized and verified weights for aggregation, Shamir's Secret Sharing(SSS) is planned to be used. SSS is used to share the distributed private information, here local weights, for aggregation to the aggregating nodes. (See Appendix B for details)

3.4 Training in the P2P Network

After the establishment of P2P network, the initiator distributes the DNN model among the training peers. The training peers are responsible for optimizing the DNN model by training it with the locally available private data. Meanwhile, the noising peers sends verifiable noise within the threshold to the training peers. The noise is incorporated with the training process to maintaining the data privacy while training. Then the local model weights are sent to the pool of verifying peers for verification. The verifying peers verifies the model updates and rejects the malicious updates, and sends the updates with signature. Then the signed local weight updates are sent to aggregating peers for aggregating. The aggregating peers are responsible for aggregating the local models and updating the global aggregated model.

3.4.1 Verification

Verification is the process of verifying whether the updated model weights are legitimate or malicious. A pool of peers are responsible for determining the legitimacy of the model updates. After the training peers updates the local model weights, it is sent to the verification pool of peers. The verification pool of peers use technique like Multi-Krum Defence to identify if the update in model weight is legitimate or malicious. After determining the legitimacy of the updated model weights, the verification peers signs the model weights.

The verified signature is returned to the training peers which signifies the verification of the model weights. It is then sent for Aggregation. Verification process is necessary to defend against different attacks like Byzantine Attack.

3.4.2 Aggregation

After the training and updating the model weights, the model weights are verified. The malicious weights are neglected and the legitimate updates are signed. The signed model updates are then sent to the pool of aggregating peers for aggregation of multiple local models into a single global model. Aggregating Algorithms like Secure Aggregation, FedAvg and so on are short listed, and the best performing among the short listed algorithms will be preferred. After the aggregation of local updates into the global model, the aggregation process is completed. The aggregating pool of peers are responsible for updating the global model present in the blockchain.

3.4.3 Blockchain

A blockchain ledger is maintained for maintaining the global model updates of the training process. The blockchain is initiated by the Initiator Node. We assume that a trusted authority facilitates and bootstraps the training process by publicly distributing the genesis block out of band to all peers in the system.

4. Proposed Experimental Setup

The experimental setup for the project is divided into two main parts viz. network setup and training process setup.

4.1 Network Setup

Since the P2P network is aimed to be built on top of 5G network, the 5G network simulation is required. For the 5G network simulation, ns-3 module for 5G RAN supporting the D2D communication is planned to be used. Apart from using simulation for establishing the P2P network, a cloud based implementation of the P2P network is planned, though not in 5G network.

4.1.1 Tools Required:

- Network Simulator 3 (ns3)
- ns3-mmWave implementation of NYU
- Cloud Service for P2P network establishment (not in 5G)

4.2 Training Process Setup

The training module is planned to be built on top of current ML module, PyTorch. A framework based on PyTorch, but distributed in nature, and optimized for 5G D2D communication is planned. For the implementation of the developed framework, a DNN model is to be trained. We plan to train a simple multi-class image classification problem, classifying the numbers ranging from 0 to 9 based on the MNIST dataset, which is considered a standard problem. For this purpose, MNIST dataset is required and a standardized model of DNN is required. This can be used for bench-marking the performance of the system.

4.2.1 Tools Required:

- PyTorch Module for Framework Development
- Standard DNN Module for evaluation of performance and bench-marking
- MNIST Dataset for evaluation of performance and bench-marking

5. Proposed System Design

The Initiator node initializes the ML Training process, while initializing the blockchain ledger simultaneously. Then, the DNN model is distributed across the pool of training peers. The training process occurs using the optimization algorithm in the DNN. Meanwhile, the Noising Peers generates random verifiable noise and is sent to the training nodes. After completing the first epoch of training, the noise is added to the weights using technique like DP, and the model is verified with the help of verifying peers. Verification is required to protect against various kind of attacks. Then, the local model updates are sent to aggregating pool of nodes using the SSS. SSS helps in securely transferring the model weights to the aggregating pool of nodes. The local model weights are then aggregated to form a global model. The global model is updated in the blockchain ledger. The weights of the global model is taken in regard by every training peer in the next epoch.

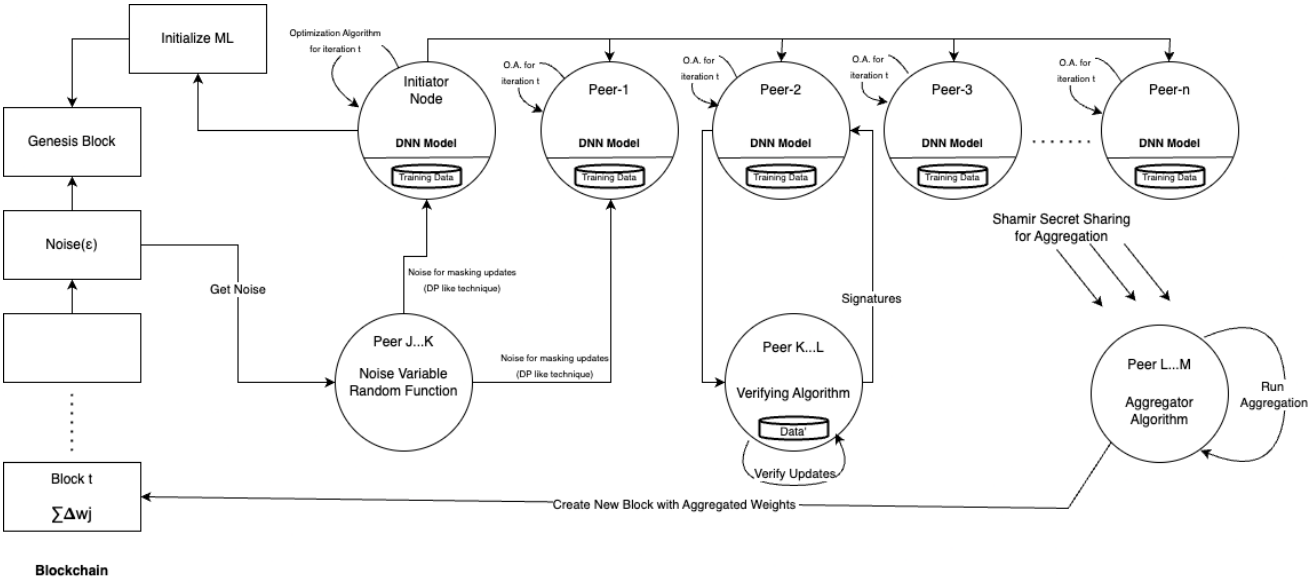

Figure 5.1: Block Diagram showing the System Overview

5.1 Nodes

5.1.1 Training Nodes

Training Nodes are responsible for optimizing the distributed DNN model from the private data they posses. They train on the distributed DNN model, using the Optimization Algorithm as provided. The data remains private, and to maintain the privacy of data, noise is added to the weights after training each epoch. After training and adding noise from the Noise Generating Nodes, the weights are then sent to verifying nodes for verification. After verification, the weights are sent for aggregation.

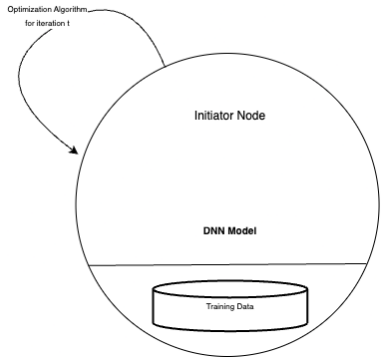

Figure 5.2: Pool of Training Nodes

5.1.2 Verifying Nodes

The verifying nodes are responsible for accepting the legitimate model updates and rejecting the malicious updates. It is conducted by the help of techniques like Multi-Krum Defence, and the accepted updates are signed. The signed updates are sent to the training nodes, where as rejected updates aren't signed.

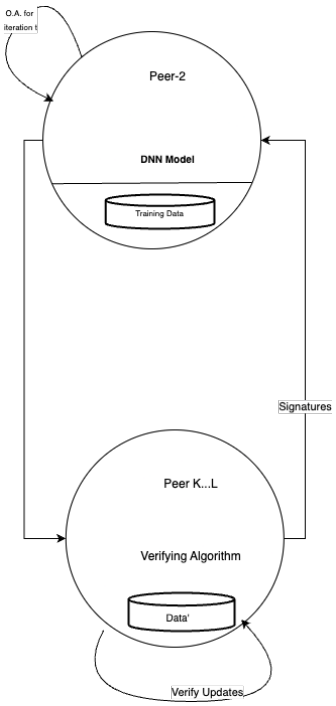

Figure 5.3: Pool of Verifying Nodes

5.1.3 Aggregating Nodes

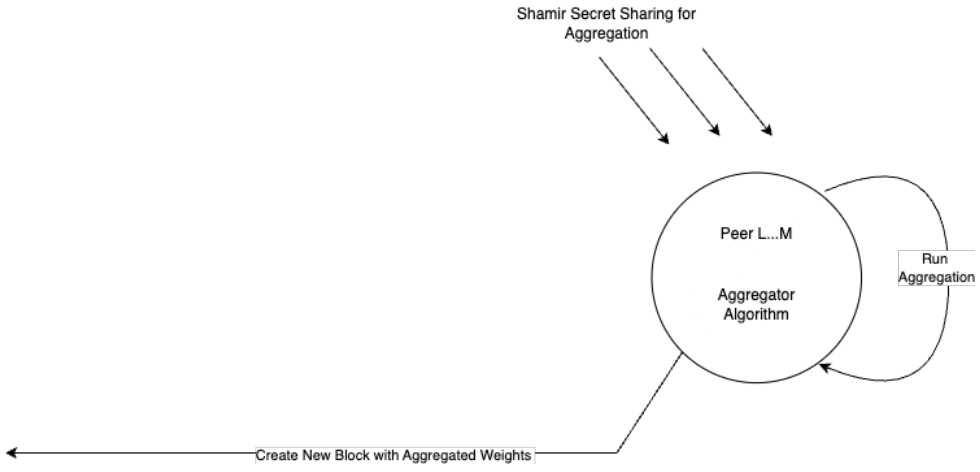

Figure 5.4: Pool of Aggregating Nodes

The aggregating nodes are responsible for aggregating the local model updates into a global model. After verification of the model weights, the training nodes forward the model weights to the aggregating nodes. With the help of aggregating algorithms like FedAvg, the aggregating nodes aggregates the local model updates. Then the global update is pushed to the last block of the blockchain ledger.

5.1.4 Noise Generating Nodes

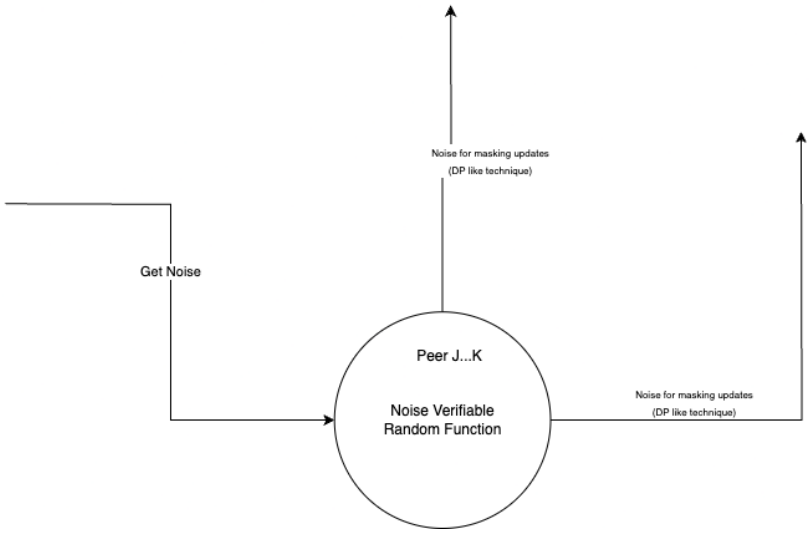

Figure 5.5: Pool of Noise Generating Peers

Noise is required in the process for the privacy protection technique. Differential Privacy like privacy preserving technique is planned to be implemented for maintaining the privacy of the private data. The Noise Generating Peers are responsible for generating Verifiable noise within the threshold of the privacy protection technique. The noise details are extracted from the blockchain block and the noise required for masking the local update is exported to the training nodes.

6. Timeline

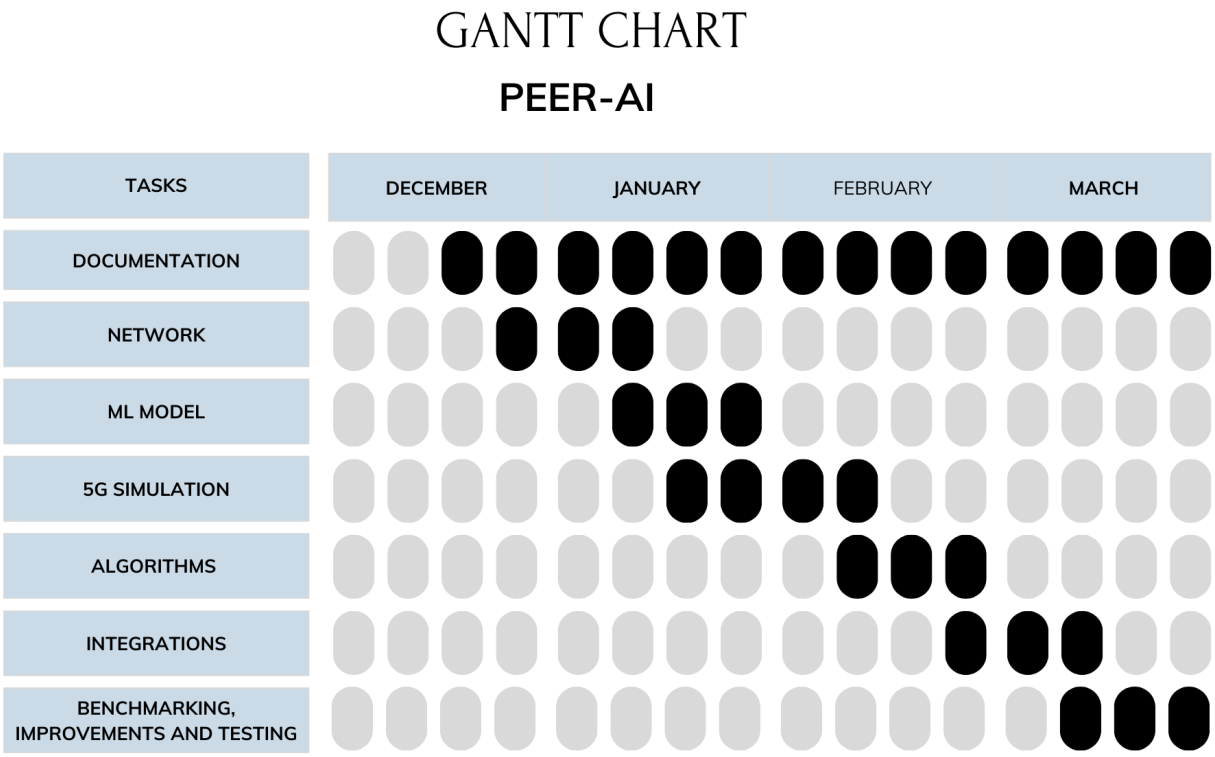

Figure 6.1: Gantt Chart for demonstrating timeline for the project

REFERENCES

[1] M. Shayan, C. Fung, C. J. M. Yoon, and I. Beschastnikh, "Biscotti: A ledger for private and secure peer-to-peer machine learning," *CoRR*, vol. abs/1811.09904, 2018. arXiv: 1811.09904. [Online]. Available: http://arxiv.org/abs/1811.09904.

[2] A. G. Roy, S. Siddiqui, S. Pölsterl, N. Navab, and C. Wachinger, "Braintorrent: A peer-to-peer environment for decentralized federated learning," *CoRR*, vol. abs/1905.06731, 2019. arXiv: 1905.06731. [Online]. Available: http://arxiv.org/abs/1905.06731.

[3] T. Wang, Y. Liu, X. Zheng, H.-N. Dai, W. Jia, and M. Xie, "Edge-based communication optimization for distributed federated learning," *IEEE Transactions on Network Science and Engineering*, vol. PP, pp. 1–1, Jun. 2021. DOI: 10.1109/TNSE.2021.3083263.

[4] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5476–5497, 2021. DOI: 10.1109/JIOT.2020.3030072.

[5] M. Singh, C. Kumar, and P. Nath, "P2p applications in 4g/5g networks using d2d communication based on social attributes of users," in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, 2020, pp. 730–738. DOI: 10.1109/WorldS450073.2020.9210297.

[6] Y. Wu, S. Wang, W. Liu, W. Guo, and X. Chu, "Iunius: A cross-layer peer-to-peer system with device-to-device communications," *IEEE Transactions on Wireless Communications*, vol. 15, no. 10, pp. 7005–7017, 2016. DOI: 10.1109/TWC.2016.2594225.

[7] P. M. Mammen, "Federated learning: Opportunities and challenges," *CoRR*, vol. abs/2101.05428, 2021. arXiv: 2101.05428. [Online]. Available: https://arxiv.org/abs/2101.05428.

[8] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography*, S. Halevi and T. Rabin, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 265–284, ISBN: 978-3-540-32732-5.

APPENDIX A

Differential Privacy and Privacy Preservation

Differential Privacy is a method of privacy protection of data by introducing a random noise in the data. It is an approach for providing privacy while sharing information about a group, by describing the patterns within the group while withholding information about specific individuals. It was introduced by Cythia Dwork, et al. in their paper titled *Calibrating Noise to Sensitivity in Private Data Analysis*[8]. It prevents the peers from membership inference attack.One of the most common ways of utilizing differential privacy is using laplace methods in which noise is added to the output of a function which masks the input of an individual in the data while still preserving the overall accuracy.

Byzantine Attack and Multi Krum

A Byzantine is a node that is not behaving as intended due to bugs, connection issues or the presence of malicious adversaries. The presence of a byzantine can cause the model to diverge. So, update verification should be done before aggregation to ensure that the update is not from a byzantine node. For this, Multi-Krum can be used which is a combination of krum and linear averaging methods for aggregation of model updates. A krum is an aggregation rule based on taking the average of only those (n-f) model updates (n-Total modal updates, f-Number of modal updates from Byzantine nodes) which have the least sum of squared difference with their average.

APPENDIX B

Shamir's Secret Sharing for distributed private information

Secret Sharing refers to the general concept of dividing confidential information(a secret) into multiple parts or shares such that no single individual has access to the full secret. The information is also encrypted so no single custodian of the shared secret fragment has any usable knowledge about the secret.

SSS is a cryptographic technique that protects sensitive data by distributing data fragments across multiple parties. The secret can be extracted if a specific number of parties combine their shares. Share threshold refers to the minimum number of shares required to unlock the main secret. So, even if a few shares are lost, the secret can still be extracted and a few shares cannot extract the secret by themselves if the few shares are within the share threshold. This also allows the secret owner to add, modify or even remove share without altering the original share or affecting other share custodian.