Blockchain Based Federated Learning system for Pneumonia detection

Vasundhhara Katoch S.A. Madhulica

Department of CSE Sri Sivasubramaniya Nadar college of Engineering
Final Year Project, May 2023

Highlights of Proposed Model

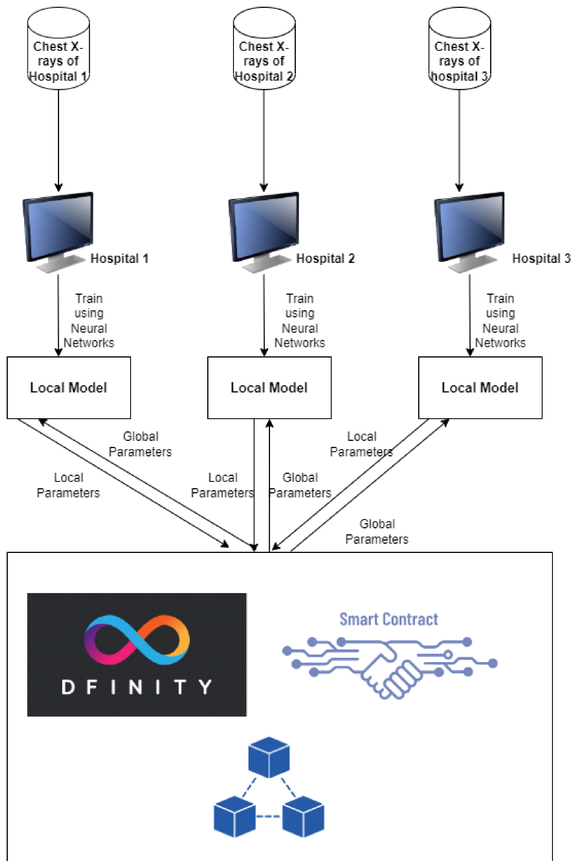
To develop a Blockchain Based Federated Learning system(BCFL) that

- Detects Pneumonia disease using chest X-rays.
- Preserves patient data using Federated Learning.
- Enables hospitals to be efficient by leveraging data of other hospitals
- Uses blockchain to decentralise the server.

Challenges in using ML models in Hospitals:

- Sensitive patient data, Data leaks and attacks
- Dependency on central server in federated learning(FL)

Proposed Model for BCFL



Functional Modules and Dataset Description

- Data Pre-processing
  - Data cleansing
  - Data augmentation
  - Feature engineering
  - Data splitting
- Pneumonia Detection (Model Building)
  - Conv2D for convolution operation
  - MaxPooling2D as pooling layer
  - RelU Sigmoid as activation functions
- Model Training
  - Adam Optimizer as optimizing functions
  - Binary Cross Entropy as loss function
  - ModelCheckpoint as callback function
- Parameter Sharing
  - Candid API for sending parameters to blockchain
- Model Aggregation
  - FedAvg, FedBoosting and Geometric Mean as aggregation functions

Dataset Description

- X-ray images of pediatric patients between the ages of one and five ,sourced from Kaggle, are used.
- All chest X-rays were initially assessed for quality before being included.
- Dataset was organized into 3 folders (train,test, val) and contains subfolders for each image category (Pneumonia/Normal).
- There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal).

Functional pipeline of Proposed System

Model Training using CNN

- This has 9 convolutional layers, where each intermediate convolutional layer gets input from both from previous layer and the layers preceding it via concatenation referred to as DenseNet.
- Local models of each hospital are trained at this level using CNN.

Sending Model Parameters to Blockchain

- The Internet Computer blockchain which was created by the DFINITY Foundation, has been used.
- Parameters are extracted and encoded to send via the network.
- Candid API and http requests library are used to send the parameters.
- The canister ID of the deployed canister is used to send the model parameters.

Deploying the Canister on Internet Computer Blockchain

- Motoko Playground is used to deploy a canister on Internet Computer Blockchain.
- The canister stores the smart contract and gets triggered only after the local model parameters are recieved from all the hospitals and after a fixed time period.
- The canister will have a unique canister ID for 20 minutes.
- The functions defined in smart contract stored in canister are called in the local python codes using Candid API.

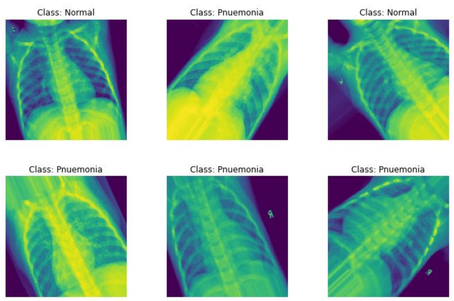


Figure 2. Preprocessed X-ray images

Performance Analysis

- Performance analysis in terms of security depends highly on the Blockchain framework we use.
- Hosting it on Internet Computer blockchain and building the canister on Motoko playground increases the security by many folds as it is completely decentralised and there is no central server as there is in the case of traditional FL.
- In the experiments conducted, the local models were found to have accuracies of 76%, 82% and 84% (Table 1).

Table 1. Local Models

Models	Accuracy
Local Model 1	76%
Local Model 2	82%
Local Model 3	84%

- Table 2 depicts the accuracy analysis of the global models built using different aggregation algorithms, the accuracy was improved.

Table 2. Global Model

Global Models	Accuracy
Global Model (FedAvg)	85%
Global Model(Geometric Mean)	85%
Global Model(FedBoosting)	87%

Canister deployed

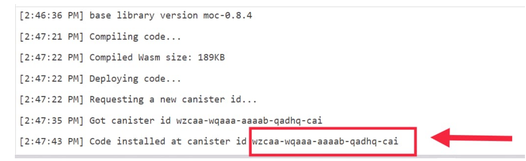


Figure 3. Unique Canister ID after deployment

Inferences

- The accuracies of global models were improved as compared to local models because of aggregation through different aggregation algorithms.
- The security of the system is improved because of the decentralisation done by the blockchain.
- Single point of failure and the possibility of malicious clients is prevented using consensus in smart contract.