NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE
School of Computer Science and Engineering
College of Engineering

CZ4079 Final Year Project – Peer to Peer Cluster Federated Learning

Presented by: Chai Li Xian
Supervised by: Anupam Chattopadhyay

Objectives

Federated Learning (FL) is a machine learning technique that enables the training of models across decentralized devices or nodes, without requiring the raw data to be centrally collected in one location. Instead, the model is trained in a distributed manner across multiple nodes, with each node only sending the model updates (and not the raw data) to a central server. Peer-to-Peer (P2P) Cluster FL is a variant of Federated Learning that involves the formation of clusters among participating nodes to improve the efficiency and accuracy of the learning process.

The project's direction was to explore the impact of various factors on the accuracy of the trained model in a P2P Cluster FL setting. The factors studied included **data quality, network bandwidth, memory bandwidth, cluster size**, and the use of the **Gale-Shapley algorithm** for node-cluster matching.

Algorithm

Peer to Peer Federated Averaging with Clustering

- Peer-to-Peer FedAvg is similar to normal FedAvg, but done in a peer-to-peer manner
- Nodes communicate with immediate neighbors instead of a central leader node
- Each node keeps track of global and local models
- Leader node trains its own local model and serves as the initial global model
- Leader node sends global and local model updates to neighbors
- Neighbors use global model to train their own local model
- Process continues until all nodes have participated and returned to leader node
- Leader node averages all local model weights to update global model.

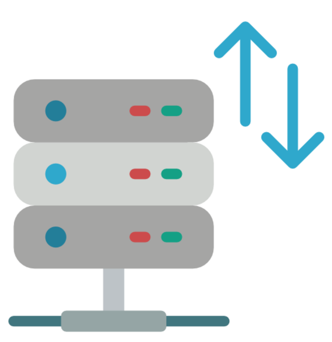
The process repeats until the model converges to a certain level of accuracy or until the end of all communication rounds. For efficiency, the shortest path between all nodes are always taken. Within each cluster, every node must be reachable from every other node within the cluster.

Algorithm 3: P2P Federated Averaging with Clustering

for *each cluster* **do**
  leader node starts off training;
  leader node appends local model weights to lists;
  leader node sends model weights to neighbours;
  for *each communication round* **do**
    for *each neighbour* **do**
      receive model weights from neighbours;
      set local model weights to be received weights ;
      do local training;
      append local model weights to lists;
      send model weights to neighbours;
    end
    if *node is leader* **then**
      leader node averages weights from weight list;
      updates global model weights;
      send global model weights to neighbours;
    end
  end
end

Factors


Data Quality


Network Bandwidth


Memory Bandwidth


Cluster Size

Stable Marriage Problem – Gale Shapley Algorithm

Algorithm 4: Gale-Shapley Algorithm for Matching Nodes to Clusters

Input : List of n nodes $N_1, N_2, ..., N_n$ and m clusters $C_1, C_2, ..., C_m$
Output: Stable matching between nodes and clusters

for i from 1 to n **do**
  N_i proposes to its most preferred cluster C_j that has not yet rejected it;
  if C_j *is available* **then**
    C_j accepts the proposal and accept node N_i;
  else
    if C_j *prefers node* N_i *more than its existing match* **then**
      C_j rejects current match;
      C_j accepts node N_i;
    else
      C_j rejects node N_i;