MACHINE LEARNING ASSIGNMENT #2

FEDERATED LEARNING MODEL FOR IMAGE CLASSIFICATION

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI, RAJASTHAN - 333031

BITS F464 - MACHINE LEARNING

GROUP - 41

Prepared By:-

Aditya Shrivastava **2022A8PS1732P**

Date of Submission:- Course I/C:-

29th April 2025 **Navneet Goyal Sir**

INDEX

1. Introduction:-

Machine Learning (ML) has traditionally relied on a centralized approach, where data from various sources is collected and stored on a central server. Models are trained on this centralized dataset to make predictions or perform classifications.

However centralized learning poses significant challenges related to **data privacy, security, and communication costs,** especially when the data originates from distributed sources such as mobile devices.

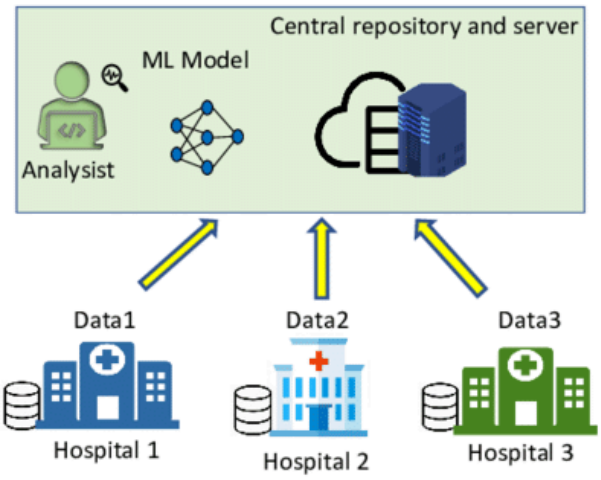


Fig.1 - Centralized Machine Learning Model

In contrast, Federated Learning (FL) is a decentralized machine learning technique. Instead of sending raw data to a central server, clients train models locally on their own data, and only the updated model parameters (like model weights, biases, etc) are sent back to the server. The server then aggregates the updates from multiple clients and improves the global model without ever seeing the raw data.

This decentralized approach offers several advantages:

- **Privacy:** Raw data never leaves the device
- **Security:** Reduces risk of central data breaches

- **Reduced Communication:** Only model updates are transmitted instead of large datasets.

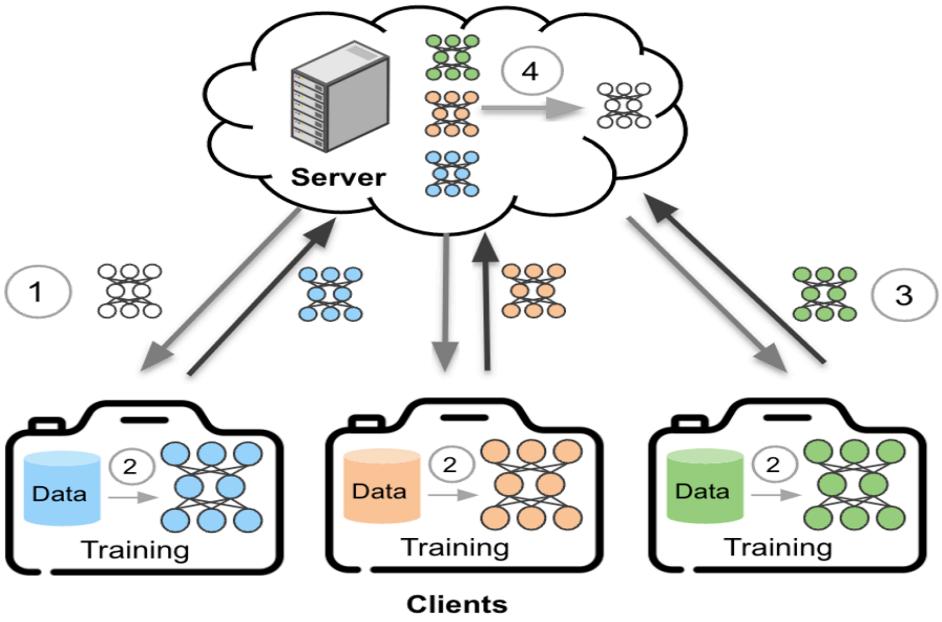


Fig.2 - Federated Learning Model

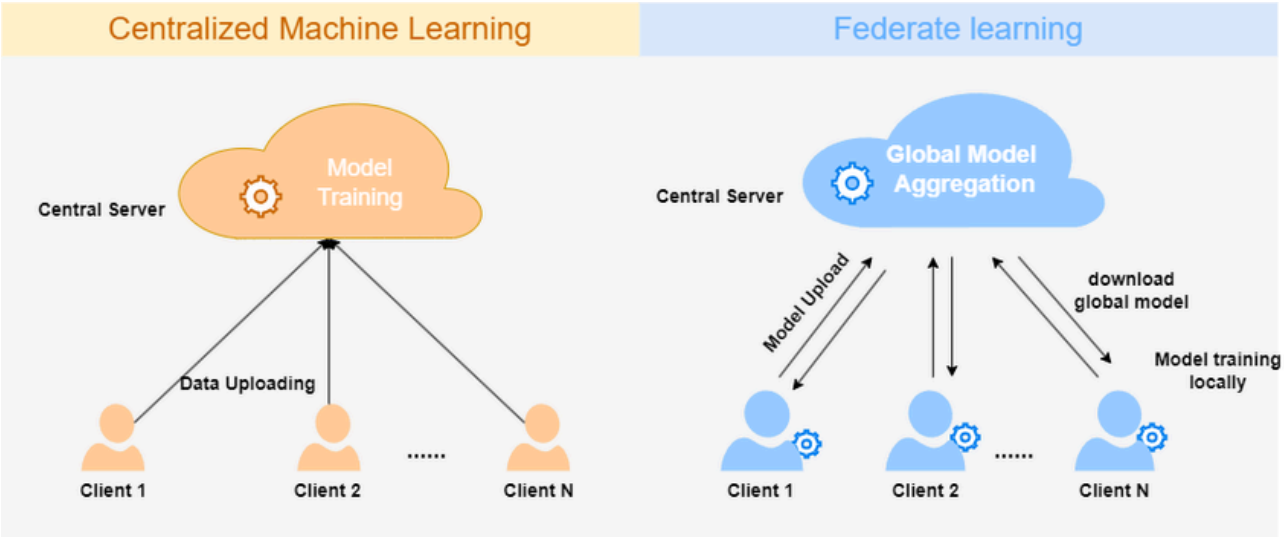


Fig.3 - Centralized v/s Federated Model

In this project we focus on solving an Animal Image Classification problem using both:

-> Centralized Machine Learning

-> Federated Learning

We simulate a federated system consisting of 1000 clients (mobile devices), each with their own local datasets, varying battery levels, network conditions and computing capabilities. We aim to:

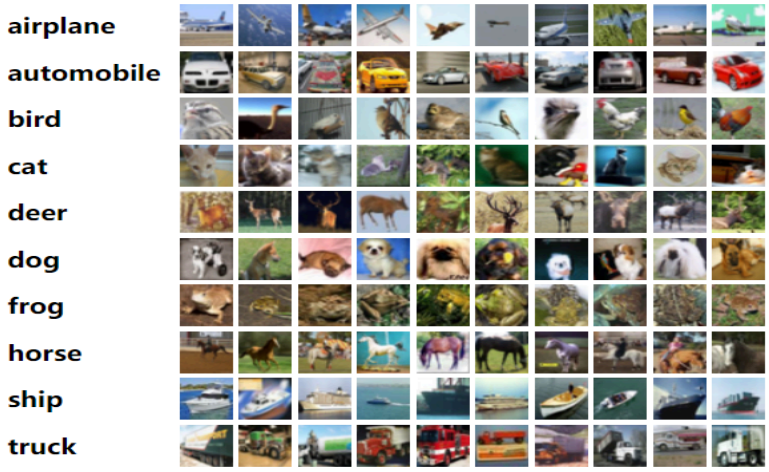
- Evaluate and compare the accuracy of both models
- Measure the communication cost involved in federated learning
- Analyze the practical challenges and trade-offs between the two models

2. Dataset Preparation:-

The dataset used for this project is the **CIFAR-10** dataset, a well-known benchmark for image classification tasks. It contains **60,000 color images** (32x32 pixels) across 10 categories, including objects like airplanes, cars, animals, etc. Since our project focuses on **animal image classification**, we filter and retain only the following **6 animal classes:**

Original Label	New Label	Class
2	0	Bird
3	1	Cat
4	2	Deer
5	3	Dog
6	4	Frog
7	5	Horse

We remap these classes to new labels (0-5) for easier processing.



2.1 Dataset Expansion:-

To simulate a realistic federated environment with 1000 clients, a larger dataset was needed. Thus the original CIFAR-10 dataset was **duplicated three times** to create an expanded dataset with approximately **150,000 samples**.

This ensures that:

- Every client can start with enough initial data (50 samples)
- Clients can receive additional new samples after every FL round

2.2 Preprocessing:-

All images undergo the following transformations:

- **Random Horizontal Flip:** Randomly flips images horizontally to augment training data.
- **Random Crop with Padding:** Crops images randomly to introduce variation.
- **Normalization:** Normalizes pixel values to [-1, 1] range across all channels.

These augmentations help the model generalize better to unseen images.

2.3 Final Data Split:-

The expanded dataset is split into:

- **Testing Set** (to evaluate final model accuracy) => 10,000
- **Server Private Data** (for pre-training skeleton model) => 5,000
- **Training Pool** (for client data assignment) => Remaining dataset

3. Centralized Machine Learning:-

In the centralized machine learning approach, all available training data is collected in one location and used to train a single model.

3.1 Model Architecture:-

A custom-designed **Convolutional Neural Network (CNN)** is used for classification. The architecture includes:

- Three Convolutional layers:
  - Each followed by Batch Normalization and ReLU activation.
  - Max Pooling after certain layers for downsampling.
- Two Fully Connected layers:
  - A dense layer with 356 neurons and ReLU
  - A final output layer with 6 neurons for the 6 classes

LAYER TYPE	DETAILS
Conv2D	32 filters, 3x3 kernel
Conv2D	64 filters, 3x3 kernel
MaxPooling	2x2
Conv2D	128 filters, 3x3 kernel
MaxPooling	2x2
FullyConnected	128 x 8 x 8 -> 256 neurons
FullyConnected	256 -> 6 output classes

3.2 Training Settings:-

Setting	Value
Batch Size	64
Learning Rate	0.0001

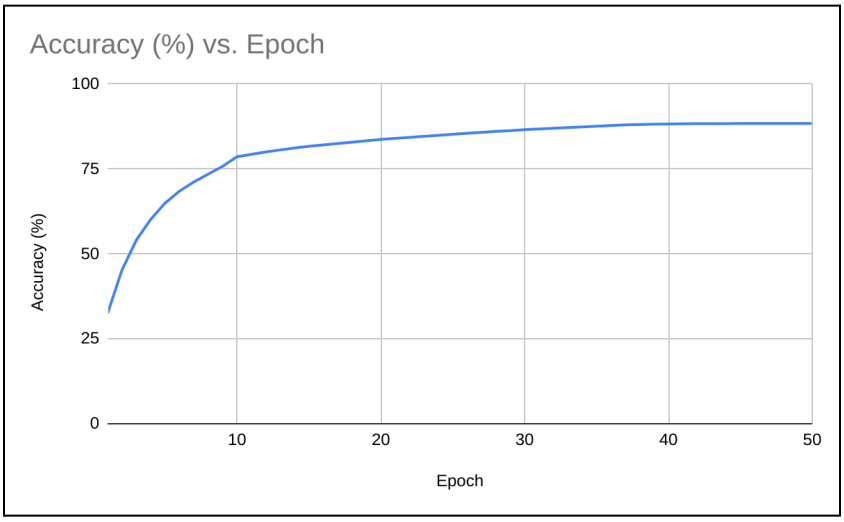
Setting	Value
Optimizer	Adam
Loss Function	Cross Entropy Loss
Epochs	50
Transformations	Random Flip, Random Crop, Normalization

The model is coded to be trained using a GPU when available, otherwise CPU (my laptop only had CPU).

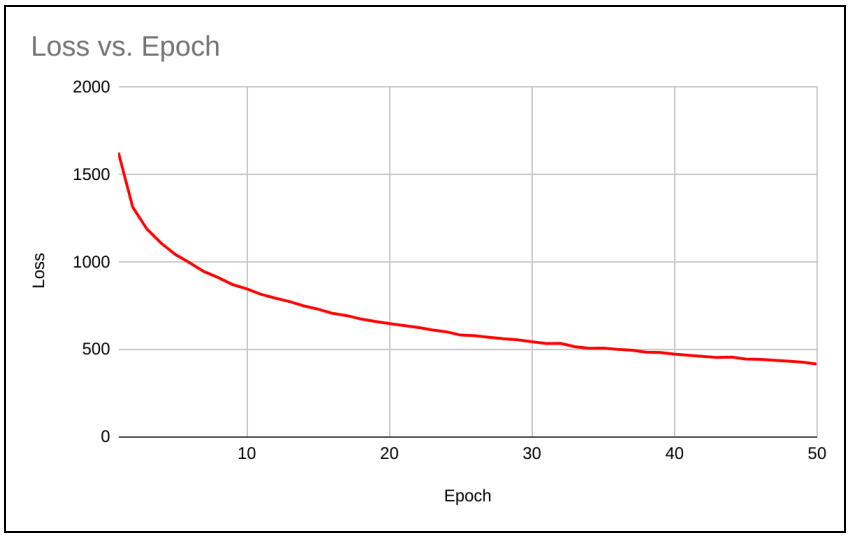
3.3 Results:-

After training the centralized model for 50 epochs:

- Final Test Accuracy: **88.39%**
- Training Loss: Gradually decreased across epochs (started from 1626.7151 in the 1st epoch to 417.5017 in the 50th epoch)
- Observations:
  - The model learns animal features well.
  - Misclassifications mostly occur between visually similar classes (eg. dogs vs cats).



Graph 1: Accuracy vs Epochs for Centralized Machine Learning Model



Graph 2: Loss vs Epochs for Centralized Machine Learning Model

4. Federated Learning:-

In Federated Learning (FL), the goal is to train a shared global model collaboratively across many decentralized clients (simulated mobile devices) without sharing raw data. Each client trains locally using its own data, and only model updates (weights) are communicated with the server. This preserves privacy and reduces risks compared to centralized approaches.

4.1 Simulation Setup:-

To simulate a realistic FL environment:

- Number of Clients: 1000
- Client Metadata:
  - Battery Level (random between 20% to 100%)
  - Network Type (Strong Free/ Strong Paid/ Weak)
  - Compute Capability (random between 20% to 100%)

Clients were selected dynamically based on:

- Battery Threshold (above 30%)
- Network Strength (Weak clients excluded)
- Compute Capability (above 30%)

Thus, only eligible clients participated in each FL round.

4.2 Client and Server Roles:-

The client.py:

- Generated 1000 clients
- Generated random metadata for each client
- Start with initial labeled data (50 samples)
- Train the model locally
- Receive new data every round
- Send updated model weights back to the server

The server.py:

- Initialized a skeleton model using private server data (5000 samples)
- Sends the model to selected clients each round
- Aggregate updates using Federated Averaging (FedAvg)
- Update the global model

4.3 Training Process:-

- **Initial Pre-training:**
  - Server trains the skeleton model before starting FL rounds.
  - 5000 labeled sample for 10 epochs.

- **Federated Training:**
    - Each FL round consists of:
        - Selecting 200 clients randomly (based on eligibility)
        - Sending the current global model to these clients
        - Clients perform local training (5 epochs)
        - Clients receive 25 new samples from the training pool each round
        - Clients send updated model weights to the server
        - Server aggregates all updates using FedAvg to update the global model

Setting	**Value**
Number of Rounds	25
Clients per Round	200
Local Training Epochs	5(per client)
Learning Rate	0.001
New Data per Round	25(per client)

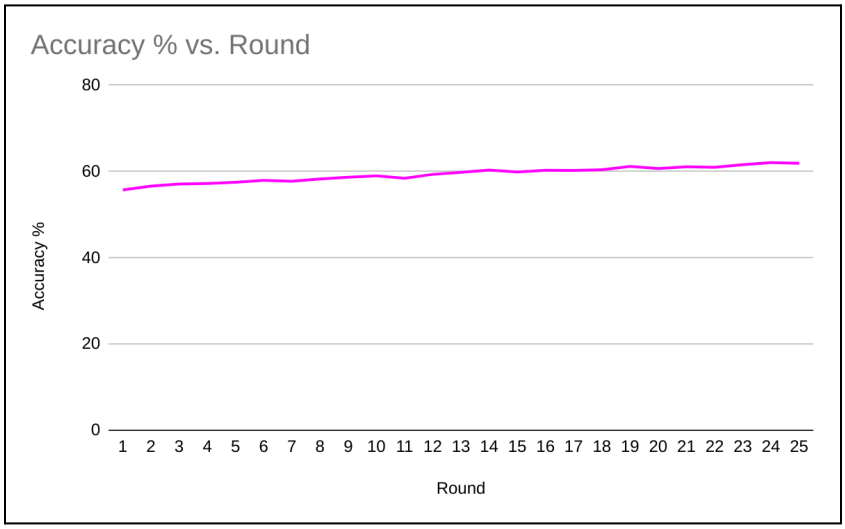
5. Results & Analysis:-

This section presents the results obtained from both centralized and federated learning approaches.
We compare them based on model accuracy and communication cost.

5.1 Accuracy Comparison:-

- **Centralized Machine Learning:**
  - Trained using the full dataset on the server.
  - Final Test Accuracy after 50 epochs: **88.39%**

- **Federated Learning:**
  - 25 rounds of training with 200 clients per round.
  - Clients started with 50 samples and received 25 new samples each round.
  - Final Test Accuracy after 25 rounds:



Graph 3: Accuracy vs Rounds for Federated Learning

This graph clearly shows a steady improvement in the global model accuracy over FL rounds, though it remains lower than centralized due to:

- Limited local data per client (max. 675 samples after 25 rounds)
- Only 5 epochs of local training
- No overlap in client datasets (non-IID)

5.2 Communication Cost Analysis:-

In Federated Learning, there is significant communication between the server and clients:

- Each round:
  - Model is sent to 200 clients
  - 200 clients send updated weights back
- Each communication involves the full model (approx. 8MB)

Calculated Cost:

- Model Size = 8MB
- Clients per Round = 200
- Number of Rounds = 25
- Total Communication (send + receive) = 2 × 200 × 25 × 8MB = 80GB

Thus, there is a clear **trade-off**.

Federated Learning saves data privacy but incurs high communication costs.

5.3 Summary Comparison:-

	Centralized	**Federated**
Accuracy	88.39%	62.06%
Communication Cost	0 MB	83642.81 MB
Data Privacy	Not Preserved	Preserved
Scalability	Requires full data access	Scales to many distributed clients
Training Location	Central Server	Clients (Mobiles)

Table: Comparison between Centralized and Federated Model

6. Conclusion:-

In this project we explored and implemented both centralized and federated learning approaches for an animal image classification task using a filtered version of the CIFAR-10 dataset.

We successfully:

- Trained a centralized CNN model using the entire dataset
- Simulated a realistic federated learning environment with 1000 clients
- Designed client selection logic based on battery, network and compute capacity
- Implemented local training, server aggregation and round wise updates
- Compared both methods in terms of accuracy and communication cost

Key Takeaways:

- Centralized Learning achieved higher accuracy due to access to the complete dataset and full-batch training.
- Federated Learning achieved **62.06%** accuracy, which is respectable given the constraints:
    - Small local datasets
    - Non-IID data distribution
    - Only 5 local epochs per round
- Communication cost in FL was significant, which is expected as model parameters are exchanged frequently between the server and clients.

Final Remarks:

Federated Learning offers a promising alternative to traditional machine learning when:

- Data privacy is critical
- Data is inherently distributed (eg. across different devices)
- Bandwidth and compute cost trade-offs are acceptable

With more advanced techniques (e.g., adaptive client selection, differential privacy, compression of updates), FL accuracy can be improved while reducing communication overhead.

This project provided hands-on insights into the practical challenges of FL, such as data heterogeneity, slower convergence, and system simulation — all of which are central themes in modern AI deployment.

7. References:-

- Krizhevsky, A. (2009). **Learning Multiple Layers of Features from Tiny Images**. *[CIFAR-10 Dataset Source]*
    - https://www.cs.toronto.edu/~kriz/cifar.html

- McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). **Communication-Efficient Learning of Deep Networks from Decentralized Data**. *Federated Averaging (FedAvg) Algorithm*.
    - https://arxiv.org/abs/1602.05629

- PyTorch Documentation
    - https://pytorch.org/docs/

- **OpenAI. (2024).** *ChatGPT Assistance in Research & Code Structuring* (Used for guidance, architecture planning, debugging, and reporting assistance in this project.)