EDS6397 - Digital Image Processing

Group-3 Project Proposal

Comparative Analysis of Custom CNN's v/s Pretrained Image Models Using Federated Learning on CIFAR-10 Dataset

Abstract

Federated learning represents a paradigm shift in machine learning, enabling decentralized model training without compromising data privacy. By distributing data processing across multiple devices, federated learning prevents the transmission of raw data, enhancing security and protecting sensitive information. This approach is particularly valuable in healthcare and mobile applications where data confidentiality is paramount. Through collaborative model improvement while maintaining data confidentiality, federated learning bridges the gap between individual device capabilities and collective knowledge.

Introduction

Federated learning (FL) is a collaborative approach to privacy-preserving that utilizes private data across multiple distributed devices. This technique allows Internet of Things (IoT) devices/secure data servers to operate without transmitting sensitive network traffic data to a central cloud server. In this project, we will be introducing a federated deep learning (FDL) method aimed at object classification. Within the realm of machine learning (ML), federated learning allows models to be trained locally on devices, with only the trainable parameters being shared with a central server, thus ensuring the privacy and security of the underlying data. A common strategy involves training models locally on each device using a datasets and the parameters from these local models are then averaged to produce the final aggregated model parameters.

Approach

1. Data Collection: For our comparative analysis of custom CNNs and pretrained models using federated learning, we selected the CIFAR-10 dataset. This dataset was chosen due to its diverse class range and its effectiveness in illustrating the principles of federated learning.

2. Model Selection: The plan is to develop three custom CNN models and use three pre-trained models: one from convolutional neural networks, another from a vision transformer, and the third from a Swin transformer. These models will then be compared in terms of performance.
3. Federated Algorithm: Implementing the federated learning algorithm is a complex and time-consuming process, here the idea is to use FedAvg algorithm to aggregate the model weights.
4. Model Evaluation: Classification metrics such as accuracy, precision, and F1-score will be used to assess the performance of each model.

Dataset

Here, we will be using "CIFAR-10 dataset", and here is the dataset link-

https://www.kaggle.com/c/cifar-10

Tools

1. Coding Language: Python
2. Framworks: Google colab, Jupyter Notebook
3. Model designing Libraries: Tensorflow, PyTorch and Scikit Learn
4. IDE: VScode
5. Version Control: Github

Group Members

1. Ashish Darshi (2339727) - Responsible for implementing the FedAvg algorithm, developing a custom CNN model, adapting a pre-trained model for our needs, and managing the GitHub repository.
2. Bhuwaneshwar Sagar (2347985) - Tasked with researching a custom CNN model to enhance object detection, understanding its architecture, and handling the project report, proposal, and presentations.
3. Kirthi Swarangi Chandrika (2298220) - Focused on researching a custom CNN model, particularly optimizing the second layer of a pre-trained architecture code work.
4. Dheeraj Vanga (2346311) - Dedicated to studying VGG16 and other transformer architectures, implementing these models within our model logic, and analyzing their performance.
5. Tharun Kshatriya (2347995) - Responsible for implementing models using the Swin Transformer and developing second-layer machine learning models to enhance overall performance.