Data Distribution Analysis

The bar plots indicate the frequency of each class for the overall dataset and five selected users. Below is the description of each distribution:

Overall Data Distribution

- The overall distribution exhibits a significant variance in class frequencies.
- Some classes are notably more frequent than others, suggesting an imbalanced dataset.
- The highest peaks correspond to classes that are most common in the dataset.

User 0 Data Distribution

- User 0's data shows a somewhat uniform distribution with a few classes peaking.
- This suggests that while User 0 has a diverse dataset, certain classes dominate.

User 1 Data Distribution

- User 1's distribution is uneven with certain classes showing higher frequencies.
- There is a visible skew towards some classes, indicating a concentration of specific data points.

User 2 Data Distribution

- User 2's plot shows a couple of classes with significantly higher frequency.
- The majority of classes have a lower and quite uniform frequency.

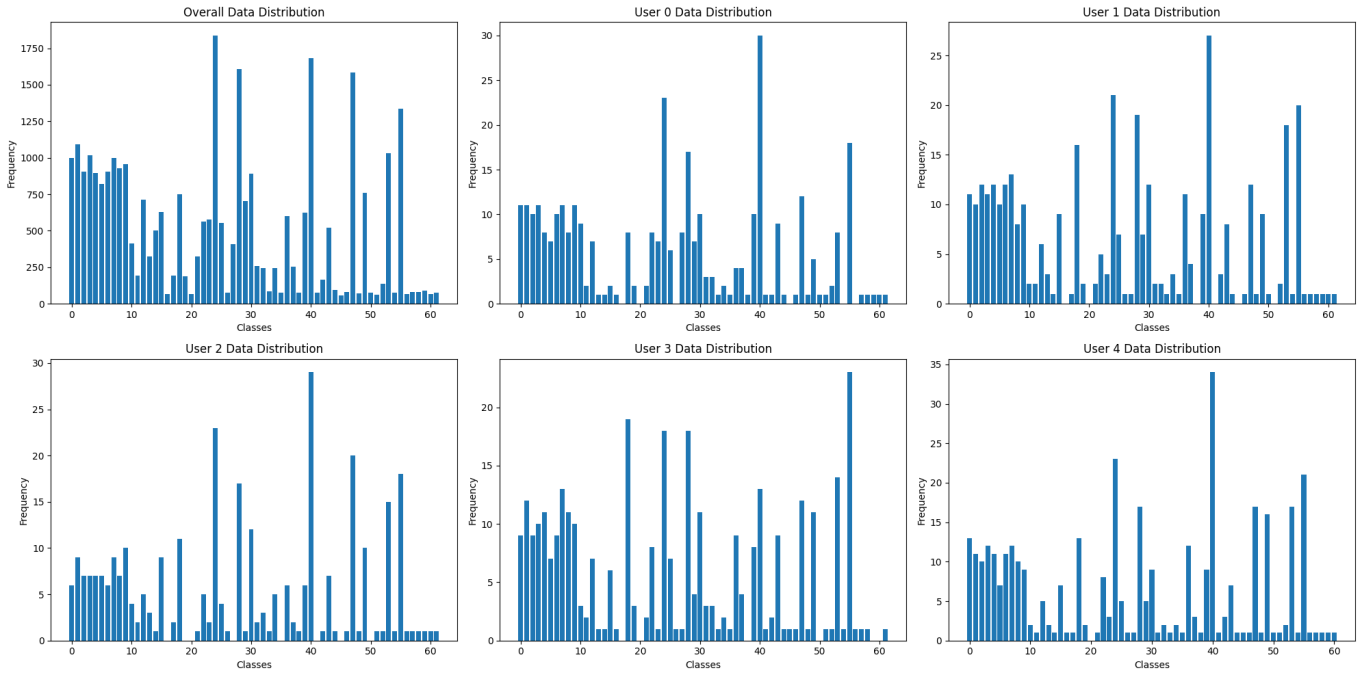
User 3 Data Distribution

- User 3's data distribution has one class with an exceptionally high frequency.
- The rest of the classes have moderate to low frequencies.

User 4 Data Distribution

- User 4's graph indicates a distribution where a handful of classes have much higher frequencies than the rest.
- This distribution is also fairly imbalanced with a focus on certain classes.

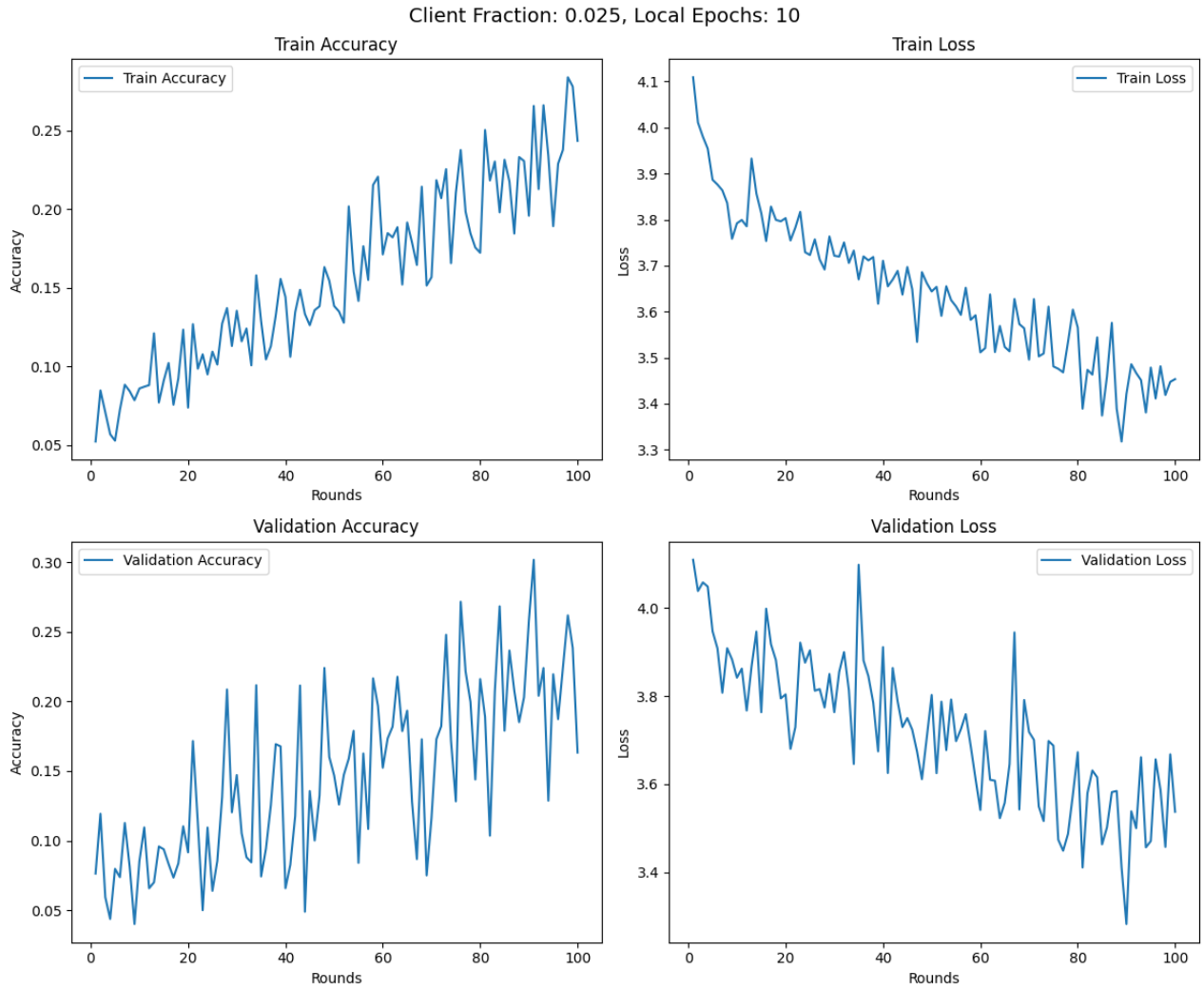
Each bar plot represents the class frequency for 62 classes, including 10 digits (0-9), 26 lowercase (a-z), and 26 uppercase (A-Z) characters. The x-axis lists the classes, and the y-axis represents the frequency of each class. These plots help in understanding the data distribution for each client, which is crucial for training models in a federated learning setup.



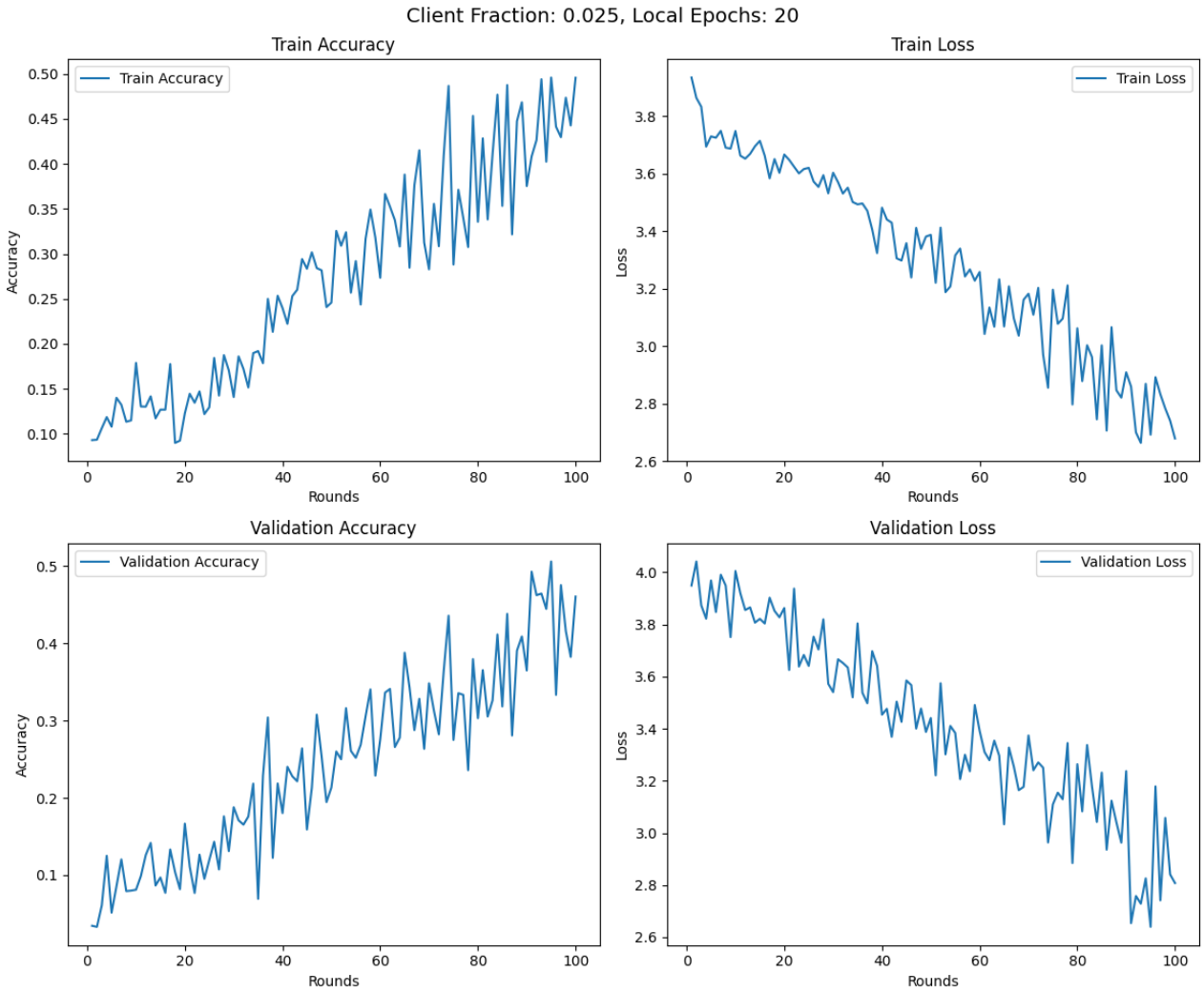
Grid Search Training Plots

The following plots represent the training progress under different epoch and client sample size combinations. These visualizations are crucial for understanding the performance of the federated learning model across various configurations.

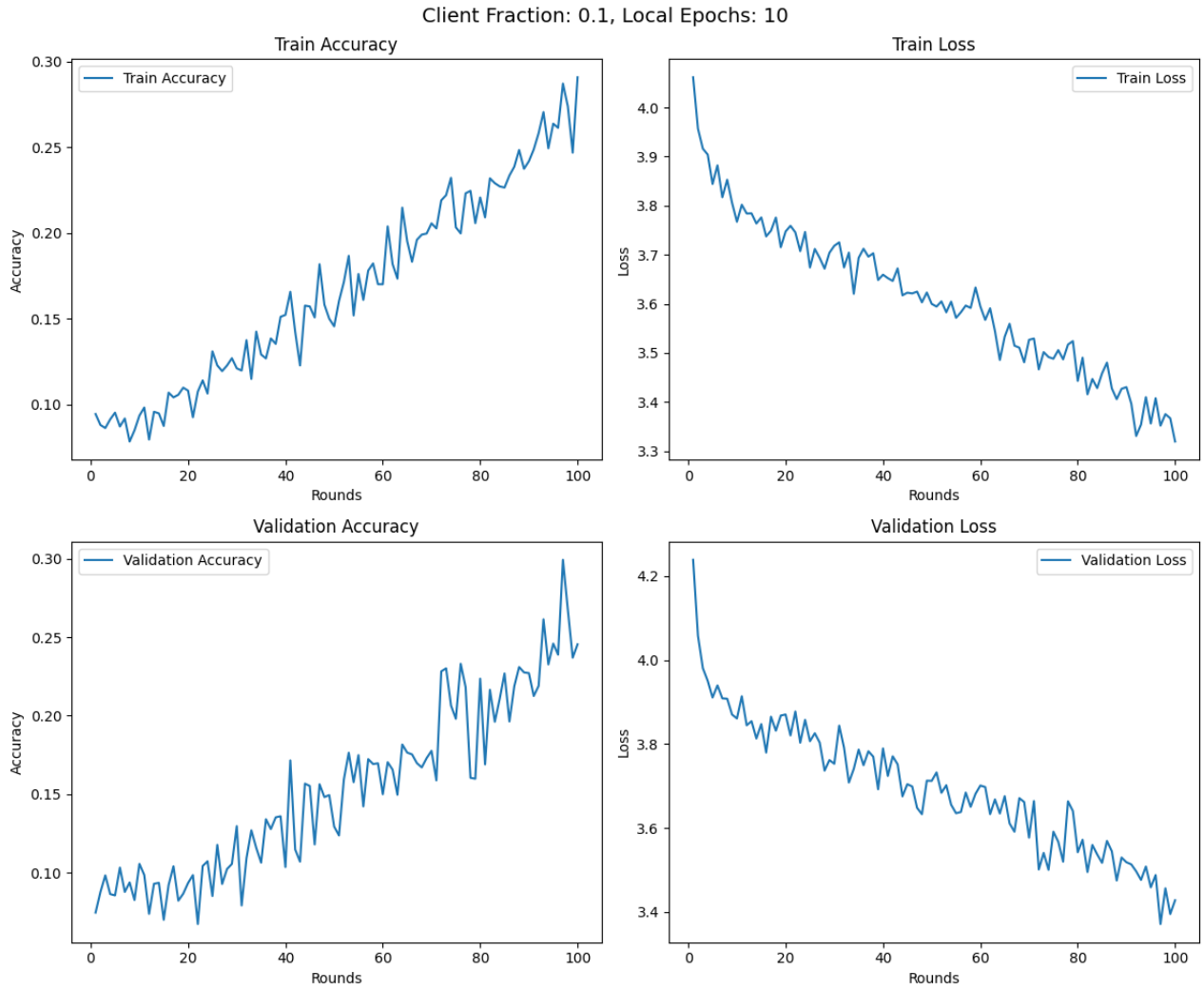
1. Client Fraction 0.025, Epochs 10:



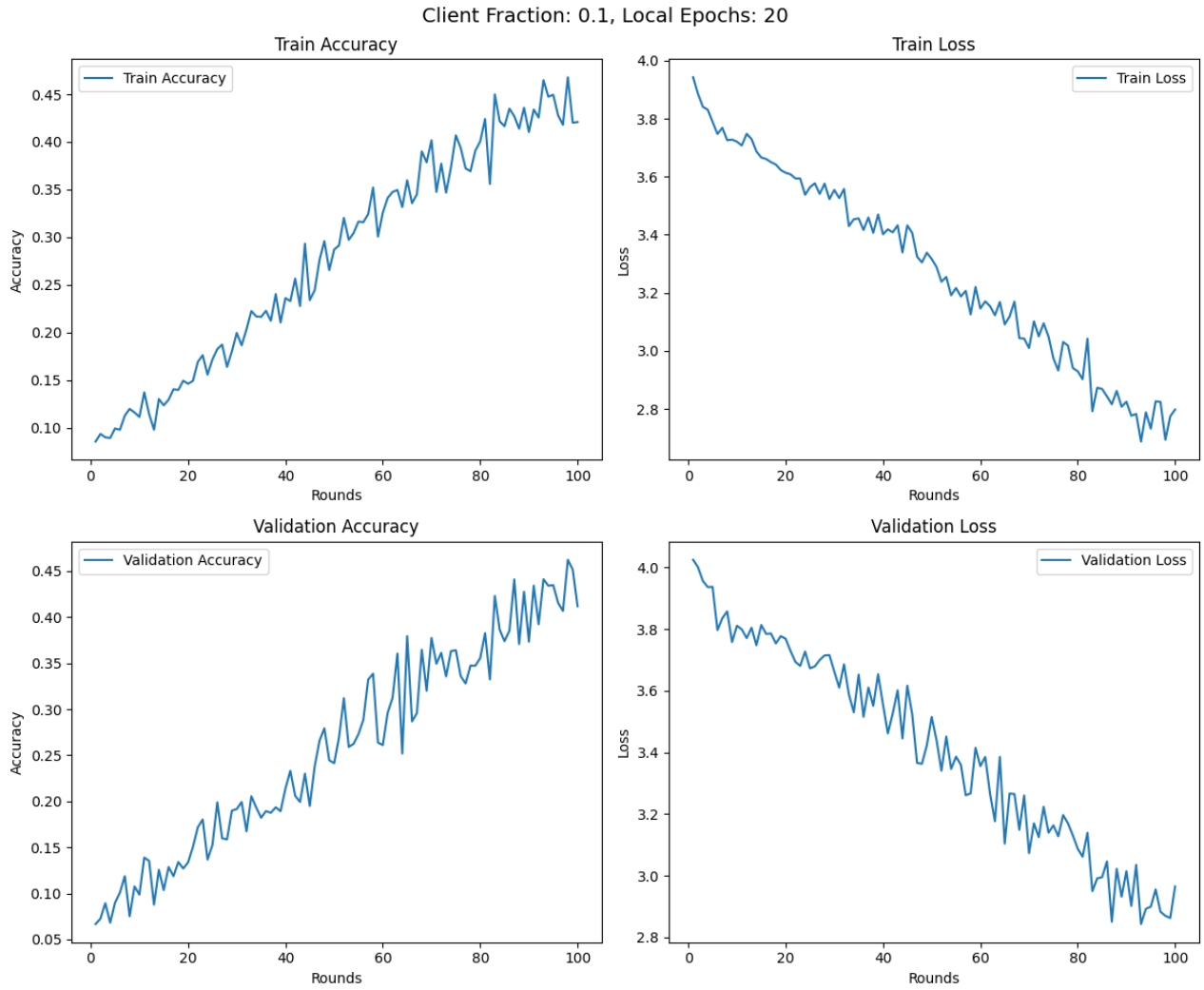
2. Client Fraction 0.025, Epochs 20:



3. Client Fraction 0.1, Epochs 10:



4. Client Fraction 0.1, Epochs 20:

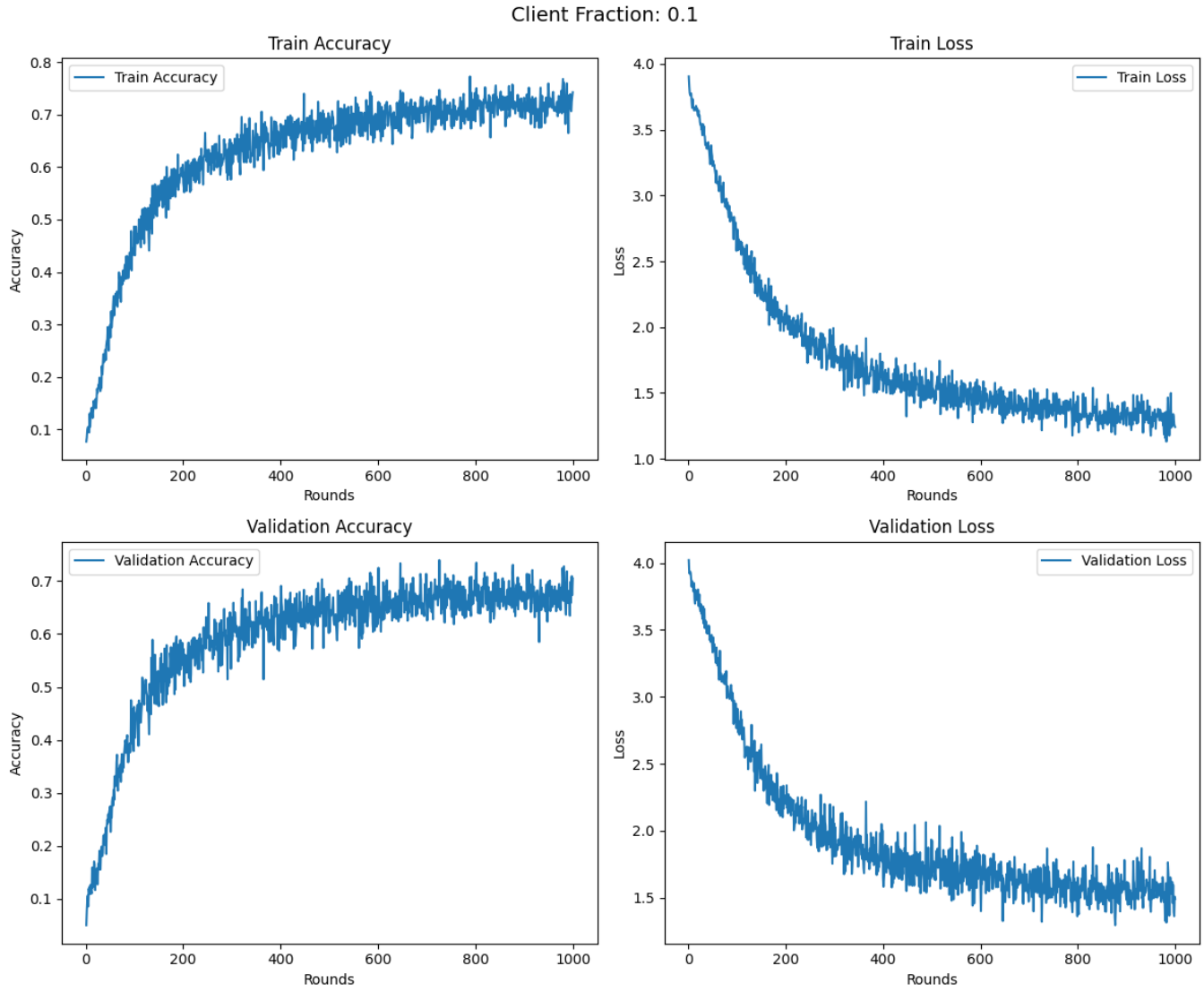


Analysis

The analysis of the plots indicates that a setup with 20 epochs and a client proportion of 0.1 yields better results and less variance. This improvement can be attributed to the extended training time, allowing more clients to contribute to the global model's learning process.

Final Trial Results (Part 1)

The final trial for Part 1 of the assignment was conducted using a client proportion of 0.1 and 20 epochs. The plot below shows the training progress over 1000 rounds.

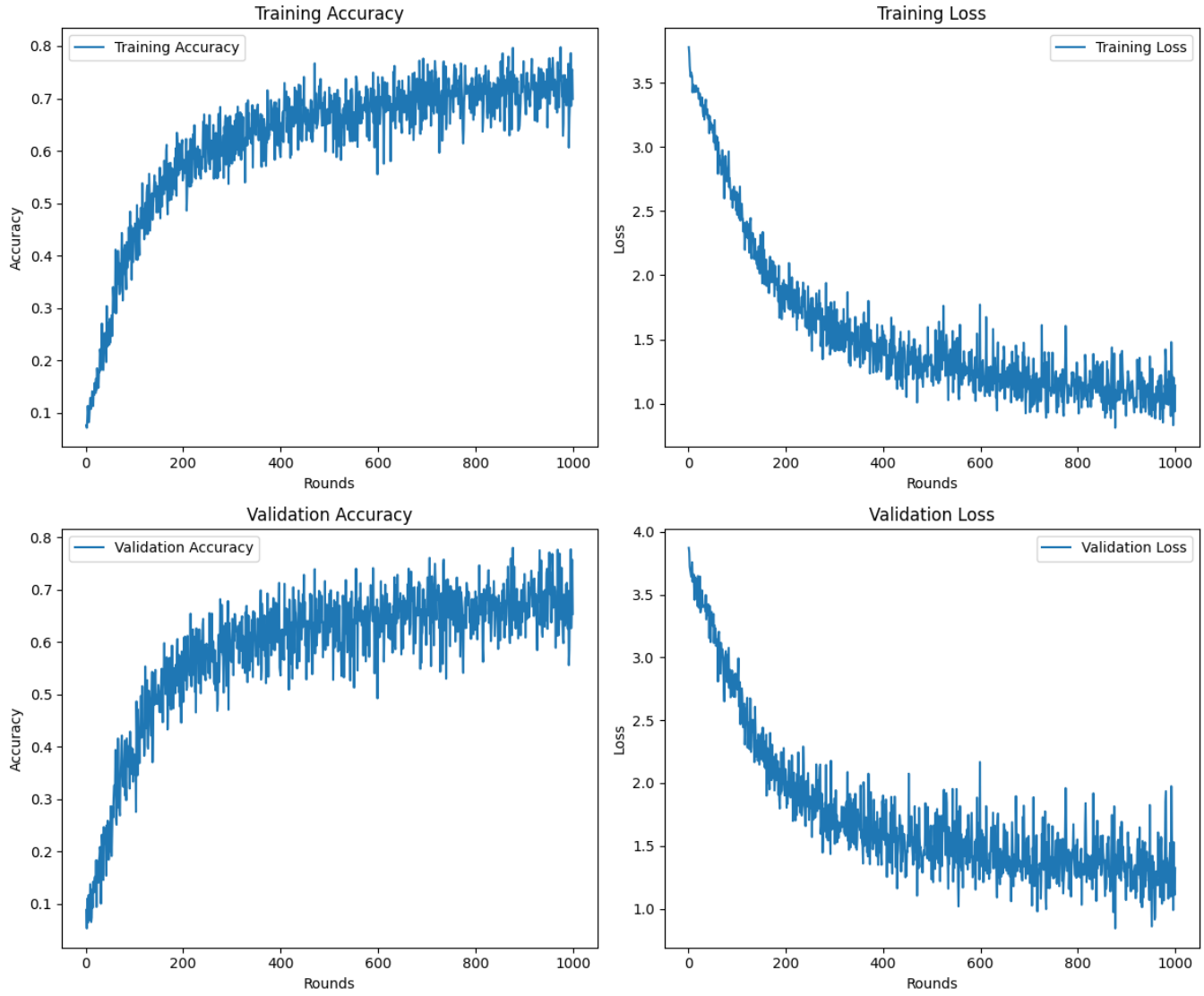


Test Accuracy

The test accuracy for the 20-epoch, 10% client model is `62.69%`.

Part 2: Parallel Clients

Following the identification of optimal hyperparameters in Part 1, Part 2 utilizes 20 epochs, with 4 clients or a 4% client proportion as mandated by the assignment requirements.



Test Accuracy for Part 2

The test accuracy for the model in Part 2 is `62.61%`.

Instructions for Running the Code

To execute the scripts, follow these steps:

1. Obtain the data from the SharePoint.
2. Create a virtual environment using `requirements.txt`.
3. For Part 1, activate the virtual environment and execute `python310 03_hw_q1_code.py`.
4. For Part 2, run `python310 03_hw_q2_code.py`.
5. To use the holdout set, adjust the script at the bottom as needed. Run the script for both the sequential (Part 01) and parallel (Part 02) parts.
    - For Part 1, use `python310 03_hw_holdout_code.py 1`.
    - For Part 2, use `python310 03_hw_holdout_code.py 2`.