KOÇ ÜNİVERSİTESİ
COMPUTER ENGINEERING

Fall 24

COMP 430: Data Privacy and Security
Project Report

Malware Detection Model using Federated Learning

Cemal Nişan

Motivation / Explanation

One of the disadvantages of Machine Learning models is the requirement to collect and process all data on a single server for training. However, this approach may cause risks in terms of data privacy and security. It is possible to develop models on the client's devices using their data, then aggregate the model parameters on a server to train a model as if it were a single model. This approach is called "Federated Learning".

In this project, a malware detection model was trained using Federated Learning. Various experiments were conducted to explore the advantages and disadvantages of FL. The details of these experiments are provided in the following section of the report.

Technical Details

Tools and technologies used during the project is listed following:

- Programing Language

  Python 3.9.12

- ML Model

  Convolutional Neural Network (CNN)

- Libraries

  PyTorch NumPy matplotlib

- Utility Tools

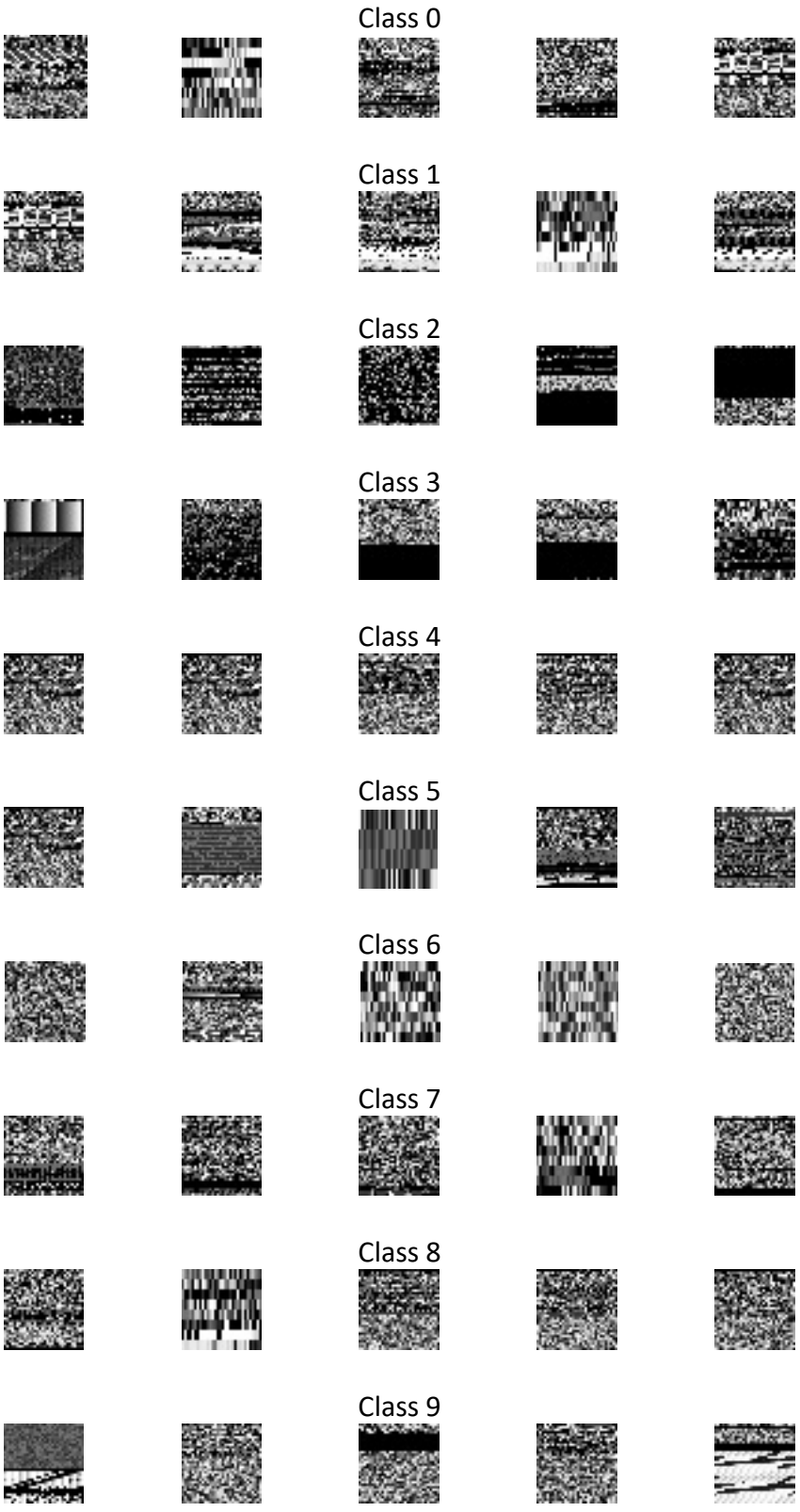
  Visual Studio Code GitHub GitHub Copilot ChatGPT

- Dataset

  Virus-MNIST from Kaggle was used.

  Size of dataset:
  - Test samples: 3,458
  - Train samples: 48,423

5 random samples were selected from each class of train dataset for illustration.



Experiments

To access the source code and experiment results, you can visit my GitHub repository: https://github.com/cemalnisan/FederatedLearning

Important Note: For all experiments, a separate GitHub branch has been created. Each branch contains the following items related to its respective experiment:

1) Model checkpoint (.pth)
2) Accuracy graph (.png)
3) Loss graph (.png)
4) Training log (.txt)
5) Changes made to the main code to enable the experiment.

This approach makes it easier and more comprehensible to observe the results. Additionally, the ML model has been saved for potential use in future experiments.

The experiments focused on two main aspects:

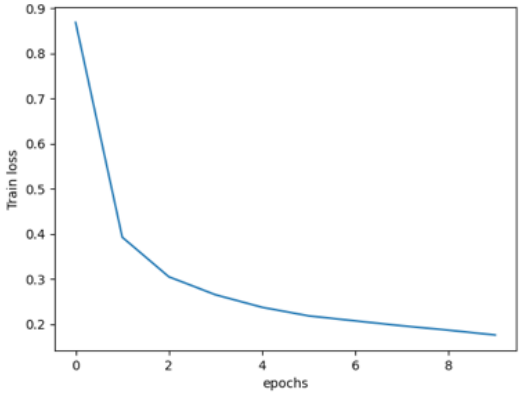
- Balanced Data Distribution
- Imbalanced Data Distribution

The data distribution of the Clients was first carried out in a balanced manner and then in an imbalanced manner. Additionally, the effect of the number of Clients was also examined.

Firstly, a single-server model was developed to compare the performance of Federated Learning (FL). The metrics for this model are provided below:

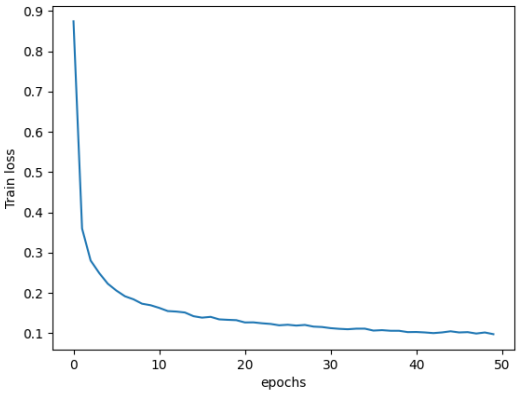
ML Model | Train Epochs = 10

Test Accuracy = 98.40%



ML Model | Train Epochs = 50

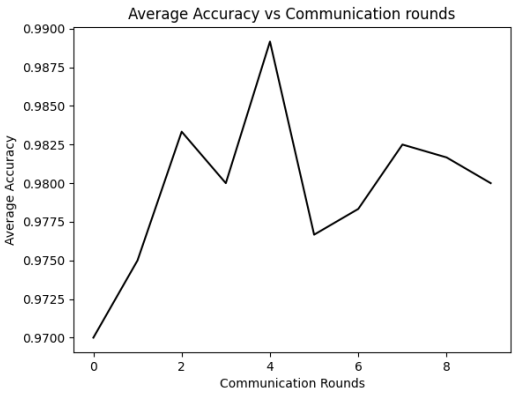
Test Accuracy = 99.07%



→ In the single-server model, as the number of epochs increases, the model's accuracy improves, and the training loss decreases.

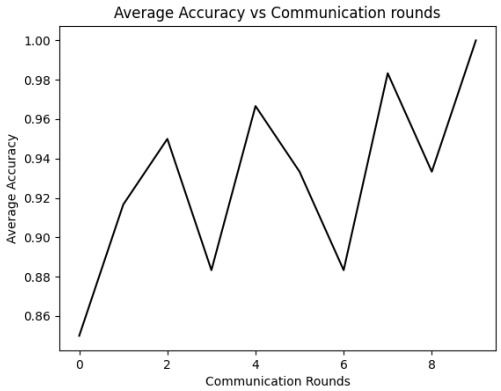
FL Model | Balanced | IID | Client Number = 5 | Local Epoch = 10 | Round = 10

Test Accuracy = 98.77%



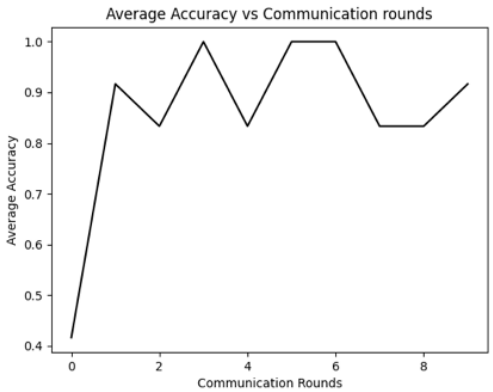
FL Model | Balanced | IID | Client Number = 100 | Local Epoch = 10 | Round = 10

Test accuracy = 97.20%



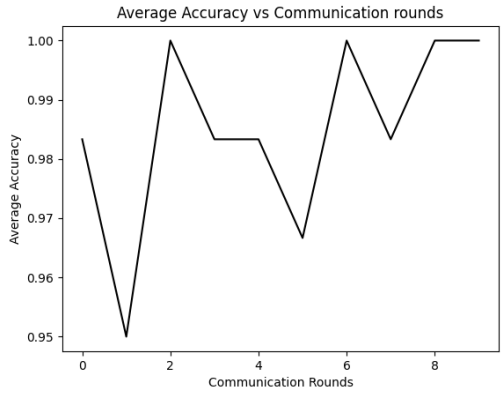
FL Model | Balanced | IID | Client Number = 500 | Local Epoch = 10 | Round = 10

Test Accuracy = 93.49%



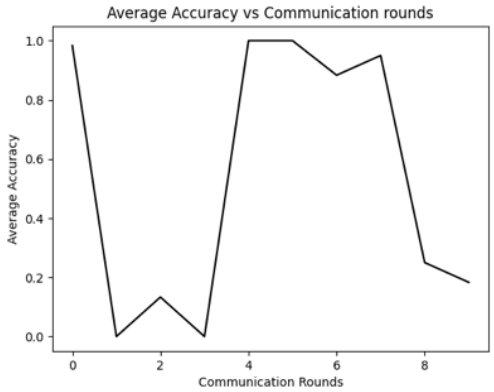
FL Model | Balanced | non-IID | Client Number = 5 | Local Epoch = 10 | Round = 10

Test Accuracy = 27.35 %



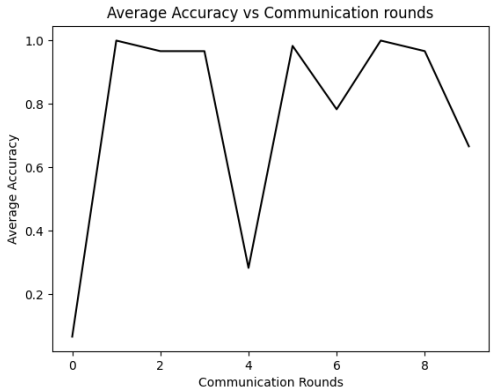
FL Model | Balanced | non-IID | Client Number = 50 | Local Epoch = 10 | Round = 10

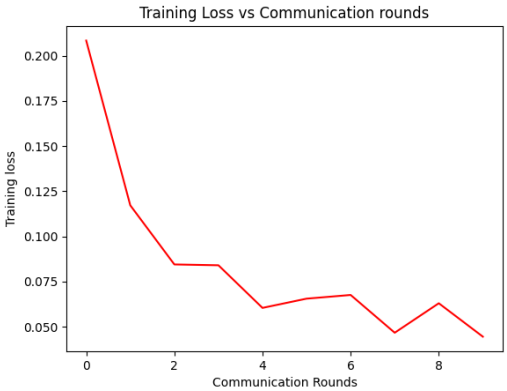
Test Accuracy = 65.02%



FL Model | Balanced | non-IID | Client Number = 100 | Local Epoch = 10 | Round = 10

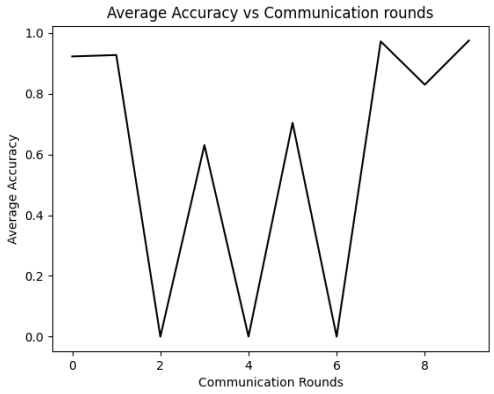
Test Accuracy = 68.61%





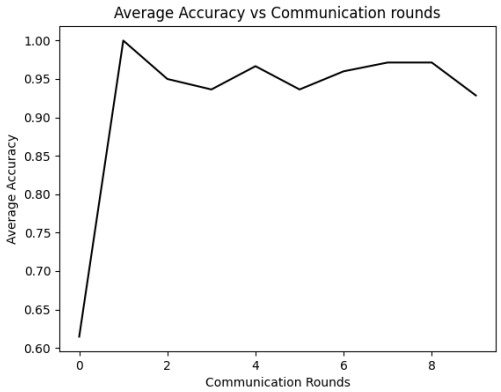
FL Model | Imbalanced | non-IID | Client Number = 5 | Local Epoch = 10 | Round = 10

Test Accuracy = 98.70%



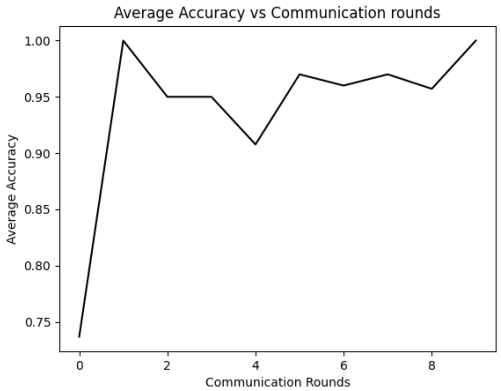
FL Model | Imbalanced | non-IID | Client Number = 50 | Local Epoch = 10 | Round = 10

Test Accuracy = 97.43%



FL Model | Imbalanced | non-IID | Client Number = 100 | Local Epoch = 10 | Round = 10

Test Accuracy = 96.68%



Comparisons

The results of the experiments will be compared based on different parameters. This approach allows for a clearer observation of how any given features (data distribution, client number, etc.) impacts the outcomes.

Model	Client Number	Test Accuracy
FL | Balanced | IID	5	**98.77%**
FL | Balanced | IID	100	**97.20%**
FL | Balanced | IID	500	**93.49%**

Table 1

⇒ Models with Balanced IID data distribution were trained with different numbers of clients. The conclusion is that as the number of clients increases, the model's accuracy decreases.

Model	Client Number	Test Accuracy
FL | Balanced | non-IID	5	**27.35%**
FL | Balanced | non-IID	50	**65.02%**
FL | Balanced | non-IID	100	**68.61%**

Table 2

⇒ Models with Balanced non-IID data distribution were trained with different numbers of clients. The conclusion is that as the number of clients increases, the model's accuracy increases.

Model	IID or non-IID	Test Accuracy
FL | Balanced | Client Number = 5	IID	**98.77%**
FL | Balanced | Client Number = 5	non-IID	**27.35%**

Table 3

Model	IID or non-IID	Test Accuracy
FL | Balanced | Client Number = 100	IID	**97.20%**
FL | Balanced | Client Number = 100	non-IID	**68.61%**

Table 4

⇒ Models with Balanced data distribution were trained using either IID or non-IID data distributions. The conclusion is that, for the same number of clients, IID data distribution achieves **significantly higher** accuracy compared to non-IID.

Model	Client Number	Test Accuracy
FL | Imbalanced | non-IID	5	**98.70%**
FL | Imbalanced | non-IID	50	**97.43%**
FL | Imbalanced | non-IID	100	**96.68%**

Table 5

⇒ Models with Imbalanced non-IID data distribution were trained with different numbers of clients. The conclusion is that as the number of clients increases, the model's accuracy decreases slightly.

Model	IID or Imbalanced	Test Accuracy
FL |Client Number = 5	IID	**98.77%**
FL |Client Number = 5	Imbalanced	**98.70%**

Table 6

Model	IID or Imbalanced	Test Accuracy
FL |Client Number = 100	IID	**97.20%**
FL |Client Number = 100	Imbalanced	**96.68%**

Table 7

⇒ Models with the same client number were trained using either Balanced IID or Imbalanced data distribution. The conclusion is that, for the same number of clients, Balanced IID data distribution achieves **slightly higher** accuracy compared to Imbalanced.

Model	Non-IID or Imbalanced	Test Accuracy
FL |Client Number = 5	Non-IID	**27.35%**
FL |Client Number = 5	Imbalanced	**98.70%**

Table 8

Model	Non-IID or Imbalanced	Test Accuracy
FL |Client Number = 100	Non-IID	**68.61%**
FL |Client Number = 100	Imbalanced	**96.68%**

Table 9

⇒ Models with the same client number were trained using either Balanced non-IID or Imbalanced data distribution. The conclusion is that, for the same number of clients, Imbalanced data distribution achieves **significantly higher** accuracy compared to Balanced non-IID.

Summary

The experiments conducted in this project showed the performance of Federated Learning (FL) under different conditions. Balanced IID distributions consistently yielded the highest accuracy. The performance of models with Balanced non-IID is significantly lower than the other. However, as the number of clients increases, there are noticeable improvement in the performance. Interestingly, imbalanced non-IID distributions showed better adaptability compared to balanced non-IID setups. It highlights FL's potential even with an imbalanced data distribution when appropriately configured.

FL holds significant promise for future applications in data-sensitive fields such as healthcare and cybersecurity. It can redefine how organizations manage distributed datasets due to its ability to train models collaboratively. However, addressing non-IID performance gaps and optimizing communication efficiency will be critical to unlocking its full potential.

References

[1] AshwinRJ, "ASHWINRJ/Federated-Learning-PyTorch: Implementation of Communication- efficient learning of Deep Networks from Decentralized Data," *GitHub*, https://github.com/AshwinRJ/Federated-Learning-PyTorch (accessed Jan. 1, 2025)

Some of the code in this repository was used as-is, while other parts were modified to suit the needs of this project:

[2] Tecperson, "Virus-mnist," *Kaggle*, https://www.kaggle.com/datasets/datamunge/virusmnist (accessed Jan. 1, 2025)