Aprenentatge Automàtic per a Xarxes (ML4Net)

Seminar 5 - Problem statement

May 22, 2025

Abstract

In this last seminar, you will be introduced to Federated Learning (FL), a popular technique to train Machine Learning (ML) models collaboratively. Once again, you will work on the pose estimation problem viewed in Seminar 2, but this time data will be distributed among multiple clients. The dataset that you are going to use in this seminar (`dataset_Seminar5.zip`) contains Wi-Fi signals (in the form of Channel State Information matrix) measured when a person was doing a given pose (e.g., walk, run, sit down, etc.).[1] Instead of having a single training dataset, you will have datasets from $K = 10$ clients, and your job will be to train a global model with the individual contributions of each client.

1 Part I: ML model definition

The dataset is known by you, so you are already able to load and process it as needed. For that reason, you will start focusing on the ML model that you will train collaboratively through FL. In this regard, you are free to choose the model architecture that you believe (you can also work with the linear regressor that you implemented in Seminar 2). Remember that we are dealing with a classification problem, which may have implications in your model architecture (e.g., selected loss function, type of activations).

2 Part II: Preparation of the FL setting

To train a model collaboratively (through FL), you have to define a central orchestrator (or parameter server) which must perform the following operations:

1. Initialize a global ML model, $\omega(t = 0)$)
2. Select a subset of clients $\mathcal{S} \subseteq \mathcal{K}$ to participate in the current training iteration.
3. Send the global model to the clients, which is retrained by the clients using their local datasets ($\mathcal{D}^k, \forall k \in \mathcal{S}$).

[1] Zhou, Y., Xu, C., Zhao, L., Zhu, A., Hu, F., & Li, Y. (2022). CSI-former: Pay more attention to pose estimation with WiFi. Entropy, 25(1), 20.

4. Retrieve the individual models $\omega^k(t), \forall k \in \mathcal{S}$ from the selected clients.

5. Aggregate the individual contributions to update the global model, $\omega(t+1)$. To perform model aggregation, consider using FedAvg, whereby a weighted average of the model weights is calculated. The weights (α) are given by the total share of data of each client, $\alpha^k = \frac{|\mathcal{D}^k|}{\sum_{k \in \mathcal{S}} |\mathcal{D}^k|}$.

6. Repeat steps 2-5 until convergence.

3 Part III: Collaborative training of the model

Once everything is ready, it is time to train and evaluate the model. For that:

- Define the required hyperparameters to collaboratively train your ML model (e.g., number of FL iterations, number of epochs for local model training, batch size, etc.).
- Train the model using FL using the training dataset (`client_k_features.csv` and `client_k_labels.csv`).
- Evaluate the model using the test data partition (`test_features.csv` and `test_labels.csv`).