AI/ML in Networking

Lecture 3: Wi-Fi Sensing & Federated Learning

Francesc Wilhelmi (francisco.wilhelmi@upf.edu)

Università degli Studi di Cagliari

Overview

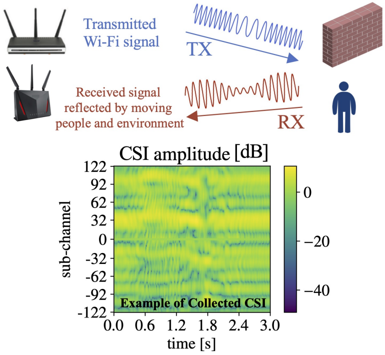
1. Wi-Fi Sensing
2. Pose Estimation
3. Federated Learning

Outline

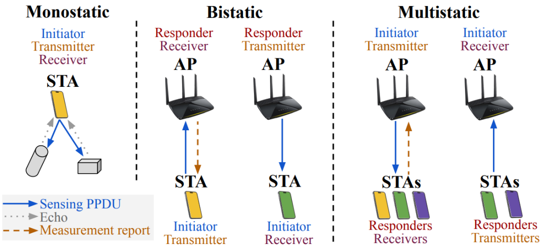
1. Wi-Fi Sensing
2. Pose Estimation
3. Federated Learning

Sensing

- Wireless signal transmission creates multipath propagation as signals reflect off objects and people in the environment.
- Channel Frequency Response (CFR) captures the signal modifications caused by these paths.
- This wireless information allows for the passive sensing of movement, presence, and other characteristics of people and objects.
    - Channel State Information (CSI): A fine-grained estimate of the CFR.
    - Received Signal Strength Indicator (RSSI): A measure of signal power.
    - Packet Arrival Time: Timing information often used for ranging or velocity.



Channel State Information (II)



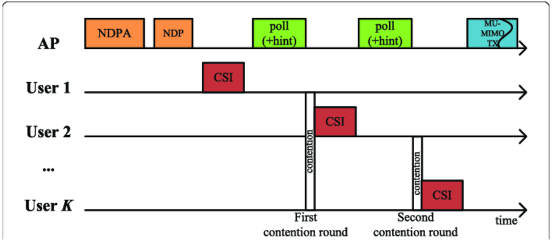
Sensing modes:

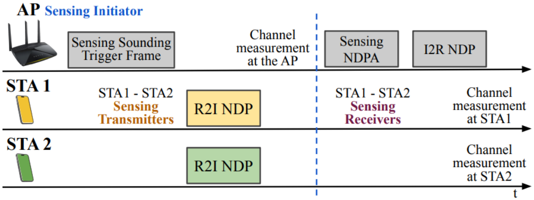
- *Monostatic*: a single device transmits a signal and collects the echoes from the environment. [1]
- *Bistatic*: the transmitter and receiver are distinct devices, and each of them can act as an initiator.
- *Multistatic*: more devices are involved.

[1] Geraci, G., Meneghello, F., Wilhelmi, F., Lopez-Perez, D., Val, I., Giordano, L. G., ... & Bellalta, B. (2026). Wi-Fi: Twenty-five years and counting. *Proceedings of the IEEE*.

IEEE 802.11bf:

- Wi-Fi sensing < 7 GHz & mmWave.
- Leverages channel estimation for data decoding and precoding (originally for MIMO & beamforming).
- Defines the procedure for obtaining channel measurements between ≥ 2 (bistatic & multistatic), or between the TX and RX antennas of a single device (monostatic).
- Sensing operations are separated from data communication.





Outline

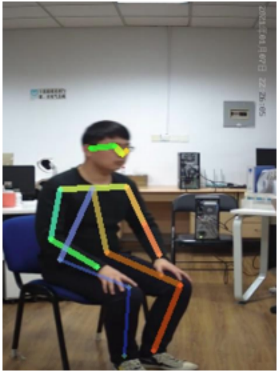
1. Wi-Fi Sensing
2. Pose Estimation
3. Federated Learning

Dataset (I)

How were the data generated?[a]

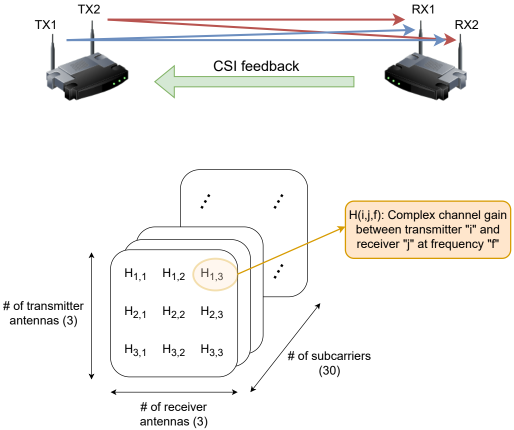
- Real Wi-Fi measurements were taken when people were doing specific poses.
    - Walk, sit down, run, stand up, bend...
- The features are the Wi-Fi measurements (the CSI, in this case).
- The labels (poses) were taken using cameras (Alphapose).

Full dataset available here.



[a] 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.

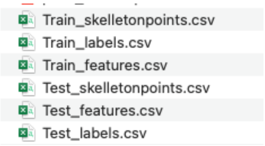
Dataset (II)



- Each element contains the channel gain for each antenna pair and subcarrier
- In the dataset, we have 5 CSI data packets of $30 \times 3 \times 3$ corresponding to 1 image (1 pose)
  - 30 are the number of subcarriers
  - 3×3 are the 3 Wi-Fi transmitter and receiver antennas

Dataset (III)

- Train features (`Train_features.csv`): CSI measurements in the training dataset.
  - 1000 samples $\times$ 270 (a flattened CSI matrix of 30$\times$3$\times$3)
- Train labels (`Train_labels.csv`): actual pose in the training dataset.
  - 1000 samples of an integer between 1 and 5 ({'wave', 'push', 'crouch', 'sitdown', 'bend'})
- Test features (`Test_features.csv`): CSI measurements in the test dataset.
  - 200 samples $\times$ 270 (a flattened CSI matrix of 30$\times$3$\times$3)
- Test labels (`Test_labels.csv`): actual pose in the test dataset.
  - 200 samples of an integer between 1 and 5 ({'wave', 'push', 'crouch', 'sitdown', 'bend'})

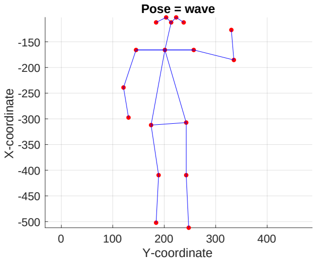


We will use a simplified version of a dataset that can be found at: `https://github.com/NjtechCVLab/Wi-PoseDataset?tab=readme-ov-file`

Dataset (IV)

In addition, `SkeletonPoints` are provided:

- Only for visualization purposes
- Each skeleton contains 54 points:
    - x coordinates (from 1 to 18)
    - y coordinates (from 19 to 36)
    - Confidence of each coordinate (from 37 to 56)
- Pose data are encoded using MPII[a]




```
% Define connections between the 18 keypoints
connections_18 = [
    1, 2;     % Nose - Neck
    2, 3;     % Neck - Right Shoulder
    3, 4;     % Right Shoulder - Right Elbow
    4, 5;     % Right Elbow - Right Wrist
    2, 6;     % Neck - Left Shoulder
    6, 7;     % Left Shoulder - Left Elbow
    7, 8;     % Left Elbow - Left Wrist
    2, 9;     % Neck - Right Hip
    9, 10;    % Right Hip - Right Knee
    10, 11;   % Right Knee - Right Ankle
    2, 12;    % Neck - Left Hip
    12, 13;   % Left Hip - Left Knee
    13, 14;   % Left Knee - Left Ankle
    1, 15;    % Nose - Right Eye
    1, 16;    % Nose - Left Eye
    15, 17;   % Right Eye - Right Ear
    16, 18;   % Left Eye - Left Ear
    3, 6;     % Right Shoulder - Left Shoulder
    9, 12;    % Right Hip - Left Hip
];
```

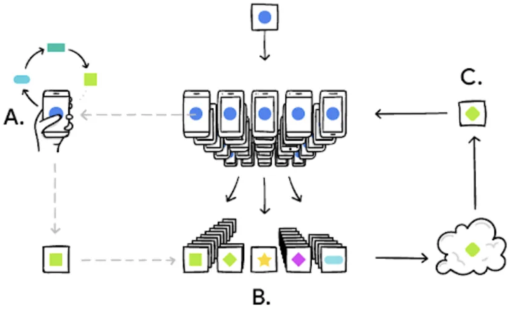
[a] `https://www.mpi-inf.mpg.de/departments/computer-vision-and-machine-learning/software-and-datasets/mpii-human-pose-dataset`

Outline

1. Wi-Fi Sensing
2. Pose Estimation
3. Federated Learning

What is Federated Learning (FL)?

- A distributed machine learning paradigm that enables models to be trained on decentralized datasets.
- Data remains on local devices (clients), while only model updates (e.g., gradients) are shared with a central server.
- Aims to achieve model training while ensuring data privacy and security.



Benefits of Federated Learning

- **Data Privacy:** Keeps sensitive user data on local devices, reducing privacy risks.
- **Data Security:** Less vulnerable to data breaches as raw data is not aggregated.
- **Generalization:** Can potentially lead to more robust global models than for centralized learning.
- **Resource Efficiency:** Reduces communication overhead by only sending model updates.
- **Collective power:** Leverages the computation/storage resources from multiple parties.

Federated Learning Applications

- **Mobile Keyboards (Google):** Improving word prediction and emoji suggestions without sending typing data.
- **Healthcare:** Training models on patient data across different hospitals while preserving patient privacy.
- **IoT Devices:** Smart homes, industrial sensors for predictive maintenance.
- **Financial Services:** Fraud detection models across different banks.
- **Autonomous Driving:** Training models on diverse vehicle sensor data.

https://research.google/blog/improving-gboard-language-models-via-private-federated-analytics/

Centralized vs Federated Learning

Centralized Learning

- All data is collected at a central point.
- A single model is trained on the entire dataset.
- **Drawbacks:** Privacy concerns, security risks, scalability.

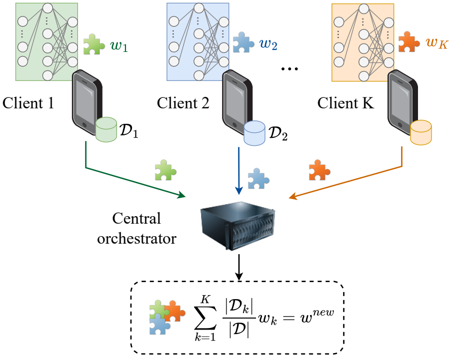
Federated Learning

- Data remains on client devices.
- Clients train local models using their own data.
- The server aggregates model updates to form a global model.

Federated Averaging (FedAvg)

Iterative Process:

1. Central server sends a **global model** $\omega(t)$ to selected clients.
2. Clients train $\omega(t)$ on their local data D_k (for E epochs) and obtain a **local model update** $\omega_k(t)$.
3. Clients send $\omega_k(t)$ back to central server.
4. Central server **aggregates** local updates to create a new global model $\omega(t+1) = \sum_{k=1}^{K} \frac{n_k}{K} \omega_{t+1}^{k}$.
5. The process is repeated for a number (T) of **iterations**.



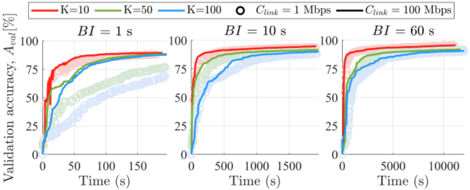
Federated Averaging (FedAvg)

Objective: $\min_\omega (F(\omega) = \sum_{k=1}^{K} \frac{|\mathcal{D}_k|}{|\mathcal{D}|} F_k(\omega))$

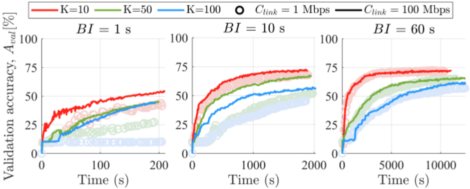
- K: total number of clients.
- $|\mathcal{D}_k|$: number of data samples on client k.
- $|\mathcal{D}| = \sum_{k=1}^{K} |\mathcal{D}_k|$: total number of data samples across all clients.
- $F_k(\omega)$: local loss function on client k's data $\mathcal{D}_k$.

Factors influencing convergence:

- **Learning Rate (η):** How much the model weights are adjusted.
- **Number of Local Epochs (E):** How many SGD steps clients take locally.
- **Client Heterogeneity (Non-IID):** How much client data distributions differ.
- **Number of Clients (K) and Participation Rate (C):** Scalability and sampling bias.
- **Variance of Gradients:** Noise from mini-batch SGD.



(b) Validation accuracy (MNIST).

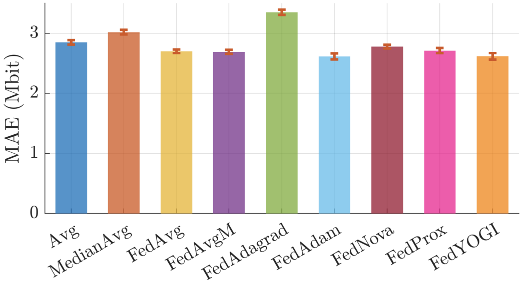


(d) Validation accuracy (CIFAR-10).

Wilhelmi, Francesc, et al. "The implications of decentralization in blockchained federated learning: Evaluating the impact of model staleness and inconsistencies." Computer Networks 245 (2024): 110361.

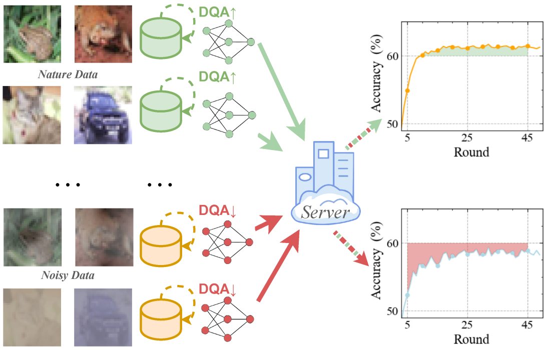
Model Aggregation Strategies

- **FedProx:** Adds a proximal term to the local objective function to constrain local updates, pushing them closer to the global model.
- **FedAdam / FedAdagrad / FedYogi:** Adaptations of adaptive optimizers (Adam, Adagrad, Yogi) for the server-side aggregation.
- **Other (e.g., Byzantine Tolerant):**
  - **Krum:** Selects clients whose updates are closest to a central point, discarding outliers.
  - **Trimmed Mean/Median:** Discards extreme values (e.g., 10% highest/lowest updates).



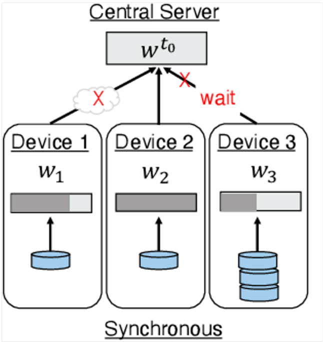
Client Selection

- In FL, not all clients might participate in every training round (e.g., not all clients are always online).
- Client selection must ensure diversity (selected clients provide diverse data to generalize well) and fairness (all clients contribute over time).
- Client selection strategies:
  - Random
  - Availability-based selection
  - Resource-aware selection (clients with more computational power or bandwidth)
  - Data-aware selection (clients with more data or diverse date)
  - Contribution-based selection (clients that contribute more to model improvement)

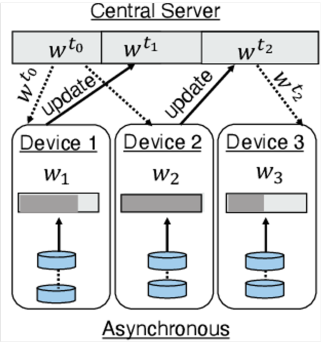


Sync vs Async Federated Learning

Synchronous FL



Asynchronous FL



Jeong, E., Zecchin, M., & Kountouris, M. (2022). Asynchronous Decentralized Learning over Unreliable Wireless Networks. arXiv preprint arXiv:2202.00955.

Federated Pose Estimation (I)

- Train features (`client_k_features.csv`, inside client_datasets folder): CSI measurements in the training dataset.
  - A random number of samples with a flattened CSI matrix of $30 \times 3 \times 3$.
- Train labels (`client_k_labels.csv`, inside client_datasets folder): actual pose in the training dataset.
  - A random number of samples with an integer between 1 and 12 (corresponding to poses {'wave', 'push', 'crouch', 'sitdown', 'bend', etc.})
- Test features (`test_features.csv`): CSI measurements in the test dataset.
  - 500 samples $\times$ 270 (a flattened CSI matrix of $30 \times 3 \times 3$)
- Test labels (`test_labels.csv`): actual pose in the test dataset.
  - 500 samples of an integer between 1 and 12 (corresponding to poses {'wave', 'push', 'crouch', 'sitdown', 'bend', etc.})

Federated Pose Estimation (II)

- Data is distributed among $N = 10$ clients (e.g., `client_1_features.csv`, `client_1_labels.csv`)
- Each client has data from [2, 8] classes.
- Between 10 and 100 samples are selected per class.
- The test dataset is apart and contains 500 samples (randomly selected among all the classes).