Federated Learning Project

OM GOVIND JHA, IISER, India
HARSH SHUKLA, IISER, India

ACM Reference Format:
Om Govind Jha and Harsh Shukla. . Federated Learning Project. 1, 1 (October), 6 pages. https://doi.org/

1 Introduction

Q1: Give a high-level overview of the research area/domain you are interested in.

Our research is focused on developing practical Federated Learning (FL) frameworks which tries to handle the primary obstacles to the real world deployment in heterogeneity which includes non-IID data heterogeneity that harms model accuracy and fairness, and system heterogeneity (e.g., varied device speeds and memory) that causes inefficiency and unfair participation.Our goal is to design holistic solutions that make FL robust, efficient, and fair enough for the vast ecosystem of resource-constrained edge devices.

Q2: Which research problem you are interested in the above mentioned area

Our specific research interest is designing a unified FL framework that overcomes the limitations of solutions that tackle heterogeneity in isolation. For instance, FedCHAR uses clustering for robustness against data heterogeneity but is limited by synchronous operations. Our proposed approach is to develop a hybrid framework that integrates robust clustering with memory-efficient techniques to manage severe memory constraints, creating a practical solution for the edge. Our objective is to make a combined framework which handles both the heterogeneity in data as well as the devices involved in the federated system.

Q3: Are there any related research works w.r.t. your problem? If yes, write a high level summary of these works after completing the Table

2 Methods

2.1 Datasets

Are there any relevant datasets available that could be used in your solution? If yes, briefly describe them

We will try to utilize some well-established datasets to evaluate our federated learning framework depending on the task we would do. **Human Activity Recognition Datasets:**

- **UCI-HAR Dataset**: Contains data from 30 subjects performing six activities (walking, walking upstairs, walking downstairs, sitting, standing, lying) using smartphone accelerometer and gyroscope sensors. The dataset includes 10,299 instances with 561 features extracted from time and frequency domain signals.
- **WISDM Dataset** [5]: Comprises accelerometer data from 36 subjects performing six activities (walking, jogging, upstairs, downstairs, sitting, standing). The dataset contains over 1 million raw sensor readings collected at 20Hz sampling frequency.
- **PAMAP2 Dataset**: Physical Activity Monitoring dataset with 18 different activities from 9 subjects wearing 3 inertial measurement units and heart rate monitors. Contains over 3.8 million instances with multivariate sensor data including accelerometer, gyroscope, and magnetometer readings.

Image Datasets:

- **MNIST**: Handwritten digit recognition dataset (28x28 grayscale images) commonly used for federated learning benchmarks.
- **CIFAR-10**: Natural image classification dataset with 60,000 32x32 color images across 10 classes, suitable for evaluating federated learning performance on complex visual data.

These datasets provide diverse characteristics like different data modalities (sensor data vs. images), varying degrees of heterogeneity, and different levels of complexity, making them ideal for the comprehensive evaluation of our federated learning approach.

2.2 Baselines

Are there any existing methods available with which you can compare your solution? If yes, briefly describe them here

We will compare our proposed framework against different state-of-the-art federated learning baselines:

Standard Federated Learning Methods:

- **FedAvg (Federated Averaging)** [4]: The fundamental federated learning algorithm that averages model weights from participating clients. Serves as the primary baseline for communication efficiency and convergence analysis.
- **FedSGD (Federated Stochastic Gradient Descent)**: A simpler baseline where clients perform single gradient updates before aggregation, useful for understanding the benefits of local training.

Clustering-based Federated Learning:

- **FedCHAR** [1]: Hierarchical clustering-based personalized federated learning specifically designed for HAR that creates specialized models for user groups while identifying malicious nodes.
- **ShuffleFL** [3]: Multi-device federated learning approach that addresses heterogeneity through data shuffling and adaptive client selection.

Authors' Contact Information: Om Govind Jha, om22@iiserb.ac.in, IISER, Bhopal, India; Harsh Shukla, harsh22@iiserb.ac.in, IISER, Bhopal, India.

Memory-Efficient Methods:

- **FedMef** [2]: Memory-efficient federated dynamic pruning framework with budget-aware parameter selection and activation pruning.

2.3 Methodology steps

Write in bullets about the steps you are planning to take in your solution

- **Data Preprocessing and Partitioning**: Prepare datasets with realistic non-IID distributions simulating real-world federated scenarios.
- **Memory-Efficient Model Architecture**: Understanding the steps where there is a lot of memory consumption and experimenting with different strategies to reduce it.
- **Adaptive Aggregation Strategy**: Implementing client contribution weighting based on data quality and quantity.
- **Comprehensive Evaluation**: Conduct extensive experiments across multiple datasets comparing accuracy, convergence speed, communication overhead, memory usage, and performance metrics.
- **Real-world Deployment Testing**: Validate the framework on resource-constrained devices to assess practical feasibility and performance under realistic constraints.

3 Work division

Fill Table 1 to show the work division among the team members.

Table 1. Group work division

Member name	Task
Om Govind Jha	Data sourcing and preprocessing, baseline implementation, lightweight clustering algorithm development, memory-efficient model optimization, Handling distribution shift in data
Harsh Shukla	Evaluation and result analysis, clustering validation, malicious node detection mechanisms, comprehensive performance benchmarking

4 Timeline

Create a timeline of your project.

- **Weeks 1-2**: Dataset identification, Literature Understanding, Problem Statement Setup
- **Weeks 3-4**: Baseline method implementation (FedAvg, FedSGD, FedCHAR) and initial experimental setup. Experimenting with the main paper code [1].
- **Weeks 5-6**: Brainstorming and implementing of lightweight clustering algorithms and memory-efficient techniques. Mid-Sem Report Preparation
- **Weeks 7-9**: Comprehensive experimentation across all datasets and baselines. Performance evaluation including accuracy, efficiency, and fairness metrics. Handling the problems in the implementation of our idea. Developing proof of concept for the proposed method.
- **Week 9-12**: Result analysis, documentation completion, and final report preparation. Comparison study, limitations and future work identification.

5 Our queries for the Meeting

- We observe that there are many directions in which we can do the work, but we are not sure which of them will be more aligned with our interests as well as the course objective.
- What are the possible problems we may face with the listed datasets? What kind of datasets will be good in general for our project?
- Are we going in right direction? What next steps can we take from here?

6 Data Formulation and Problem Setup

6.1 Dataset Description

We utilize the WISDM (Wireless Sensor Data Mining) Activity Recognition dataset [5], which contains smartphone accelerometer data from 36 users performing six daily activities: Walking (0), Jogging (1), Upstairs (2), Downstairs (3), Sitting (4), and Standing (5). The raw dataset comprises 1,086,465 valid sensor readings collected at 20Hz sampling frequency.

Preprocessing Pipeline:

- **Sliding Window Creation**: We apply a sliding window approach with window size of 200 timesteps (10 seconds at 20Hz) and 50% overlap (step size of 100), generating 10,591 windows.
- **Feature Normalization**: Three-axis accelerometer data (x, y, z) is normalized using StandardScaler to ensure consistent feature scales across all clients.
- **Data Shape**: Each sample has shape (200, 3), representing 200 timesteps with 3 features per timestep.

6.2 Non-IID Data Distribution

To simulate realistic federated learning scenarios where different users exhibit distinct activity patterns, we implement a **hard clustering** approach that introduces three types of heterogeneity:

1. Feature Distribution Skew (Hard Clustering):

- **Cluster 0** (Clients 0-9): Only Walking and Jogging activities (7,443 samples)
- **Cluster 1** (Clients 10-19): Only Upstairs and Downstairs activities (2,136 samples)
- **Cluster 2** (Clients 20-29): Only Sitting and Standing activities (1,012 samples)

This clustering creates natural groups based on activity similarity: locomotion activities (Cluster 0), stair navigation (Cluster 1), and stationary activities (Cluster 2).

2. Label Distribution Skew (Dirichlet Distribution): Within each cluster, we use Dirichlet distribution with $\alpha = 0.5$ to create

imbalanced label distributions between the two activities. For example, in Cluster 0, one client might have 80% Walking and 20% Jogging, while another has 30% Walking and 70% Jogging.

3. Quantity Skew (Log-Normal Distribution): Sample counts per client follow a log-normal distribution: $\text{samples}_i \sim \text{LogNormal}(\mu = 4.5, \sigma = 0.8) + 50$, ensuring each client has at least 50 samples but with significant variation (ranging from 50 to several hundred samples).

6.3 Problem Formulation

Given $K = 30$ clients with heterogeneous data distributions $\mathcal{D}_k$, the objective is to learn personalized models θ_k for each client k that maximize accuracy while ensuring fairness across clients. The optimization problem can be formulated as:

$$\min_{\{\theta_k\}_{k=1}^{K}} \frac{1}{K} \sum_{k=1}^{K} \mathcal{L}_k(\theta_k; \mathcal{D}_k) \tag{1}$$

subject to constraints on model size (for memory efficiency) and fairness (low variance in accuracy across clients).

7 Methodology

7.1 Baseline: FedCHAR

FedCHAR (Federated Clustering Hierarchical Adaptive Regularization) [1] addresses data heterogeneity through hierarchical clustering and personalized training with regularization.

Algorithm Overview:

(1) **Initial Training Phase** (10 rounds): All clients train with global model using FedAvg aggregation
(2) **Clustering Phase**: Clients are clustered based on cosine similarity of model updates using Agglomerative Hierarchical Clustering with complete linkage
(3) **Personalized Training Phase** (40 rounds): Within each cluster, clients train dual models:
- **Personal Model** (v_k): Tailored to client's local data
- **Group Model** (w_g): Shared within cluster

Loss Function: FedCHAR employs a regularized objective that balances local and cluster objectives:

$$\mathcal{L}_k^{\text{FedCHAR}} = \mathcal{L}_k(v_k; \mathcal{D}_k) + \frac{\lambda}{2} \|v_k - w_g\|^2 \tag{2}$$

where $\lambda = 0.01$ controls the trade-off between personalization and cluster coherence. The regularization term ensures personal models don't drift too far from the cluster model, enabling knowledge sharing while maintaining personalization.

Key Advantages:

- Exploits similarity between users to improve accuracy
- Enhances fairness by creating specialized models per cluster
- Naturally robust to attacks by isolating malicious nodes through clustering

7.2 Baseline: FedMef

FedMef (Federated Memory-efficient Framework) [2] focuses on reducing memory footprint through dynamic pruning while maintaining model accuracy.

Algorithm Overview:

(1) **Initialization**: Create randomly pruned model at target sparsity (80%)
(2) **Training Rounds**: Alternate between standard training and adjustment rounds
(3) **Standard Training**: Clients train sparse models normally
(4) **Adjustment Rounds** (every 10 rounds, up to round 30):
- Apply Budget-Aware Extrusion (BaE) training
- Aggregate client gradients
- Adjust masks: prune low-magnitude weights, grow high-gradient weights
(5) **Final Training** (rounds 31-50): Continue training with fixed masks

Budget-Aware Extrusion (BaE) Loss: BaE preserves information from weights marked for pruning by penalizing low-magnitude parameters:

$$\mathcal{L}_k^{\text{BaE}} = \mathcal{L}_k^{\text{CE}}(y, \hat{y}) + \lambda_{\text{BaE}} \sum_{w \in \Theta_{\text{low}}} w^2 \tag{3}$$

where Θ_{low} represents the bottom 20th percentile of weight magnitudes, and $\lambda_{\text{BaE}} = 0.01$. This forces information transfer from pruned weights to remaining weights before removal.

Dynamic Mask Adjustment: At adjustment rounds, FedMef:

- **Prunes** 20% of active weights with lowest magnitudes
- **Grows** same number of inactive weights with highest gradient magnitudes

This dynamic adjustment allows the sparse structure to adapt to changing data distributions during training.

Key Advantages:

- Significant memory reduction (80% sparsity achieved)
- Dynamic pruning adapts to data characteristics
- BaE prevents catastrophic accuracy loss during aggressive pruning

7.3 Proposed Method: CA-AFP

CA-AFP (Cluster-Aware Adaptive Federated Pruning) combines the strengths of clustering-based personalization and memory-efficient pruning through two novel components.

Architecture Overview:

(1) **Phase 1-2**: Initial training (10 rounds) + Clustering (same as FedCHAR)
(2) **Phase 3**: Train dense specialized models per cluster (30 rounds)
(3) **Phase 4**: Apply cluster-aware adaptive pruning
(4) **Phase 5**: Fine-tune pruned models (3 epochs per client)

Novel Component 1: Cluster-Aware Importance Scoring

Traditional magnitude-based pruning treats all weights independently. CA-AFP introduces a hybrid importance score that considers cluster-level information:

$$\text{Score}_w = \alpha \cdot \text{Magnitude}_w + \beta \cdot \text{Coherence}_w + \gamma \cdot \text{Consistency}_w \tag{4}$$

where $\alpha = 0.5$, $\beta = 0.25$, $\gamma = 0.25$, and:

- **Magnitude**: $\frac{|w|}{\max(|W|)}$ (normalized weight magnitude)
- **Coherence**: $\frac{1}{1+\text{Var}_{\text{clients}}(w)}$ (inverse of variance across cluster clients)

- **Consistency**: Similarity of gradients across cluster members (placeholder: 1.0)

Intuition: Weights with low variance across cluster members are consistently important and should be preserved. Weights that vary wildly across clients are candidates for pruning.

Novel Component 2: Adaptive Pruning Scheduler

Instead of uniform sparsity across all clusters, CA-AFP calculates cluster-specific target sparsity based on data complexity and cluster size:

$$s_c = s_{\text{base}} - 0.2 \cdot \frac{H(y_c)}{\log(6)} - 0.1 \cdot \min\left(\frac{|C_c|}{10}, 1\right) \tag{5}$$

where:

- $s_{\text{base}} = 0.7$ (base sparsity)
- $H(y_c) = -\sum_i p_i \log p_i$ (label entropy measuring data complexity)
- $|C_c|$ is the number of clients in cluster c
- Result is clipped to $[0.5, 0.9]$

Intuition: Complex clusters (high entropy) need more model capacity, so we reduce sparsity. Larger clusters with more diverse data also benefit from lower sparsity.

Pruning Algorithm:

(1) For each cluster c, calculate adaptive target sparsity s_c
(2) Compute hybrid importance scores for all weights in cluster model
(3) For each layer:
   - Sort weights by importance score
   - Prune bottom $s_c \times |\Theta|$ weights (set to zero)
(4) Fine-tune pruned models with 3 epochs of training per client

Key Innovations:

- **Cluster-aware scoring**: Preserves weights important across cluster, not just individual clients
- **Adaptive sparsity**: Different clusters get different compression ratios based on need
- **Information preservation**: Hybrid scoring prevents loss of critical cluster-level features

8 Experimental Results and Analysis

8.1 Experimental Setup

Model Architecture: LSTM-based model with architecture: LSTM(64) → Dropout(0.3) → LSTM(64) → Dropout(0.3) → Dense(32, ReLU) → Dropout(0.2) → Dense(6, Softmax)

Training Configuration:

- **FedCHAR**: 50 rounds (10 initial + 40 personalized), 3 epochs/round, $\lambda = 0.01$
- **FedMef**: 50 rounds, 3 epochs/round, target sparsity 80%, adjustment every 10 rounds
- **CA-AFP**: 100 rounds (10 initial + 30 dense training + pruning + fine-tuning), 3 epochs/round

Evaluation Metrics:

- **Average Accuracy**: Mean test accuracy across all clients
- **Fairness (Standard Deviation)**: Lower values indicate more uniform performance
- **Model Sparsity**: Percentage of zero weights (memory efficiency)
- **Per-Cluster Performance**: Accuracy breakdown by true clusters

8.2 Overall Performance Comparison

Table 2 presents the comprehensive comparison of all three methods across key metrics.

Table 2. Performance comparison of federated learning methods

Metric	FedCHAR	FedMef	CA-AFP (Ours)
Average Accuracy	91.62%	36.95%	**82.92%**
Std Dev (Fairness)	12.58%	39.70%	**28.14%**
Min Accuracy	53.85%	0.00%	**6.67%**
Max Accuracy	100.0%	100.0%	100.0%
Sparsity	0% (Dense)	**89.64%**	50-61% (Adaptive)
Cluster 0 Accuracy	83.14%	—	75.30%
Cluster 1 Accuracy	**99.14%**	—	97.30%
Cluster 2 Accuracy	**98.72%**	—	99.05%
Memory Efficiency	Low	**High**	Medium-High
Fairness Ranking	**1st**	3rd	2nd

8.3 Detailed Analysis

8.3.1 FedCHAR Performance. **Strengths**:

- **Highest Overall Accuracy** (91.62%): Benefits from dense models without pruning constraints
- **Best Fairness** (12.58% std dev): Clustering with personalized training ensures consistent performance
- **Excellent Cluster-wise Performance**: Cluster 1 (99.14%) and Cluster 2 (98.72%) achieve near-perfect accuracy

Limitations:

- **No Memory Efficiency**: Full dense model requires complete parameter storage on all devices
- **Cluster 0 Underperformance** (83.14%): Locomotion activities (Walking/Jogging) are harder to distinguish, even with personalization
- **Practical Deployment**: Not suitable for severely memory-constrained edge devices

8.3.2 FedMef Performance. **Strengths**:

- **Excellent Memory Efficiency** (89.64% sparsity): Achieves aggressive compression
- **Some High-Performing Clients**: A few clients (0, 1, 6, 9, 24, 27, 29) achieve 70-100% accuracy

Critical Limitations:

- **Catastrophic Failure** (36.95% avg accuracy): Aggressive uniform pruning destroys model capacity
- **Severe Unfairness** (39.70% std dev): 13 out of 30 clients have 0% accuracy (complete model collapse)

- **Ignores Data Heterogeneity**: Uniform pruning across all clients ignores clustering structure
- **BaE Insufficient**: Budget-aware extrusion alone cannot prevent information loss at 80% sparsity

Root Cause Analysis: FedMef's failure stems from applying uniform 80% sparsity without considering:

(1) Cluster-specific data complexity (e.g., Cluster 0 needs more capacity for similar activities)
(2) Importance of cluster-level features that need preservation
(3) Distribution of pruned parameters across cluster boundaries

8.3.3 CA-AFP Performance (Proposed). **Strengths**:

- **Balanced Trade-off** (82.92% accuracy): Only 8.7% accuracy loss compared to FedCHAR while achieving significant compression
- **Moderate Fairness** (28.14% std dev): Better than FedMef, though not matching FedCHAR
- **Adaptive Sparsity**: Cluster-specific compression preserves critical features
  - Cluster 0: 50.00% sparsity (more capacity for difficult locomotion activities)
  - Cluster 1: 60.62% sparsity
  - Cluster 2: 60.31% sparsity
- **Strong Cluster 1 & 2 Performance**: 97.30% and 99.05% accuracy are very close to FedCHAR accuracies
- **No Complete Failures**: Minimum accuracy 6.67% vs 0% in FedMef

Limitations:

- **Cluster 0 Performance** (75.30%): Still challenges remain in distinguishing similar activities even with adaptive pruning
- **Fairness Gap**: Higher variance than FedCHAR (though much better than FedMef)
- **Complex Training Pipeline**: Requires careful coordination of clustering, dense training, and adaptive pruning phases

8.4 Key Insights

1. Clustering is Essential for Non-IID Data: Both FedCHAR and CA-AFP leverage clustering to achieve cluster-specific performance (97-99% for Clusters 1 & 2), while FedMef's global approach fails catastrophically.

2. Adaptive Pruning Outperforms Uniform Pruning: CA-AFP's cluster-aware adaptive sparsity (50-61%) achieves 82.92% accuracy, while FedMef's uniform 89.64% sparsity collapses to 36.95% accuracy. This demonstrates that *where* we prune matters more than *how much* we prune.

3. Accuracy-Memory Trade-off:

- FedCHAR: 91.62% accuracy, 0% sparsity (baseline)
- CA-AFP: 82.92% accuracy, 50-61% sparsity (**9% accuracy loss for 50%+ memory savings**)
- FedMef: 36.95% accuracy, 89.64% sparsity (very high accuracy loss)

4. Cluster-Aware Importance Scoring Works: CA-AFP's hybrid scoring (magnitude + coherence + consistency) successfully identifies weights critical to cluster-level performance, preventing the catastrophic failures seen in FedMef.

5. Activity Complexity Varies by Type:

- **Stationary Activities** (Sitting/Standing): Easiest (98-99% accuracy)
- **Stair Navigation** (Upstairs/Downstairs): Medium (97-99% accuracy)
- **Locomotion** (Walking/Jogging): Hardest (75-83% accuracy) - require more model capacity

8.5 Conclusions

Where to Select which model:

- **Choose FedCHAR if**: Memory is not constrained, and maximum accuracy/fairness is priority
- **Choose CA-AFP(ours) if**: Memory constraints exist, but reasonable accuracy (80%+) is required
- **Avoid FedMef for**: Intense Non-IID federated settings without clustering integration

Our Contribution: CA-AFP successfully bridges the gap between accuracy and memory efficiency through:

(1) Cluster-aware hybrid importance scoring that preserves critical cluster-level features
(2) Adaptive sparsity targets based on data complexity and cluster size
(3) Information preservation through fine-tuning of pruned models

The results validate that our method of **cluster-aware pruning is better than uniform compression**, achieving practical memory savings while maintaining usable accuracy in highly data heterogeneous federated learning scenarios.

9 Next Steps - Work to be Done

While CA-AFP demonstrates promising results in combining clustering-based personalization with adaptive pruning, we plan to explore further for improvement and better experimentation of our model :

9.1 Algorithm Enhancements

- **Improved Gradient Consistency Scoring**: Currently, the consistency component in our hybrid importance scoring uses a placeholder value of 1.0. Future work should implement actual gradient similarity computation across cluster members to better identify weights that contribute consistently to cluster performance.
- **Dynamic Cluster Reassignment**: Investigate adaptive clustering mechanisms that allow clients to migrate between clusters as their data distributions evolve over training rounds, rather than maintaining fixed cluster assignments after initial clustering.
- **Layer-wise Adaptive Sparsity**: Extend the adaptive sparsity mechanism to operate at the layer level, allowing different layers within the same cluster model to have different sparsity targets based on their importance to task performance.

9.2 Scalability and Efficiency

- **Communication Overhead Analysis**: Conduct detailed analysis of communication costs associated with cluster-aware training and compare with baseline methods. Explore compression techniques to reduce model transmission overhead.
- **Larger-Scale Experiments**: Evaluate CA-AFP on larger federated settings with 100+ clients and multiple datasets (CIFAR-10, UCI-HAR) to assess scalability and generalization capabilities.
- **Real Device Deployment**: Test the framework on actual resource-constrained devices (smartphones, wearables) to measure real-world memory usage, inference latency, and energy consumption.

9.3 Robustness and Security

- **Robustness**: Evaluate CA-AFP's resilience to malicious clients attempting to poison cluster models or disrupt the pruning process. Investigate integration with robust aggregation techniques.
- **Privacy Analysis**: Conduct formal privacy analysis of cluster-aware pruning, particularly examining whether pruning patterns leak information about cluster membership or data distributions.

9.4 Theoretical Analysis

- **Convergence Guarantees**: Develop theoretical convergence bounds for CA-AFP under non-IID data distributions, analyzing how adaptive pruning affects convergence rates compared to dense models.
- **Fairness Metrics**: Investigate additional fairness metrics beyond standard deviation, such as worst-case performance guarantees and equitable resource allocation across clusters.

9.5 Application Domains

- **Cross-Domain Evaluation**: Apply CA-AFP to other federated learning domains beyond HAR, such as medical imaging, natural language processing, and IoT sensor networks.
- **Continual Learning**: Explore integration with continual learning techniques to handle concept drift and evolving activity patterns over extended deployment periods.

10 Related Work

Table 3 presents a comparative analysis of existing federated learning approaches that address heterogeneity and efficiency challenges. FedCHAR [1] leverages hierarchical clustering for personalized HAR models but lacks memory efficiency considerations. FedMef [2] achieves significant memory reduction through dynamic pruning but struggles with severe data heterogeneity. ShuffleFL [3] addresses heterogeneity through adaptive client selection but provides limited personalization mechanisms. Our proposed CA-AFP framework combines the strengths of clustering-based personalization with adaptive pruning strategies to achieve both accuracy and memory efficiency in heterogeneous federated learning scenarios.

Table 3. Comparative analysis of related works

Paper	Summary	Limitations
FedCHAR [1]	Uses hierarchical clustering to create personalized models for user groups in HAR, improving accuracy and fairness while identifying malicious nodes.	Clustering introduces computational overhead; limited evaluation on memory-constrained devices and scalability concerns for large deployments.
FedMef [2]	Introduces memory-efficient federated dynamic pruning with budget-aware extrusion, achieving 28.5% memory reduction while maintaining accuracy.	Pruning strategy may not suit all architectures; requires careful parameter tuning that varies across scenarios. Struggles with severe non-IID data.
ShuffleFL [3]	Addresses heterogeneity through data shuffling and adaptive client selection, improving convergence for non-IID data distributions.	Shuffling may introduce privacy risks and complexity; limited analysis of personalization versus global performance trade-offs.

References

[1] Youpeng Li, Xuyu Wang, and Lingling An. 2023. Hierarchical Clustering-based Personalized Federated Learning for Robust and Fair Human Activity Recognition. *ACM Transactions on Intelligent Systems and Technology* (2023).

[2] Hong Huang, Weiming Zhuang, Chen Chen, and Lingjuan Lyu. 2024. FedMef: Towards Memory-efficient Federated Dynamic Pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 27538–27547.

[3] Ran Zhu, Mingkun Yang, and Qing Wang. 2024. ShuffleFL: addressing heterogeneity in multi-device federated learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(2), 1–34.

[4] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[5] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. 2011. Activity Recognition using Cell Phone Accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2), 74–82.