MANTECH Publications
Journal of Artificial Intelligence, Machine Learning and Soft Computing
Volume 10, Issue 2, July-December 2025

Federated Learning & Collaborative Machine Learning: Concepts, Challenges, and Applications

Ankur S Tyagi[1], Sohan Chauhan[2]
Assistant Professor[1], Associate Professor[2]
Department of AI for Smart Cities
Blue Ridge University, India
Email Id: *Ankurssstyagi@rediffmail.com[1], Chauhan_sohan02@gmail.com[2]*

ABSTRACT

The rapid growth of data in decentralized environments, coupled with privacy concerns, has driven the development of ***Federated Learning (FL)*** *and* ***Collaborative Machine Learning (CML)*** *paradigms. These approaches allow multiple parties to collaboratively train machine learning models without sharing raw data, thereby preserving privacy, improving security, and reducing communication overhead. This paper provides a comprehensive review of FL and CML, highlighting their architectures, algorithms, optimization techniques, privacy mechanisms, and real-world applications. We also discuss the challenges, recent advances, and future directions, emphasizing the importance of efficient model aggregation, robustness against attacks, and regulatory compliance. Tables and figures illustrate system architectures, communication frameworks, and performance comparisons across various domains. This review aims to serve as a reference for researchers, practitioners, and policy-makers working in decentralized and privacy-preserving machine learning.*

KEYWORDS: *Federated Learning, Collaborative Machine Learning, Privacy-Preserving AI, Decentralized Learning, Model Aggregation, Data Security*

INTRODUCTION

Machine learning (ML) has become an integral part of modern technology, driving applications from healthcare to finance. Traditional ML approaches rely on centralized datasets, requiring sensitive data to be collected and stored in a single location. This centralization raises

significant concerns related to **data privacy**, **regulatory compliance**, and **data ownership**.

Federated Learning (FL) and **Collaborative Machine Learning (CML)** emerge as solutions to these challenges. In FL, multiple clients train a shared global model collaboratively while keeping their data local. The model parameters, rather than raw data, are communicated and aggregated at a central server or through decentralized protocols. CML extends the concept to broader collaborative frameworks, including multi-institutional research networks and cross-organizational learning systems.

This paper systematically reviews the concepts, methods, challenges, and applications of FL and CML. We focus on the architectures, learning algorithms, privacy-preserving techniques, and recent advances in the field.

BACKGROUND

The background section establishes the foundational concepts of **Federated Learning (FL)** and **Collaborative Machine Learning (CML)** by comparing them with traditional machine learning approaches. Understanding these distinctions is crucial to appreciate why decentralized learning paradigms are necessary in today's privacy-sensitive environment.

2.1 Traditional Machine Learning vs Collaborative Paradigms

Traditional Machine Learning (ML) relies heavily on centralized data collection. The standard workflow in traditional ML typically includes three major steps:

1. **Centralized Data Collection:**

Data from multiple sources is aggregated into a central repository, such as a cloud server or a data center. This step allows ML models to access large volumes of labeled or unlabeled data. Centralized storage simplifies data preprocessing, feature engineering, and model development.

2. **Model Training on Consolidated Datasets:**

Once the data is collected, ML algorithms (e.g., deep neural networks, decision trees, support vector machines) are trained on the combined dataset. Centralization allows high computational efficiency and better model accuracy due to the availability of large and diverse datasets.

3. **Deployment of the Trained Model:**

After training, the model is deployed for inference. Users or devices interact with the trained model through APIs or embedded software.

While this centralized approach has proven effective, it faces several critical limitations:

- **Privacy Concerns:** Sensitive data, such as medical records or financial transactions, must be shared with central servers, increasing the risk of breaches.
- **Regulatory Constraints:** Laws such as the **General Data Protection Regulation (GDPR)** in Europe or **HIPAA** in the United States restrict cross-border data sharing and impose strict consent requirements.
- **Data Ownership Issues:** Organizations and individuals are often unwilling to transfer proprietary or personal data to external servers.
- **Communication Overhead:** Transmitting massive amounts of raw data to a central server can be costly and slow, especially in distributed systems like IoT networks.

Collaborative paradigms, including FL and CML, have emerged as solutions to these challenges by enabling **decentralized model training**. Instead of moving data to the model, these paradigms move the model to the data. Key advantages include:

- **Preservation of Privacy:** Raw data never leaves the local environment, reducing exposure to unauthorized access.
- **Compliance with Regulations:** Organizations can maintain local control of sensitive data while still participating in joint model training.
- **Resource Efficiency:** Only model parameters or gradients are communicated, lowering network bandwidth requirements.
- **Enhanced Collaboration:** Multiple entities can jointly improve model performance without compromising data confidentiality.

2.2 Federated Learning

Federated Learning (FL) is a decentralized machine learning paradigm introduced by **Google** in 2016. It was designed initially to enable mobile devices to collaboratively improve predictive models, such as **next-word prediction** on keyboards, without transmitting user data to central servers. Over time, FL has expanded to other domains, including healthcare, finance, and IoT. The **standard FL workflow** includes three main stages:

1. **Local Training:**

Each client (e.g., a smartphone, IoT device, or organizational server) trains a local model using its private dataset. Training occurs independently on each client, allowing the model to capture unique patterns in local data.

2. **Parameter Aggregation:**

After local training, clients send only the **model updates** (e.g., gradients or weights) to a central aggregation server. The raw data never leaves the client. The server aggregates these updates using algorithms such as **Federated Averaging (FedAvg)**. FedAvg computes a weighted average of client updates based on their dataset sizes:

wt+1=∑k=1Knknwtkw_{t+1} = \sum_{k=1}^K \frac{n_k}{n} w_t^kwt+1=k=1∑Knnkwtk

Where:

- wtkw_t^kwtk is the local model weight for client kkk at iteration ttt,
- nkn_knk is the number of samples in client kkk,
- nnn is the total number of samples across all clients.

This aggregation step ensures that the global model benefits from all client datasets without exposing any raw data.

3. **Model Update:**

The aggregated global model is redistributed to all participating clients. Clients use the updated global model for the next round of local training. This iterative process continues until the model converges.

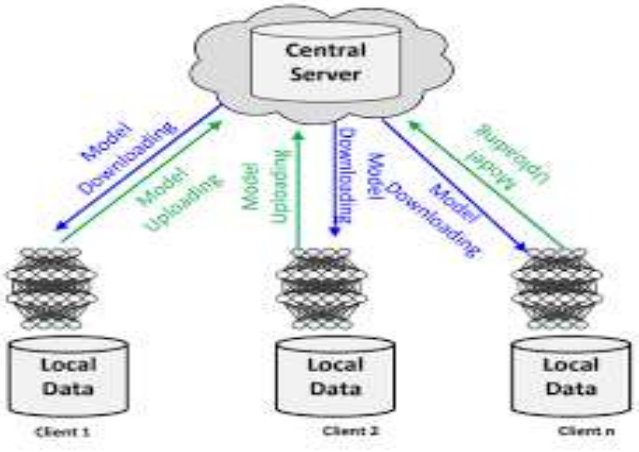

Figure 1: Federated Learning Architecture

2.3 Collaborative Machine Learning

Collaborative Machine Learning (CML) is a broader paradigm that builds on the principles of Federated Learning (FL) but emphasizes **cooperation, knowledge sharing, and joint optimization** among multiple entities or organizations. While FL focuses primarily on **privacy-preserving distributed training**, CML allows different organizations to leverage each other's models, insights, or partially overlapping datasets to improve global model performance, often across heterogeneous environments.

CML can be particularly valuable in scenarios where:

- Data cannot be directly shared due to privacy, security, or competitive concerns.
- Collaborative insights are required across organizations or institutions with complementary data.
- Complex tasks require combining multiple expertise domains.

Key Characteristics of CML:

1. **Decentralized Knowledge Sharing:** Unlike traditional ML, where a single centralized dataset drives learning, CML enables multiple organizations to share model knowledge (e.g., parameters, feature representations, or predictions) without exchanging raw data.
2. **Cooperative Optimization:** Participants collaboratively optimize a shared global objective, often balancing local model performance with overall system accuracy.
3. **Heterogeneous Data Handling:** CML frameworks are designed to handle **cross-silo environments**, where datasets differ in size, distribution, and feature space across participants.
4. **Flexibility in Collaboration:** Depending on the requirements, CML can implement **full model sharing, ensemble learning, or knowledge distillation**, allowing participants to contribute without compromising sensitive information.

Applications of CML Frameworks:

1. **Multi-Institutional Research Collaborations:**
   Academic and research institutions often have complementary datasets but cannot share raw data due to privacy regulations. CML allows these institutions to jointly train models, such as:
   - Predictive models for **disease diagnosis** across hospitals.
   - Collaborative climate or environmental modeling using region-specific data.

2. **Industry Consortiums for Safety-Critical Systems:**
   Industries like **automotive, aerospace, and energy** require high reliability and safety in their AI systems. Companies within an industry consortium can:
   - Share model insights for anomaly detection or predictive maintenance.
   - Jointly optimize AI models to improve safety and reduce system-wide failures without exposing proprietary data.

3. **Cross-Silo Learning with Heterogeneous Datasets:**
   In cross-silo CML, participants maintain large but distinct datasets. For example:
   - Banks collaborating on fraud detection models where transaction patterns differ across institutions.
   - Pharmaceutical companies training drug discovery models without disclosing sensitive proprietary datasets.

3. Architectures and Algorithms

Federated Learning (FL) relies on specific **architectural designs and algorithms** to enable collaborative model training while maintaining privacy and efficiency. FL architectures are primarily categorized based on the **scale of participants**, **size and type of local datasets**, and **network conditions**. Understanding these architectures is essential for selecting appropriate FL strategies in practical applications.

3.1 Federated Learning Architectures

Federated Learning architectures are generally classified into **Cross-Device FL** and **Cross-Silo FL**. Each type has distinct characteristics, design considerations, and challenges.

1. Cross-Device Federated Learning

Definition:

Cross-Device FL involves a **large number of devices**, often ranging from thousands to millions, such as smartphones, wearable devices, IoT sensors, and edge devices. Each device contributes to model training using its **small local dataset**.

Characteristics:

- **Participants:** Millions of potentially unreliable devices with intermittent connectivity.

- **Data Size:** Each device has a small, often non-IID dataset. For example, a smartphone may have personal text, app usage, or health sensor data.
- **Computation:** Devices perform lightweight model training due to limited processing power.
- **Communication:** High communication cost due to frequent model updates from a large number of devices. Efficient compression and update strategies are necessary.
- **Privacy Sensitivity:** High, as the data often contains personal user information.

Use Cases:

- **Mobile Keyboard Prediction:** Improving next-word prediction without sending personal messages to a central server.
- **Smart Health Devices:** Aggregating heart rate or step-count data for global activity models.
- **IoT Networks:** Optimizing energy consumption across smart homes without exposing individual household data.

Challenges:

- Device dropouts due to network instability.
- Non-IID data leading to model divergence.
- Limited computational and memory resources on devices.

2. Cross-Silo Federated Learning

Definition:

Cross-Silo FL involves a **smaller number of participants**, typically 2–100 entities, such as hospitals, banks, or large organizations. Each entity holds a **large local dataset**, which can be heterogeneous in features or distribution.

Characteristics:

- **Participants:** Fewer, but highly reliable organizations with stable connectivity and sufficient computational resources.
- **Data Size:** Large datasets per participant, often containing sensitive or regulated information.
- **Computation:** Organizations can perform more complex model training using local GPU or cloud resources.

- **Communication:** Lower frequency and volume of updates compared to Cross-Device FL. Updates are often larger due to complex models.
- **Privacy Sensitivity:** Extremely high, as data may include patient records, financial transactions, or proprietary business information.
- **Use Cases:**
- **Healthcare Collaborations:** Hospitals training diagnostic models for rare diseases without sharing patient records.
- **Financial Fraud Detection:** Banks jointly training models to detect fraudulent transactions without exposing sensitive client data.
- **Industrial Predictive Maintenance:** Manufacturing plants sharing knowledge about machinery failures without revealing proprietary process data.

Challenges:

- Handling heterogeneous datasets across organizations (different features, distributions, and formats).
- Achieving fair contribution and reward mechanisms for participating entities.
- Ensuring compliance with local regulations (e.g., GDPR, HIPAA).

Table 1: compares these architectures.

Feature	Cross-Device FL	Cross-Silo FL
Number of Participants	Thousands to millions	Few (2–100)
Data Size per Client	Small	Large
Network Reliability	Unstable, intermittent	Stable
Privacy Focus	High	High
Communication Cost	High	Moderate

3.2 Federated Learning Algorithms

Federated Learning relies on specialized algorithms that enable multiple clients to collaboratively train a global model without sharing raw data. These algorithms are designed to address **data heterogeneity, communication efficiency, and privacy preservation**. The following are the most prominent algorithms in FL:

3.2.1 Federated Averaging (FedAvg)

FedAvg is the most widely used FL algorithm, introduced by McMahan et al. (2017). It extends standard stochastic gradient descent (SGD) to the federated setting. The workflow involves:

1. **Local Training:** Each client trains its local model using its private dataset.
2. **Model Update Transmission:** Clients send the updated model weights (or gradients) to the central server.
3. **Weighted Aggregation:** The server combines the client models using a weighted average to produce the updated global model.

The global model update is given by:

wt+1=∑k=1Knknwtkw_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_t^kwt+1=k=1∑Knnkwtk

Where:

- wtkw_t^kwtk is the local model weight of client kkk at iteration ttt,
- nkn_knk is the number of local data samples of client kkk,
- n=∑k=1Knkn = \sum_{k=1}^{K} n_kn=∑k=1Knk is the total number of samples across all clients,
- KKK is the total number of clients participating in that round.

Advantages:

- Simple and easy to implement.
- Efficient in homogeneous and moderately heterogeneous data scenarios.

Limitations:

- Performance can degrade when client data distributions are highly non-IID.
- Sensitive to stragglers or clients with delayed updates.

3.2.2 FedProx

FedProx extends FedAvg to handle **heterogeneous data and system environments**. Key features include:

- Adds a **proximal term** to the local objective function:

minwfk(w)+μ2∥w−wt∥2\min_{w} f_k(w) + \frac{\mu}{2} \| w - w_t \|^2wminfk(w)+2μ∥w−wt∥2

Where fk(w)f_k(w)fk(w) is the local loss function, wtw_twt is the current global model, and μ\muμ controls the penalty for deviating from the global model.

- This modification stabilizes training when clients have different data distributions, local computation capabilities, or inconsistent update frequencies.

Use Cases:

- Cross-silo FL with hospitals or banks, where dataset sizes and distributions vary significantly.

3.2.3 MOON (Model-Contrastive Federated Learning)

MOON improves the **generalization of FL models** by applying **contrastive learning** between local and global representations:

- Each client minimizes a contrastive loss between its local model output and the global model output.
- Encourages local models to align with the global representation, mitigating divergence due to non-IID data.

Advantages:

- Improves convergence and performance on heterogeneous datasets.
- Particularly effective in cross-device FL with highly diverse data.

3.2.4 Secure Aggregation Algorithms

Privacy is a core requirement in FL. Secure aggregation ensures that **model updates cannot be reverse-engineered** to infer local data:

- **Homomorphic Encryption (HE):** Aggregates encrypted updates without decrypting them.
- **Secure Multi-Party Computation (SMC):** Clients collaboratively compute global updates without exposing local updates.
- **Differential Privacy (DP):** Adds noise to updates, providing strong guarantees that individual data points cannot be inferred.

Advantages:

- Enables privacy-preserving FL in sensitive domains like healthcare and finance.
- Reduces risks of data leakage from malicious aggregation servers or adversaries.

3.3 Collaborative Machine Learning Approaches

Collaborative Machine Learning (CML) extends FL by emphasizing **knowledge transfer, joint optimization, and multi-organization collaboration**. CML algorithms focus on

improving global model performance without requiring raw data exchange.

3.3.1 Transfer Learning-based CML

Concept:

- Pre-trained models from one organization or domain are **fine-tuned** on local data at another organization.
- Enables **cross-domain knowledge sharing** and accelerates model training when local data is limited.

Example:

- A hospital with a small dataset of MRI scans uses a pre-trained model trained by a consortium of hospitals to improve disease diagnosis.

Advantages:

- Reduces the need for large local datasets.
- Leverages domain knowledge from other organizations.

3.3.2 Knowledge Distillation

Concept:

- Local models generate predictions on either public or synthetic datasets.
- Predictions from multiple models are aggregated into a **teacher model**, which distills knowledge back to the participants.

Workflow:

1. Each client trains a local model.
2. Predictions (soft labels) are sent to a central teacher model.
3. Teacher model refines knowledge and shares it back with local models for improved learning.

Advantages:

- Preserves data privacy while enabling knowledge transfer.
- Useful for heterogeneous datasets and model architectures.

3.3.3 Multi-Agent Reinforcement Learning (MARL) for CML

Concept:

- Each participant is modeled as an **agent** optimizing a shared global objective.

- Agents use **reinforcement learning techniques** to coordinate their actions and updates.

Applications:

- Collaborative traffic signal optimization among multiple cities.
- Energy grid optimization across distributed microgrids.
- Cross-organization collaborative supply chain optimization.

Advantages:

- Suitable for dynamic, sequential decision-making tasks.
- Enables decentralized cooperation while maximizing overall system performance.

4. PRIVACY AND SECURITY IN FEDERATED LEARNING

Privacy preservation is central to FL and CML. Techniques include:

4.1 Differential Privacy (DP)

DP adds noise to local updates before sharing, ensuring that individual data points cannot be inferred from model updates. The privacy guarantee is defined by a parameter ϵ\epsilonϵ, representing the privacy budget.

4.2 Secure Multiparty Computation (SMC)

SMC enables computations over encrypted data, ensuring that model updates are aggregated without exposing individual client data.

4.3 Homomorphic Encryption

Allows mathematical operations directly on encrypted data, providing end-to-end privacy during model aggregation.

4.4 Adversarial Threats

FL is vulnerable to:

- **Poisoning attacks:** Malicious clients send manipulated updates.
- **Inference attacks:** Global model is exploited to infer private data.

Mitigation strategies involve robust aggregation techniques, anomaly detection, and client validation.

5. APPLICATIONS

5.1 Healthcare

- Collaborative training of diagnostic models across hospitals without sharing patient data.

- Example: Federated models for **COVID-19 detection from CT scans**.

5.2 Finance

- Fraud detection models shared across banks without exposing sensitive transaction data.
- Example: Cross-institution credit scoring using CML.

5.3 Internet of Things (IoT)

- Smart devices collaboratively train models for energy optimization and predictive maintenance.
- FL reduces communication costs and preserves privacy.

5.4 Autonomous Vehicles

- Multi-agent collaborative learning allows vehicles to share knowledge of road conditions without transmitting raw sensor data.

CHALLENGES AND LIMITATIONS

1. **Data Heterogeneity:** Non-IID data distributions among clients reduce model performance.
2. **Communication Overhead:** Frequent model updates across large networks can be expensive.
3. **System Scalability:** Cross-device FL must handle millions of clients with intermittent connectivity.
4. **Regulatory Compliance:** Ensuring adherence to GDPR, HIPAA, and other policies in cross-border collaborations.
5. **Security Threats:** Poisoning and inference attacks remain challenging to fully mitigate.

RECENT ADVANCES

Optimization Techniques

- **Adaptive Federated Optimization:** Incorporates momentum and adaptive learning rates in FedAvg.
- **Clustered FL:** Groups clients with similar data distributions to improve convergence.

Hybrid FL Models

- **Neurosymbolic FL:** Combines symbolic reasoning with neural federated models for explainability.
- **Graph FL:** Leverages graph neural networks to model relationships between clients and tasks.

Federated Transfer Learning

- Enables knowledge transfer across domains with partially overlapping features, addressing the data heterogeneity problem.

FUTURE DIRECTIONS

1. **Cross-Modality Federated Learning:** Combining text, image, and sensor data in decentralized training.
2. **Explainable Federated Models:** Developing interpretable models for critical applications such as healthcare.
3. **Integration with Edge Computing:** Reducing latency and improving scalability by performing local model updates on edge devices.
4. **Blockchain-enabled FL:** Ensures immutable and transparent audit trails for model updates.

CONCLUSION

Federated Learning and Collaborative Machine Learning represent transformative approaches to decentralized and privacy-preserving AI. By enabling model training without sharing sensitive data, they address critical privacy, regulatory, and efficiency concerns. This review highlights key architectures, algorithms, privacy mechanisms, applications, and challenges. Although challenges like data heterogeneity, communication costs, and security risks remain, ongoing research in optimization, hybrid models, and cross-domain collaboration promises to make FL and CML more robust and scalable. These paradigms are poised to become foundational components of AI systems in healthcare, finance, IoT, and autonomous systems.

REFERENCES

1. Kairouz, P., McMahan, H. B., et al. (2021). *Advances and Open Problems in Federated Learning*. Foundations and Trends in Machine Learning, 14(1–2), 1–210.
2. McMahan, H. B., Moore, E., Ramage, D., et al. (2017). *Communication-Efficient Learning of Deep Networks from Decentralized Data*. AISTATS.
3. Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). *Federated Learning: Challenges, Methods, and Future Directions*. IEEE Signal Processing Magazine, 37(3), 50–60.
4. Bonawitz, K., et al. (2017). *Practical Secure Aggregation for Privacy-Preserving*

Machine Learning. CCS.

5. Smith, V., Chiang, C. K., Sanjabi, M., & Talwalkar, A. (2017). *Federated Multi-Task Learning*. NIPS.
6. Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). *Federated Machine Learning: Concept and Applications*. ACM Transactions on Intelligent Systems and Technology, 10(2), 12.
7. Geyer, R. C., Klein, T., & Nabi, M. (2017). *Differentially Private Federated Learning: A Client Level Perspective*. NIPS Workshop.
8. Li, X., Huang, K., Yang, W., Wang, S., & Zhang, Z. (2020). *On the Convergence of FedAvg on Non-IID Data*. ICLR.
9. Sheller, M. J., Reina, G. A., Edwards, B., et al. (2020). *Federated Learning in Medicine: Facilitating Multi-Institutional Collaborations without Sharing Patient Data*. Scientific Reports, 10, 12598.
10. Rieke, N., Hancox, J., Li, W., et al. (2020). *The Future of Digital Health with Federated Learning*. NPJ Digital Medicine, 3, 119.