Detection and Mitigation Data Poisoning Attacks in Multimodal Online Federated Learning

Heqiang Wang, Xiaoxiong Zhong, Weihong Yang, Hualong Wu, Fangming Liu, Weizhe Zhang

***Abstract*—The Internet of Things (IoT) ecosystem produces vast quantities of multimodal data from diverse sources such as sensors, cameras, and microphones. With the growing integration of edge intelligence, IoT devices have evolved beyond simple data collection units into intelligent edge nodes, necessitating distributed learning paradigms to handle heterogeneous and dynamic multimodal data effectively. Moreover, the real-time nature of data generation and the limited storage capabilities of edge devices call for an online learning framework. To address these challenges, Multimodal Online Federated Learning (MMO-FL) has emerged as a promising solution, enabling decentralized and real-time model training across multiple modalities. However, existing MMO-FL studies largely assume idealized environments and overlook security threats. As a result, the security landscape of MMO-FL remains severely underexplored. In practice, MMO-FL introduces unique and complex security vulnerabilities across three dimensions: spatial (federated), temporal (online), and modal (multimodal). These characteristics make the distributed and online data collection process highly vulnerable to adversarial manipulation. To bridge this gap, we present the first systematic study of data poisoning attacks within the MMO-FL scenario. We begin with a theoretical analysis quantifying the impact of such attacks on learning performance. To defend against them, we propose a novel detection and mitigation algorithm tailored specifically for MMO-FL systems. Extensive experiments conducted on two real-world multimodal datasets, UCI-HAR and USC-HAD, demonstrate that our approach effectively detects and mitigates data poisoning attacks.**

***Index terms*—** Federated Learning, Multimodal Learning, Online Learning, Poisoning Attack Detection and Mitigation.

I. Introduction

The rapid expansion of the Internet of Things (IoT) [1] has led to an unprecedented surge in data generated by a multitude of interconnected devices, including smart home appliances [2], wearable health monitors [3], and industry sensors [4]. To enable intelligent services and applications across the IoT ecosystem, artificial intelligence techniques, particularly machine learning and deep learning, has become a fundamental tool for model training on large-scale IoT data. Traditionally, such training has been performed in centralized cloud platforms or data centers. However, this centralized paradigm faces significant challenges as both the scale of IoT data and the number of IoT devices continue to expand. Transferring large volumes of raw data to centralized servers imposes significant demands on network bandwidth and leads to substantial communication overhead, rendering it impractical for latency-sensitive applications such as autonomous driving [5] and real-time healthcare monitoring [6]. Additionally, uploading sensitive user data to the cloud raises serious privacy concerns [7]. With the gradual evolution of IoT devices from mere data collectors to intelligent edge nodes, there is increasing potential to harness their computational capabilities to address the scalability and efficiency challenges of massive IoT deployments. In this context, federated learning (FL) [8] has emerged as a promising distributed learning paradigm. FL enables collaborative model training across devices while keeping raw data local, offering a cost-effective and privacy-preserving alternative to traditional centralized learning. By significantly reducing data transmission and ensuring local data privacy, FL presents a natural and scalable solution for deploying intelligent applications in IoT environments.

Traditional FL frameworks for IoT have primarily been designed for unimodal data. However, in practice, IoT environments are inherently multimodal, involving a diverse range of sensors that generate data across multiple modalities [9], such as images from cameras, audio from microphones, and structured text or signals from various functional sensors. These multimodal inputs offer richer and more informative representations, which are crucial for enabling robust decision-making in complex scenarios. To accommodate this complexity, Multimodal Federated Learning (MFL) has emerged as an extension of standard FL, designed to support collaborative learning across distributed multimodal sources [10]. A typical approach in MFL involves deploying modality-specific encoder networks, each tailored to a particular data type, to extract meaningful feature representations from high-dimensional raw inputs. These features are then fused and passed through a head encoder, usually composed of deep neural network (DNN) layers followed by a softmax classifier, to produce the final prediction decision.

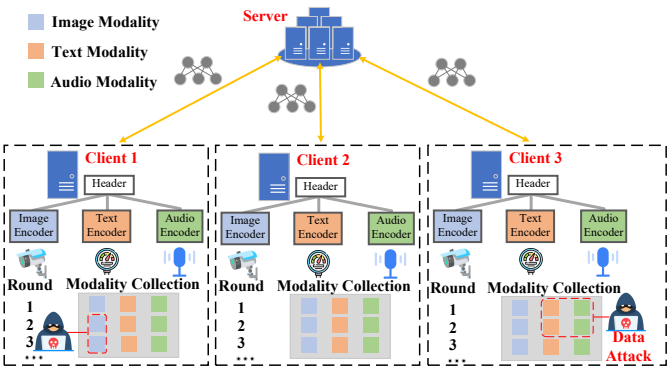


Fig. 1: MMO-FL with Data Poisoning Attacks

With the rise of dynamic, real-time data collection in IoT

H. Wang, X. Zhong, W. Yang, H. Wu, F. Liu and W. Zhang are with the Department of New Network, Peng Cheng Laboratory, Shenzhen, 518066, China. (Corresponding Authors: Xiaoxiong Zhong). Acknowledge grants: Peng Cheng Laboratory Project (Grant No. PCL2025A13).

environments, Multimodal Online Federated Learning (MMO-FL) has gained increasing attention as a promising framework for distributed, privacy-preserving intelligence. However, as an emerging research area, MMO-FL lacks thorough investigation into its security vulnerabilities, with most prior works assuming ideal, attack-free conditions. In traditional offline unimodal FL settings, security-related issues have been widely examined [11]–[15]. Nevertheless, these methods cannot be directly extended to the MMO-FL scenario for two primary reasons. First, offline FL assumes that all training data are collected before training, resulting in static data distributions and poisoning behaviors that can be analyzed offline without requiring real-time attack detection mechanisms. In contrast, MMO-FL continuously receives streaming multimodal data across rounds, leading to dynamically evolving attack patterns and requiring online detection and mitigation mechanisms. Second, unimodal FL lacks cross-modal interactions and therefore cannot address modality-specific threats inherent to multimodal learning. In particular, the Modality Misalignment Attack considered in this work deliberately disrupts semantic consistency across modalities by mismatching modality pairs, a type of attack that does not exist in unimodal settings. To bridge this gap, we initiate the first systematic exploration of data poisoning attacks (DPA) [16] within MMO-FL, a threat that significantly degrades learning performance.

Compared with conventional offline unimodal FL, MMO-FL introduces more challenging security threats due to its online and multimodal nature. In particular, these attacks present unique challenges due to MMO-FL's spatiotemporal and multimodal characteristics: First, the temporal nature of streaming data allows adversarial behavior to evolve across rounds. Second, the multimodal structure enables attackers to selectively poison specific modalities rather than entire samples. Third, the federated setting permits localized attacks targeting only a subset of clients. These factors collectively complicate both the detection and mitigation of DPA and highlight the urgent need for robust, adaptive defense strategies. A schematic illustration in Fig. 1 highlights the diverse manifestations of DPA across the spatial, temporal, and modality dimensions in MMO-FL. Motivated by these challenges, this work conducts the first systematic investigation of data poisoning attack detection and mitigation in the MMO-FL framework from a security perspective. To the best of our knowledge, no prior study has explicitly examined security threats in MMO-FL. The main contributions of this work are summarized as follows:

1) We investigate the problem of data poisoning attacks in the MMO-FL framework, where the inherent heterogeneity across spatial (federated), temporal (online), and modality (multimodal) dimensions introduces substantially greater complexity than in conventional centralized, unimodal, or offline learning settings. These challenges highlight the necessity for a systematic research effort.
2) We provide a comprehensive theoretical analysis of MMO-FL under the influence of data poisoning attacks. The resulting regret bound explicitly captures the performance degradation caused by such attacks, incorporating additional analytical terms that characterize their impact.
3) To address data poisoning attacks in MMO-FL, we propose the Prototypical Attack Detection and Mitigation (PADM) algorithm. PADM identifies attacks at the client level by examining post-attack feature deviations, and subsequently reduces their impact through adaptive mitigation strategies guided by attack classification.
4) We conduct extensive empirical evaluations of the proposed PADM algorithm within the MMO-FL framework using two representative multimodal datasets, UCI-HAR and USC-HAD. The results show that PADM consistently outperforms baseline methods, both in accurately identifying the presence of attacks and in effectively mitigating the impact of data poisoning within MMO-FL settings.

The remainder of this paper is organized as follows. Section II reviews related work on multimodal FL, online FL, and data poisoning attack in FL. Section III presents the system model and formulates the problem. Section IV details the workflow of MMO-FL under data poisoning attack. In Section V, we conduct a regret analysis of MMO-FL and examine the impact of data poisoning attack on learning performance. Section VI introduces the proposed PADM algorithms under MMO-FL scenario. Section VII reports experimental results that validate the effectiveness of the proposed PADM algorithms within MMO-FL framework. Finally, Section VIII concludes the paper.

II. Related Work

A. Online Federated Learning

Online learning is designed to process data sequentially and update models incrementally, making it well-suited for applications involving continuously arriving data and the need for real-time model adaptation [17]. These methods offer computational efficiency and eliminate the requirement of having access to the full dataset in advance, rendering them particularly suitable for memory-constrained IoT environments. In the context of FL, online federated learning (OFL) has emerged as a promising paradigm that extends online learning principles to distributed networks of decentralized learners [18]. A distinguishing feature of OFL, compared to traditional offline FL, is its emphasis on minimizing long-term cumulative regret rather than static optimization objectives during local updates. Although OFL remains relatively underexplored, several notable studies have recently advanced the field. For instance, [19] proposes a communication-efficient OFL algorithm that balances reduced communication overhead with strong learning performance. Similarly, [20] introduces FedOMD, an OFL method designed for uncertain environments, capable of handling streaming data without relying on assumptions about loss function distributions. While these works focus primarily on the horizontal federated learning setting, [21] explores the vertical federated learning (VFL) context [22], proposing an online VFL framework tailored to cooperative spectrum sensing and achieving sublinear regret. Further extending to real-world industrial applications, [4] addresses challenges such as noise interference and device

heterogeneity in online VFL systems. However, all the aforementioned online learning approaches are limited to unimodal online FL. In practice, multimodal data is pervasive in IoT applications, where information from diverse types of sensors must be jointly leveraged. To bridge this gap, this work pioneers the study of MMO-FL, with the goal of enhancing the robustness and security of online FL in complex, multimodal IoT environments.

B. Multimodal Federated Learning

MFL aims to train task-relevant models on multimodal data distributed across multiple clients, thereby enabling the effective utilization of diverse data sources. With the growing interest in MFL, a variety of algorithms have been proposed to address its unique challenges and improve learning performance. One prominent challenge is modality heterogeneity, where different clients possess access to varying subsets of modalities. This inconsistency complicates model aggregation and hampers effective knowledge sharing. Several studies, such as [23], [24], have explored strategies for heterogeneous modality fusion to address this issue. Another critical challenge involves optimizing modality selection for training under constrained computational and communication resources. To tackle this, [25] proposes MPriorityFed, an adaptive resource allocation framework that improves computational efficiency by prioritizing modality encoders based on their relevance and training requirements. Beyond above two challenges, a particularly pressing challenge in MFL is maintaining robust performance in the presence of missing modalities [26]. Missing data can result from incomplete data collection [27], sensor failures [28], or privacy restrictions [26], all of which degrade the effectiveness of conventional MFL frameworks. For example, [29] introduces a cluster-enhanced method that utilizes feature clustering to address missing modalities in brain imaging analysis, while [30] proposes the MFCPL framework, which leverages cross-modal prototypes to enhance knowledge transfer at both modality-shared and modality-specific levels. Despite these advancements, existing approaches are predominantly designed for offline learning scenarios with static datasets. They fail to address the challenges of online learning settings, where data arrives sequentially and models must adapt in real time. Besides, prior research on multimodal FL has largely focused on algorithmic performance and data heterogeneity, security concerns remain underexplored. To address this gap, our work is the first to examine MMO-FL from a security perspective, focusing on how to ensure the integrity of the learning process and maintain model performance in the face of the combined challenges posed by distributed, online, and multimodal data.

C. Data Poisoning Attack in Federated Learning

Data poisoning attacks have been extensively studied in the context of multimodal learning, as demonstrated by works such as [31], [32], however, these approaches are exclusively designed for centralized learning settings. Recent studies have extensively examined data poisoning attacks in FL, wherein malicious clients intentionally manipulate their local training data, such as through label flipping or injection of crafted examples, to degrade the performance of the global model. One of the earliest works to formalize and evaluate targeted data poisoning in FL systems is [33], which demonstrated that even a small number of malicious participants can significantly reduce accuracy on targeted classes. They also proposed strategies to detect such malicious clients. Comprehensive surveys, such as those by [34] and [16], provide in-depth analyses of data poisoning threats in FL. In data poisoning attacks, adversaries do not directly modify local model parameters but instead compromise training through data tampering. Data poisoning techniques are typically categorized into two types: clean-label attacks [35], where the attacker cannot alter the labels, and dirty-label attacks [36], where manipulations include actions like label flipping. However, the majority of existing studies focus on unimodal and offline learning settings, which differ substantially from the challenges presented in multimodal and online federated learning scenarios. Only a limited number of works have begun to explore data poisoning in multimodal or online contexts. For example, [32] were the first to investigate data poisoning in multimodal models, targeting both visual and textual modalities. Their study revealed modality-specific vulnerabilities and proposed both pre-training and post-training defenses to mitigate such attacks. Additionally, [37] modeled data poisoning in online learning as a stochastic optimal control problem, solved using model predictive control and deep reinforcement learning, and provided both theoretical and empirical evidence for near-optimal attack strategies. Overall, existing work either neglects the combined challenges of multimodal and online settings or only partially addresses them, rendering the corresponding methods difficult to apply directly to the MMO-FL scenario.

III. System Model

Before introducing the system model, we first present the abbreviations used throughout this work in Table I and summarize the key notations in Table II for clarity.

TABLE I: Key Abbreviations

Acronyms	Full terms
MFL	Multimodal Federated Learning
MMO-FL	Multimodal Online Federated Learning
DPA	Data Poisoning Attacks
MCA	Modality Corruption Attacks
MMA	Modality Misalignment Attacks
PADM	Prototypical Attack Detection and Mitigation
PCSD	Prototype Cosine Similarity Detection
PCE	Prototype Cross-Entropy
AF	Attack Free
AO	Attack Only
PD	Passive Defense
GM	Geometric Median

To illustrate the security challenges in MMO-FL, consider a smart factory setting comprising a central cloud server and K distributed workstations acting as clients. Each workstation is equipped with heterogeneous sensors, such as vision cameras, acoustic microphones, and environmental sensors for temperature and humidity, to monitor operations across different factory zones, as depicted in Fig. 1. These sensors

TABLE II: Key Notations

Symbol	Semantics
K	The number of clients
M	The number of modalities
T	The number of global rounds
N	The number of data samples collected by client
E	The number of local iterations
C	The number of classes
θ^m	The modality encoder
θ^0	The head encoder
η	The learning rate
Z	The feature extractor
Θ	The overall model
$\mathbf{G}_k^t$	The overall gradient
$v_{k,c}^{t,m}$	The local prototype
$\bar{v}_c^{t,m}$	The cumulative global prototype
$\bar{\mathcal{V}}^t$	The cumulative global prototype collection
$R_k^{t,m}$	The cosine similarity vector
λ	The attacks occurrence probability
κ	The Non-IID level

continuously generate multimodal data streams that are used to collaboratively train a global model across clients in real time. However, in this dynamic and decentralized environment, the data collection and communication processes are vulnerable to adversarial interference. Malicious actors may launch data poisoning attacks that corrupt the collected streaming data, thereby degrading the overall learning performance. This scenario exemplifies the core challenge of MMO-FL with security concerns, where ensuring robust and trustworthy collaborative learning becomes critical in the presence of multimodal, temporal, and spatial threats. During factory operations, each client's sensors continuously gather new data over time, with the entire timeline segmented into discrete intervals denoted as $t = 1, 2, ..., T$. For ease of analysis, each interval is also treated as a global round within the MMO-FL process.

In each global round t, each client $k \in \mathcal{K}$ collects the current local training dataset consisting of N data samples from M modalities. The dataset is denoted as $\mathcal{D}_k^t = \left(X_k^{t,1}, X_k^{t,2}, \ldots, X_k^{t,M}; Y_k^t\right) = \left\{\left(x_{k,n}^{t,1}, x_{k,n}^{t,2}, \ldots, x_{k,n}^{t,M}; y_{k,n}^t\right)\right\}_{n=1}^{|\mathcal{D}_k^t|}$. Here, $x_{k,n}^{t,m}$ represents the m^{th} modality data of the n^{th} sample in client k collected at global round t, and $y_{k,n}^t$ denotes the corresponding label. We also define $\mathcal{D}^t = \sum_{k=1}^K \mathcal{D}_k^t$ as overall dataset aggregated across all clients at global round t. Without loss of generality, we assume that each client k collects exactly N training samples per global round t.

In the MFL, the model trained collaboratively across clients and aggregated at the server consists of two primary components: a set of modality-specific encoders $\theta^1, \ldots, \theta^M$, and a shared head encoder θ^0. Each modality encoder θ^m is responsible for extracting meaningful feature representations from the raw input data of modality m, potentially using architectures tailored to the unique characteristics of that modality. The extracted feature vector for the m^{th} modality of client k at round t is denoted as $Z_k^{t,m} = \theta^m(X_k^{t,m})$. These modality-specific features are then fused by the head encoder θ^0 to generate the final prediction. Based on this architecture, the loss function for the distributed training process at global round t can be defined over the collective training data as:

$$F_t(\Theta, \mathcal{D}^t) = \frac{1}{K}\sum_{k=1}^{K} f_t\left(\theta^0\left(Z_k^{t,1}, \ldots, Z_k^{t,M}\right), Y_k^t\right) \quad (1)$$

where $\Theta = \left\{\theta^0, \theta^1, \ldots, \theta^M\right\}$ represents the overall model, comprising both the head encoder and modality-specific encoders. The term $\theta^0\left(Z_k^{t,1}, \ldots, Z_k^{t,M}\right)$ denotes the predicted labels through head encoder, and f_t is the loss function that measures the discrepancy between the predicted labels and the actual labels. It is important to note that the above loss function corresponds to a single global round. Since the training process is based on dynamically collected real-time data rather than a static dataset, adopting an online learning paradigm is essential. Let $\Theta^1, \ldots, \Theta^T$ denote the sequence of models over T global rounds. To evaluate the performance of the online learner, we define the cumulative learning regret Reg_T, which quantifies the difference between the total loss incurred by the learner and that of the best fixed model in hindsight. Specifically:

$$\text{Reg}_T = \sum_{t=1}^{T} F_t(\Theta^t; \mathcal{D}^t) - \sum_{t=1}^{T} F_t(\Theta^*; \mathcal{D}^t) \quad (2)$$

Where $\Theta^* = \arg\min_\Theta \sum_{t=1}^T F_t(\Theta; \mathcal{D}^t)$ represents the optimal fixed model selected in hindsight. Our objective is to minimize the learning regret, which equates to minimizing the cumulative loss. Importantly, if the learning regret grows sublinearly with respect to T, it indicates that the online learning algorithm can progressively reduce the training loss asymptotically.

The fixed optimal strategy in hindsight refers to an idealized solution computed by a hypothetical centralized oracle that has full knowledge of all loss functions across time. Achieving such a strategy would require access to future data specifically, the complete sequence of streaming datasets that have not yet been observed. However, due to the inherent stochasticity and temporal variability of real-time data collection, this information is fundamentally inaccessible in practice. Consequently, the loss functions evolve over time and cannot be predetermined. In light of this, our theoretical framework adopts regret as the primary performance metric, quantifying the deviation between the proposed algorithm and this unattainable, oracle-based optimal benchmark.

It is crucial to note that the previously defined notion of learning regret is established under idealized conditions, assuming the absence of adversarial interference. However, our problem setting explicitly accounts for data poisoning attacks, which pose significant challenges to learning performance. Consequently, our objective is to design detection and mitigation mechanisms that enable the algorithm to maintain a regret level comparable to that of the ideal, attack-free scenario. Achieving this goal forms the core of the theoretical analysis presented in the subsequent sections.

IV. MMO-FL With Data Poisoning Attacks

In this section, we explore the distinct security challenges introduced by data poisoning attacks in the MMO-FL setting,

which differ significantly from those in conventional FL scenarios. These challenges primarily arise from two key aspects: the heterogeneity of multimodal data and the dynamic nature of online learning. To build a comprehensive understanding, we first introduce the foundational concepts behind data poisoning attack methodology and outline the corresponding threat model. We then present a detailed examination of how the MMO-FL training process operates under adversarial conditions.

A. Attack Methodology

Given the complexity of industrial environments, we consider scenarios that involve multiple types of data poisoning attacks. In particular, this study focuses on two primary categories of attacks within the MMO-FL framework. The first category, known as Modality Corruption Attacks, targets the integrity of training data by introducing significant noise or distortions into specific modalities, thereby directly degrading model performance. The second category, termed Modality Misalignment Attacks, disrupts the semantic alignment between different modalities and their associated ground-truth labels. This misalignment undermines cross-modal consistency and hampers the model's ability to learn effective representations. Notably, Modality Misalignment Attacks are unique to multimodal learning scenarios and have not been addressed in prior FL literature concerning data poisoning. The essential distinctions between these two attack types are illustrated in Fig. 2, offering a comparative visualization of their respective impacts on the MMO-FL system. To facilitate understanding, the UCI-HAR dataset is used as a representative example to demonstrate the mechanics of each attack.

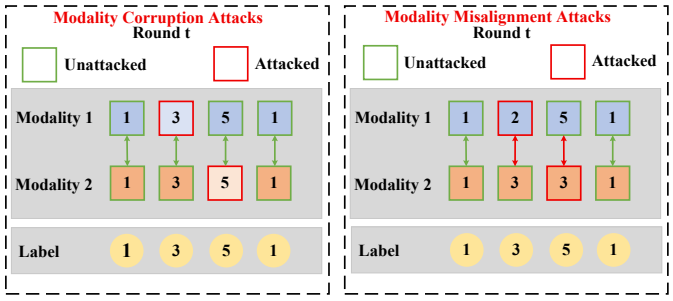


Fig. 2: Attack Methodology in MMO-FL

Modality Corruption Attacks: In a Modality Corruption Attack, the adversary degrades the learning process by injecting noise directly into the raw data of one or more modalities, thereby corrupting the input representation and impairing model training [16]. We denote the compromised dataset at client k during round t as $\hat{\mathcal{D}}_{k,a_1}^t$, where a_1 refers to this specific attack type. Each poisoned sample in this set is expressed as $\left\{\left(\hat{x}_{k,n\to c}^{t,1}, \hat{x}_{k,n\to c}^{t,2}\right); y_{k,n\to c}^t\right\}$, indicating that while the class label remains consistent across modalities, the input of one or more modalities has been corrupted. This corruption may affect a single modality or span multiple modalities. As shown in the left panel of Fig. 2, consider samples belonging to class 3, modality 1 is corrupted, resulting in significant distortion in its feature space. In practical scenarios, MCA may occur when adversaries intentionally interfere with specific sensing devices, compromise edge sensors, or physically perturb the sensing environment. Such interference severely undermines the model's capacity to learn consistent multimodal representations, ultimately compromising prediction accuracy.

Modality Misalignment Attacks: In a Modality Misalignment Attack, the adversary deliberately poisons training samples by mismatching modalities from different classes [32]. For example, pairing accelerometer data labeled as "Walking" with gyroscope data from the "Sitting" class. This cross-modal mismatch disrupts the semantic alignment necessary for effective multimodal learning. We denote the poisoned dataset under this attack as $\hat{\mathcal{D}}_{k,a_2}^t$, where a_2 identifies the misalignment attack. Each poisoned sample in $\hat{\mathcal{D}}_{k,a_2}^t$ can be expressed as $\left\{\left(x_{k,n\to c}^{t,1}, x_{k,n\neq c}^{t,2}\right); y_{k,n\to c}^t\right\}$, meaning that while modality 1 corresponds to the correct class, modality 2 originates from a different class, introducing semantic inconsistency. As shown in the right panel of Fig. 2, consider samples labeled as class 3. Here, modality 1 remains intact, but modality 2 is replaced with data from class 2. MMA is particularly specific to multimodal learning environments and may occur when adversaries intentionally manipulate modality synchronization or semantic correspondence. For example, attackers may tamper with timestamps, or modality associations to induce semantic misalignment across modalities. Such attacks impairs the model's ability to learn coherent relationships across modalities, leading to degraded overall performance in the MMO-FL setting. Such attacks are unique to multimodal learning and have been largely overlooked in prior FL literature.

B. Threat Model

In the context of MMO-FL, each client continuously collects new multimodal data using various modality-specific sensors at every global round. To model this setting realistically, we make a practical assumption: if a client is attacked in a given round, it is subjected to only one type of data poisoning attack. Additionally, we consider a black-box adversarial setting, wherein the attacker has no knowledge of model parameters or internal training dynamics. As a result, the identity of the attacked client and the specific status of its modalities remain unknown at each global round. Therefore, in each round, depending on the type of attack encountered, client k may receive one of three possible datasets: $\mathcal{D}_k^t$, $\hat{\mathcal{D}}_{k,a_1}^t$, or $\hat{\mathcal{D}}_{k,a_2}^t$, corresponding to the clean (unattacked) case, Modality Corruption Attacks, and Modality Misalignment Attacks, respectively. To distinguish between the models trained under these varying conditions, we denote the model trained on potentially compromised data as $\hat{\Theta}$, while the clean, attack-free counterpart is denoted as Θ. In this work, we focus on training-stage data poisoning attacks in MMO-FL, where adversaries compromise the learning process by injecting malicious or corrupted training samples to contaminate the global model. In contrast, the test dataset used during inference is assumed to remain clean and is only employed to evaluate the effectiveness and robustness of the proposed algorithm.

C. Workflow of MMO-FL under Data Poisoning Attacks

In this part, we will introduce the basic workflow of MMO-FL under data poisoning attacks. During each global round $t \in \mathcal{T}$, where $\mathcal{T} = \{0, 1, 2, \ldots, T-1\}$, all clients execute a fixed number of local training iterations, represented by the parameter E. The index $\tau = 0, 1, 2, ..., E$ is used to track these local iterations. Then each global round t consists of a sequence of coordinated steps carried out across clients and the server.

*1) **Client - New Data Collect**:* At the beginning of each global round t, each client k prepares to collect new training data for that round. However, due to the potential presence of data poisoning attacks, the collected dataset may belong to one of three categories: $\tilde{\mathcal{D}}_k^{t,m} = \left\{\mathcal{D}_k^{t,m}, \hat{\mathcal{D}}_{k,a_1}^{t,m}, \hat{\mathcal{D}}_{k,a_2}^{t,m}\right\}$. Before the application of specialized detection and mitigation mechanisms, clients must continue training using the collected data, regardless of its integrity, which may lead to significant degradation in performance.

*2) **Client - Local Model Update**:* Each client k uses the current global model Θ^t, provided by the server, as the initial model to train a new local model based on the current collected multimodal training datasets $\tilde{\mathcal{D}}_k^t$. Each client performs E iterations of online gradient descent (OGD) using the full training dataset. The update process described by the following equations:

$$\begin{aligned} &\Theta_k^{t,0} = \Theta^t \\ &\tilde{\Theta}_k^{t,\tau+1} = \tilde{\Theta}_k^{t,\tau} - \eta \tilde{\mathbf{G}}_k^{t,\tau}, \quad \forall \tau = 1, ..., E \\ &\tilde{\Theta}_k^{t+1} = \tilde{\Theta}_k^{t,E} \end{aligned} \tag{3}$$

Here, $\tilde{\mathbf{G}}_k^{t,\tau} = \nabla F_t(\tilde{\Theta}_k^{t,\tau}, \tilde{\mathcal{D}}_k^t)$ denotes the gradient computed on the current local dataset $\tilde{\mathcal{D}}_k^t$. This notation is introduced to distinguish it from the clean gradient $\mathbf{G}_k^{t,\tau}$. For clients whose datasets are not affected by any poisoning attack case, $\tilde{\mathbf{G}}_k^{t,\tau}$ naturally equals $\mathbf{G}_k^{t,\tau}$.

*3) **Client - Local Model Upload**:* Each client will upload the corresponding local model $\tilde{\Theta}_k^{t+1}$ to the server after finishing the E iterations of local model update.

*4) **Server - Global Model Update**:* The server updates the global model by using the local model updates from the clients, as given by the following equation:

$$\Theta^{t+1} = \frac{1}{K} \sum_{k \in \mathcal{K}} \tilde{\Theta}_k^{t+1} \tag{4}$$

The server then distributes the updated global model to all clients for the subsequent round. This iterative process repeats until either the predefined number of global rounds is completed or the target test accuracy is achieved.

To develop effective detection and mitigation strategies for data poisoning attacks, it is essential to first understand their theoretical impact on learning performance. In this context, we perform a detailed regret analysis to quantify how these attacks affect learning across spatial, temporal, and modality dimensions in the MMO-FL framework. This theoretical insight serves as a critical foundation for the design and justification of our proposed algorithms.

V. Theoretical Analysis

In this section, we present a comprehensive regret analysis of the proposed MMO-FL algorithm under the influence of data poisoning attacks. Our theoretical assumptions and corresponding proofs are established under a more general data poisoning case, rather than being confined to the two specific attack types proposed in this study. This generalization ensures broader applicability and robustness of the derived results across various attack scenarios. The analysis accounts for variations across rounds, clients, and modalities, and examines how these factors collectively impact the resulting regret bound. To clearly assess the influence of adversarial interference, we first derive the regret bound under ideal, attack-free conditions, and then contrast it with the regret bound in the presence of data poisoning attacks.

A. MMO-FL with local iterations $E > 1$ and without data poisoning attacks

To facilitate our analysis, we first introduce several additional definitions. After applying some basic transformations to the global model update equation above, we derive an alternative form of the global model update equation for the case where the local iteration is $E > 1$ and no data poisoning attacks happen, as follows:

$$\Theta^{t+1,0} = \Theta^{t,0} - \frac{\eta}{K} \sum_{k=1}^{K} \sum_{\tau=0}^{E-1} \mathbf{G}_k^{t,\tau} \tag{5}$$

where $\mathbf{G}_k^{t,\tau}$ denotes the gradient of the local overall model for client k across all M modalities for round t and local iteration τ, which equals:

$$\mathbf{G}_k^{t,\tau} = \left[\left(\mathbf{G}_k^{t,\tau,0}\right)^\top, \ldots \left(\mathbf{G}_k^{t,\tau,m}\right)^\top, \ldots \left(\mathbf{G}_k^{t,\tau,M}\right)^\top\right]^\top \tag{6}$$

In the following, we introduce a set of standard assumptions commonly adopted in the analysis of online convex optimization, as referenced in prior works such as [4], [21], [38]. To support our theoretical development, several assumptions are formulated at the modality level, reflecting the structure of the MMO-FL setting. Similar definition have also been employed in existing literature, ensuring the validity and generality of our analytical framework.

Assumption 1. *For any $\mathcal{D}^t$, the loss function $F_t(\Theta; \mathcal{D}^t)$ is convex with respect to Θ and differentiable.*

Assumption 2. *The loss function is L-Lipschitz continuous, the partial derivatives for each modality satisfies: $\|\nabla_m F_t(\Theta)\|^2 \leq L^2$.*

Assumption 3. *The partial derivatives, corresponding to the consistent loss function, fulfills the following condition:*

$$\left\|\boldsymbol{G}_k^{t,\tau'} - \boldsymbol{G}_k^{t,\tau}\right\| \leq \varphi \left\|\Theta_k^{t,\tau'} - \Theta_k^{t,\tau}\right\|$$

where τ' and τ indicate that they correspond to different local iterations.

For the theoretical analysis that follows, we consider each modality to be represented by a D-dimensional vector within

both the overall model and its corresponding gradient. Specifically, let $\mathbf{G}_{k,d}^m$ denote the d-th element of the gradient vector for modality m at client k, and $\Theta_{k,d}^m$ denote the d-th element of the corresponding model vector. Here, $d \in [1, D]$, and $m \in 0, 1, \ldots, M$, where $m = 0$ refers to the head encoder. Consequently, each full gradient or model across all modalities comprises a total of $(M+1)D$ vector elements.

Assumption 4. *The arbitrary vector element d in the overall model $\Theta_{k,d}^m$ for any modality m is bounded by: $\left|\Theta_{k,d}^m\right| \leq \sigma$.*

Assumption 1 ensures the convexity of the function, enabling us to leverage the properties of convex optimization. Assumption 2 constrains the magnitude of the loss function's partial derivatives at the modality level. Assumption 3 ensures that the variation in the partial derivatives is confined within a specific range, which aligns with the model variation over two different local iterations that maintain a consistent loss function. This approach effectively utilizes the concept of smoothness. Lastly, Assumption 4 specifies the permissible range for any vector element in the overall model. Then we can obtain the following Theorem 1.

Theorem 1. *Under Assumption 1-4, MMO-FL with local iterations $E > 1$ and excluding the impact of data poisoning attacks, achieves the following regret bound:*

$$
\begin{aligned}
Reg_T &= \sum_{t=1}^{T}\sum_{k=1}^{K}\mathbb{E}_t\left[F_t(\Theta^{t,0};\mathcal{D}_k^t)\right] - \sum_{t=1}^{T}\sum_{k=1}^{K}F_t(\Theta^*;\mathcal{D}_k^t) \\
&\leq \frac{K\left\|\Theta^{1,0}-\Theta^*\right\|^2}{2\eta E} + \frac{\eta TKE(M+1)L^2}{2} \\
&+ 2\eta TDEK(M+1)^2\varphi\sigma L
\end{aligned}
$$

Proof. The proof can be found in Appendix B. □

As established in Theorem. 1, by choosing a learning rate of $\eta = \mathcal{O}(1/\sqrt{T})$, the MMO-FL framework achieves a sublinear regret bound of $\mathcal{O}(\sqrt{T})$ under ideal, attack-free conditions. In the follows, we extend this result by examining how the presence of data poisoning attacks impacts the regret performance.

B. MMO-FL with local iterations $E > 1$ and with data poisoning attacks

In this section, we extend our theoretical analysis to the more general case where the number of local iterations satisfies $E > 1$, while explicitly accounting for the presence of data poisoning attacks. Unlike simplified settings that assume uniform attack patterns, our analysis considers a more realistic and flexible scenario in which arbitrary subsets of data across different clients may be compromised in each round. This includes cases where only a portion of the clients or partial segments of their data are affected. As a result, the derived conclusions offer greater generalizability and practical relevance for MMO-FL under adversarial conditions.

To account for the impact of data poisoning attacks, we redefine the global model update equation in this setting as follows:

$$
\Theta^{t+1} = \frac{1}{K}\left(\sum_{k\in\mathcal{S}_{t+1}}\Theta_k^{t+1} + \sum_{k\in\mathcal{K}\backslash\mathcal{S}_{t+1}}\hat{\Theta}_k^{t+1}\right) \tag{7}
$$

where $\mathcal{S}_{t+1}$ denotes the set of clients that not attacked in round t, and $\mathcal{K}\backslash\mathcal{S}_{t+1}$ corresponds to the set of clients that are subjected to attacks. After transforming the formula, we can obtain the following equation:

$$
\Theta^{t+1,0} = \Theta^{t,0} - \frac{\eta}{K}\left(\sum_{k\in\mathcal{S}_{t+1}}\sum_{\tau=0}^{E-1}\mathbf{G}_k^{t,\tau} + \sum_{k\in\mathcal{K}\backslash\mathcal{S}_{t+1}}\sum_{\tau=0}^{E-1}\hat{\mathbf{G}}_k^{t,\tau}\right)
$$

Given that the attack targets only a subset of clients and affects only a portion of the data per round, we introduce additional assumptions to facilitate a more accurate and tractable theoretical analysis.We begin by making an assumption on the distance between the gradient computed on an attacked data sample and the gradient computed on its clean data sample, as follows:

Assumption 5. *For any arbitrary component d of the overall gradient corresponding to modality m, the difference between the gradients computed with and without the influence of an attack is assumed to be bounded within a finite range, defined as:*

$$
\left|\hat{g}_{k,d}^{t,\tau,m} - g_{k,d}^{t,\tau,m}\right| \leq \phi_m \tag{8}
$$

Here, $g_{k,d}^{t,\tau,m}$ denotes the gradient computed from a single data sample, while $\boldsymbol{G}_{k,d}^{t,\tau,m} = \frac{1}{N}\sum_{n=1}^{N} g_{k,d}^{t,\tau,m}$ represents the corresponding batch gradient averaged over N samples. The term ϕ_m denotes the maximum deviation induced by an attack in modality m.

Next, we introduce an assumption regarding the number of clients attacked in each round. Specifically, we assume that the number of compromised clients follows a binomial distribution.

Assumption 6. *We model the number of clients that remain uncompromised in each global round as a binomial random variable, denoted by $S_t \sim Binomial(K, p)$, where K represents the total number of clients and $p \in [0, 1]$ denotes the probability that an individual client is not subjected to a data poisoning attack.*

Assumption 5 constrains the difference between attacked and unattacked gradients within a bounded range. This assumption is commonly adopted in robust learning and adversarial optimization analysis to ensure tractable theoretical derivations [39]. Assumption 6 models the number of unattacked clients using a binomial distribution, which captures the realistic stochastic nature of distributed attacks in MMO-FL systems, where each client may independently experience attacks with a certain probability during online data collection. It is important to emphasize that the aforementioned assumptions are introduced purely for the purpose of facilitating theoretical analysis. They are not prerequisites for practical

algorithm design or experimental implementation. Based on these assumptions, we derive Theorem 2, which extends the regret analysis to account for the presence of data poisoning attacks. This result explicitly captures the effects of partial client participation and selective data corruption within each training round, thereby offering a more realistic and robust theoretical bound.

Theorem 2. *Under Assumption 1-6, MMO-FL with local iterations $E > 1$ and including the impact of data poisoning attacks, achieves the following regret bound:*

$$
\begin{aligned}
Reg_T &= \sum_{t=1}^{T}\sum_{k=1}^{K}\mathbb{E}_t\left[F_t(\Theta^{t,0};\tilde{\mathcal{D}}_k^t)\right] - \sum_{t=1}^{T}\sum_{k=1}^{K}F_t(\Theta^*;\tilde{\mathcal{D}}_k^t) \\
&\leq \frac{K\left\|\Theta^{1,0}-\Theta^*\right\|^2}{2\eta E} + \frac{2T(1-p)\sigma D(M+1)EN_{max}\phi_{max}}{N} \\
&+ 2(1-p)^2\eta TKE(M+1)L^2 + 2T\eta DEK(M+1)^2\sigma\varphi L \\
&+ \frac{2(1-p)^2\eta TKE(M+1)DN_{max}\phi_{max}}{N} \\
&+ \eta p^2 TKE(M+1)L^2
\end{aligned}
$$

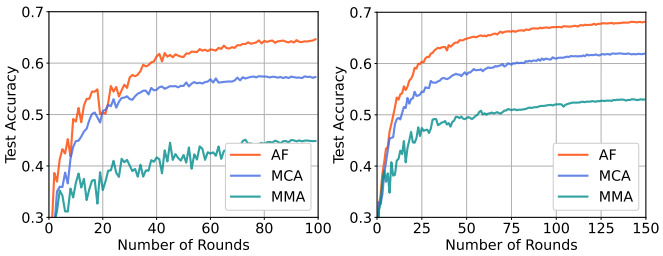
Where $\phi_{\max} = \max_{m\in\mathcal{M}}\phi_m$ and $N_{\max} = \max_{m,t,k} N_k^{t,m}$.

Proof. The proof can be found in Appendix D. □

Based on Theorem 2, by choosing the learning rate as $\eta = \mathcal{O}(1/\sqrt{T})$, MMO-FL with the impact of data poisoning attacks can achieve a regret bound of $\mathcal{O}(\sqrt{T}+\frac{T(1-p)N_{max}\phi_{max}}{N})$ over T time rounds. Notably, the term $\frac{T(1-p)N_{\max}\phi_{\max}}{N}$ emerges as the critical factor determining whether sublinear regret remains achievable. When no attacks are present (i.e., $p = 1$ and $N_{\max} = 0$), this term vanishes entirely, recovering the ideal case. Conversely, when $N_{\max} = N$, we revert to the worst-case scenario in which the entire dataset at each client is compromised per round. In general, the presence of poisoning introduces an additional regret component. Since the extent of attacked data cannot be controlled directly, the only viable approach to suppress this term is by minimizing $\phi_{\max}$, which necessitates designing effective detection and mitigation strategies. In the following section, we introduce our proposed algorithm specifically tailored to address this challenge.

VI. Data Poisoning Attack Detection and Mitigation Algorithm

Building upon the theoretical analysis presented in the previous section, we have examined the impact of data poisoning attacks on learning performance through the derived regret bounds. However, the specific influence of Modality Corruption Attacks and Modality Misalignment Attacks on empirical performance remains insufficiently explored. To address this, we evaluate the effects of each attack type on learning performance using the UCI-HAR and USC-HAD datasets. The results presented in Fig. 3 provide a deeper understanding of how different attack patterns contribute to the degradation of the learning process in the MMO-FL setting. Specifically, the experiments are conducted in a Non-IID MMO-FL environment with 5 clients, where each client experiences an attack with probability 0.5 at each global round. For each experimental setting, only a single attack type is applied throughout the entire training process. The reported results are averaged over 10 experimental runs to ensure statistical stability. Detailed experimental configurations are provided in the subsequent experiments section.



(a) Test Accuracy with UCI-HAR (b) Test Accuracy with USC-HAD

Fig. 3: Impact of Data Poisoning Attacks on Learning Performance. (Corruption SNR = 10 dB, Misalignment rate = 0.5.)

Building upon the theoretical analysis presented in the previous section and the experimental results discussed above, it is evident that data poisoning attacks significantly affect the regret bound and, consequently, degrade overall learning performance. Although the severity of their effects may vary depending on hyperparameter configurations, all types of data poisoning attacks negatively impact learning performance. Overall, experimental results across both datasets consistently indicate that Modality Misalignment Attacks tend to cause more pronounced degradation, likely due to the semantic inconsistency introduced across modalities. Therefore, it is necessary to design dedicated algorithms that first detect the presence of data poisoning attacks and subsequently apply mitigation strategies to minimize their impact on learning performance. In this section, we introduce the Prototypical Attack Detection and Mitigation (PADM) Algorithm which leverages prototype learning techniques to enable efficient attack defense during the learning process. To enable effective attack detection and mitigation, the initial prototypes are constructed from a small set of trusted benign samples. These prototypes serve as the foundation for identifying deviations caused by adversarial behavior. During subsequent training, they are continuously refined through an online update mechanism to reflect the evolving data distribution. The entire PADM process is primarily divided into three parts: Online Prototype Generation, Poisoning Attacks Detection, and Poisoning Attacks Mitigation. The following subsections provide a detailed introduction of each component. The main workflow of the PADM algorithm is shown in Fig. 4.

A. Online Prototype Generation

The first stage of the proposed PADM algorithm involves constructing the prototype set. This process begins with the initialization of a set of clean prototypes derived from benign samples. During training, the prototype set is continuously and synchronously updated to accommodate the evolving nature of online data, thereby supporting real-time adaptation. Let $c \in \{1,\ldots,C\}$ denote the class labels. Let $X_c^{t_0,m}$ represent the set of initial benign sample for class c and modality m, which are collected on the server prior to the start of training.

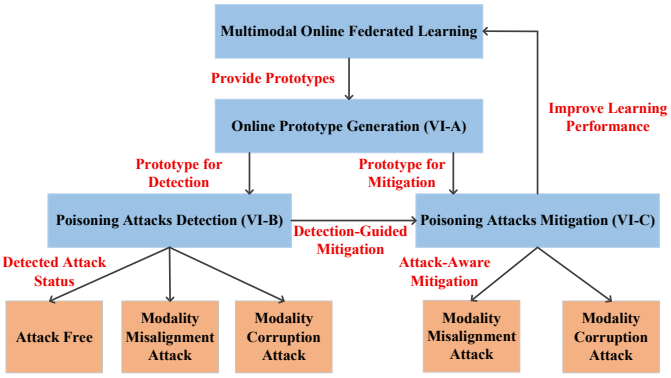


Fig. 4: Architecture of the PADM Algorithm

The initial prototype is then defined as the average value of the features extracted by the modality encoder:

$$v_c^{t_0,m} = \frac{1}{|X_c^{t_0,m}|} \sum_{n \in \mathcal{N}_{t_0}(\Phi_0)} \theta^m(x_n^{t_0,m}) \tag{9}$$

$$\textit{where} \quad \Phi_0 = \{y_{k,n}^t = c\} \tag{10}$$

This initial prototype serves as the foundation for subsequent prototype updates, as well as for the detection and mitigation of data poisoning attacks. In subsequent stages, the prototype must be updated synchronously with the training process. Since new data is continuously collected in a distributed manner across clients in the FL environment, updating the cumulative global prototype requires aggregating local prototypes computed by individual clients. The local prototype is computed as the mean of the feature representations extracted by the modality encoder from client with clean dataset, provided these samples are identified as clean and unaffected by attacks. The corresponding formulation is given as follows:

$$v_{k,c}^{t,m} = \frac{1}{\sum_c Q_{k,c}^{t,m}} \sum_{n \in \mathcal{N}(\Phi_{NA})} \theta^m(x_{k,n}^{t,m}) \tag{11}$$

$$\textit{where} \quad \Phi_{NA} = \{q_k^{t,m} = 1, y_{k,n}^t = c\} \tag{12}$$

Here, $q_k^{t,m} = 1$ indicates that the modality m on client k at round t is considered a clean dataset. However, whether the dataset is clean must be determined through the detection algorithm, as it is not known in advance. Accordingly, $Q_{k,c}^{t,m}$ denotes the total number of clean samples belonging to class c that satisfy this criterion. To promote balanced representation across different modalities, we introduce two guiding principles. First, it is essential that local prototypes, regardless of modality, adhere to a unified structural format. This ensures that the features produced by the modality-specific encoders, expressed as $\theta^m(x_{k,n}^{t,m})$, are aligned within a common latent space, even if the input data varies in form. By transforming raw inputs through their respective encoders, the extracted features can be harmonized for cross-modal consistency. Second, to ensure comparability and maintain stability throughout training, all local prototypes should be normalized. This normalization step guarantees consistent magnitude across modalities, thereby facilitating robust learning dynamics.

At the end of each training round, every client with clean dataset generates local prototypes for each modality and transmits them to the server. Upon receiving these local prototypes, the server constructs a cumulative global prototype for modality m and class c at global round t as follows:

$$\bar{v}_c^{t,m} = \frac{v_c^{t_0,m} + (t-1)\bar{v}_c^{t-1,m} + \frac{1}{\sum_k q_k^{t,m}} \sum_k v_{k,c}^{t,m}}{t+1} \tag{13}$$

The cumulative global prototype $\bar{v}_c^{t,m}$ is continuously updated and maintained on the server, this will serve as a key indicator for subsequent attack detection and mitigation. The server retains the full set of cumulative global prototypes, organized by modality and class, denoted as $\bar{\mathcal{V}}^t$, and defined as follows:

$$\bar{\mathcal{V}}^t = \begin{bmatrix} \bar{v}_1^{t,1} & \dots & \bar{v}_c^{t,1} & \dots & \bar{v}_C^{t,1} \\ \dots & \dots & \dots & \dots & \dots \\ \bar{v}_1^{t,m} & \dots & \bar{v}_c^{t,m} & \dots & \bar{v}_C^{t,m} \\ \dots & \dots & \dots & \dots & \dots \\ \bar{v}_1^{t,M} & \dots & \bar{v}_c^{t,M} & \dots & \bar{v}_C^{t,M} \end{bmatrix} \tag{14}$$

In the subsequent sections, we demonstrate how the accumulated global prototype set $\bar{\mathcal{V}}^t$ can be leveraged for the detection and mitigation of data poisoning attacks during the learning process.

B. Poisoning Attacks Detection

In this section, we detail how cumulative global prototypes can be utilized to detect data poisoning attacks that arise during the learning process. Building upon the previously defined attack taxonomy, our detection algorithm is designed not only to identify the presence of poisoning but also to distinguish between different attack types specifically, Modality Corruption Attacks (MCA) and Modality Misalignment Attacks (MMA). This classification is essential for enabling the application of appropriate mitigation strategies.

MCA typically involve the injection of noise, often drawn from a specific distribution, into all data collected during a particular training round. In contrast, MMA usually disrupt the semantic alignment between different modalities within the collected data. Drawing on these distinctive characteristics, we introduce a heuristic method named Prototype Cosine Similarity Detection (PCSD). This approach leverages feature extraction from the locally gathered dataset and corresponding prototypes. By calculating the cosine similarity between prototypes across modalities, PCSD provides a principled basis for identifying the presence of data poisoning attacks. To intuitively illustrate how cosine similarity behaves under clean and different attack conditions, we utilize heatmaps to visualize the results across three cases on the accelerometer modality of the UCI-HAR dataset, as depicted in Fig. 5. The heatmaps are generated after a short warm-up stage of training rather than after full convergence. Each small colored block in the figure represents the distance between the extracted feature and its corresponding class prototype.

The results reveal that MMA lead to prominent large-scale anomalies, evident as light-colored regions, stemming from disrupted alignment between modalities. In contrast, MCA introduce noise that results in more localized and less fragmented anomalies. The corresponding cosine similarity values remain positive, indicating that the feature directions remain

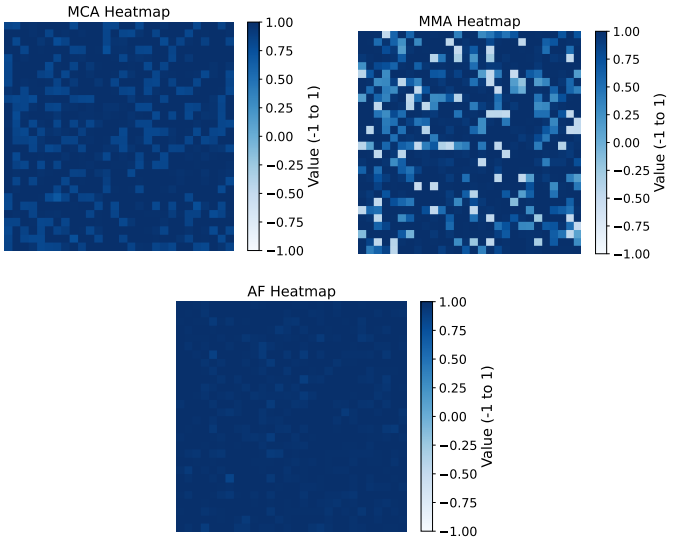


Fig. 5: Cosine similarity heatmap under different conditions.

generally consistent with those of the prototypes. The attack-free (AF) condition generally results in stable and coherent representations. Motivated by these observations, we now elaborate on the specific workflow of the PCSD method. First, the newly collected data undergoes feature extraction through the current round's feature model. Subsequently, the cosine similarity vector $R_k^{t,m}$ between these extracted features and the corresponding cumulative global prototypes is computed using the following formula:

$$\begin{aligned} r_{k,n}^{t,m} &= \cos\left(\theta^m(x_{k,n}^{t,m}), \bar{v}_{c\to n}^{t-1,m}\right) \\ &= \frac{\left\langle \theta^m(x_{k,n}^{t,m}), \bar{v}_{c\to n}^{t-1,m} \right\rangle}{\left\|\theta^m(x_{k,n}^{t,m})\right\| \left\|\bar{v}_{c\to n}^{t-1,m}\right\| + \epsilon} \end{aligned} \tag{15}$$

Where $R_k^{t,m} = \left[r_{k,1}^{t,m}, \cdots, r_{k,n}^{t,m}, \cdots, r_{k,N}^{t,m}\right]$. A small constant ϵ is introduced to avoid division by zero in the denominator. After transforming each vector $R_k^{t,m}$ into an appropriate input format and passing it through the pre-trained CNN, we obtain the corresponding inference results. This CNN model was initially trained on synthetically generated data that mimic various attack patterns, derived heuristically by analyzing characteristic heatmap patterns under different attack conditions. Notably, if the attack patterns evolve over time, the model can be incrementally updated in an online manner during training to adapt to these changes. The detailed configuration of the detection CNN will be presented in the subsequent experimental section.

C. Poisoning Attacks Mitigation

Once the client's modality status has been identified through the preceding detection process, it becomes necessary to design targeted mitigation strategies to alleviate the impact of data poisoning attacks. Specifically, distinct mitigation methods are adopted for MCA and MMA, tailored to the unique characteristics of each attack type. Furthermore, since each client may experience varying intensities of attack during different rounds, it is desirable for the mitigation algorithm to possess adaptive compensation capabilities that dynamically adjust to the severity of the threat.

Mitigation for MCA: To mitigate the impact of MCA, our approach focuses on aligning the corrupted feature representations with their expected semantic prototypes. This alignment is guided by the relative distances among modality-specific prototypes, enabling effective correction of degraded modality inputs. Drawing inspiration from prior work [40], we introduce a Prototype Cross-Entropy (PCE) loss, which quantifies the discrepancy between the extracted feature of a modality and its corresponding prototype at each global round. The PCE loss is formally defined as follows:

$$\mathcal{L}_{PCE}^m(f_t) = \mathbb{E}\left[-\log \frac{\exp(-d(\theta^m(x_{k,n}^{t,m}), \bar{v}_{c\to n}^{t-1,m}))}{\sum_{c=1}^{C}\exp(-d(\theta^m(x_{k,n}^{t,m}), \bar{v}_c^{t-1,m}))}\right]$$

Here, $d(\cdot,\cdot)$ denotes the distance function, which is instantiated as the Euclidean distance. The MCA mitigation loss, denoted by $\mathcal{L}_{MCA}$, is defined as a weighted combination of the standard cross-entropy (CE) loss and PCE loss. It is given as follows:

$$\mathcal{L}_{MCA} = \mathcal{L}_{CE} + \bar{\alpha}_k^{t,m}\beta\mathcal{L}_{PCE}^m \tag{16}$$

In this formulation, $\bar{\alpha}_k^{t,m} = \frac{1}{N}\sum_n \frac{1-r_{k,n}^{t,m}}{2}$ is an adaptive parameter that assigns weights to the PCE loss, calculated based on cosine similarity values reused from the detection phase. A higher value signifies stronger MCA attack intensity, thereby assigning greater weight to the PCE loss component to more effectively mitigate the attack's impact. β is a hyper-parameter to control the degree of modulation. A higher value indicates greater sensitivity to the impact of attacks, while a lower value suggests diminished emphasis on accounting for attack effects.

Mitigation for MMA: To counter the effects of MMA, we introduce an adaptive soft correction mechanism guided by modality-specific prototypes. This approach balances simplicity with flexibility, allowing the model to dynamically adjust to misaligned inputs during training. The core idea involves detecting misaligned samples and leveraging prototype information to gently nudge their feature representations toward the correct semantic direction. To achieve this, we implement an adaptive residual shrinkage strategy on the affected modality, formulated:

$$\theta^m(x_{k,n}^{t,m})' = \bar{v}_{c\to y_{k,n}^t}^{t-1,m} + (1-\alpha_{k,n}^{t,m})\left(\theta^m(x_{k,n}^{t,m}) - \bar{v}_{c\to y_{k,n}^t}^{t-1,m}\right) \tag{17}$$

where $\alpha_{k,n}^{t,m} = \frac{1-r_{k,n}^{t,m}}{2}$ serves as an adaptive weighting parameter that reflects the correlation between the misaligned feature and its corresponding class prototype. This parameter dynamically modulates the distance between the extracted feature of the misaligned sample and the true prototype, enabling fine-grained correction. By employing this adaptive soft correction instead of direct prototype replacement, the approach effectively mitigates misalignment while reducing the risk of overfitting and making full use of all newly collected data.

The aforementioned steps form the core components of the proposed PADM algorithm. In the next section, we present a comprehensive set of experimental results to evaluate and demonstrate the effectiveness of PADM in mitigating data poisoning attacks within the MMO-FL framework.

VII. Experiment

In this section, we will experimentally evaluate the performance of the MMO-FL algorithm. The experiments were conducted on an Ubuntu 22.04 machine equipped with an Intel Core i7-10700KF 3.8GHz CPU and a GeForce RTX 3070 GPU. The detailed experimental settings are provided below.

A. Datasets

To emulate the MMO-FL setting in IoT scenarios, we utilize two real-world multimodal datasets: UCI-HAR and USC-HAD. Both datasets are derived from IoT sensor data and feature multiple modalities along with diverse class labels, making them well-suited for evaluating DAP in MMO-FL scenario.

UCI-HAR: The UCI-HAR dataset is a widely recognized resource for human activity recognition research. It includes 10299 data samples collected from 30 participants (average age: 24) engaging in six activities (six classes): walking, walking upstairs, walking downstairs, sitting, standing, and lying down. These activities were recorded using smartphone sensors, specifically accelerometers and gyroscopes, which capture three-dimensional motion data. The sensors sampled data at 50 Hz, producing 128 readings per sensor axis within each time window. This dataset is utilized in our experiments to analyze sensor-based human activity recognition using three-dimensional motion data.

USC-HAD: The USC-HAD dataset is curated to support research in activity recognition based on motion sensor data. It comprises 27205 samples collected from 14 subjects performing 12 common daily activities, including walking forward, walking left, walking right, walking upstairs, walking downstairs, running, jumping, sitting, standing, sleeping, riding an elevator up, and riding an elevator down. Data were gathered using a wearable inertial measurement unit (IMU) motion node that integrates both triaxial accelerometers and triaxial gyroscopes, enabling precise capture of 3D movement patterns. Sensor readings were recorded at a sampling rate of 100 Hz, offering fine-grained temporal resolution for each activity. Compared to the UCI-HAR dataset, USC-HAD provides a richer set of activity classes and incorporates additional sensors for data collection.

Both of the original datasets are static and designed for offline learning. To align with the requirements of online learning, they must be transformed into dynamic datasets. The transformation process is described in detail in the following.

B. Online Data Generation

In the experiment, operating within an online learning scenario requires the training dataset to be dynamic, with data collected at the start of each global round. To ensure sufficient data samples for good training performance, we collect the initial dataset at the beginning of the training process. Considering the differences in dataset types and sizes, distinct online data generation details are employed for the UCI-HAR and USC-HAD datasets.

UCI-HAR: For the UCI-HAR dataset, training involves a total of five clients. Initially, each client is assigned 2000 data samples drawn according to a Dirichlet distribution with a Non-IID degree κ, representing the client's long-term data source. In subsequent global rounds, each client maintains an online local dataset of 500 samples. At every round, 20 new samples are drawn from the long-term data source and appended to the local dataset, while the oldest 20 samples are simultaneously removed. This real-time update mechanism ensures that the local datasets evolve dynamically, satisfying the requirements of online learning.

USC-HAD: For the USC-HAD dataset, the training process is conducted over five clients. Each client is initially allocated 4,000 data samples drawn from a Dirichlet distribution with a Non-IID parameter κ, representing its long-term data source. During subsequent global rounds, each client maintains an online local dataset containing 800 samples. At every round, 20 new samples are drawn from the long-term data source and added to the local dataset, while the oldest 20 samples are removed. This streaming update mechanism ensures that the client datasets evolve continuously over time, thereby conforming to the requirements of online learning.

C. Poisoning Attacks Simulation

To improve the clarity and interpretability of the experimental outcomes, we assume that each client is affected by only one type of attack during any given global round, as defined in the threat model section. To evaluate the impact of these poisoning attacks, we introduce the parameter set $\lambda = \{\lambda_0, \lambda_{a_1}, \lambda_{a_2}\}$, where $\lambda_0 + \lambda_{a_1} + \lambda_{a_2} = 1$. These parameters represent the probabilities of experiencing no attack (0), a Modality Corruption Attack (a_1), or a Modality Misalignment Attack (a_2), respectively. To control confounding factors and minimize the effects of modality interactions and client heterogeneity on performance, we impose two constraints in our simulations. First, all clients share the same λ configuration across global rounds. Second, because the datasets contain only two modalities, at most one modality can be compromised in any given round. Under this controlled setting, we systematically investigate how varying degrees of data poisoning attack λ, influence the overall learning performance.

D. Model Details

In the following, we detail the model architectures and key parameters used in our experiments, presented separately for the two datasets.

UCI-HAR: The dataset includes two distinct modalities: accelerometer and gyroscope signals, necessitating the use of modality-specific encoder models. For the accelerometer data, we use a CNN-based model as the encoder. This model consists of five convolutional layers and one fully connected layer. For the gyroscope data, we use an LSTM-based model

with one LSTM layer and one fully connected layer. Both encoders generate a 128-dimensional feature representation. The shared header model is implemented using two fully connected layers. The learning rate is 0.08, with a decay factor of 0.95 until it reaches 0.001. The batch size is set to 128.

USC-HAD: The dataset includes two distinct modalities: accelerometer and gyroscope signals, need modality-specific encoder models. For the accelerometer data, a four-layer CNN is utilized, with a modified output layer designed to produce a 128-dimensional feature vector. For gyroscope data, a single-layer LSTM network is employed, also generating a 128-dimensional output. The modality-specific features are subsequently processed by a shared header model consisting of two fully connected layers with ReLU activation and dropout, producing the final predictions over twelve activity categories. The learning rate is 0.1, with a decay factor of 0.99 until it reaches 0.001. The batch size is set to 128.

Attacks Detection: For the detection model, we adopt a 1D CNN architecture composed of two sequential blocks, each containing a convolutional layer, batch normalization, ReLU activation, and max pooling. These are followed by a flattening layer and two fully connected layers. The model is trained using the Adam optimizer with a learning rate of 0.001 and a batch size of 32.

E. Benchmarks

In our experiments, we employ several baseline methods for performance comparison. As this work is the first to address data poisoning attacks in the context of MMO-FL, there are no established benchmarks available for direct comparison. To demonstrate the effectiveness of our proposed algorithm, we designed a set of baseline methods for comparative evaluation.

Attack Free (AF). In this setting, standard MMO-FL training is performed without any injected attacks, serving as an upper-bound baseline that reflects ideal clean training performance.

Attack Only (AO). In this setting, attacks are introduced into the training process, but no detection or mitigation mechanisms are applied. This configuration serves as a lower-bound baseline, reflecting the performance degradation caused by adversarial interference.

Passive Defense (PD). In this setting, we adopt a widely used baseline for data poisoning attack scenarios, which replaces the conventional weighted average aggregation with a coordinate-wise median. This approach effectively reduces the influence of extreme malicious updates and enhances the robustness of the global model. Under this configuration, there is no need to explicitly detect or identify which clients have been attacked, as the aggregation mechanism itself mitigates the impact of adversarial contributions.

Geometric Median (GM). In this setting, we adopt another widely used robust aggregation baseline for data poisoning attack scenarios, namely Geometric Median aggregation. Instead of using conventional weighted average aggregation, this approach computes the geometric median of all client updates at the server side, thereby reducing the influence of malicious or abnormal updates on the global model. Under this configuration, there is no need to explicitly detect attacked clients, as the robust aggregation mechanism itself mitigates the impact of adversarial contributions during FL process.

F. Simulation Results

In this section, we assess the performance of the proposed MMO-FL framework under various data poisoning attack scenarios. We begin by evaluating the detection capabilities of the PADM algorithm, demonstrating its effectiveness in identifying whether an attack has occurred and in distinguishing between different types of attacks. Following this, we present a comprehensive evaluation of the full detection and mitigation pipeline by comparing the end to end performance of PADM against the baseline methods. Finally, we conduct ablation studies to investigate the sensitivity of PADM to key parameter configurations, with the aim of understanding how different settings influence both detection accuracy and mitigation effectiveness. The experimental evaluation is conducted on both the UCI-HAR and USC-HAD datasets. All reported results represent the average performance over 10 independent runs.

Evaluation of Attack Detection Performance. We begin by evaluating the detection performance of the proposed PADM algorithm. Clients without attacks are labeled as 0, those under MCA as 1, and those under MMA as 2. The "Real" row represents the actual attack status experienced by each client at the current global round, while the "Detect" row denotes the corresponding attack status predicted by the proposed PADM detection algorithm. Different colors are used to represent different attack states, including attack-free status, MCA, and MMA. If the colors in the "Real" and "Detect" rows are consistent at a given position, this indicates that the proposed algorithm correctly identifies the client's attack status. Conversely, inconsistent colors indicate incorrect detection results. We visualize the actual and detected attack states of three clients throughout the entire training process. As shown in Fig. 6(a), based on the UCI-HAR dataset with configuration $[\lambda_0 = 0.4, \lambda_{a_1} = 0.3, \lambda_{a_2} = 0.3]$, several key findings emerge: Firstly, PADM consistently demonstrates strong detection capability, successfully identifying whether an attack occurred and distinguishing its type. Secondly, although a few false positives appear in the early rounds, the detection accuracy stabilizes in later stages; and finally the detection performance is robust and largely unaffected by variations in local data distributions across clients.

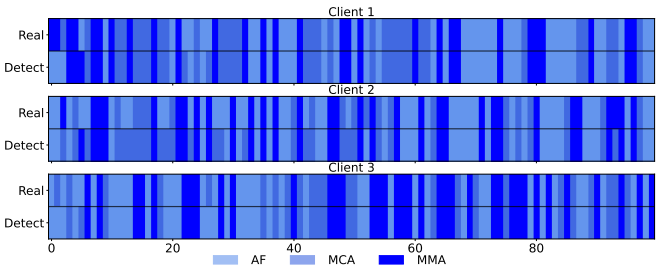
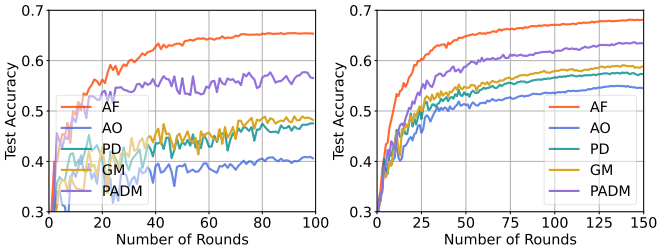


Fig. 6: Detection performance of PADM algorithm.

Evaluation of Overall PADM Performance. We then integrate the detection and mitigation stages to comprehensively evaluate the overall performance of the proposed PADM

algorithm with baseline algorithms. The test accuracy comparison under data poisoning attack between proposed PADM algorithm and baseline algorithms are shown in Fig. 7(a) and Fig. 7(b) based on UCI-HAR dataset with configuration $[\lambda_0 = 0.4, \lambda_{a_1} = 0.3, \lambda_{a_2} = 0.3, \beta = 0.5]$ and USC-HAD dataset with $[\lambda_0 = 0.4, \lambda_{a_1} = 0.3, \lambda_{a_2} = 0.3, \beta = 0.5]$, respectively. Based on the simulation results, we can summarize the following findings. First, the AF achieves the highest test accuracy and exhibits the most stable learning curve, as expected due to the absence of any data poisoning attacks. In contrast, both AO, PD and GM scenarios, which respectively represent no defense and a simple aggregation-based defense, suffer noticeable performance degradation under attack conditions. Second, the proposed PADM algorithm substantially outperforms both AO, PD and GM. This demonstrates the effectiveness of our combined detection-and-mitigation strategy, which significantly enhances the model's resilience against data poisoning attacks. Third, although the PD and GM method lacks explicit attack detection or correction mechanisms, it still shows moderate robustness. This reduces the influence of malicious model updates from compromised clients.



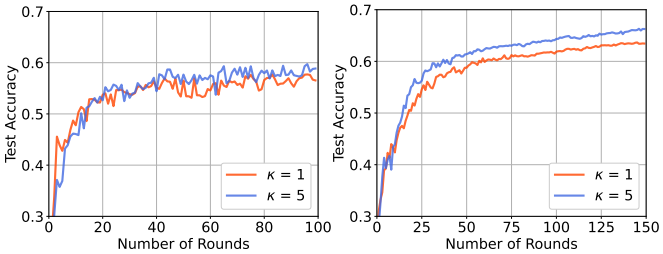
(a) Test Accuracy with UCI-HAR (b) Test Accuracy with USC-HAD

Fig. 7: Overall performance of PADM algorithm and benchmarks under data poisoning attacks.

Next, we conduct a series of ablation studies to explore how various parameter settings influence the performance of the proposed PADM algorithm. We begin by analyzing the effects of different non-IID levels, a common challenge in FL. Following this, we examine how varying the probability of data poisoning attacks impacts the learning performance of PADM, providing deeper insight into its robustness under different threat intensities.

Impact of Non-IID Level κ. In this section, we investigate how varying levels of data heterogeneity, controlled by the Non-IID parameter κ, influence the learning performance of the PADM algorithm. The results for the UCI-HAR and USC-HAD datasets are shown in Fig. 8(a) and Fig. 8(b), respectively. Across both datasets, we observe that a higher κ value (e.g., $\kappa = 5$), which corresponds to a more balanced class distribution and thus a lower degree of Non-IID, leads to improved learning performance compared to the highly skewed case ($\kappa = 1$). This observation aligns with established findings in traditional FL, where increasing data homogeneity typically enhances convergence and model accuracy. Although the extent of this effect varies between datasets, the overall trend remains consistent: reduced data heterogeneity consistently contributes to better performance.

Impact of Attacks Occurrence Probability λ . In this section, we investigate how varying the attack occurrence



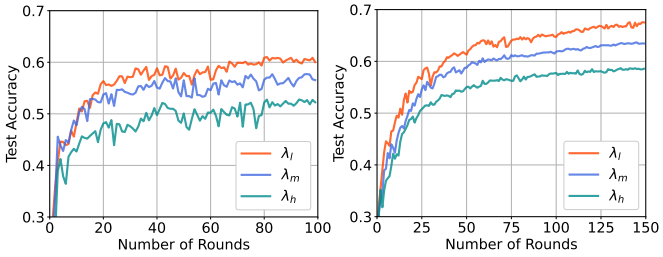
(a) Test Accuracy with UCI-HAR (b) Test Accuracy with USC-HAD

Fig. 8: Performance evaluation of PADM with different Non-IID level.

probability λ affects the learning performance of the proposed PADM algorithm. To systematically analyze this impact, we categorize λ into three levels: low (λ_l), medium (λ_m), and high (λ_h), with the specific configurations detailed in Table III. The experimental results for the UCI-HAR and USC-HAD datasets are illustrated in Fig. 9(a) and Fig. 9(b), respectively. Across both datasets, we observe that λ_l yields the highest learning performance, while λ_h results in the most significant performance degradation. This trend is primarily due to the increased frequency of data poisoning attacks at higher λ values. On one hand, frequent attacks necessitate repeated detection and mitigation efforts, yet these corrections cannot fully recover the fidelity of clean data. On the other hand, a high attack ratio disrupts the stability and accuracy of prototype updates, further impairing the model's convergence and generalization capabilities. Overall, while the PADM algorithm demonstrates strong resilience under all λ settings, scenarios with high attack rates (λ_h) may benefit from enhanced mitigation strategies to sustain optimal performance.

TABLE III: Parameter Detail of λ

Symbol	Value
λ_l	$\lambda_0 = 0.8, \lambda_{a_1} = 0.1, \lambda_{a_2} = 0.1$
λ_m	$\lambda_0 = 0.4, \lambda_{a_1} = 0.3, \lambda_{a_2} = 0.3$
λ_h	$\lambda_0 = 0.2, \lambda_{a_1} = 0.4, \lambda_{a_2} = 0.4$



(a) Test Accuracy with UCI-HAR (b) Test Accuracy with USC-HAD

Fig. 9: Performance evaluation of PADM with different λ.

VIII. Conclusion

In this work, we examined data poisoning attacks within the framework of Multimodal Online Federated Learning (MMO-FL) from a security standpoint. The intrinsic characteristics of MMO-FL, namely its online, distributed, and multimodal structure, introduce unique challenges for both theoretical analysis and practical defense. To address these challenges, we proposed the Prototypical Attack Detection and Mitigation (PADM) algorithm, specifically designed to detect and

defend against such adversarial threats. The proposed method is theoretically grounded, with rigorous regret analysis, and empirically validated through extensive experiments. Looking ahead, we plan to broaden our investigation into other emerging security threats in MMO-FL, such as model poisoning attacks. Furthermore, we aim to deploy MMO-FL in real-world Industrial IoT environments to identify and resolve security vulnerabilities that may only arise in operational systems. Overall, this work lays a foundational step toward enhancing the robustness of MMO-FL.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] H. Wang, J. Bian, and J. Xu, "On the local cache update rules in streaming federated learning," *IEEE Internet of Things Journal*, vol. 11, no. 6, pp. 10 808–10 816, 2023.

[3] T. Wu, F. Wu, C. Qiu, J.-M. Redouté, and M. R. Yuce, "A rigid-flex wearable health monitoring sensor patch for iot-connected healthcare applications," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6932–6945, 2020.

[4] H. Wang, X. Zhong, K. Liu, F. Liu, and W. Zhang, "Denoising and adaptive online vertical federated learning for sequential multi-sensor data in industrial internet of things," *IEEE Transactions on Mobile Computing*, 2025.

[5] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 163–168.

[6] F. Al-Turjman and S. Alturjman, "Context-sensitive access in industrial internet of things (iiot) healthcare applications," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2736–2744, 2018.

[7] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 843–859, 2012.

[8] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[9] Z. Huang, X. Xu, J. Ni, H. Zhu, and C. Wang, "Multimodal representation learning for recommendation in internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 675–10 685, 2019.

[10] L. Che, J. Wang, Y. Zhou, and F. Ma, "Multimodal federated learning: A survey," *Sensors*, vol. 23, no. 15, p. 6986, 2023.

[11] J. Zhang, C. Zhu, X. Sun, C. Ge, B. Chen, W. Susilo, and S. Yu, "Flpurifier: Backdoor defense in federated learning via decoupled contrastive training," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 4752–4766, 2024.

[12] Y. Miao, X. Yan, X. Li, S. Xu, X. Liu, H. Li, and R. H. Deng, "Rfed: Robustness-enhanced privacy-preserving federated learning against poisoning attack," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 5814–5827, 2024.

[13] K. Gai, D. Wang, J. Yu, L. Zhu, and W. Meng, "Fedamm: Federated learning against majority malicious clients using robust aggregation," *IEEE Transactions on Information Forensics and Security*, 2025.

[14] C. Zhu, J. Zhang, X. Sun, B. Chen, and W. Meng, "Adfl: Defending backdoor attacks in federated learning via adversarial distillation," *Computers & Security*, vol. 132, p. 103366, 2023.

[15] S. Wang, K. Gai, J. Yu, L. Zhu, W. Meng, and B. Xiao, "Easter: Embedding aggregation-based heterogeneous models training in vertical federated learning," *IEEE Transactions on Mobile Computing*, 2025.

[16] J. Fan, Q. Yan, M. Li, G. Qu, and Y. Xiao, "A survey on data poisoning attacks and defenses," in *2022 7th IEEE International Conference on Data Science in Cyberspace (DSC)*. IEEE, 2022, pp. 48–55.

[17] D. Sahoo, Q. Pham, J. Lu, and S. C. Hoi, "Online deep learning: learning deep neural networks on the fly," *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, 2018.

[18] S. Hong and J. Chae, "Communication-efficient randomized algorithm for multi-kernel online federated learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9872–9886, 2021.

[19] D. Kwon, J. Park, and S. Hong, "Tighter regret analysis and optimization of online federated learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[20] A. Mitra, H. Hassani, and G. J. Pappas, "Online federated learning," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 4083–4090.

[21] H. Wang and J. Xu, "Online vertical federated learning for cooperative spectrum sensing," *IEEE Transactions on Cognitive Communications and Networking*, 2024.

[22] S. Wang, K. Gai, J. Yu, Z. Zhang, and L. Zhu, "Pravfed: Practical heterogeneous vertical federated learning via representation learning," *IEEE Transactions on Information Forensics and Security*, 2025.

[23] H. Chen, Y. Zhang, D. Krompass, J. Gu, and V. Tresp, "Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, 2024, pp. 11 285–11 293.

[24] J. Chen and A. Zhang, "Fedmsplit: Correlation-adaptive federated multi-task learning across multimodal split networks," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2022, pp. 87–96.

[25] J. Bian, L. Wang, and J. Xu, "Prioritizing modalities: Flexible importance scheduling in federated multimodal learning," *arXiv preprint arXiv:2408.06549*, 2024.

[26] M. Ma, J. Ren, L. Zhao, S. Tulyakov, C. Wu, and X. Peng, "Smil: Multimodal learning with severely missing modality," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 2302–2310.

[27] C. Zhang, X. Chu, L. Ma, Y. Zhu, Y. Wang, J. Wang, and J. Zhao, "M3care: Learning with missing modalities in multimodal healthcare data," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2418–2428.

[28] H. Maheshwari, Y.-C. Liu, and Z. Kira, "Missing modality robustness in semi-supervised multi-modal semantic segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 1020–1030.

[29] X. Wang, R. Zhou, H. Xie, X. Tang, L. He, and C. Yang, "Clusmfl: A cluster-enhanced framework for modality-incomplete multimodal federated learning in brain imaging analysis," *arXiv preprint arXiv:2502.12180*, 2025.

[30] H. Q. Le, C. M. Thwal, Y. Qiao, Y. L. Tun, M. N. Nguyen, E.-N. Huh, and C. S. Hong, "Cross-modal prototype based multimodal federated learning under severely missing modality," *Information Fusion*, p. 103219, 2025.

[31] H. Bansal, N. Singhi, Y. Yang, F. Yin, A. Grover, and K.-W. Chang, "Cleanclip: Mitigating data poisoning attacks in multimodal contrastive learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 112–123.

[32] Z. Yang, X. He, Z. Li, M. Backes, M. Humbert, P. Berrang, and Y. Zhang, "Data poisoning attacks against multimodal encoders," in *International Conference on Machine Learning*. PMLR, 2023, pp. 39 299–39 313.

[33] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," in *European symposium on research in computer security*. Springer, 2020, pp. 480–501.

[34] G. Xia, J. Chen, C. Yu, and J. Ma, "Poisoning attacks in federated learning: A survey," *IEEE Access*, vol. 11, pp. 10 708–10 722, 2023.

[35] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein, "Poison frogs! targeted clean-label poisoning attacks on neural networks," *Advances in neural information processing systems*, vol. 31, 2018.

[36] V. Shejwalkar, A. Houmansadr, P. Kairouz, and D. Ramage, "Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning," in *2022 IEEE symposium on security and privacy (SP)*. IEEE, 2022, pp. 1354–1371.

[37] X. Zhang, X. Zhu, and L. Lessard, "Online data poisoning attacks," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 201–210.

[38] H. Wang, X. Liu, X. Zhong, L. Chen, F. Liu, and W. Zhang, "Multimodal online federated learning with modality missing in internet of things," *IEEE Transactions on Mobile Computing*, 2025.

[39] Y. Wang, P. Mianjy, and R. Arora, "Robust learning for data poisoning attacks," in *International conference on machine learning*. PMLR, 2021, pp. 10 859–10 869.

[40] Y. Fan, W. Xu, H. Wang, J. Wang, and S. Guo, "Pmr: Prototypical modal rebalance for multimodal learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 029–20 038.

Heqiang Wang received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from the University of Kentucky, the University of Connecticut, and the University of Miami in 2016, 2019, and 2024, respectively. He is currently an Assistant Researcher with Peng Cheng Laboratory, Shenzhen, China. His research focuses on AI-enabled network systems and intelligent industrial Internet, with research interests including federated learning, AI for networks, and industrial Internet of Things.

Xiaoxiong Zhong received his Ph.D degree in Computer Science from Harbin Institute of Technology, China, in 2015. From 2016 to 2018, he was a Postdoctoral Research Fellow with Tsinghua University, China. He is currently an associate professor with Peng Cheng Laboratory, Shenzhen, China. His general research interests include network protocol, IoT, edge computing and AI for networks.

Weihong Yang received the Ph.D. degree in computer science and technology from Harbin Institute of Technology, China, in 2022. He is currently an Assistant Researcher with the Department of New Networks, Pengcheng Laboratory, Shenzhen, China. From 2022 to 2024, he was a Senior Engineer with Huawei, where he worked on network protocols and related technologies. His research interests include network protocols, network optimization, and distributed systems.

Hualong Wu is currently an Engineer with the Department of New Networks, Pengcheng Laboratory, Shenzhen, China. His research interests include mobile edge computing, federated learning, and intelligent optimization.

Fangming Liu (S'08, M'11, SM'16) received the B.Eng. degree from the Tsinghua University, Beijing, and the Ph.D. degree from the Hong Kong University of Science and Technology, Hong Kong. He is currently a Full Professor with the Huazhong University of Science and Technology, Wuhan, China. His research interests include cloud computing and edge computing, datacenter and green computing, SDN/NFV/5G and applied ML/AI. He received the National Natural Science Fund (NSFC) for Excellent Young Scholars, and the National Program Special Support for Top-Notch Young Professionals. He is a recipient of the Best Paper Award of IEEE/ACM IWQoS 2019, ACM e-Energy 2018 and IEEE GLOBECOM 2011, the First Class Prize of Natural Science of Ministry of Education in China, as well as the Second Class Prize of National Natural Science Award in China.

Weizhe Zhang received the PhD degree from the Harbin Institute of Technology in 2006. He is currently a professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China, and director with the Department of New Networks, Peng Cheng Laboratory, Shenzhen, China. His research interests include cyberspace security, cloud computing, and high-performance computing. He is a lifetime member of the ACM and seinoir Member of the IEEE.