PFAE: Personalized Federated Learning for Anomaly Detection Over Heterogeneous IoT Domains

Phai Vu Dinh[1,2], Marwan Krunz[1], Diep N. Nguyen[2], Hoang Thai Dinh[2]
[1] Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721, USA
[2]School of Electrical and Data Engineering, University of Technology Sydney, NSW 2007, Australia

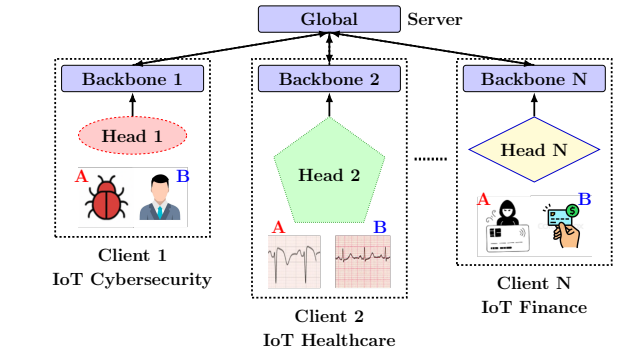
***Abstract*—Existing federated learning (FL)-based anomaly detection (AD) methods suffer from performance degradation due to domain heterogeneity, e.g., differences in features, data distributions, and data types across clients from different domains. This challenge is particularly pronounced in IoT systems and is more challenging than the well-known problem of non-i.i.d. data issues. To address this problem, we propose Personalized Federated Heterogeneous Autoencoder (PFAE), an AD framework that introduces a private head for processing domain-specific inputs and a public backbone for shared representation learning. PFAE design enables each client to effectively detect anomalies within its own data domain while leveraging global knowledge through model aggregation. We theoretically prove that the difference in the expected anomaly score of PFAE calculated using the public backbone for benign samples from any pair of clients is bounded. This implies that the distribution of benign samples is shared across domains, so benign samples from a client may not be misidentified as anomalies by other clients during inference. PFAE has lower training complexity than existing autoencoder-based methods. Experiments on eight non-i.i.d. datasets from different domains, where each client is trained on a single dataset, show that PFAE enhances the generalization of anomaly scores for benign samples across domains, especially under high anomaly ratios.**

***Index Terms*—Federated learning, autoencoder, anomaly detection, domain heterogeneity, IoT.**

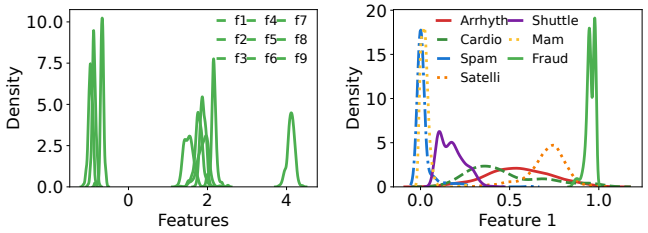
I. Introduction

FEDERATED learning (FL) offers a promising solution for anomaly detection (AD), a broad class of techniques for identifying data points or patterns that deviate from expected behavior. By enabling collaborative model training across decentralized devices while keeping clients' raw data local, FL not only preserves data privacy but also minimizes communication overhead and enhances model generalization. These advantages make FL suitable for distributed Internet of Things (IoT) environments where data are often not independent and identically distributed (non-i.i.d.) [1], [2].

However, IoT applications are quite diverse, resulting in *heterogeneous* domains and datasets. On the one hand, this poses a great challenge for FL modeling, far more than the conventional challenges of dealing with non-i.i.d. data within a single domain. On the other hand, domain heterogeneity may help anomaly detection models learn the distribution of benign samples shared across domains, thereby increasing anomaly detection accuracy. Specifically, for non-i.i.d. data within a single domain, features can be grouped so that each group is i.i.d. (see Fig. 1(b)). In contrast, domain heterogeneity refers to mismatched feature spaces, different data distributions, and varying data dimensionalities across different IoT application domains, as shown in Fig. 1(c). For example, ECG signals from IoT healthcare devices have 279 features per sample



(a) IoT applications in different domains



(b) Non-i.i.d data in IoT finance (Fraud [5])

(c) Distributions of seven non-i.i.d. domains

Fig. 1: Heterogeneous IoT domains.

(Arrhythmia) [3], network traffic logs in IoT cybersecurity have 122 features (NSL-KDD) [4], and transaction records in IoT finance have 29 features (Credit Card Fraud) [5], as shown in Fig. 1. Such discrepancies disrupt the consistency of the distribution learned by the global model, particularly when clients are trained on entirely different datasets, ultimately degrading AD accuracy for individual clients.

Averaging-based FL models have been extended to handle AD under domain heterogeneity by incorporating auxiliary models, such as Principal Component Analysis (PCA) [6]. The authors in [7] introduced federated principal components analysis (FedPE) and federated PCA on the Grassmann manifold (FedPG), which extract key eigenvectors to capture dominant patterns in each client's data and aggregate them at the server. Similarly, clients in Federated Autoencoder (FAE-Avg) share the weights of a neural network based on an autoencoder (AE) [8]. The authors in [9] introduced the Deep Autoencoder for Federated Learning (DAEF), which uses regularized one-layer neural networks to train individual layers of the client's AE model, so combines them into the global model. Another notable work is Federated with Noisy Global

Density Estimation and Self-supervised Ensemble Distillation (FADngs), which aims to align knowledge of the distribution across clients by sharing processed density functions used for AD [10]. In all these approaches (FedPE, FedPG, FAE-Avg, and DAEF), the anomaly score is typically derived from the reconstruction error between the original data and its latent-space reconstruction. Moreover, the feature alignment from different domains leads to information loss from the original data and adds significant complexity to both the training and inference stages.

Personalized FL (PFL) [11] attempts to address domain heterogeneity by tailoring models to individual clients, thereby improving local AD accuracy while benefiting from shared knowledge. Per-FedAvg [12] finds an initial shared model that users can easily adapt to their local datasets by performing a few steps of gradient descent (GD) on their data. Building on this the authors in [11] proposed pFedMe, which minimizes the Moreau envelopes of clients' loss functions while training a global model on the server's data. Similarly, the authors in [13] proposed FactorizedFL, which breaks down model parameters into shared vectors that capture common knowledge and client-specific vectors that represent individual clients' data. However, FactorizedFL requires a consistent architecture across clients' models. To address this limitation, the authors in [14] used optimized masking vectors to train client models (perFedMask), allowing clients to have different architectures. FedMD [15] uses transfer learning and knowledge distillation to aggregate client knowledge, such as digits, without requiring model sharing or consistent architectures.

It is worth noting that all the above PFL methods use supervised classification models to learn from known distributions. Because anomalies are continuously evolving, e.g., in cybersecurity, these models struggle to detect unseen anomalies due to their bias toward known data patterns. Zooming in on AD using unsupervised learning, the authors in [16] introduced FedP-OSVM, a PFL algorithm for one-class support vector machines that separates data into benign samples and anomalies. The authors in [17] proposed pFedHMD, a personalized and differentially private FL approach for AD. pFedHMD uses the Laplace mechanism to add noise to clients' gradients for enhanced privacy. Layer-wise PFL (LPFL), designed for transformer neural networks, was also applied to AD [18]. FedP-OSVM may be sensitive to the choice of K-nearest neighbors and the specific kernel method, whilst pFedHMD faces a trade-off between adding noise to the client's model and achieving high detection accuracy. Notably, these methods do not address domain heterogeneity.

In this paper, we propose a novel Personalized Federated Heterogeneous Autoencoders (PFAE) for AD under domain heterogeneity. Each PFAE client has two components: (i) a private head for domain-specific personalization; and (ii) a public backbone with shared parameters used in the global aggregation, as illustrated in Fig. 1(a). The private head projects data from different domains onto a common representation space, while the public backbone is trained locally and shared with the server, resulting in aligned distributions across different domains. Each client reconstructs the original input data at the private head and the representation data at the public backbone. Finally, each client combines the reconstruction errors from both and assigns an anomaly score to its data samples. We theoretically prove that the difference in the expected anomaly score of PFAE, calculated using the public backbone model for benign samples from any pair of clients, is bounded. This ensures that benign samples from one domain are not misclassified as anomalies by models trained for other domains. To validate the performance of PFAE, we apply eight non-i.i.d. datasets from different domains: Arrhythmia [3], Cardio [19], SpamBase [20], Satellite [21], Shuttle [22], Mammography [23], NSL-KDD [4], and Fraud [5], where each client uses one dataset. The MNIST [24] dataset is also used to evaluate the AD performance of various FL models as the number of clients increases, and the ratio of anomalies within benign samples varies. Results show that PFAE outperforms existing FL models for AD. Our main contributions are summarized as follows:

- We propose PFAE, a novel FL framework for AD in heterogeneous IoT domains. PFAE is designed to learn the distribution of benign samples across different domains. PFAE consists of a private head for processing domain-specific inputs and a public backbone for shared representation learning. A client's private head is used to map data input from a domain onto a consistent latent space. The public backbone is then shared with the server for global aggregation to leverage global knowledge from different domains. Finally, each client combines its private head with the public backbone for AD.
- We theoretically prove that the difference in the expected anomaly score of PFAE, calculated using the public backbone for benign samples from any pair of clients, is bounded. This means that the distribution of benign samples is shared across domains, so benign samples from a client will not be misidentified as anomalies by other clients during inference. In addition, PFAE has lower training complexity than common AE-based methods.
- Experiments on eight non-i.i.d. datasets from different domains show that PFAE trained on heterogeneous domains can achieve higher anomaly detection AUC in a given domain than centralized AD models trained only on that domain. In addition, PFAE improves the generalization of anomaly scores for benign samples across domains compared to related FL models and centralized models, particularly when the ratio of anomalies to benign samples is high (i.e., $\geq 100\%$). Finally, we conduct an ablation study to demonstrate the superior performance of the proposed PFAE over other methods.

II. Personalized Federated Heterogeneous Autoencoder

A. Problem Statement

Consider N heterogeneous datasets, $D^{(1)}, D^{(2)}, \ldots, D^{(N)}$, representing N different domains. let $D^{(i)} = \{\mathbf{x}^{(i,j)}\}_{j=1}^{n_i}$ denote the dataset of the ith client or device, where n_i is the number of data samples in $D^{(i)}$, $\mathbf{x}^{(i,j)} = \{x_t^{(i,j)}\}_{t=1}^{d_i} \in \mathbb{R}^{d_i}$ is the jth sample in $D^{(i)}$, and d_i is the dimensionality of the dataset $D^{(i)}$, i.e., number of features of each sample. The goal is to identify m_i anomalous samples within $D^{(i)}$. This task can be approached by learning a scoring function $\mathcal{A}_i \colon \mathbb{R}^{d_i} \rightarrow \mathbb{R}$ that assigns an anomaly score to each data sample $\mathbf{x}^{(i,j)}$. A higher value of $\mathcal{A}_i(\mathbf{x}^{(i,j)})$ indicates a higher likelihood that $\mathbf{x}^{(i,j)}$ is anomalous. Consequently, the m_i most

anomalous samples from $D^{(i)}$ are selected as those with the highest anomaly score.

To maintain the privacy of $D^{(i)}$, it can only be used to learn the function $\mathcal{A}_i$ locally. As a result, a benign data sample $\mathbf{x}^{(i,j)} \in D^{(i)}$ identified as such by $\mathcal{A}_i$ may be considered anomalous by the function $\mathcal{A}_k$ learned by another client k, due to differences between the two domains i and k. To address this issue, an FL approach can be employed, which involves collaborative learning of the functions $\{\mathcal{A}_i\}_{i=1}^N$ without requiring clients to share their private datasets. Finally, this collaborative approach aims to:

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}} \quad f(\boldsymbol{w}) \triangleq \frac{1}{N}\sum_{i=1}^{N} f_i(\boldsymbol{w}) \tag{1}$$

where $f_i(\boldsymbol{w})$ denotes the local objective function (loss function) of client i, which is used to learn $\mathcal{A}_i$, and $\boldsymbol{w}$ is the set of parameters of the public backbone. In standard FL, all clients share the same model, i.e., $f_i(\cdot) = f_j(\cdot)$ [25], allowing Federated Averaging (FedAvg) to learn a global model. In personalized FL (pFL), clients have different models or objectives ($f_i(\cdot) \neq f_j(\cdot)$).

B. Background: Federated Autoencoder For AD (FedAE)

An autoencoder (AE) consists of two components: an encoder and a decoder. The encoder maps an input sample $\mathbf{x}^{(i,j)} \in \mathbb{R}^{d_i}$ to a latent representation $\mathbf{z}^{(i,j)} \in \mathbb{R}^{d_{i,z}}$, where $d_{i,z} < d_i$. The decoder then maps $\mathbf{z}^{(i,j)}$ back to a reconstruction vector $\hat{\mathbf{x}}^{(i,j)} \in \mathbb{R}^{d_i}$ in the original input space. AE is trained to reconstruct $\hat{\mathbf{x}}^{(i,j)}$ as closely as possible to the original input $\mathbf{x}^{(i,j)}$. The reconstruction error can be used as the anomaly score of the data sample $\mathbf{x}^{(i,j)}$ [7]–[9]:

$$\mathcal{A}_i(\mathbf{x}^{(i,j)}) \triangleq \frac{1}{d_i}\sum_{t=1}^{d_i}\left(x_t^{(i,j)} - \hat{x}_t^{(i,j)}\right)^2. \tag{2}$$

The AE attempts to minimise the following loss function:

$$f_i(\boldsymbol{w}) \triangleq \frac{1}{n_i}\sum_{j=1}^{n_i}\mathcal{A}_i(\mathbf{x}^{(i,j)}) \tag{3}$$

where $\boldsymbol{w}$ now denotes the set of weights and biases of the AE for client i. To minimize $f(\boldsymbol{w})$ in (1), a traditional FedAvg approach requires the parameter sets $\boldsymbol{w}$ to be consistent across all clients' AEs, i.e., the dimensionality of the datasets $D^{(i)}$ must be the same for all clients ($d_i = d$ for all i). This allows for designing AE models with a shared architecture. In cases where $d_i \neq d_j$, dimensionality reduction techniques such as PCA can be applied to project the datasets onto the same feature space [6]. However, PCA can increase the computational complexity for all clients, and the transformed representations may lose important characteristics of the original data, resulting in a lower AD accuracy.

C. Proposed Personalized Federated Heterogeneous Autoencoder (PFAE)

1) Problem Formulation: Instead of addressing the traditional problem of FedAvg in (1), which assumes a consistent model across all clients, we take a different approach by localizing each client's model to be trained on non-i.i.d. data

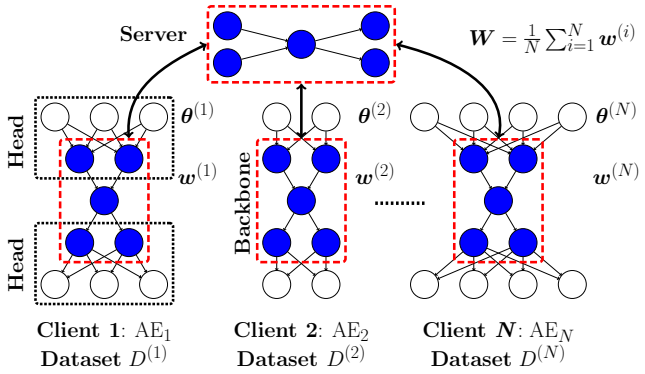


Fig. 2: PFAE architecture: Each client, represented by an AE, uses a private head $\boldsymbol{\theta}^{(i)}$ to map heterogeneous inputs onto a consistent latent space. This latent representation is then processed by the client's public backbone $\boldsymbol{w}^{(i)}$. The public backbones (shown in red) are shared with the server for global aggregation (global model $\boldsymbol{W}$).

with different input dimensionalities. This is achieved by defining a separate loss function for each client i:

$$\mathcal{L}_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)}) \triangleq g_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)}) + \lambda f_i(\boldsymbol{w}^{(i)}) \tag{4}$$

$$g_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)}) \triangleq \frac{1}{n_i}\sum_{j=1}^{n_i}\mathcal{A}_i(\mathbf{x}^{(i,j)}) \tag{5}$$

$$f_i(\boldsymbol{w}^{(i)}) \triangleq \frac{1}{n_i}\sum_{j=1}^{n_i}\mathcal{B}_i(\mathbf{y}^{(i,j)}) \tag{6}$$

where $g_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)})$ and $f_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)})$ are the private and public objectives of client i, respectively; $\boldsymbol{\theta}^{(i)}$ denotes the private head model, i.e., the set of trained parameters of client i, and $\boldsymbol{w}^{(i)}$ is the public backbone model, i.e., the set of shared parameters. In practice, each client uses its private encoder, $\boldsymbol{\xi}^{(i,e)}$, to map a heterogeneous domain, which may have different dimensionalities compared to other clients, onto a consistent latent space that shares the same dimensionality across all clients. The output of each client's private encoder is then passed through the public backbone, consisting of a public encoder $\boldsymbol{\phi}^{(i,e)}$ and the public decoder $\boldsymbol{\phi}^{(i,d)}$. The output of the public decoder is then fed into the private decoder of the private head, $\boldsymbol{\xi}^{(i,d)}$. The public backbone model $\boldsymbol{w}^{(i)} = (\boldsymbol{\phi}^{(i,e)}, \boldsymbol{\phi}^{(i,d)})$ is sent to the server for global aggregation into $\boldsymbol{w}^{(g)}$, as illustrated in Fig. 2. Note that $\boldsymbol{\theta}^{(i)} = (\boldsymbol{\xi}^{(i,e)}, \boldsymbol{\xi}^{(i,d)})$ is the set of parameters of the client's private head. In (4), λ is a hyperparameter that balances the contributions of the private head and public backbone models. When $\lambda = 0$, FL is not performed. In (6) $\mathcal{B}_i(\mathbf{y}^{(i,j)})$ is the reconstruction error between the input to the public encoder, $\mathbf{y}^{(i,j)}$, and the output of the public decoder, $\hat{\mathbf{y}}^{(i,j)}$. This reconstruction error, which enforces consistency across all clients, is given by:

$$\mathcal{B}_i(\mathbf{y}^{(i,j)}) \triangleq \frac{1}{d_y}\sum_{t=1}^{d_y}\left(y_t^{(i,j)} - \hat{y}_t^{(i,j)}\right)^2 \tag{7}$$

where d_y is the dimensionality of the input to the public backbone model. In practice, we design $\mathbf{y}^{(i,j)}$ as the output

of the first hidden layer, as it captures the most important information from the input $\mathbf{x}^{(i,j)}$ (see Fig. 2):

$$\mathbf{y}^{(i,j)} \triangleq \sigma(\boldsymbol{\xi}^{(i,e)}, \mathbf{x}^{(i,j)}), \hat{\mathbf{y}}^{(i,j)} \triangleq \sigma(\boldsymbol{\xi}^{(i,e)}, \boldsymbol{w}^{(i)}, \mathbf{x}^{(i,j)}) \quad (8)$$

where σ is a sigmoid activation function. The learning objective function of PFAE is formulated as follows:

$$\underset{\{\boldsymbol{\theta}^{(i)}\}_{i=1}^{N},\, \boldsymbol{w}}{\text{minimize}} \quad \frac{1}{N}\sum_{i=1}^{N}\left[g_i(\boldsymbol{\theta}^{(i)}, \boldsymbol{w}^{(i)}) + \lambda \cdot f_i(\boldsymbol{w}^{(i)})\right]. \quad (9)$$

2) Algorithm: A computational procedure for PFAE is given in Algorithm 1. Similar to FedAvg [25], at the beginning of each round t, the server broadcasts the global model $\boldsymbol{W}_t$ to all clients. After completing R local update steps, a uniformly sampled subset of clients $\mathcal{K}_t$, where $K = |\mathcal{K}_t|$, sends their updated public backbone models back to the server for global aggregation. Each client integrates its private head and public backbone model and computes gradients for both using the Adam optimizer [26].

3) Anomaly Detection at Individual Clients: After training PFAE, each client integrates its private head and public backbone model. The anomaly score for a data sample $\mathbf{x}^{(i,j)}$ of client i is calculated as follows:

$$\mathcal{S}_i(\mathbf{x}^{(i,j)}) \triangleq \mathcal{A}_i(\mathbf{x}^{(i,j)}) + \gamma \mathcal{B}_i(\mathbf{y}^{(i,j)}) \quad (11)$$

$$\gamma \triangleq \frac{\sum_{j=1}^{n_i} \mathcal{A}_i(\mathbf{x}^{(i,j)})}{\sum_{j=1}^{n_i} \mathcal{B}_i(\mathbf{y}^{(i,j)})} \quad (12)$$

where γ is a balancing parameter that controls the contribution of the two components.

III. Analysis

In this section, we analyze the difference in the expected anomaly scores calculated by the proposed PFAE, using the public backbone for benign samples across any pair of clients. This ensures that the distribution of benign samples is consistent across domains, reducing the likelihood that a benign sample from one client is misidentified as an anomaly by another client. Additionally, we analyze the training complexity of PFAE and compare it with baseline models.

A. Anomaly Detection Evaluation

Hypothesis 1. *By using the reconstruction error from AE variants, anomalies exhibit higher expected reconstruction errors compared to benign samples. Formally,* $\mathbb{E}\left[\mathcal{A}_b(\mathbf{x}^{(i,j)})\right] < \mathbb{E}\left[\mathcal{A}_a(\mathbf{x}^{(i,k)})\right]$, *where* $\mathbf{x}^{(i,j)} \in D_b^{(i)}$ *and* $\mathbf{x}^{(i,k)} \in D_a^{(i)}$. *Thus,* $D^{(i)} = D_b^{(i)} \cup D_a^{(i)}$, $D_a^{(i)} \cap D_b^{(i)} = \emptyset$. $D_b^{(i)}$ *and* $D_a^{(i)}$ *denote the subsets of benign samples and anomalies, respectively, in the dataset* $D^{(i)}$. *The number of benign samples* $n_b = |D_b^{(i)}|$ *is significantly greater than the number of anomalies,* $n_a = |D_a^{(i)}|$, *i.e.,* $n_b \gg n_a$. $\mathcal{A}_a(\cdot)$ *and* $\mathcal{A}_b(\cdot)$ *denote the anomaly scores of the anomalous and benign samples, respectively.*

Remark 1 (Hardt et al. [27])**.** *Let* $\mathcal{D}^{(i)}$ *and* $\mathcal{D}^{(j)}$ *be two datasets of size* n *that differ in exactly one data sample. Let* $\boldsymbol{w}_R^{(i)}$ *and* $\boldsymbol{w}_R^{(j)}$ *be the model parameters obtained by running* R *steps of stochastic gradient descent (SGD) on* $\mathcal{D}^{(i)}$ *and* $\mathcal{D}^{(j)}$, *respectively. Define* $\delta_R \triangleq \left\|\boldsymbol{w}_R^{(i)} - \boldsymbol{w}_R^{(j)}\right\|$. *Then,* $\mathbb{E}[\delta_R] \leq \frac{2L}{n}\sum_{t=1}^{R}\alpha_t$, *where* L *is the Lipschitz constant of the loss function and* α_t *is the learning rate at step* t.

Remark 2 (Union Bound [28])**.** *Let* $\{E_i\}_{i=1}^{\infty}$ *be a finite or countable collection of events. Then,* $\Pr(\bigcup_i E_i) \leq \sum_i \Pr(E_i)$.

Remark 3 (Hoeffding's Inequality [29])**.** *Let* $X_1, X_2, \ldots, X_n$ *be independent random variables such that* $\Pr[a \leq X_i \leq b] = 1$, $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. *Then, for any* $\varepsilon > 0$, $\Pr\left(\left|\bar{X} - \mathbb{E}[\bar{X}]\right| \geq \varepsilon\right) \leq 2\exp\left(-\frac{2n^2\varepsilon^2}{\sum_{i=1}^{n}(b-a)^2}\right)$.

Theorem 1 (PFAE: Benign Score Generalization Consistency)**.** *Let* $\mathcal{Y}_b^{(k)}$ *and* $\mathcal{Y}_b^{(j)}$ *be the outputs of the private*

Algorithm 1 PFAE: Personalized Federated Autoencoder

Require: T (number of global rounds), R (number of local updates), K (number of selected clients), η learning rate, Adam's parameters (decay rate for the first moment β_1 and second moment β_2, small constant for numerical stability ϵ). Initial models: server's global model $\boldsymbol{W}_0$, clients' private head model $\{\boldsymbol{\theta}_0^{(i)}\}_{i=1}^{N}$

function ADAM($\mathbf{x}, \mathbf{g}, \mathbf{m}, \mathbf{v}, r$) [26]
  $\mathbf{m} \leftarrow \beta_1\mathbf{m} + (1-\beta_1)\mathbf{g}$ ▷ 1^{st} moment estimate
  $\mathbf{v} \leftarrow \beta_2\mathbf{v} + (1-\beta_2)\mathbf{g}^2$ ▷ 2^{nd} moment estimate
  $\hat{\mathbf{m}} \leftarrow \mathbf{m}/(1-\beta_1^{r+1})$ ▷ Bias-corrected estimates
  $\hat{\mathbf{v}} \leftarrow \mathbf{v}/(1-\beta_2^{r+1})$ ▷ Bias-corrected estimates
  return $\mathbf{x} - \eta\hat{\mathbf{m}}/(\sqrt{\hat{\mathbf{v}}} + \epsilon)$
end function
for $t = 0$ to $T-1$ **do** ▷ Global rounds
  Server sends global model $\boldsymbol{W}_t$ to all clients
  for all clients $i = 1$ to N **do**
    Initialize: $\boldsymbol{w}_{t,0}^{(i)} \leftarrow \boldsymbol{W}_t$
    Load private head model: $\boldsymbol{\theta}_{t,0}^{(i)} \leftarrow \boldsymbol{\theta}_t^{(i)}$
    Initialize Adam moments: $\boldsymbol{m}_\theta^{(i)} \leftarrow \mathbf{0}$, $\boldsymbol{v}_\theta^{(i)} \leftarrow \mathbf{0}$, $\boldsymbol{m}_w^{(i)} \leftarrow \mathbf{0}$, $\boldsymbol{v}_w^{(i)} \leftarrow \mathbf{0}$
    for $r = 0$ to $R-1$ **do** ▷ Local updates
      Sample mini-batch $D^{(i)}$ from client i
      $\mathbf{g}_\theta^{(i)} \leftarrow \nabla_\theta \mathcal{L}_i(\boldsymbol{\theta}_{t,r}^{(i)}, \boldsymbol{w}_{t,r}^{(i)}; D^{(i)})$ ▷ gradients
      $\mathbf{g}_w^{(i)} \leftarrow \nabla_w \mathcal{L}_i(\boldsymbol{\theta}_{t,r}^{(i)}, \boldsymbol{w}_{t,r}^{(i)}; D^{(i)})$ ▷ gradients
      $\boldsymbol{\theta}_{t,r+1}^{(i)} \leftarrow$ ADAM($\boldsymbol{\theta}_{t,r}^{(i)}, \mathbf{g}_\theta^{(i)}, \boldsymbol{m}_\theta^{(i)}, \boldsymbol{v}_\theta^{(i)}, r$)
      $\boldsymbol{w}_{t,r+1}^{(i)} \leftarrow$ ADAM($\boldsymbol{w}_{t,r}^{(i)}, \mathbf{g}_w^{(i)}, \boldsymbol{m}_w^{(i)}, \boldsymbol{v}_w^{(i)}, r$)
    end for
    Final private head model: $\boldsymbol{\theta}_{t+1}^{(i)} \leftarrow \boldsymbol{\theta}_{t,R}^{(i)}$, $\boldsymbol{w}_{t+1}^{(i)} \leftarrow \boldsymbol{w}_{t,R}^{(i)}$
    Save private head model: $\boldsymbol{\theta}_{t+1}^{(i)}$
  end for
  Server selects subset $\mathcal{K}_t \subseteq \{1, \ldots, N\}$ of $K = |\mathcal{K}_t|$ clients uniformly at random, $1 \leq K \leq N$
  Each selected client $i \in \mathcal{K}_t$ sends $\boldsymbol{w}_{t+1}^{(i)}$ to server
  Server updates global model:

$$\boldsymbol{W}_{t+1} \leftarrow \frac{1}{K}\sum_{i \in \mathcal{K}_t} \boldsymbol{w}_{t+1}^{(i)}. \quad (10)$$

end for

encoders of the private heads when the network is fed with the empirical training distributions (datasets) $\mathcal{D}_b^{(k)}$ and $\mathcal{D}_b^{(j)}$, respectively, from benign samples of clients k and j. Assume that the datasets $\mathcal{Y}_b^{(k)}$ and $\mathcal{Y}_b^{(j)}$ are i.i.d. samples from the same distribution, but they differ in exactly one data sample, each dataset of size $n_j = n_k = n$. Suppose that clients j and k train their public backbone models $\boldsymbol{w}^{(k)}$ and $\boldsymbol{w}^{(j)}$ using R epochs of SGD with learning rate η. Assume that the loss function $f_i(\boldsymbol{w}^{(i)}, \mathbf{y}) \in [0, M]$, as defined in (6), is L-Lipschitz continuous with respect to $\boldsymbol{w}$. Then, with probability at least $1-\rho$, the difference in expected anomaly scores of the benign samples across the two clients satisfies:

$$\begin{aligned}\delta &\triangleq \left|\mathbb{E}_{\mathbf{y}\sim\mathcal{Y}_b^{(k)}} f(\boldsymbol{w}^{(k)}, \mathbf{y}) - \mathbb{E}_{\mathbf{y}\sim\mathcal{Y}_b^{(j)}} f(\boldsymbol{w}^{(j)}, \mathbf{y})\right| \\ &\leq \frac{4(K-1)\eta L^2 R}{nK} + \frac{M}{\sqrt{2n}}\sqrt{\ln(\frac{4}{\rho})} \triangleq \delta^*.\end{aligned} \tag{13}$$

Theorem 1 suggests that the expected anomaly score of PFAE for benign samples remains nearly consistent across similar clients when using the public backbone model, calculated using (7). This supports the consistent generalization capability of the global model across benign samples from different clients. In other words, the average anomaly score for benign samples, as calculated by the public backbone model, is expected to differ slightly across clients and should be significantly lower than the average anomaly scores of abnormal samples for each client, as observed in Fig. 4. Next, we denote $|\cdot|$ and $\|\cdot\|$ as the absolute value and norm, respectively.

Proof. Applying the triangle inequality to δ on the left-hand side of (13), we obtain:

$$\delta \leq \delta_1 + \delta_2 \tag{14}$$

$$\delta_1 \triangleq \left|\mathbb{E}_{y\sim\mathcal{Y}^{(k)}} f(\boldsymbol{w}^{(k)}, \mathbf{y}) - \mathbb{E}_{y\sim\mathcal{Y}^{(k)}} f(\boldsymbol{w}^{(j)}, \mathbf{y})\right| \tag{15}$$

$$\delta_2 \triangleq \left|\mathbb{E}_{y\sim\mathcal{Y}^{(k)}} f(\boldsymbol{w}^{(j)}, \mathbf{y}) - \mathbb{E}_{y\sim\mathcal{Y}^{(j)}} f(\boldsymbol{w}^{(j)}, \mathbf{y})\right|. \tag{16}$$

If $\mathbf{X}$ and $\mathbf{Y}$ are two random variables, then following [30], we have $\mathbb{E}[\mathbf{X}-\mathbf{Y}] = \mathbb{E}[\mathbf{X}] - \mathbb{E}[\mathbf{Y}]$. Therefore,

$$\delta_1 = \left|\mathbb{E}_{y\sim\mathcal{Y}^{(k)}}[f(\boldsymbol{w}^{(k)}, \mathbf{y}) - f(\boldsymbol{w}^{(j)}, \mathbf{y})]\right| \tag{17}$$

$$\leq \left|\mathbb{E}_{y\sim\mathcal{Y}^{(k)}}[L||\boldsymbol{w}^{(k)} - \boldsymbol{w}^{(j)}||]\right| = L\|\boldsymbol{w}^{(k)} - \boldsymbol{w}^{(j)}\| \tag{18}$$

since the function $f(\boldsymbol{w}, \mathbf{y})$ is L-Lipschitz in $\boldsymbol{w}$ for every $\mathbf{y}$, i.e., $|f(\boldsymbol{w}^{(k)}, \mathbf{y}) - f(\boldsymbol{w}^{(j)}, \mathbf{y})| \leq L\cdot\|\boldsymbol{w}^{(k)} - \boldsymbol{w}^{(j)}\| \quad \forall \mathbf{y}$. Applying the triangle inequality to $\|\boldsymbol{w}^{(k)} - \boldsymbol{w}^{(j)}\|$, we have:

$$\|\boldsymbol{w}^{(k)} - \boldsymbol{w}^{(j)}\| \leq \|\boldsymbol{w}^{(k)} - \boldsymbol{w}\| + \|\boldsymbol{w} - \boldsymbol{w}^{(j)}\|$$

where $\boldsymbol{w} = \frac{1}{K}\sum_{i=1}^{K}(\boldsymbol{w}^{(i)})$ as observed in (10). We have

$$\begin{aligned}\delta_3 &\triangleq \|\boldsymbol{w} - \boldsymbol{w}^{(j)}\| = \|\frac{1}{K}\sum_{i=1}^{K}(\boldsymbol{w}^{(i)}) - \boldsymbol{w}^{(j)}\| \\ &= \|\frac{1}{K}(\sum_{i=1}^{K}\boldsymbol{w}^{(i)} - K\boldsymbol{w}^{(j)})\| = \|\frac{1}{K}\sum_{i=1}^{K}(\boldsymbol{w}^{(i)} - \boldsymbol{w}^{(j)})\| \\ &= \|\frac{1}{K}\sum_{\substack{i=1\\ i\neq j}}^{K}(\boldsymbol{w}^{(i)} - \boldsymbol{w}^{(j)})\| \leq \frac{1}{K}\sum_{\substack{i=1\\ i\neq j}}^{K}\|\boldsymbol{w}^{(i)} - \boldsymbol{w}^{(j)}\| \\ &\leq \frac{K-1}{K}\max_{i\neq j}\|\boldsymbol{w}^{(i)} - \boldsymbol{w}^{(j)}\|.\end{aligned}$$

Using Remark 1, we have $\delta_3 \leq \frac{2(K-1)\eta LR}{nK}$, where $\alpha_t = \eta$ for all t. Similarly, define $\delta_4 \triangleq \left\|\boldsymbol{w}^{(k)} - \boldsymbol{w}\right\|$; then $\delta_4 \leq \frac{2(K-1)\eta LR}{nK}$. Therefore,

$$\delta_1 \leq \frac{4(K-1)\eta L^2 R}{nK}. \tag{19}$$

Next, we aim to quantify the difference in the expected anomaly scores of benign samples, denoted as δ_2 as observed in (16), under the two distributions $\mathcal{Y}_b^{(k)}$ and $\mathcal{Y}_b^{(j)}$, as follows:

$$\begin{aligned}&\delta_2 \triangleq |\hat{\mu}^{(j)} - \hat{\mu}^{(k)}| \\ &\hat{\mu}^{(j)} \triangleq \frac{1}{n}\sum_{t=1}^{n} f\left(\mathbf{y}_t^{(j)}\right), \hat{\mu}^{(k)} \triangleq \frac{1}{n}\sum_{t=1}^{n} f\left(\mathbf{y}_t^{(k)}\right)\end{aligned}$$

where $\mathbf{y}_t^{(j)}$ and $\mathbf{y}_t^{(k)}$ denote samples drawn from the benign data distributions $\mathcal{Y}_b^{(j)}$ and $\mathcal{Y}_b^{(k)}$ of clients j and k, respectively. Since $\mathcal{Y}_b^{(k)}$ and $\mathcal{Y}_b^{(j)}$ are i.i.d. and the nearly same distribution, i.e., $\mathbb{E}[f(x^{(j)})] - \mathbb{E}[f(x^{(k)})] \approx 0$, applying triangle inequality, we have:

$$\begin{aligned}\delta_2 \leq &\left|\hat{\mu}^{(j)} - \mathbb{E}[f(\mathbf{y}^{(j)})]\right| + \left|\mathbb{E}[f(\mathbf{y}^{(j)})] - \mathbb{E}[f(\mathbf{y}^{(k)})]\right| \\ &+ \left|\hat{\mu}^{(k)} - \mathbb{E}[f(\mathbf{y}^{(k)})]\right|.\end{aligned}$$

Following to Remarks, i.e., 2 and 3, we have

$$\begin{aligned}\Pr(\delta_2 \geq \varepsilon) &\leq \Pr\left(|\hat{\mu}^{(j)} - \mathbb{E}[f(\mathbf{y}^{(j)})]| \geq \frac{\varepsilon}{2}\right) \\ &\quad + \Pr\left(|\hat{\mu}^{(k)} - \mathbb{E}[f(\mathbf{y}^{(k)})]| \geq \frac{\varepsilon}{2}\right) \\ &\leq 2\exp\left(-\frac{2n\varepsilon^2}{(b-a)^2}\right) + 2\exp\left(-\frac{2n\varepsilon^2}{(b-a)^2}\right).\end{aligned}$$

For $a = 0$ and $b = M$, we have

$$\Pr(\delta_2 \geq \varepsilon) \leq 4\exp\left(-\frac{2n\varepsilon^2}{M^2}\right) \triangleq \rho. \tag{20}$$

Calculate ε in (20), we have $\varepsilon = \frac{M}{\sqrt{2n}}\sqrt{\ln(\frac{4}{\rho})}$. Therefore, with probability at least $1-\rho$, we have

$$\delta_2 \leq \frac{M}{\sqrt{2n}}\sqrt{\ln(\frac{4}{\rho})}. \tag{21}$$

From Eqs. (19) and (21), Theorem 1 is proven. □

TABLE I: Training complexity comparison.

Methods	Complexity
PCA-FAE-Avg [25]	$\mathcal{O}\left(2KTnd^2HR + Knd^2_{\max} + Kd^3_{\max}\right)$
PCA-FedPE [7]	$\mathcal{O}\left(KTnd^2R + KTd^3R + Knd^2_{\max} + Kd^3_{\max}\right)$
PCA-FedPG [7]	$\mathcal{O}\left(KTnd^2R + KTd^3R + Knd^2_{\max} + Kd^3_{\max}\right)$
Our proposed PFAE	$\mathcal{O}\left(2KTnd^2HR + 2KTndd_{\max}R\right)$

From Theorem 1, the average anomaly score of benign samples calculated by the public backbone model (as defined in 7) does not differ significantly across clients. According to Hypothesis 1, the average anomaly score of anomalous samples is often substantially higher than that of benign ones. Therefore, PFAE can effectively differentiate anomalies from benign samples, as shown in (11). Theorem 1 holds under the assumption that the datasets $\mathcal{Y}_b^{(k)}$ and $\mathcal{Y}_b^{(j)}$ are i.i.d. samples drawn from the same distribution and they differ in exactly one data sample. In practice, $\mathcal{Y}_b^{(k)}$ represents the outputs of the private encoder of the private head, i.e., the first layer of the client model after applying the sigmoid function. This transformation may help to bring originally non-i.i.d. data closer to being i.i.d. When $n \to \infty$, δ^* in (13) approaches zero. This means that the average anomaly score of benign samples calculated by the public backbone model is nearly the same across clients.

B. Training Complexity

We analyze the training complexity of the proposed PFAE and compare it to PCA-FAE-Avg [6], [8], [25], PCA-FedPE [7], and PCA-FedPG [7]. These models require PCA [6] to transform heterogeneous data of different dimensions onto a consistent representation. Assume that PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG apply PCA [6] to project their input data onto lower-dimensional and consistent feature spaces of dimension $d \triangleq \min\{d_i\}_{i=1}^N$. In contrast, the proposed PFAE uses its private head model to project data onto a representation with d-dimensional space. For PCA-FAE-Avg, the number of hidden layers is H, and each hidden layer has $q_t \triangleq \alpha_t d$ neurons, where $t \in \{1, \ldots, H\}$, $\alpha_t \in (0, 1]$. The public backbone structure of PFAE is the same as that of PCA-FAE-Avg. The training samples of all K clients are n.

Lemma 1. *Let a single AE be trained on a dataset of size n for R epochs. Suppose the input dimension is d, and the encoder consists of H hidden layers, $q_t = \alpha_t d$, where $\alpha_t \in (0, 1]$. Then, the total training cost of the AE using a feedforward network is $\mathcal{O}(2nd^2HR)$.*

Proof. Set $q_0 \triangleq d$ and $q_{H+1} \triangleq d$. The total parameters used for training the AE is $P_{AE} \triangleq \sum_{t=0}^{H}(q_t q_{t+1}) = d^2 \sum_{t=0}^{H}(\alpha_t \alpha_{t+1})$, where $\alpha_0 = \alpha_{H+t} = 1$. The training of the AE depends on the forward pass and the backwards propagation on each single neuron. Therefore, the training complexity of one epoch of the AE for n samples is $\mathcal{O}\left(2nd^2 \sum_{t=0}^{H}(\alpha_t \alpha_{t+1})\right) = \mathcal{O}\left(2nd^2H\right)$. For training R epochs, the training complexity of the AE is $\mathcal{O}\left(2nd^2HR\right)$. □

Assume that $d_{\max} \triangleq \max\{d_i\}_{i=1}^N$. For PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG, they need to use PCA with the complexity, i.e., $\mathcal{O}\left(nd^2_{\max} + d^3_{\max}\right)$ [31], to make the consistent input for K clients. From Lemma 1, we have the training complexity comparison for four FL models, as observed in Table I. Take a comparison between PCA-FAE-Avg and PCA-FedPE (or PCA-FedPG), we have $\mathcal{O}_{\textit{FAE-Avg}} - \mathcal{O}_{\textit{FedPE}} = \mathcal{O}\left(2KTnd^2HR - KTnd^2R - KTd^3R\right) = \mathcal{O}\left(KTR(2nd^2H - nd^2 - d^3)\right)$. Therefore, for high-dimensional datasets with $d \gg 2nH$, the training complexity of PCA-FedPE and PCA-FedPG is much higher than that of PCA-FAE-Avg. Take a comparison between PFAE and PCA-FAE-Avg, we have $\mathcal{O}_{\textit{PFAE}} - \mathcal{O}_{\textit{PCA-FAE-Avg}} = \mathcal{O}\left(2KTndd_{\max}R - Knd^2_{\max} - Kd^3_{\max}\right) = \mathcal{O}\left(Kd^2(2TnR - n - d)\right)$, when $d_{\max} = d$. For $2TnR > n + d$, the training complexity of PFAE is greater than that of PCA-FAE-Avg. In contrast, for $n + d \gg 2TnR$, the training complexity of PCA-FAE-Avg is much higher than that of the proposed PFAE.

TABLE II: Datasets.

Datasets' name	Short	N	n_a	n_b	$\frac{n_a}{n_b}$	d	CV
Arrhythmia	M1	451	65	386	14.4%	279	38.4
Cardio	M2	1830	176	1654	9.6%	21	25.6
Spambase	M3	4600	1812	2788	39.4%	57	22.1
Satellite	M4	6434	2036	4398	31.6%	36	8.0
Shuttle	M5	49096	3510	45586	7.1%	9	4.3
Mammography	M6	11182	260	10922	2.3%	6	42.9
NSLKDD	M7	148516	77053	71463	51.9%	122	87.4
Fraud	M8	284806	492	284314	0.2%	29	8.8

IV. Experimental Settings

A. Metric Evaluation

We use the area under the ROC curve (*AUC*), *F1-score*, false alarm rate (*FAR*), and miss detection rate (*MDR*) to evaluate the performance of AD models [32], where $\textit{MDR} \triangleq \frac{\textit{FN}}{\textit{FN+TP}}$ and $\textit{FAR} \triangleq \frac{\textit{FP}}{\textit{FP+TN}}$, with *TP*, *TN*, *FP*, and *FN* representing *True Positive*, *True Negative*, *False Positive*, and *False Negative*, respectively. We evaluate federated models as follows:

$$\textit{AUC} \triangleq \frac{1}{N}\sum_{i=1}^{N}(\textit{AUC}_i), \textit{F1-score} \triangleq \frac{1}{N}\sum_{i=1}^{N}(\textit{F1-score}_i) \quad (22)$$

$$\textit{FAR} \triangleq \frac{1}{N}\sum_{i=1}^{N}(\textit{FAR}_i), \textit{MDR} \triangleq \frac{1}{N}\sum_{i=1}^{N}(\textit{MDR}_i) \quad (23)$$

where N is the number of clients. In addition to assessing model performance, we measure the heterogeneity of datasets using the coefficient of variation *CV* [33], defined as $\textit{CV} \triangleq 100\frac{1}{d}\sum_{t=1}^{d}\frac{\sigma_t}{\mu_t}$, where $\mu_t > 0$. Here, σ_t is the standard deviation, μ_t is the mean of the tth feature, and d represents the dimensionality of the dataset. A high *CV* indicates that the dataset is highly heterogeneous.

B. Datasets

We use eight real-world datasets for AD in IoT, which are different domains with non-i.i.d. and different dimensionalities, as shown in Table II [32], [34], [35]. In our FL setup, each dataset is assigned to a client. This allows a client to identify anomalies in one dataset after the training process is complete. The datasets come from different AD domains, including Arrhythmia (M1) [3], Cardio Disease (M2) [19], Spambase (M3) [20], Satellite (M4) [21], Shuttle

TABLE III: *AUC* obtained by PFAE is compared with that of the FL methods on eight non-i.i.d. datasets.

No. Clients	PCA-FAE-Avg [8]	PCA-FedPE [7]	PCA-FedPG [7]	PFAE
2	0.767	**0.895**	0.895	0.864
3	0.760	0.768	0.775	**0.778**
4	0.723	0.735	0.736	**0.740**
5	0.686	0.739	0.768	**0.770**
6	0.671	0.744	0.762	**0.771**
7	0.632	**0.726**	0.691	0.709
8	0.733	0.679	0.707	**0.783**
Avg	0.710	0.755	0.762	**0.774**

TABLE IV: *AUC* obtained by PFAE is compared with that of the FL methods on MNIST datasets with 21 clients.

Anomaly Ratio	FAE-Avg [8]	PCA-FedPE [7]	PCA-PedPG [7]	PFAE
0.001	0.672	0.726	0.787	**0.957**
0.005	0.589	0.732	0.790	**0.943**
0.010	0.608	0.775	0.828	**0.938**
0.050	0.539	0.794	0.830	**0.882**
0.100	0.521	0.791	0.813	**0.831**
0.200	0.492	0.755	0.733	**0.756**
0.400	0.444	**0.708**	0.649	0.655
0.600	0.447	**0.638**	0.615	0.585
0.800	0.424	**0.596**	0.584	0.554
1.000	0.411	**0.557**	0.551	0.524
Avg	0.515	0.707	0.718	**0.762**

(M5) [22], Mammography (M6) [23], Network Security NSL-KDD (M7) [4], and Credit Card Fraud (M8) [5]. These are described in detail in Table II. The dataset's coefficient of variation (*CV*) is also different. We use the MNIST image dataset [24] to evaluate performance with a large number of clients. Anomalies are selected from classes 0, 8, and 9, while the remaining classes are benign samples. Each client's training dataset combines samples from one benign class with one anomaly class. For example, Client 1's dataset includes samples from class 1 (benign) and class 0 (anomalous). The anomaly-to-benign sample ratio $\frac{n_a}{n_b}$ is adjusted using the list $(0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0)$. A high ratio, such as $\frac{n_a}{n_b} = 1$, often results in low AD accuracy for centralized models without FL [32].

C. Experimental Setup

Experiments are conducted using TensorFlow for the proposed model (PFAE) and the related model (PCA-FAE-Avg), and PyTorch for PCA-FedPE and PCA-FedPG [7], on a Red Hat Enterprise Linux system with 188 GB of RAM and an Intel 32-core CPU. For the eight non-i.i.d. datasets, each client is assigned one dataset, leading to seven scenarios with client counts of $(2, 3, 4, 5, 6, 7, 8)$. In the MNIST dataset, three labels are designated as anomalies, allowing the creation of up to 21 distinct datasets with the remaining seven labels. Therefore, we observe scenarios with client counts of $(3, 6, 9, 12, 16, 21)$. The learning rate, number of client epochs R, number of server epochs T, and batch size are set to 0.0001, 5, 2000, and 100, respectively [32]. To tune the model, we select the latent space sizes as $(d_z, 2d_z, 3d_z, 4d_z, 5d_z)$ and choose the best *AUC* for each model, where $d_z \triangleq \lfloor \sqrt{d_{min}} \rfloor$ and $d_{min} \triangleq \min\{d_i\}_{i=1}^{N}$. For eight non-i.i.d. datasets, PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG use PCA to project data into the space with dimensionality d_{min}.

For MNIST, high dimensionality caused extreme computational costs, resulting in *NaN* values during the first two global epochs. Therefore, it is necessary to reduce the input dimension of each training client to $0.1d \approx 78$ using PCA [6] before applying FedPE and FedPG. In contrast, FAE-Avg [8] and the proposed PFAE do not require using PCA. For FAE-Avg and PFAE, their global models share the same structure with clients' public backbone: $\{d_{min}, 0.9d_{min}, 0.8d_{min}, d_z, 0.8d_{min}, 0.9d_{min}, d_{min}\}$ [40]. For MNIST, we set $d_{min} = 0.1d$. In the PFAE, we use a leaky ReLU activation function for the output of the client (AE), while the output of the private head and the public backbone of each client applies a sigmoid function, and the remaining layers of the private head and public backbone use the tanh function [32].

TABLE V: *AUC* obtained by PFAE is compared with that of locally trained models on eight non-i.i.d. datasets.

D	LOF [36]	AE [37]	VAE [37]	AnoGAN [38]	SkipGAN [39]	PFAE
M1	0.792	0.742	**0.802**	0.741	0.742	0.745
M2	0.743	0.761	**0.935**	0.802	0.858	0.775
M3	0.498	**0.607**	0.539	0.530	0.512	0.541
M4	0.565	**0.676**	0.606	0.625	0.616	0.644
M5	0.566	**0.987**	0.986	0.933	0.964	0.974
M6	0.821	0.866	0.860	0.786	0.813	**0.902**
M7	0.507	0.267	0.051	0.628	0.408	**0.746**
M8	0.519	**0.950**	0.949	0.916	0.920	0.938
Avg	0.626	0.732	0.716	0.745	0.729	**0.783**

TABLE VI: *F1-score*, *FAR*, and *MDR* obtained by PFAE on eight non-i.i.d. datasets.

No. Clients	2	3	4	5	6	7	8	Avg
F1-score	0.879	0.777	0.738	0.788	0.814	0.748	0.825	0.796
FAR	0.437	0.437	0.480	0.390	0.462	0.539	0.452	0.457
MDR	0.121	0.223	0.262	0.212	0.186	0.252	0.175	0.204

V. Simulation Results

A. Performance of FL Models for AD

First, the *AUC* obtained by PFAE is higher than PCA-FAE-Avg, PCA-FedPE, and PCA-Fed-PG on eight heterogeneous datasets with two to eight clients, as shown in Table III. For instance, the average *AUC* for PFAE, PCA-FAE-Avg, PCA-FedPE, and PCA-Fed-PG across seven experiments is 0.774, 0.710, 0.755, and 0.762, respectively. For these datasets, the *AUC* achieved by PFAE is up to 5% higher than that of the other methods. This demonstrates the effectiveness of PFAE in processing the original heterogeneous datasets compared to the other three methods that apply PCA to map the data onto the same space.

Second, PFAE outperforms PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG on MNIST, as shown in Table IV. PFAE, PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG achieve average AUCs of 0.762, 0.515, 0.707, and 0.718, respectively, over 10 experiments with an anomaly ratio from 0.001 to 1.0. PCA-FedPE and PCA-FedPG struggle to capture the structure of the nonlinear dataset at high dimensionality. PFAE leverages both private and global reconstruction errors for anomaly scoring, enhancing its ability to identify anomalies on a per-client basis.

Third, we compare PFAE with centralized anomaly detection (AD) models, including LOF [36], AE [37], VAE [37], AnoGAN [38], and SkipGAN [39], all of which are trained locally without global aggregation. Note that LOF is a non-parametric method, while AE and VAE use reconstruction error as the anomaly score to identify anomalies. AnoGAN and SkipGAN, based on generative adversarial networks (GANs), aim to identify anomalies by leveraging both the generator and discriminator reconstruction errors. The results for these centralized models are taken from [32], [34], [37]. In general, PFAE outperforms centralized AD models (LOF, AE, VAE,

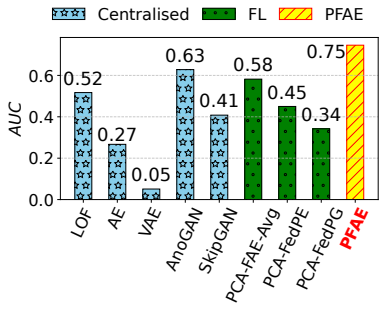


Fig. 3: Comparison of *AUC* on the NSL-KDD dataset [4] when Hypothesis 1 is violated ($\frac{n_a}{n_b} = 51.9\%$ as shown in Table II).

TABLE VII: *AUC* obtained by PFAE is compared to that of an AE trained on one dataset and evaluated on eight non-i.i.d. datasets.

No. Clients	2	3	4	5	6	7	8	Avg
PCA-AE	0.708	0.713	0.728	0.694	0.657	0.384	0.676	0.651
PFAE	**0.864**	**0.778**	**0.740**	**0.770**	**0.771**	**0.709**	**0.783**	**0.774**

AnoGAN, and SkipGAN), as observed in Table V. The average AUCs achieved by LOF [36], AE [37], VAE [37], AnoGAN [38], SkipGAN [39], and PFAE across the eight non-i.i.d. datasets are 0.626, 0.732, 0.716, 0.745, 0.729, and 0.783, respectively. These results demonstrate that PFAE trained on heterogeneous domains can increase the anomaly detection AUC compared to centralized AD models trained on data from a single domain. The average *F1-score*, *FAR*, and *MRD* obtained by PFAE on the eight datasets are 0.796, 0.457, and 0.204, respectively (Table VI). Notably, when Hypothesis 1 is violated with $\frac{n_a}{n_b} = 51.59\%$, NSL-KDD dataset as shown in Table II, PFAE outperforms the above methods by up to 12% in terms of AUC, as shown in Fig. 3. The *AUC* achieved by the FL-based methods, i.e., PCA-FAE-Avg, PCA-FedPE, and PCA-FedPG, is not higher than that of methods trained in a centralized manner without global aggregation (AnoGAN [38]). This highlights the effectiveness of the proposed PFAE compared to other federated and centralized methods for anomaly detection.

B. PFAE's Model Performance Evaluation

First, we compare the *AUC* obtained by PFAE with a baseline model, an AE trained independently by a single client without data from others, as shown in Table VII. This trained AE model is then used to evaluate AD performance on other clients' datasets. Without training data from other clients, the AE model struggles to differentiate benign samples from anomalies in other datasets. For example, the average *AUC* achieved by PCA-AE is only 0.651, about 12% lower than that of PFAE.

We apply PCA for dimensionality reduction on the eight non-i.i.d. datasets before training PFAE. The average *AUC* obtained by PFAE is slightly higher than that of PCA-PFAE, 0.774 vs. 0.766, as shown in Table VIII. Both models achieve high results, highlighting the benefits of a personalized model for preserving local data characteristics for AD. However, PCA-PFAE may suffer performance degradation due to information loss during the PCA transformation.

We compare the *AUC* obtained using the reconstruction errors of the private head model $\mathcal{A}(\mathbf{x}^{(i,j)})$, the public backbone model $\mathcal{B}(\mathbf{x}^{(i,j)})$, and the anomaly score of PFAE, $\mathcal{S}(\mathbf{x}^{(i,j)})$, as shown in Table IX. The average *AUC* for $\mathcal{S}(\mathbf{x}^{(i,j)})$ is higher than those for $\mathcal{A}(\mathbf{x}^{(i,j)})$ and $\mathcal{B}(\mathbf{x}^{(i,j)})$ across the eight experiments: 0.774 vs. 0.766 vs. 0.738. Additionally, the result for the public backbone model $\mathcal{B}(\mathbf{x}^{(i,j)})$ is lower than that for the private head model $\mathcal{A}(\mathbf{x}^{(i,j)})$. These findings demonstrate the effectiveness of the personalized FL model and indicate that incorporating the public backbone model enhances generalization in AD.

TABLE VIII: *AUC* obtained by PCA-PFAE and PFAE.

No. Clients	2	3	4	5	6	7	8	Avg
PCA-PFAE	**0.886**	0.768	0.729	**0.782**	**0.784**	**0.736**	0.673	0.766
PFAE	0.864	**0.778**	**0.740**	0.770	0.771	0.709	**0.783**	**0.774**

TABLE IX: *AUC* obtained by using PFAE's private head, public backbone, and both.

No. Clients	2	3	4	5	6	7	8	Avg
$\mathcal{A}_i(\mathbf{x}^{(i,j)})$	**0.872**	**0.788**	0.723	0.761	0.754	0.702	0.761	0.766
$\mathcal{B}_i(\mathbf{x}^{(i,j)})$	0.844	0.731	0.701	0.754	0.756	0.636	0.743	0.738
$\mathcal{S}_i(\mathbf{x}^{(i,j)})$	0.864	0.778	**0.740**	**0.770**	**0.771**	**0.709**	**0.783**	**0.774**

TABLE X: Average anomaly score obtained by PFAE for anomalies and benign samples.

No. Clients	2	3	4	5	6	7	8	Avg
S_b (benign)	0.039	0.013	0.013	0.003	0.003	0.004	0.004	0.011
S_a (anomaly)	0.088	0.029	0.025	0.006	0.008	0.006	0.007	0.024
$\nu = \frac{S_a}{S_b}$	2.271	2.157	1.930	2.352	2.492	1.686	1.850	2.158

We report the average anomaly scores of benign samples, S_b, and anomalies, S_a, obtained by PFAE, as shown in Table X. To provide clearer insight, we compute the ratio $\nu = \frac{S_a}{S_b}$. The results show that ν is always greater than 1, indicating that Hypothesis 1 holds in every case across these experiments. In addition, we discuss the average anomaly scores calculated using (6) on the MNIST dataset, as shown in Fig. 4. The average scores for benign samples calculated by the backbone model ($\mathcal{B}_b$) are lower than those of anomalies ($\mathcal{B}_a$) across all nine clients. This indicates that the average anomaly scores of benign samples among the clients are quite similar and significantly lower than those of anomalies, providing empirical evidence that Theorem 1 holds.

We observe that as the ratio of anomalies to benign samples on each client increases, the average *AUC* obtained by PFAE decreases, as shown in Fig. 5 (a). When the ratio reaches 1.0, the AD performance based on the reconstruction error of the AE approaches 0.5, indicating that the model cannot differentiate anomalies from benign samples. However, interestingly, when the number of clients increases, the average *AUC* obtained by PFAE can still improve for some clients with an anomaly-to-benign ratio of 1.0, as observed in Fig. 5 (b). This highlights the effectiveness of using an FL model to leverage data across all clients, thereby improving the generalization of the final AD model.

Finally, we present the data representation of PFAE's public backbone compared to the original data. Benign samples from the Arrhythmia and Cardio domains do not overlap in the original space (Fig. 6 a) but mostly overlap in the backbone space of PFAE (Fig. 6 b). Thus, benign samples from both domains may share a similar distribution in the backbone space, while their anomalies remain distinct from the benign samples. Next, we present the distributions of benign samples from the public backbone of PFAE compared to the original data. As shown

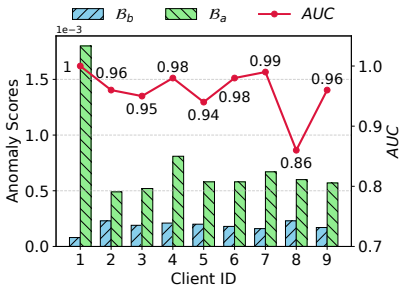
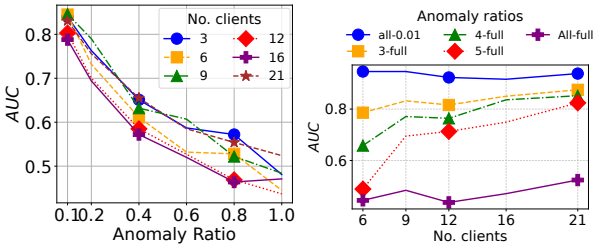


Fig. 4: Average anomaly scores of the public backbone model, calculated using (6) for anomaly samples ($\mathcal{B}_a$) and benign samples ($\mathcal{B}_b$), for each client on the MNIST dataset.



(a) *AUC* by anomaly ratios **(b)** *AUC* by the number of clients

Fig. 5: *AUC* of PFAE with varying anomaly ratios and client numbers on the MNIST dataset.

TABLE XI: AUC obtained by PFAE using only the reconstruction error from the input $\mathbf{x}$ (PFAE-X) or from $\mathbf{y}$ (PFAE-Y) on eight non-i.i.d. datasets.

No. Clients	2	3	4	5	6	7	8	Avg
PFAE-X	0.774	0.734	0.661	0.727	0.750	0.650	0.746	0.720
PFAE-Y	0.641	0.631	0.523	0.580	0.609	0.529	0.560	0.582
PFAE	**0.864**	**0.778**	**0.740**	**0.770**	**0.771**	**0.709**	**0.783**	**0.774**

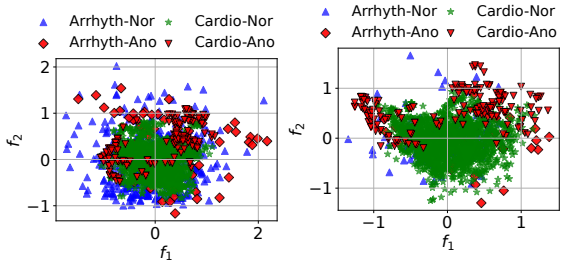
in Fig. 7b, the means of the seven distributions are closer together, with most of them being Gaussian distributions. In contrast, the distributions of the seven original datasets differ significantly in Fig. 7a. This demonstrates PFAE's effectiveness in aligning data distributions across different domains.

C. PFAE's Ablation Evaluation

First, we evaluate the detection accuracy of the proposed PFAE using local and global anomaly scores. To achieve this, we train the proposed PFAE under three configurations: using only the loss for the private head model in (5) (denoted as PFAE-X), only the loss for the public backbone model in (6) (denoted as PFAE-Y), and the full PFAE model using (4), as shown in Table XI. The average *AUC* obtained by PFAE is significantly higher than those of PFAE-X and PFAE-Y (0.774 vs. 0.720 vs. 0.582). Second, we train the PFAE using Adam [26] and GD optimizers for all clients, as shown in Fig. 8. All eight clients achieve a loss value below 0.05 after 25 training epochs when using Adam in Fig. 8 a., whereas Client 1 struggles to converge with GD in Fig. 8 b. This demonstrates that the proposed PFAE converges effectively with Adam.

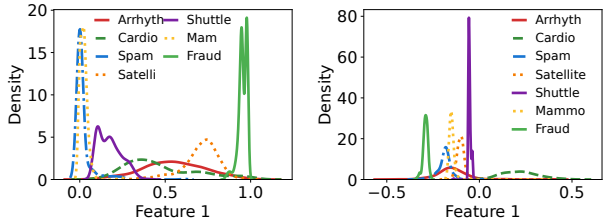
VI. Conclusion

This work identified and addressed the challenges of domain heterogeneity for anomaly detection in IoT systems. It presented a novel FL framework (PFAE) that is designed to learn



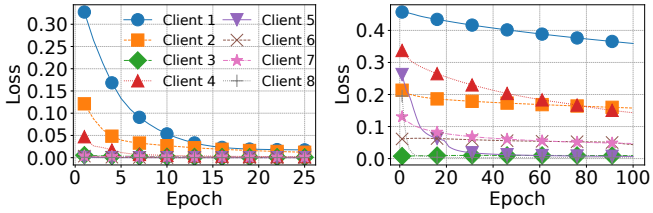
(a) Original data. **(b)** PFAE: backbone representation.

Fig. 6: Simulation of data on Arrhythmia (Arrhyth) [3] and Cardio [19].



(a) Original distributions. **(b)** PFAE: backbone distributions.

Fig. 7: Distributions of original and backbone representation data.



(a) Adam (b) GD

Fig. 8: Training epochs of the proposed PFAE when clients use Adam [26] and gradient descent (GD) [41].

the distribution of benign samples across different domains. PFAE is comprised of a private head for processing domain-specific inputs and a public backbone for shared representation learning. This design enabled each client to detect anomalies effectively within its own data domain while leveraging global knowledge through global aggregation to leverage global knowledge from different domains. We theoretically proved that the difference in the expected anomaly score of PFAE, calculated using the public backbone for benign samples from any pair of clients, was bounded. This meant that the distribution of benign samples was shared across domains, so benign samples from a client might not be misidentified as anomalies by other clients during inference. In addition, PFAE had lower training complexity than existing AE-based methods. Experiments on eight non-i.i.d. datasets, with each client trained on a single dataset, showed that PFAE improved the generalization of anomaly scores for benign samples across domains.

ACKNOWLEDGEMENT

"This research was supported in part by the Australia-Vietnam Strategic Technologies Centre, NSF (awards # 2434021, 2413009, 2229386, 2425535, and 2432139), and by the WISPER Center. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of NSF."

REFERENCES

[1] X. Zhang, R. Zhao, Z. Jiang, Z. Sun, Y. Ding, E. C. Ngai, and S.-H. Yang, "Aoc-ids: Autonomous online framework with contrastive learning for intrusion detection," in *Proc. INFOCOM*, Las Vegas, Nevada, 2024, pp. 581–590.

[2] R. Li, Q. Li, Y. Zhang, D. Zhao, X. Xiao, and Y. Jiang, "Genos: general in-network unsupervised intrusion detection by rule extraction," in *Proc. INFOCOM*, Las Vegas, Nevada, 2024, pp. 561–570.

[3] D. Dua and C. Graff, "Uci machine learning repository: Arrhythmia data set," https://archive.ics.uci.edu/ml/datasets/arrhythmia, accessed: 2025-06-18.

[4] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *Proc. Symposi. Computati. Intellige. Securi. Defens. Applicat.*, Ottawa, ON, Canada, 2009, accessed: 2025-06-18. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

[5] A. D. Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection dataset," https://www.kaggle.com/mlg-ulb/creditcardfraud, 2015, accessed: 2025-06-18.

[6] W. Zuo, D. Zhang, and K. Wang, "Bidirectional pca with assembled matrix distance metric for image recognition," *IEEE Trans. Syst. Man. Cybern. B Cybern.*, vol. 36, no. 4, pp. 863–872, Aug. 2006.

[7] T.-A. Nguyen, J. He, L. T. Le, W. Bao, and N. H. Tran, "Federated pca on grassmann manifold for anomaly detection in iot networks," in *Proc. INFOCOM*, Hoboken, NJ, USA, 2023, pp. 1–10.

[8] W. Zixuan, M. Cheng, X. Yuhua, L. Zeyi, S. Zhixin, and W. Pan, "Trusted encrypted traffic intrusion detection method based on federated learning and autoencoder," *China Communicati.*, vol. 21, no. 8, pp. 211–235, Aug. 2024.

[9] D. Novoa-Paradela, O. Fontenla-Romero, and B. Guijarro-Berdiñas, "Fast deep autoencoder for federated learning," *Pattern Recognition*, vol. 143, p. 109805, Nov. 2023.

[10] B. Dong, D. Chen, Y. Wu, S. Tang, and Y. Zhuang, "Fadngs: Federated learning for anomaly detection," *IEEE Tran. Neura. Netw. Learn. System.*, vol. 36, no. 2, pp. 2578–2592, Feb. 2025.

[11] C. T Dinh, N. Tran, and J. Nguyen, "Personalized federated learning with moreau envelopes," *NeurIPS*, vol. 33, pp. 21 394–21 405, Dec. 2020.

[12] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning: A meta-learning approach," *arXiv preprint arXiv:2002.07948*, 2020. [Online]. Available: https://arxiv.org/abs/2002.07948

[13] W. Jeong and S. J. Hwang, "Factorized-fl: Personalized federated learning with parameter factorization & similarity matching," *NeurIPS*, vol. 35, pp. 35 684–35 695, Dec. 2022.

[14] M. Setayesh, X. Li, and V. W. Wong, "Perfedmask: Personalized federated learning with optimized masking vectors," in *Proc. ICLR*, Kigali, Rwanda, 2023, pp. 1–27.

[15] D. Li and J. Wang, "Fedmd: Heterogenous federated learning via model distillation," *arXiv preprint arXiv:1910.03581*, 2019. [Online]. Available: https://arxiv.org/pdf/1910.03581

[16] A. Anaissi, B. Suleiman, and W. Alyassine, "A personalized federated learning algorithm for one-class support vector machine: An application in anomaly detection," in *Internati. Conf. Comput. Sci.*, London, United Kingdom, 2022, pp. 373–379.

[17] Z. Zhang, W. Zhang, Z. Bao, Y. Miao, Y. Liu, Y. Zhao, R. Zhang, and W. Zhu, "A personalized and differentially private federated learning for anomaly detection of industrial equipment," *IEEE J. Radio Freq. Identificat.*, pp. 468–475, Apr. 2024.

[18] L. Barbieri, M. Brambilla, and M. Roveri, "A layer-wise personalization approach for transformer-based federated anomaly detection," in *Proc. FLTA*, Valencia, Spain, 2024, pp. 32–38.

[19] D. Campos and J. Bernardes, "Cardiotocography dataset," https://archive.ics.uci.edu/dataset/193/cardiotocography, 2000, accessed: 2025-06-18.

[20] D. Dua and C. Graff, "Uci machine learning repository: Spambase data set," https://archive.ics.uci.edu/ml/datasets/spambase, accessed: 2025-06-18.

[21] ——, "Uci machine learning repository: Statlog (landsat satellite) data set," https://archive.ics.uci.edu/ml/datasets/statlog+(landsat+satellite), accessed: 2025-06-18.

[22] ——, "Uci machine learning repository: Statlog (shuttle) data set," https://archive.ics.uci.edu/ml/datasets/statlog+(shuttle), accessed: 2025-06-18.

[23] M. Zwitter and M. Soklic, "Mammographic mass dataset," https://archive.ics.uci.edu/dataset/161/mammographic+mass, 1988, accessed: 2025-06-18.

[24] H. Gao, W. Jiang, Q. Ran, and Y. Wang, "Vision-language interaction via contrastive learning for surface anomaly detection in consumer electronics manufacturing," *IEEE Tran. Consume. Electroni.*, vol. 3, no. 70, pp. 6119–6130, Aug. 2024.

[25] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, Fort Lauderdale, FL, USA, 2017, pp. 1273–1282.

[26] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diego, CA, USA, 2015.

[27] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *Proc. ICML*. New York City, NY, USA: PMLR, 2016, pp. 1225–1234.

[28] M. Mitzenmacher and E. Upfal, *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

[29] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *The Collected Works of Wassily Hoeffding*, pp. 409–426, 1994.

[30] D. Bertsekas and J. N. Tsitsiklis, *Introduction to Probability*, 1st ed. Belmont, MA: Athena Scientific, 2008, vol. 1.

[31] I. T. Jolliffe, *Principal component analysis for special types of data*. Springer, 2002.

[32] P. V. Dinh, D. N. Nguyen, D. T. Hoang, Q. U. Nguyen, and E. Dutkiewicz, "Multiple-input variational auto-encoder for anomaly detection in heterogeneous data," *arXiv preprint arXiv:2501.08149*, 2025. [Online]. Available: https://arxiv.org/pdf/2501.08149

[33] S. Verrill and R. A. Johnson, "Confidence bounds and hypothesis tests for normal distribution coefficients of variation," *Comm. Statist. Theory Methods*, vol. 36, no. 12, pp. 2187–2206, May 2007.

[34] P. V. Dinh, D. T. Hoang, N. Q. Uy, D. N. Nguyen, S. P. Bao, and E. Dutkiewicz, "Multiple-input auto-encoder for iot intrusion detection systems with heterogeneous data," in *Proc. ICC*, Denver, CO, USA, 2024, pp. 2707–2712.

[35] Y. Zhou, X. Song, Y. Zhang, F. Liu, C. Zhu, and L. Liu, "Feature encoding with autoencoders for weakly supervised anomaly detection," *IEEE Tran. Neur. Netw. Learni. Syste.*, vol. 33, no. 6, pp. 2454–2465, Jun. 2021.

[36] A. Adesh, G. Shobha, J. Shetty, and L. Xu, "Local outlier factor for anomaly detection in hpcc systems," *J. Paralle. Distribute. Comput.*, vol. 192, p. 104923, Oct. 2024.

[37] P. V. Dinh, D. N. Nguyen, H. D. Thai, Q. U. Nguyen, S. P. Bao, and E. Dutkiewicz, "A dual-decoder variational auto-encoder for anomaly detection," in *Proc. WCNC*, Milan, Italy, 2025, pp. 1–6.

[38] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-anogan: Fast unsupervised anomaly detection with generative adversarial networks," vol. 54, May 2019, pp. 30–44.

[39] S. Akçay, A. Atapour-Abarghouei, and T. P. Breckon, "Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection," in *Proc. IJCNN*, Budapest, Hungary, 2019, pp. 1–8.

[40] P. V. Dinh, D. N. Nguyen, H. D. Thai, Q. U. Nguyen, T. H. Le, S. P. Bao, and E. Dutkiewicz, "A deep learning approach for outlier detection in heterogeneous/non-iid data," in *Proc. GLOBECOM*, Cape Town, South Africa, 2024, pp. 1215–1220.

[41] W. Liu, L. Chen, Y. Chen, and W. Zhang, "Accelerating federated learning via momentum gradient descent," *IEEE Transactions. Paralle. Distribut. Syste.*, vol. 31, no. 8, pp. 1754–1766, Feb. 2020.