Anomaly Detection and Attack Mitigation in Federated Learning

Haoqi Huang

A Thesis Submitted to the Faculty of Graduate Studies
In Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science

Graduate Program in Electrical Engineering and Computer Science

York University
Toronto, Ontario

November 2025

Abstract

Federated Learning (FL) enables collaborative model training across multiple participants without sharing raw data, achieving a balance between privacy protection and data utility. However, the decentralized nature of FL also introduces new security threats, among which model poisoning attacks are the most critical. Malicious clients can upload manipulated model updates to disrupt the global aggregation process, leading to performance degradation or even system failure.

This thesis systematically investigates anomaly detection (AD) and attack mitigation in FL, aiming to enhance system robustness and security from both detection and defense perspectives. First, a comprehensive review and categorization of AD methods are presented, focusing on reconstruction-based and prediction-based deep learning frameworks and their applications to different data types. Second, existing defense mechanisms are analyzed in terms of their effectiveness and limitations under various attack scenarios, providing the theoretical foundation for the proposed framework.

Based on these analyses, this thesis proposes an unsupervised defense framework named Dual-VAE with Truncated Gaussian (DVTG). The framework follows a three-stage structure to model and filter client updates. In Stage 1, a variational autoencoder (VAE) is trained to estimate reconstruction errors and identify a set of potentially clean updates. In Stage 2, a second VAE with a truncated Gaussian prior is trained on this refined subset to obtain a more stable latent representation. In Stage 3, the trained model evaluates incoming client updates and filters those with high reconstruction errors before aggregation. The method enables

effective anomaly detection without requiring labeled or clean data and remains stable under both adversarial and stochastic disturbances.

Experiments conducted on the MNIST dataset under non-independent and identically distributed (non-IID) conditions show that DVTG outperforms the baseline model across different attack scenarios. The framework effectively detects malicious clients while maintaining stable convergence and comparable accuracy to the non-attack scenario.

Finally, this thesis discusses several future directions. These include extending the defense framework to hierarchical FL architectures and developing more interpretable and efficient AD models. The goal is to build a reliable and practical FL framework with stronger defense capability and better adaptability to real-world environments.

Acknowledgements

I would like to express my heartfelt gratitude to my supervisor, Prof. Ping Wang, for her exceptional guidance, patience, and encouragement throughout my research. Her patient guidance and continuous encouragement made it possible for me to overcome difficulties and maintain confidence in my work. I am truly grateful for her kindness and support, which have had a lasting impact on my academic progress.

I would also like to thank Prof. Liang Xue and Prof. Manos Papagelis for kindly agreeing to be my defense examiners. I greatly appreciate the time they dedicated to reviewing my thesis and participating in my oral defense, as well as the valuable feedback and constructive suggestions they provided during the examination.

Finally, I am deeply thankful to my family for their unwavering love, support, and understanding during my studies.

Contents

List of Tables

List of Figures

List of Abbreviations

AE	Autoencoder
AI	Artificial Intelligence
AAE	Adversarial Autoencoder
CNN	Convolutional Neural Network
DP	Differential Privacy
FedAvg	Federated Averaging
FL	Federated Learning
GAN	Generative Adversarial Network
GNN	Graph Neural Network
LSTM	Long Short-Term Memory
MLP	Multilayer Perceptron
non-IID	non-independent and identically distributed
RNN	Recurrent Neural Network
GRU	Gated Recurrent Unit
VAE	Variational Autoencoder

Chapter 1

1 Introduction

1.1 Overview of Federated Learning

With the rapid advancement of artificial intelligence (AI) and big data analytics, data-driven models have achieved remarkable progress across diverse domains, including computer vision, natural language processing, and autonomous systems. These breakthroughs have been largely powered by deep learning architectures trained on massive, high-quality centralized datasets. However, in many real-world applications such as healthcare, finance, and mobile services, data is often distributed across multiple devices or organizations and cannot be directly shared due to privacy regulations, security concerns, and communication limitations [1]. As data privacy bekcomes increasingly important, organizations and individuals are less willing or legally permitted to share raw data with external entities. At the same time, modern data ecosystems are highly fragmented, with information distributed across different devices, institutions, and geographic locations. These two factors make it difficult to apply traditional centralized learning, which depends on collecting and storing all data in a single location for model training. In such a paradigm, all participants must transfer their local data to a central server, where the model is trained using the combined dataset [2].

To overcome these limitations, Federated Learning (FL) has emerged as an ideal paradigm for collaborative model training without data centralization [3]. As illustrated in Figure 1,

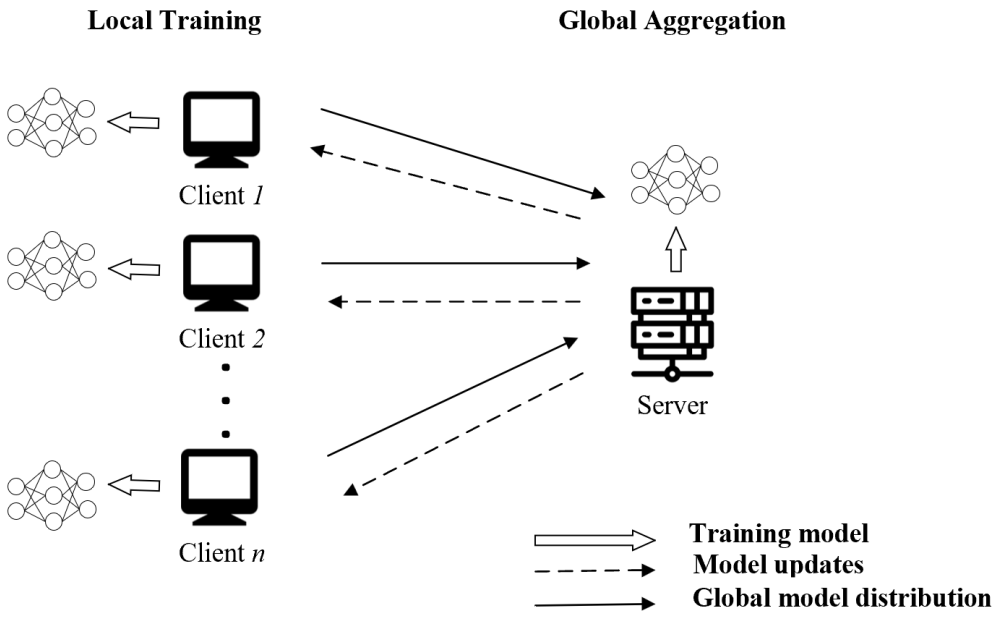

Figure 1: Illustration of the Federated Learning Process: Local Training and Global Aggregation.

in a typical FL framework, a central server coordinates a group of distributed clients, each of which trains a local model using its private data and transmits only model parameters or gradients to the server. The server then aggregates these updates to construct a global model, which is subsequently redistributed to all clients for the next training iteration. Through this iterative process, FL enables knowledge sharing among distributed clients while ensuring that raw data never leaves local devices. This architecture, consisting of a single central server and multiple clients, effectively preserves data privacy, reduces communication costs, and leverages distributed computational resources, making FL a cornerstone framework for privacy-preserving and scalable machine learning in modern data-sensitive environments.

1.2 Security Challenges in Federated Learning

Despite its advantages, FL remains highly vulnerable to adversarial threats, particularly model poisoning attacks [4]. In such attacks, malicious clients intentionally manipulate their local model updates by flipping gradient signs or amplifying noise to disrupt the training

process. These poisoned updates exploit the decentralized and trust-based nature of FL. When aggregated by the server, they can degrade model accuracy, hinder convergence, or even steer the global model toward the attacker's objective [5]. Since the aggregation process assumes that clients behave honestly, even a small number of poisoned updates can have a disproportionately harmful impact on the overall training outcome.

The challenge is further exacerbated by the non-independent and identically distributed (non-IID) nature of client data, which is common in real-world FL scenarios. Clients often possess data drawn from different underlying distributions due to variations in user behavior, device type, or geographic context. This statistical heterogeneity leads to substantial variation in local updates even among benign clients, making it difficult to distinguish between honest but atypical updates and genuinely malicious ones. As a result, poisoned updates can appear statistically plausible and are often accepted into the global aggregation, gradually corrupting the model's parameters over multiple communication rounds. These challenges highlight the urgent need for effective defense strategies that can ensure reliable and secure model aggregation in federated environments.

1.3 Anomaly Detection in Federated Learning

To mitigate the impact of model poisoning, researchers have explored a variety of defense strategies. These methods can be broadly classified into passive and proactive approaches. Passive defenses, such as robust aggregation techniques, attempt to reduce the influence of abnormal updates during aggregation without explicitly identifying which clients are malicious [6]. By statistically down-weighting or discarding extreme values, these methods can enhance the tolerance of the global model to outliers and improve overall stability. However, such approaches treat all updates equally and may inadvertently remove useful information from benign but diverse clients, ultimately reducing model accuracy.

In contrast, proactive defenses aim to explicitly identify and isolate malicious clients before aggregation. Among these approaches, anomaly detection (AD) has emerged as one of the most effective strategies for maintaining model integrity in FL [7]. The core idea of AD-based defense is to analyze the behavioral patterns of client updates and detect deviations that indicate potential adversarial activity. In general, AD learns what constitutes normal client behavior based on the statistical characteristics or latent representations of model updates. Clients that exhibit unusual update magnitudes, directions, or feature distributions are considered suspicious and can be excluded from the aggregation process. Through this mechanism, AD enables the server to identify poisoned updates before they affect the global model, thereby preserving both model integrity and convergence stability.

However, traditional AD techniques often rely on strong assumptions that are incompatible with FL environments. Many existing AD frameworks require access to clean or labeled data to train a reliable detection model, assuming a known distribution of benign and malicious behaviors. In FL, these assumptions rarely hold because privacy constraints prevent direct access to raw data, client updates are high-dimensional and non-IID, and the server lacks a verified reference dataset to define what constitutes normal [8]. Furthermore, local data itself may contain bias, noise, or small irregularities unrelated to attacks, causing supervised detectors to misclassify honest clients as malicious. The dynamic nature of FL, where client participation and data distribution may change over time, further complicates the task of maintaining an accurate detection boundary. To address these challenges, there is a growing need to develop unsupervised AD mechanisms that can operate effectively without relying on labeled or clean datasets. Such methods should be capable of autonomously learning the intrinsic structure of normal client updates, adapting to non-IID data distributions, and maintaining robustness under privacy and communication constraints. Developing these adaptive, data-driven AD frameworks is essential for building secure and reliable FL systems

capable of resisting sophisticated poisoning attacks.

1.4 Contributions

This thesis makes contributions from both analytical and methodological perspectives to the study of AD and attack mitigation in FL. The main contributions are summarized as follows:

- **Systematic analysis of AD and defense methods in FL.** This work provides a systematic review and summary of existing approaches for AD and defense in FL. It outlines the current research landscape and key technical routes, highlighting the major challenges faced under non-IID data distributions and limited labeled data. This analysis establishes a theoretical foundation for the design of subsequent unsupervised defense framework.

- **Proposal of an unsupervised defense framework.** A novel unsupervised defense framework, namely DVTG, is proposed to detect and filter malicious client updates without requiring labeled data or prior knowledge of attack types. The framework consists of two VAEs: The first VAE performs preliminary filtering based on reconstruction errors, while the second VAE incorporates a truncated Gaussian prior to refine the latent representation, thereby enhancing the separability between benign and poisoned updates and improving robustness and stability.

- **Experimental validation and performance evaluation.** Extensive experiments on the MNIST dataset under various attack scenarios, including sign-flipping and additive Gaussian noise, demonstrate that the proposed DVTG framework achieves higher AD accuracy and better global model stability than the conventional baselines, confirming its effectiveness and generalization under non-IID conditions.

1.5 Organization of Thesis

This thesis is organized as follows. Chapter 2 introduces the fundamental background of AD and attack mitigation in FL, providing the theoretical foundation for this study. Chapter 3 defines the system model and formalizes the problem setting, describing the adversarial

behavior and defense objectives. In Chapter 4, we propose the DVTG framework, which employs a three-stage unsupervised mechanism for AD and defense. Chapter 5 evaluates the proposed framework on the MNIST dataset under various attack scenarios, presenting detailed experimental results and analysis. Finally, Chapter 6 concludes the thesis and outlines several promising directions for future research.

1.6 Related Publication

[1] Huang, H., Wang, P., Pei, J., Wang, J., Alexanian, S., and Niyato, D. Deep Learning Advancements in Anomaly Detection: A Comprehensive Survey. *IEEE Internet of Things Journal*, vol. 12, no. 21, pp. 44318–44342, November 1, 2025. DOI: 10.1109/JIOT.2025.3585884.

[2] Huang, H., Wang, P., and Kumar, O. Dual-VAE with Truncated Gaussian: An Unsupervised Defense Against Model Poisoning in Federated Learning. *Paper presented at the IEEE Vehicular Technology Conference (VTC-Fall 2025)*, Chengdu, China. Accepted (forthcoming publication).

Chapter 2

2 Preliminaries and Literature Review

2.1 Federated Learning and Related Techniques

FL has rapidly become a key paradigm for privacy-preserving and distributed machine learning since its introduction by McMahan *et al.* [9]. Each client performs several rounds of local updates using its own dataset and then transmits model parameters or gradients to a central server. The server aggregates these updates, typically through weighted averaging, and redistributes the global model for the next training iteration. This process allows knowledge to be shared among participants while maintaining data locality, making FL particularly suitable for data-sensitive domains such as healthcare [10], finance [11], and mobile applications [12].

Among various FL approaches, the Federated Averaging (FedAvg) algorithm serves as the standard framework for model aggregation. It iteratively averages locally optimized client models to construct a global model, effectively coordinating distributed learning under privacy constraints. Despite its simplicity and scalability, FedAvg suffers from significant performance degradation under non-IID data distributions, unbalanced client participation, and system heterogeneity [13]. FedProx [14] introduces a proximal regularization term and a mechanism to tolerate partial work across clients, effectively enhancing stability and convergence in heterogeneous federated environments. FedNova [15] and FedDyn [16] further refine the aggregation process to address client drift and gradient inconsistency caused by heterogeneous data and unbalanced local training. FedNova introduces normalization of local updates to eliminate objective inconsistency between clients with varying numbers of local steps, thereby ensuring that each client's contribution to the global model remains

proportionally fair. In contrast, FedDyn dynamically adjusts local regularization terms by incorporating a correction mechanism into the local objective, which compensates for the bias introduced by data heterogeneity and improves the stability of convergence across communication rounds.

Beyond algorithmic convergence, personalization and client-specific adaptation have emerged as another major research direction. Since a single global model often fails to generalize well across clients with highly diverse data distributions, personalized FL approaches such as pFedMe [17], FedBN [18], and FedPer [19] propose strategies that allow each client to maintain individualized components or locally fine-tuned layers. These techniques improve local generalization while maintaining partial model sharing, striking a balance between collaboration and autonomy. Besides, communication efficiency is another critical concern in large-scale federated networks. Frequent parameter exchanges between clients and the server introduce considerable communication overhead, particularly in bandwidth-constrained environments. To mitigate this issue, Karimireddy *et al.* [20] propose Stochastic Controlled Averaging algorithm, an algorithm that mitigates client drift caused by data heterogeneity by introducing control variates for variance reduction, thereby stabilizing local updates and significantly reducing the number of communication rounds required for convergence. Similarly, Hsu *et al.* [21] develope FedAvgM, which incorporates server-side momentum into the aggregation process to smooth global updates under highly non-IID conditions. This design accelerates convergence and consequently reduces the communication overhead needed to achieve comparable accuracy. In addition, Wu *et al.* [22] propose Federated Adaptive Weighting, an adaptive aggregation algorithm that dynamically adjusts client weights based on their contribution to the global model, measured by the angle between local and global gradients. By reinforcing positively aligned updates and suppressing negatively correlated ones, this method accelerates convergence under non-IID data distributions and substantially reduces the number of communication rounds required for model convergence.

Parallel to these algorithmic advancements, privacy and security mechanisms have become integral components of FL. Techniques such as Differential Privacy (DP) [23] ensure that individual client data cannot be inferred from transmitted model parameters, while Secure Aggregation [24] allows the server to aggregate encrypted updates without accessing

individual contributions. These methods significantly strengthen confidentiality but do not inherently prevent adversarial manipulation. Indeed, the decentralized and trust-agnostic nature of FL makes it vulnerable to various attacks, particularly model poisoning, where malicious clients intentionally upload corrupted updates to degrade or bias the global model. Such vulnerabilities have driven the development of robust and trustworthy FL frameworks that incorporate AD, reputation-based reweighting, and robust aggregation to defend against adversarial behaviors.

2.2 Anomaly Detection Methods

As discussed in the previous section, the presence of malicious or unreliable clients in FL can lead to severe model degradation, which necessitates effective mechanisms to detect and mitigate such adversarial behaviors. Among various defensive strategies, AD plays a crucial role as the first line of defense, enabling the identification of abnormal or inconsistent updates before aggregation. In deep learning–based approaches, anomalies are detected by learning the representation of normal data and measuring deviations between the learned distribution and new observations. Existing studies generally classify these methods into two major paradigms: reconstruction-based and prediction-based. Reconstruction-based methods focus on reproducing normal data instances and use reconstruction errors to indicate abnormality, while prediction-based methods learn temporal or contextual dependencies within the data to forecast future states and identify inconsistencies between predicted and observed outcomes. These two categories capture different aspects of anomaly characterization, with reconstruction-based methods emphasizing spatial consistency and prediction-based methods focusing on temporal continuity. The following subsections review representative models under each category.

2.2.1 Reconstruction-based Anomaly Detection

Reconstruction-based approaches operate by training a model to learn the underlying distribution of normal data [25]. Once trained, the model attempts to reconstruct incoming data. The reconstruction error, which is the difference between the original data and its reconstruction, is then used as an indicator of anomaly. A high reconstruction error suggests that the

data is anomalous, as it deviates from the learned normal patterns. Deep learning-based reconstructive models have become prominent due to their ability to capture complex patterns in high-dimensional data. In recent years, most reconstruction-based AD models have been developed using techniques such as Generative Adversarial Networks (GANs), Autoencoder (AE), and diffusion models.

(1) Generative Adversarial Networks methods. GANs have been extensively applied in AD owing to their strong capability for data generation and representation learning. Two main structures are commonly employed when using GANs for AD. The first structure focuses on the generator, which is trained on normal data to reproduce normal patterns. During inference, anomalies are identified by reconstruction errors, since the generator tends to fail when reconstructing abnormal samples that deviate from the learned distribution. For instance, Dong *et al.* [26] propose a semi-supervised visual AD framework using a dual-discriminator GAN architecture that emphasizes representation learning, where the generator predicts future frames for normal events and anomalies are detected based on prediction quality. Similarly, Guo *et al.* [27] introduce graph GAN, which leverages GAN-based representation learning to encode complex spatiotemporal relationships, achieving more accurate detection in dynamic graph structures. The second structure relies on the discriminator, which directly distinguishes between real (normal) and fake (anomalous) samples. A well-trained discriminator can evaluate the likelihood that an input belongs to the normal data distribution. During testing, samples that receive low likelihood scores are flagged as anomalous. Liu *et al.*[28] propose a multi-generator GAN framework in which multiple generators create potential outliers, and the discriminator learns to separate them from normal data. This design not only improves the robustness of anomaly scoring but also mitigates mode collapse by providing a more comprehensive reference distribution. Beyond detection, GANs are also widely used for data augmentation to alleviate data imbalance and improve model generalization. In many real-world scenarios, anomaly samples are scarce or lack diversity, leading to poor generalization and overfitting. GAN-based augmentation enables the generation of synthetic anomalies that mimic real-world characteristics, thus enriching the training set. Miao *et al.* [29] introduce an unsupervised AD framework that uses data augmentation through contrastive learning and GANs to mitigate overfitting. In [30],

Anomaly-GAN addresses data augmentation by using a mask pool, anomaly-aware loss, and local-global discriminators to generate high-quality, realistic synthetic anomalies with diverse shapes, angles, spatial locations, and quantities in a controllable manner. Similarly, Li *et al.* [31] propose augmented time regularized generative adversarial network that combines an augmented filter layer and a novel temporal distance metric to generate high-quality and diverse artificial data, addressing the limitations of existing GAN approaches in handling limited training data and temporal order.

(2) Autoencoder-based methods. Traditional AE models have been widely used in AD due to their ability to capture compact representations of normal data. However, their limitations in handling complex and noisy data have motivated the development of enhanced variants to improve robustness and detection accuracy. For instance, Fan *et al.* [32] introduce an AE framework incorporating $\ell_{2,1}$-norm regularization, which enhances resistance to noise and outliers during training. Wang *et al.* [33] propose an adaptive-weighted loss function to suppress the reconstruction of anomalous features, thereby improving anomaly discrimination. Liu *et al.* [34] designe a multiscale convolutional AE architecture that reconstructs backgrounds at multiple scales, effectively isolating anomalies of different sizes. Lin *et al.* [35] further combine a soft calibration mechanism with AE to mitigate the effect of contaminated training data. These improvements reflect the general trend of enhancing AE robustness and sensitivity to diverse anomaly patterns in complex datasets.

VAEs extend AEs by introducing a probabilistic latent space that models the underlying data distribution, providing stronger generalization and uncertainty estimation. Several studies have demonstrated the advantages of VAEs in AD. Huang *et al.* [36] propose an Autoencoding Transformation framework that enhances the semantic representation of normal images, enlarging the anomaly score gap between normal and abnormal samples. Yin *et al.* [37] combine convolutional neural networks (CNNs) and VAEs with a two-stage sliding window preprocessing technique to learn richer representations for AD in sequential data. Zhang *et al.* [38] develope the Graph Relational Learning Network (GReLeN), which integrates VAE and graph dependency modeling for multivariate time series AD. Zhou *et al.* [39] introduce a Variational Long Short-Term Memory (LSTM) model for high-dimensional and imbalanced datasets, where LSTM and variational components jointly balance data compres-

sion and feature retention. These works demonstrate that VAE-based approaches effectively capture latent data structure while improving robustness to heterogeneity and noise.

Adversarial Autoencoders (AAEs) combine the generative principles of VAEs with the adversarial training strategy of GANs to enforce a flexible prior distribution in the latent space. Unlike VAEs, which employ Kullback–Leibler divergence for latent regularization, AAEs use a discriminator to align the latent representation with a predefined prior via adversarial learning. Wu *et al.* [40] propose FGPAA, an end-to-end AAE model with dual discriminators for machine fault detection, achieving accurate equipment health monitoring through probabilistic attention mechanisms. Su *et al.* [41] introduce contamination-immune BiGAN models that combine the characteristics of VAE and BiGAN to detect anomalies from impure datasets, significantly outperforming conventional approaches under data contamination scenarios. Yu *et al.* [42] further extend this research with the Adversarial Contrastive Autoencoder, which employs adversarial and contrastive learning to enhance feature representation for multivariate time-series AD. These developments highlight the growing trend of hybrid AAE-based frameworks that integrate adversarial learning, contrastive objectives, and distribution purification to achieve more robust and generalizable AD performance.

(3) Diffusion-based methods. Diffusion models represent a recent class of generative frameworks that learn data distributions through a forward diffusion process and a reverse denoising process. Trained on normal data, these models iteratively remove noise to reconstruct clean samples, and anomalies are typically detected by residual reconstruction errors or unstable denoising patterns. Several studies have explored diffusion models for AD. Zhang *et al.* [43] utilize diffusion-based image generation in DiffAD to improve reconstruction quality using noisy condition embedding and channel interpolation. Li *et al.* [44] apply diffusion models to learn normal data distributions and incorporated auxiliary tasks to better distinguish anomalies. Zeng *et al.* [45] improve denoising diffusion probabilistic models for radio AD by combining them with an autoencoder to model normal signal characteristics. Other extensions include the Controlled Graph Neural Network based on diffusion for semi-supervised AD [46], and the DiAD framework [47], which integrates semantic guidance and spatial-aware feature fusion for multiclass anomaly localization. Pei *et al.* [48] also propose a two-stage diffusion model for smart grids, where the diffusion-based stage filters abnormal

patterns before classification.

Although diffusion models have achieved impressive reconstruction performance, they face challenges in efficiency and practical deployment. Their iterative denoising process requires many sampling steps, resulting in high computational cost and latency. Performance is also sensitive to hyperparameter settings such as noise schedules, which complicates tuning and scalability. Consequently, while diffusion models show potential in modeling complex distributions, simpler probabilistic approaches such as VAEs remain more suitable for scenarios that demand a balance between accuracy, interpretability, and computational efficiency.

2.2.2 Prediction-based Anomaly Detection

Prediction-based AD methods identify anomalies by forecasting future values or estimating expected behaviors and comparing these predictions with actual observations. Significant deviations indicate abnormal patterns, as they deviate from the model's learned expectations. These methods are particularly effective in capturing temporal, contextual, or relational dependencies within data. In general, three main categories of prediction-based models are widely adopted: recurrent neural networks (RNNs), attention-based architectures, and graph neural networks (GNNs).

(1) Recurrent Neural Networks methods. RNNs and their variants such as LSTM and Gated Recurrent Unit (GRU) are widely used in prediction-based AD for their ability to model temporal dependencies and dynamic patterns in sequential data. Current RNN-based AD research primarily focuses on enhancing recurrent architectures for anomaly-sensitive learning and integrating them with other deep learning methods to improve robustness and accuracy. Several studies have optimized LSTM-based models to better handle domain-specific challenges. For example, a pruning algorithm was applied to reduce false data points in railway traffic datasets, enabling LSTM-based detection to address data imbalance and noise more effectively [49]. Hybrid models combining LSTM with AE [50], VAE [51], and singular value decomposition [52] have also been explored for AD in controller area networks, electrocardiograms, and internet traffic monitoring [53]. Furthermore, adversarial learning has been integrated into time-series modeling, where GAN-enhanced LSTM architectures demonstrate high resilience to feature sparsity, imbalance, and noise interference [54, 55].

CNNs have also been incorporated with LSTM in serial [56], parallel [57], or hierarchical [58] configurations to capture spatiotemporal dependencies in multidimensional time series, further improving anomaly localization and detection accuracy.

GRUs, compared to LSTMs, offer a more compact architecture and lower computational complexity while maintaining strong temporal modeling capability. They have shown superior efficiency in handling shorter or less complex sequences. For instance, GRUs have been applied to uncover latent correlations in multivariate time-series data from industrial control systems [59]. Similar to LSTMs, GRUs are often combined with AE [60] or VAE [61] architectures to suppress noise effects and improve reconstruction-based prediction accuracy. Overall, RNN-based models remain a core component of prediction-based AD, with ongoing research emphasizing hybrid architectures and improved training efficiency to address issues of scalability, imbalance, and noise robustness.

(2) Attention-based methods. Attention mechanisms have increasingly been adopted in AD to enhance the capability of deep learning models in capturing long-range dependencies and global correlations within sequential data. The Transformer, as a representative attention-based model, has demonstrated strong performance in prediction-based time series AD by leveraging self-attention to model dynamic patterns. Compared with recurrent architectures, Transformer-based approaches achieve superior detection accuracy and training efficiency [62, 63]. However, attention-based models may overfit when data is insufficient, prompting research on hybrid and regularized frameworks. For instance, the method in [29] integrates contrastive learning and GAN into the Transformer architecture, employing data augmentation and geometric distribution masking to enhance data diversity and improve robustness under noisy conditions.

Beyond Transformers, attention mechanisms have been combined with convolutional and recurrent models to capture both local and global dependencies. For example, Zhu *et al.* [64] apply attention mechanisms to extract global contextual representations, enhancing overall detection effectiveness. Sun *et al.* [65] propose a sequential CNN–BiLSTM–attention model that first extracts abstract features through one-dimensional convolution, followed by bidirectional LSTM processing, and finally applies attention to focus on locally important time steps. Similarly, Le *et al.* [66] integrate convolutional, LSTM, and self-attention layers

within an autoencoder framework to effectively extract complex multivariate features. Pei *et al.* [56] further incorporate an additional SVM classifier to interpret attention weights in a CNN–LSTM–attention model for cyber-attack detection in energy systems.

Attention mechanisms have also been incorporated into GNN frameworks for modeling spatiotemporal correlations in structured or multimodal data. Methods such as [67, 68, 69, 70] combine attention-based reconstruction or prediction with graph learning to capture inter-node dependencies and emphasize critical temporal relationships. These works demonstrate that attention-based architectures can flexibly adapt to various data modalities, achieving effective anomaly localization and classification. Nonetheless, attention-based AD still faces challenges such as high data dependency and overfitting risks, suggesting a continued need for lightweight and interpretable designs.

(3) Graph Neural Networks methods. GNNs have recently gained increasing attention in AD, as many types of data, such as sensor networks, traffic systems, and social interactions, can be naturally represented as graph structures [71]. By modeling the spatial and relational dependencies between nodes, GNNs provide a powerful framework for capturing both local and global patterns that may indicate abnormal behavior. Wu *et al.* [72] demonstrate the effectiveness of GNNs in detecting anomalies within complex graph-structured environments, highlighting their adaptability to heterogeneous data sources.

In prediction-based AD, GNNs are commonly employed to forecast future node or system states and identify deviations between predicted and observed values. A representative example is the Graph Deviation Network [73], which combines structure learning with GNNs and incorporates attention weights to predict time-series values, detecting anomalies based on prediction residuals. Similar approaches, such as the Graph Transformer for AD [74] and Correlation-aware Spatial-Temporal Graph Learning [75], further enhance prediction accuracy by modeling higher-order spatial-temporal dependencies. Liu *et al.* [76] introduce a contrastive learning-based GNN framework that learns discriminative node representations from high-dimensional attributes and local structures, outperforming several state-of-the-art prediction-based methods across multiple benchmark datasets. Overall, GNN-based approaches effectively capture complex spatial-temporal dependencies that traditional sequence models struggle to represent. However, their performance heavily depends on graph construc-

tion quality and computational scalability, which remain open challenges for real-world AD applications.

2.3 Defense Mechanisms Against Model Poisoning in FL

The AD methods discussed in the previous section provide essential tools for identifying abnormal patterns and irregular behaviors in data or model outputs. These techniques establish the theoretical foundation for detecting deviations that may indicate potential adversarial manipulations. However, in the context of FL, the sources and impacts of anomalies become significantly more complex, as malicious clients can deliberately inject poisoned model updates during the training process. Such attacks directly compromise the integrity of the global model and cannot be fully mitigated by conventional AD approaches originally designed for centralized or static data settings.

To enhance the resilience of FL systems against model poisoning, researchers have developed a variety of defense mechanisms that integrate statistical robustness with AD principles. These mechanisms aim to prevent, detect, or neutralize the influence of adversarial updates while maintaining model performance and preserving data privacy. Broadly, existing defenses can be classified into two major paradigms: passive defenses, which focus on tolerating and mitigating the effect of poisoned updates during aggregation, and proactive defenses, which explicitly identify and remove suspicious clients prior to aggregation. While both paradigms share the same ultimate goal of improving FL robustness, they differ substantially in their design assumptions, computational complexity, and adaptability to non-IID data distributions. The following subsections provide a detailed review of each defense category.

2.3.1 Passive Defense for FL systems

Passive defense methods aim to mitigate the influence of unreliable or adversarial client updates during the aggregation phase without altering the standard FL workflow or explicitly identifying attackers. These strategies enhance the robustness of global model aggregation by applying statistically robust rules that tolerate or suppress anomalous updates, thereby improving resilience against poisoning attempts. Common techniques include Trimmed Mean [77], which discards extreme coordinate values before averaging; Krum [78], which selects the

client update closest to the majority by minimizing pairwise distances among updates; and Geometric Median [79], which computes the median point minimizing the total distance to all clients. These methods reduce the influence of Byzantine or unreliable updates by relying on the assumption that the majority of participants behave honestly.

While robust aggregation improves tolerance to outliers and remains computationally efficient, it has fundamental limitations. Most methods cannot distinguish between benign but divergent updates and truly malicious ones, especially under non-IID or adaptive attack scenarios, where poisoned updates can mimic the statistical characteristics of honest ones. Consequently, passive defenses may inadvertently suppress valuable updates or fail to remove subtle malicious contributions. These challenges motivate the exploration of more adaptive and detection-oriented defense strategies, which are discussed in the following section.

2.3.2 Proactive Defenses for FL System

Proactive defenses, by contrast, aim to detect and isolate malicious clients before the aggregation process. These approaches typically analyze client update behaviors such as gradient similarity, parameter deviation, or latent feature embeddings to identify anomalies that may indicate adversarial manipulation. Common strategies include clustering based methods [80] and distance based techniques [81], which group or measure client updates according to similarity metrics and flag outliers as potentially poisoned.

Beyond statistical approaches, machine learning based detection frameworks have also been widely explored. For instance, Idrissi *et al.* [82] train AE exclusively on benign updates to reconstruct normal client behavior, treating updates with large reconstruction errors as anomalies. Similarly, Ma *et al.* [83] pretrain a one class neural network on clean model updates to learn a compact decision boundary that separates reliable from unreliable clients. While these unsupervised models are effective in detecting subtle deviations, their performance can deteriorate under data contamination or distributional shifts because they rely on clean training distributions. In addition, supervised learning frameworks have been applied to proactive defense. Chen *et al.* [84] introduce a GRU-SVM (Gated Recurrent Unit–Support Vector Machine) based intrusion detection system for FL, where a labeled dataset is used to train the model to classify client updates. This method incorporates attention mechanisms

to adaptively weigh client contributions, suppressing low importance or potentially harmful updates. Although this approach achieves high detection accuracy and strong robustness against label flipping attacks, it requires explicit labels during training, limiting its scalability to real world scenarios where labeled data are scarce or not available. To address this limitation, Lyu *et al.* [85] propose an unsupervised hybrid defense, GeminiGuard, which combines weight space deviation analysis with latent space representations to detect poisoned updates under non-IID settings. It is lightweight and has demonstrated strong robustness across various poisoning attacks. However, its integrated framework introduces additional computational complexity, and the latent representations are not explicitly regularized, which may reduce stability under severe heterogeneity.

Overall, proactive defenses offer higher detection precision than passive counterparts by directly identifying adversarial participants. However, they usually involve additional computational and communication costs, and their effectiveness may depend on the availability of clean data or labeled information. These limitations continue to motivate research into more efficient, unsupervised, and adaptive AD frameworks for FL.

2.4 Limitations in Existing Methods

Existing anomaly detection methods have achieved strong results in general anomaly detection tasks. However, many of their underlying assumptions do not hold in federated learning environments. A large portion of reconstruction-based and prediction-based approaches rely on clean reference data, trusted clients, stable data distributions, or centralized access to training samples. However, in federated learning, the server cannot access raw data, client behavior cannot be guaranteed to be benign, no labeled anomalies exist, and client updates are highly non-IID. These violations make traditional anomaly detection methods unstable and unreliable when applied directly to federated environments, since benign and malicious updates may overlap significantly under heterogeneous data distributions.

Furthermore, federated learning typically operates on resource-constrained client devices with limited computation capability, high communication costs, and only a small number of local training rounds. As a result, anomaly detection models that are highly complex, computationally expensive, or sensitive to training instability cannot be effectively deployed

in such environments. In contrast, VAE models have relatively simple architectures, stable training behavior, and modest computational requirements. Their probabilistic latent representations allow them to capture the distributional characteristics of benign updates without relying on labeled data or centralized access to training samples, which makes them naturally suitable for the unlabeled and decentralized nature of federated learning. Based on these considerations, this thesis adopts the VAE as the core modeling component in the proposed three-stage anomaly detection framework in order to achieve efficient, stable, and scalable identification of malicious updates.

Chapter 3

3 Problem Formulation

3.1 System Model and Problem Definition

We consider a two-layer FL system consisting of a central server and a set of N distributed clients, denoted as $\mathcal{C} = \{1, 2, \ldots, N\}$. Each client i holds a private dataset $\mathcal{D}_i$ that follows its local distribution $P_i(x, y)$. At each communication round t, the central server maintains a global model parameter w_t and broadcasts it to all participating clients. Each client then performs local training on its dataset and returns the updated model w_i^{t+1} to the server. The server aggregates all received updates to obtain the new global model using the standard *FedAvg* rule [9]:

$$w_{t+1} = \sum_{i=1}^{N} \frac{n_i}{n_{\text{total}}} w_i^{t+1}, \tag{1}$$

where $n_i = |\mathcal{D}_i|$ and $n_{\text{total}} = \sum_{i=1}^{N} n_i$.

The overall optimization objective of FL is defined as:

$$\min_{w} \ \mathcal{L}(w) = \sum_{i=1}^{N} \frac{n_i}{n_{\text{total}}} \, \mathbb{E}_{(x,y)\sim\mathcal{D}_i}[\ell(f_w(x), y)], \tag{2}$$

where $\ell(\cdot)$ denotes the sample-wise loss function and $f_w(\cdot)$ represents the global model.

However, in practical scenarios, some clients may behave maliciously—either unintention-

ally, due to corrupted local data, or intentionally, by launching model-poisoning attacks. Let $\mathcal{A} \subset \mathcal{C}$ denote the set of adversarial clients and $\mathcal{B} = \mathcal{C} \setminus \mathcal{A}$ the set of benign clients. When adversaries are present, the global aggregation can be written as:

$$w_{t+1} = \sum_{i \in \mathcal{B}} \frac{n_i}{n_{\text{total}}} w_i^{t+1} + \sum_{j \in \mathcal{A}} \frac{n_j}{n_{\text{total}}} \tilde{w}_j^{t+1}, \tag{3}$$

where $\tilde{w}_j^{t+1}$ represents a poisoned or manipulated update. Such abnormal updates may distort the global model and hinder convergence.

3.2 Adversarial Behavior Modeling

Model poisoning attacks pose one of the most critical security threats in FL. In such attacks, malicious clients deliberately modify their local model updates before sending them to the central server. Since the server aggregates parameters or gradients without validating individual contributions, even a small number of compromised clients can significantly disrupt global training. By injecting biased or corrupted updates, adversaries can reduce model accuracy, slow convergence, or insert hidden backdoors [4]. Model poisoning attacks are generally categorized into two types: untargeted and targeted. Untargeted attacks aim to reduce the overall performance or stability of the global model, whereas targeted attacks seek to implant specific malicious behaviors such as class-specific misclassification or backdoor triggers. This study focuses on two representative untargeted attack types—sign-flipping and Gaussian noise injection, which are widely used as benchmark threats in FL defense research.

In a sign-flipping attack, adversarial clients invert the direction of their gradient updates. This pushes the global optimization in the opposite direction, causing divergence and unstable convergence. In a Gaussian noise attack, adversarial clients perturb model parameters by adding random noise sampled from a normal distribution. These perturbations can represent either unintentional stochastic errors or intentionally crafted noise, making them difficult to detect. Both attack types are simple to implement yet highly disruptive, making them

suitable for evaluating the robustness of defense mechanisms.

The decentralized and privacy-preserving nature of FL further increases the difficulty of defending against such attacks. Because client data are non-IID, benign updates may naturally diverge across participants, making it challenging to distinguish between statistical variability and malicious manipulation. As a result, poisoned updates may appear legitimate and be incorporated into the global aggregation, gradually degrading the model over time. These characteristics motivate the design of defense mechanisms capable of identifying and mitigating malicious updates without accessing raw client data.

We assume that a fraction $\rho = |\mathcal{A}|/N$ of all clients are adversarial. Each malicious client $j \in \mathcal{A}$ modifies its local update before transmitting it to the server, expressed as:

$$\tilde{w}_j^{t+1} = w_t + \delta_j^t, \tag{4}$$

where δ_j^t denotes the injected perturbation, and its magnitude is controlled by the attack strength parameter α.

Two representative attack formulations are considered in this work:

(1) Sign-Flipping Attack. The adversarial client reverses and amplifies its update direction:

$$\tilde{w}_j^{t+1} = w_t - \alpha(w_i^{t+1} - w_t), \tag{5}$$

where $\alpha > 0$ controls the degree of manipulation, and w_i^{t+1} denotes the genuine update that would have been generated by the client before manipulation.

(2) Additive Gaussian Noise Attack. The adversarial client injects random noise into its local update:

$$\tilde{w}_j^{t+1} = w_i^{t+1} + \mathcal{N}(0, \sigma^2 I), \tag{6}$$

where $\mathcal{N}(0, \sigma^2 I)$ denotes Gaussian noise with zero mean and variance σ^2.

The objective of adversarial clients is to maximize the global loss after aggregation,

thereby degrading the overall model performance. This objective can be expressed as:

$$\max_{\tilde{w}_j} \mathcal{L}(w_{t+1}), \quad \text{s.t.} \quad w_{t+1} = \text{Agg}(\{w_i^{t+1}, \tilde{w}_j^{t+1}\}), \tag{7}$$

where $\text{Agg}(\cdot)$ denotes the aggregation operator (e.g., weighted averaging). This formulation captures the disruptive nature of adversarial updates and emphasizes the need for robust defense mechanisms in FL.

3.3 Defense Objective

The primary objective of the defense mechanism is to maintain robust global convergence by identifying and filtering out abnormal client updates before aggregation. Since the ground-truth labels indicating benign or malicious clients are unavailable, the detection must be performed in an *unsupervised* manner.

Let a scoring function $s_\theta(\cdot)$ parameterized by θ quantify the abnormality of each client update based on its representation or reconstruction error:

$$s_\theta(w_i^{t+1}) = \|w_i^{t+1} - \hat{w}_i^{t+1}\|, \tag{8}$$

where $\hat{w}_i^{t+1}$ denotes the reconstructed update produced by a feature extractor or autoencoder model, and $\|\cdot\|$ represents the Euclidean (L2) norm.

Clients whose scores exceed a predefined threshold τ are regarded as anomalies:

$$\mathcal{A} = \{\, i \in \mathcal{C} \mid s_\theta(w_i^{t+1}) > \tau \,\}. \tag{9}$$

After removing the detected abnormal clients, the global optimization objective becomes:

$$\min_{w} \sum_{i \in \mathcal{C} \setminus \mathcal{A}} \frac{n_i}{n_{\text{total}}} \mathcal{L}_i(w), \tag{10}$$

where $\mathcal{L}_i(w)$ denotes the local loss function of client i.

Specifically, the proposed defense aims to achieve the following objectives:

(1) Detect abnormal client behaviors. Identify suspicious updates by analyzing their gradient patterns, statistical deviations, or latent representations prior to aggregation. This enables early detection of anomalous behaviors that may indicate poisoning activities.

(2) Isolate malicious clients. Accurately distinguish and exclude adversarial clients from the FL process to prevent the propagation of corrupted updates, thereby maintaining the integrity of the global model.

(3) Preserve learning efficiency and accuracy. Minimize false detections to retain benign clients whose updates contribute valuable knowledge, ensuring that the defense mechanism does not compromise model convergence speed or overall accuracy.

The joint problem of robust learning and unsupervised AD can therefore be formulated as:

$$\min_{w,\theta} \sum_{i \in \mathcal{C} \setminus \mathcal{A}} \frac{n_i}{n_{\text{total}}} \mathcal{L}_i(w), \quad \text{s.t.} \quad \mathcal{A} = \{i : s_\theta(w_i^{t+1}) > \tau\}. \tag{11}$$

This joint optimization problem simultaneously learns the model parameters w and the AD parameters θ.

3.4 Summary

This chapter formulates the FL problem under adversarial conditions and defines the defense objectives. It models the system setup, characterizes representative model poisoning attacks such as sign-flipping and Gaussian noise injection, and formalizes the detection and mitigation of malicious updates as an unsupervised AD task. The goal is to ensure robust and accurate global convergence by identifying and excluding abnormal client behaviors without relying on labeled data.

Chapter 4

4 DVTG-based Defense Framwork for Federal Learning

4.1 VAE for Anomaly Detection in Federated Learning

With the growing complexity of FL systems, identifying malicious or abnormal client updates has become a major challenge. VAEs provide an effective unsupervised mechanism to detect anomalies through latent representation reconstruction. In this section, we first review the fundamental principle of VAEs, then explain their application to AD in federated environments, and finally discuss the limitations of conventional VAEs and the motivation for adopting a truncated Gaussian prior.

4.1.1 Principle of Variational Autoencoder

VAE is a probabilistic generative model that learns the underlying distribution of input data by encoding it into a latent variable representation. It consists of two main components: an encoder, which maps input data into a latent space characterized by mean and variance parameters, and a decoder, which reconstructs the input from sampled latent variables.

As illustrated in Fig. 2, the encoder compresses an input sample x into the parameters of a Gaussian distribution, mean μ and variance σ^2, from which a latent variable z is sampled

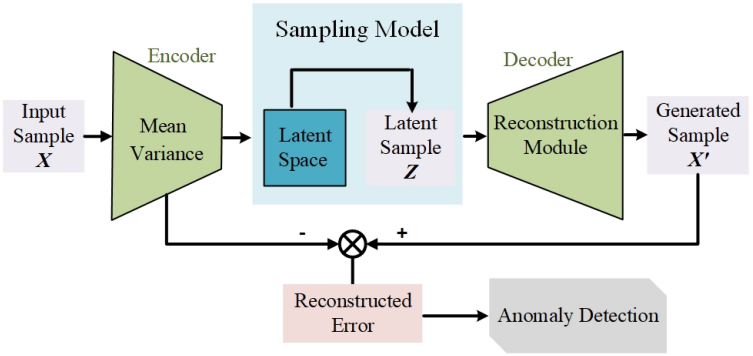

Figure 2: Conceptual illustration of a VAE for AD. The encoder maps input data into a latent space represented by mean and variance, while the decoder reconstructs the input from latent samples drawn from a Gaussian distribution. Anomalies are identified through high reconstruction errors.

as

$$z \sim \mathcal{N}(\mu, \sigma^2). \tag{12}$$

The decoder reconstructs the input as $\hat{x} = f_{\text{dec}}(z)$. The training objective of the VAE balances reconstruction accuracy and latent regularization, which can be expressed as

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q_\phi(z|x)}[\|x - \hat{x}\|^2] + \beta D_{\text{KL}}(q_\phi(z|x) \,\|\, \mathcal{N}(0, I)) \,. \tag{13}$$

Here, $D_{\text{KL}}(\cdot\|\cdot)$ denotes the Kullback–Leibler (KL) divergence, which measures how much the latent distribution $q_\phi(z|x)$ deviates from the prior distribution. The first term minimizes the reconstruction error between the input and its reconstruction, while the second term constrains the latent distribution $q_\phi(z|x)$ to approximate a standard Gaussian prior $p(z) = \mathcal{N}(0, I)$. The hyperparameter β controls the trade-off between these two terms. Through this formulation, VAEs capture the intrinsic structure of high-dimensional data on a compact latent manifold, enabling them to reconstruct or evaluate new samples according to the learned distribution.

4.1.2 VAE for Anomaly Detection in Federated Learning

The ability of VAEs to perform AD arises from their reconstruction property. When trained primarily on normal data, a VAE learns to accurately reconstruct the typical patterns of benign behavior. In contrast, samples that deviate from the learned distribution, such as adversarial or poisoned updates, yield higher reconstruction errors, which can be used as an anomaly indicator. The reconstruction error for an input x and its reconstruction $\hat{x}$ is defined as

$$\mathrm{RE}(x) = \|x - \hat{x}\|^2. \tag{14}$$

A sample is regarded as anomalous if its reconstruction error exceeds a threshold τ. This unsupervised detection principle is particularly suitable for FL, where obtaining labeled data is often infeasible due to privacy and communication constraints.

In FL, each client i periodically uploads its local model update $w_i^{(t)}$ to the central server. By encoding these updates into latent representations and measuring their reconstruction errors, the server can identify clients whose updates deviate significantly from the global distribution. This allows the detection of potentially malicious participants without requiring access to their private local data, preserving both privacy and scalability.

Although VAEs provide a convenient unsupervised mechanism for detecting abnormal client updates, directly applying a standard VAE in federated learning is often suboptimal. The core limitation is that a VAE relies on learning the distribution of benign updates, yet federated systems do not guarantee clean or homogeneous training data. When trained on mixed benign and malicious updates, the VAE tends to encode both types of patterns, which reduces the reconstruction gap between normal and adversarial inputs and weakens anomaly discrimination. In addition, the unbounded latent space imposed by the standard Gaussian prior makes the learned representation sensitive to extreme or noisy client updates, especially

under non-IID data distributions typical of FL. These factors collectively lead to unstable reconstruction behavior and degraded anomaly detection performance in practice.

4.2 The Proposed DVTG Framework

Building upon the limitations of conventional VAEs, this work introduces the proposed DVTG framework to enhance AD robustness in FL. As shown in Fig. 3, the framework operates through three sequential stages. Stage 1 and Stage 2 form the training phase, where two VAEs are trained in succession to progressively purify the latent representation. Stage 3 performs inference during each communication round of FL process, filtering malicious client updates based on reconstruction errors. This multi-stage design enables an unsupervised and adaptive defense mechanism against model poisoning attacks without requiring labeled data or modifications to client-side training.

4.2.1 Preliminary Training with Standard Gaussian Prior

The first stage aims to obtain a coarse approximation of the benign update distribution. During early communication rounds, the server collects model updates ω_i from all clients, which may contain both benign and malicious contributions. A standard VAE (denoted as VAE-1) is trained to reconstruct these updates and learn a general latent representation of client behavior. The objective function of VAE-1 follows the standard VAE formulation in Eq. (13), combining a reconstruction term and a KL divergence term that regularizes the latent distribution toward a standard Gaussian prior.

After training, reconstruction errors $\|\omega - \hat{\omega}\|^2$ are computed for each client. Clients with smaller reconstruction errors are presumed to behave normally, as their updates align closely with the learned latent manifold. The bottom $K\%$ of clients with the lowest errors are selected to form a clean subset $\mathcal{C}_{\text{clean}}$. This subset serves as the purified training data for the

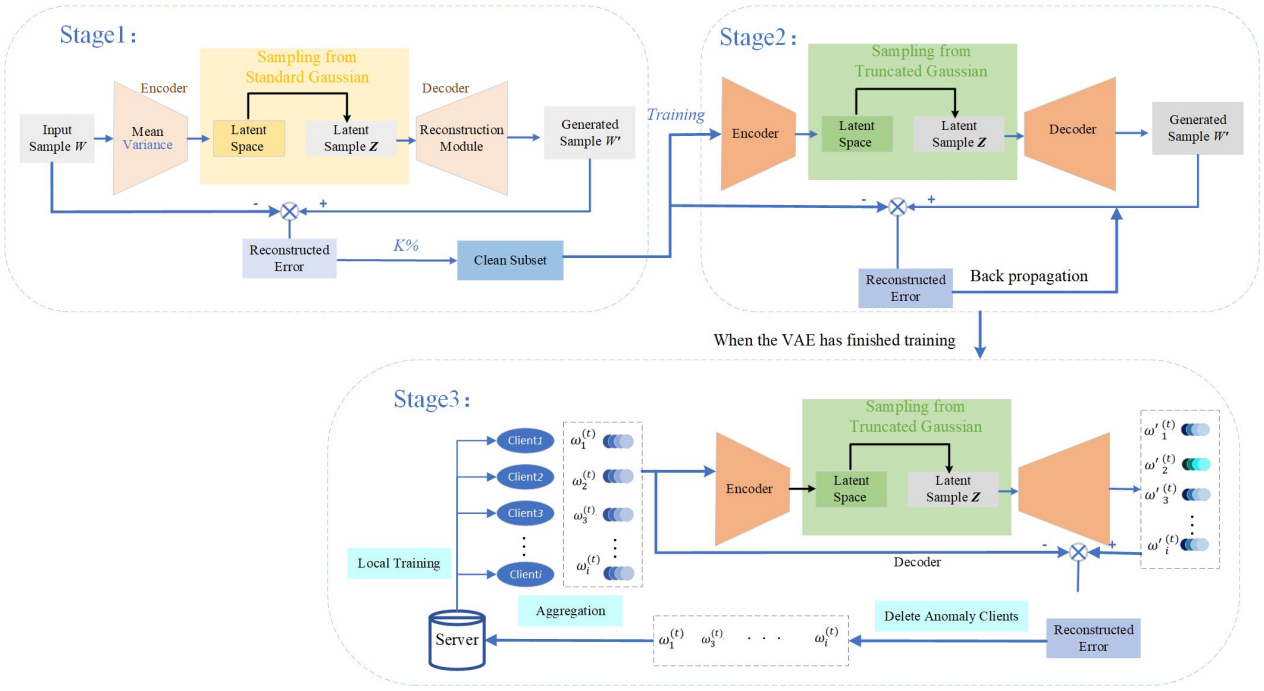

Figure 3: Overall framework of the proposed DVTG approach in FL. Stage 1 employs a standard VAE to compute reconstruction errors for all client updates and selects the lowest $K\%$ as a clean subset. Stage 2 trains a second VAE with a truncated Gaussian prior on this subset to model benign distributions. In Stage 3, the trained model evaluates incoming updates, filtering those with high reconstruction errors before aggregation.

next stage. In this way, Stage 1 functions as a coarse-grained filter that separates obvious anomalies from normal behaviors, providing a foundation for refined modeling in Stage 2.

4.2.2 Robust Retraining with Truncated Gaussian Prior

The second stage refines the learned latent representation by retraining a new VAE (VAE-2) on the clean subset obtained from Stage 1. Instead of the standard Gaussian prior, VAE-2 adopts a truncated Gaussian prior, which constrains the latent space within a bounded region to suppress extreme or adversarial latent activations:

$$z \sim \mathcal{N}_T(0, I), \quad z \in [-c, c]^d. \tag{15}$$

Here, $\mathcal{N}_T(0, I)$ denotes a multivariate Gaussian truncated within $[-c, c]^d$, and d represents the latent dimensionality. This modification limits latent values to a fixed range, reducing the influence of outliers and enhancing robustness against noisy or poisoned updates. The training objective for VAE-2 is defined as:

$$L_{\text{VAE-2}} = \mathbb{E}_{q(z|\omega)}[\|\omega - \hat{\omega}\|^2] + D_{\text{KL}}(q(z|\omega)\|\mathcal{N}_T(0, I)). \tag{16}$$

Compared with the previous stage, the only difference lies in replacing the unbounded Gaussian prior with a truncated one. This bounded prior enforces compact and stable latent representations, ensuring that minor variations in benign updates do not cause large deviations in latent space. As a result, VAE-2 becomes more sensitive to subtle anomalies and more resistant to latent contamination. Essentially, this stage serves as a fine-grained purification process that solidifies the model's understanding of benign update distributions.

4.2.3 Defense via Anomaly Scoring and Client Filtering

After the dual-stage training, the refined VAE (VAE-2) is deployed for AD during subsequent FL rounds. For each new round t, the server applies VAE-2 to reconstruct the received client updates $\omega_i^{(t)}$ and calculates their reconstruction errors:

$$s_i = \|\omega_i^{(t)} - \hat{\omega}_i^{(t)}\|^2. \tag{17}$$

This score measures the deviation of each client update from the benign distribution encoded in Stage 2. To convert this continuous score into a binary anomaly decision, an appropriate threshold τ is required.

Threshold Selection. Selecting a suitable threshold is a key step in unsupervised anomaly detection, and existing works adopt fundamentally different strategies. Du *et al.* [86] select the top-n largest reconstruction errors as anomalies. This avoids explicit thresholding, but it requires prior knowledge of the anomaly ratio, which is unavailable in real deployments. Peng *et al.* [87] determine the threshold through cross-validation on labeled validation sets. This approach works for benchmark evaluation, but it contradicts the unsupervised setting of federated learning and cannot be applied when labels are unavailable. Prediction-based models, such as the LSTM method by Malhotra *et al.* [88], derive thresholds from training prediction errors using the empirical rule $\mu + 3\sigma$, where μ and σ denote the mean and standard deviation of the prediction errors computed on the training set. This relies on the weak assumption that benign errors form a compact region while anomalies lie in the statistical tail.

Inspired by this statistical perspective, we adopt a similar thresholding approach but adjust it to the characteristics of FL. Normal prediction errors in time-series forecasting tend to be stable and tightly concentrated, which makes a conservative 3σ threshold suitable for avoiding false positives. In contrast, benign client updates in federated learning exhibit substantially larger variability due to data heterogeneity and differences in local training, leading to a much wider distribution of normal reconstruction errors. Using a 3σ threshold in this setting would therefore be overly permissive, causing many malicious updates to be incorrectly classified as normal. As a result, adopting a more sensitive 2σ cutoff provides a better balance between detection sensitivity and robustness in federated learning.

To ensure a fully unsupervised design, the threshold is obtained from the reconstruction-error statistics of the pseudo-clean subset generated in Stage 1. This subset is constructed by selecting the bottom 50% of updates with the smallest reconstruction errors, based on the standard assumption that fewer than half of the clients are malicious [82]. This selection

strategy keeps the pseudo-clean subset predominantly benign, even when some normal clients are inadvertently excluded.

Formally, the threshold is defined as

$$\tau = \mu_{\text{clean}} + 2\sigma_{\text{clean}}, \tag{18}$$

where μ_{clean} and σ_{clean} denote the mean and standard deviation of reconstruction errors within the pseudo-clean set. This strategy requires no labels, no prior knowledge of anomaly proportion, and no strict assumptions on error distribution beyond benign dominance, making it both fully unsupervised and practically deployable for FL anomaly detection.

Client Filtering and Aggregation. With the threshold established, a client is classified as malicious if $s_i > \tau$, and benign otherwise. Only benign clients are included in global model aggregation:

$$w^{(t+1)} = \frac{1}{|\mathcal{C}_{\text{benign}}|} \sum_{i \in \mathcal{C}_{\text{benign}}} w_i^{(t)}. \tag{19}$$

Through this three-stage process, DVTG achieves progressive purification of latent representations: Stage 1 performs coarse filtering, Stage 2 achieves refined modeling with a truncated Gaussian prior, and Stage 3 delivers real-time AD. Together, these components form an unsupervised, server-side defense mechanism that effectively mitigates model poisoning while maintaining scalability and compatibility with standard FL protocols.

To summarize the complete operational workflow of the proposed framework, the overall training and defense procedure of DVTG is presented in Algorithm 1.

Algorithm 1: DVTG

Input: Client updates $\{\boldsymbol{\omega}_i^{(t)}\}_{i=1}^N$, selection ratio $K\%$, truncation bound c, threshold τ
Output: Filtered normal updates for aggregation
Train VAE-1 using loss defined in Eq. (13)
foreach *client* $i \in \{1, ..., N\}$ **do**
  Compute reconstruction error: $\mathcal{L}_{\text{rec}}^{(i)} = \|\boldsymbol{\omega}_i^{(t)} - \hat{\boldsymbol{\omega}}_i^{(t)}\|^2$
Select bottom $K\%$ of clients with lowest $\mathcal{L}_{\text{rec}}$ to form clean subset $\mathcal{C}_{\text{clean}}$
Train VAE-2 on $\mathcal{C}_{\text{clean}}$ with prior $z \sim \mathcal{N}_T(0, I),\ z \in [-c, c]^d$
Use loss defined in Eq. (16)
foreach *client* $i \in \{1, ..., N\}$ **do**
  Compute reconstruction error using VAE-2
  if $\mathcal{L}_{rec}^{(i)} > \tau$ **then**
    Mark client i as malicious ;
  else
    Retain client i as benign ;
Aggregate normal clients using dataset-size-weighted FedAvg:

$$\boldsymbol{\omega}^{(t+1)} = \sum_{i \in \mathcal{C}_{\text{benign}}} \frac{|D_i|}{\sum_{j \in \mathcal{C}_{\text{benign}}} |D_j|} \boldsymbol{\omega}_i^{(t)}$$

4.3 Summary

This chapter introduced the DVTG framework for AD in FL, addressing the limitations of conventional VAEs by incorporating a truncated Gaussian prior to stabilize the latent space and enhance robustness. The proposed framework operates through three sequential stages: Stage 1 performs coarse filtering using a standard VAE, Stage 2 refines latent modeling with a truncated Gaussian prior, and Stage 3 conducts real-time anomaly screening and selective aggregation. The training and defense procedures demonstrate how DVTG progressively purifies model representations and filters adversarial updates in an unsupervised, server-side manner without modifying client operations. Although the framework introduces moderate server-side computation proportional to the number of clients and latent dimensionality, inference requires only forward passes through VAE-2, ensuring scalability and efficiency. Overall, DVTG provides a lightweight yet effective defense mechanism that enhances the

resilience of FL systems against model poisoning and adversarial manipulation.

Chapter 5

5 Experiments and Results

5.1 Experiment Setup

This section describes the experimental setup used to evaluate the proposed DVTG framework, including the dataset configuration, attack models, baseline methods, and training settings.

We evaluate the proposed DVTG framework on the MNIST dataset, which contains 60,000 training and 10,000 testing grayscale digit images. To simulate a realistic non-IID setting, the training data was sorted by label and divided into 200 shards, each containing 300 samples. These shards were randomly assigned to 100 clients, with each client receiving two shards. As a result, most clients hold data from only two digit classes, forming a highly label-skewed distribution that reflects practical FL scenarios. A simple multilayer perceptron (MLP) with two fully connected layers and ReLU activation is employed as the global model. For the truncated Gaussian prior used in Stage 2 of DVTG, the truncation bound was set to $c = 2.5$, which restricts the latent variables to the range $[-c, c]$ and improves robustness against extreme or adversarial activations. All models were implemented in PyTorch 2.2.1, trained with a learning rate of 0.001, and batch size of 128. Each client performs one local training epoch per round before transmitting its model update to the central server.

To assess robustness, adversarial clients were injected into the FL system with proportions $p \in \{10\%, 20\%, 30\%\}$. Two representative model-poisoning attacks were simulated:

- **Sign-Flipping Attack:** malicious clients inverted their local model updates by mul-

tiplying all parameters by -1, introducing severe disruption during aggregation.

- **Additive Gaussian Noise Attack:** each malicious client added random noise sampled from $\mathcal{N}(0, \sigma^2)$ with $\sigma = 0.1$ to its update, creating subtle but cumulative distortions.

Both attack types capture extreme (sign-flipping) and stealthy (additive-noise) adversarial behaviors commonly observed in decentralized learning environments.

To demonstrate the advantage of the proposed DVTG framework, we compared it with three baseline settings:

- **No Attack:** an ideal reference scenario in which all clients are benign and training proceeds without adversarial interference.

- **No Defense (FedAvg):** the standard federated averaging aggregation without any AD or defense mechanism, representing the lower bound of performance under poisoning attacks.

- **Single-VAE Defense:** a one-stage unsupervised defense method in which a single VAE is trained on all client updates to model normal behavior. Clients whose reconstruction errors exceed a fixed threshold are identified as abnormal and excluded from aggregation.

5.2 Performance Metrics

We evaluate the effectiveness of the proposed DVTG framework from two complementary perspectives: **(1)** AD performance, and **(2)** global model accuracy during federated training. The detection metrics quantify how effectively DVTG identifies and removes malicious clients, while the global test accuracy reflects the overall learning quality and the impact of the defense mechanism on model convergence.

To assess AD capability, four standard metrics are employed: Precision (P), Recall (R), F1-score (F1), and Accuracy (A). They are defined as follows:

$$P = \frac{TP}{TP + FP}, \tag{20}$$

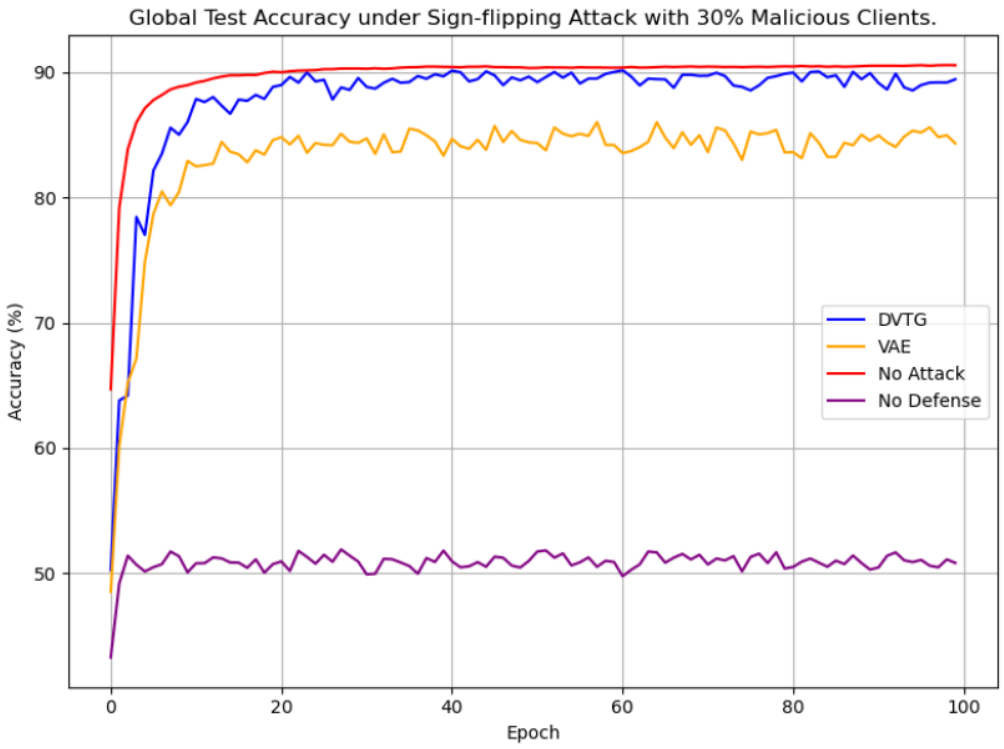

Figure 4: Global test accuracy under sign-flipping attack with 30% malicious clients.

$$R = \frac{TP}{TP + FN}, \tag{21}$$

$$F1 = \frac{2 \times P \times R}{P + R}, \tag{22}$$

$$A = \frac{TP + TN}{TP + TN + FP + FN}, \tag{23}$$

where TP, FP, FN, and TN denote the numbers of true positives, false positives, false negatives, and true negatives, respectively. These values are computed based on the known client identities (benign or malicious). A higher precision indicates fewer false alarms, while a higher recall implies a lower miss rate in detecting attackers. The F1-score provides a balanced assessment between precision and recall, and the overall accuracy measures the proportion of correctly classified clients among all participants. These metrics evaluate how well DVTG detects adversarial clients and how stably it maintains the global model.

We report both the AD results and the global model performance under different at-

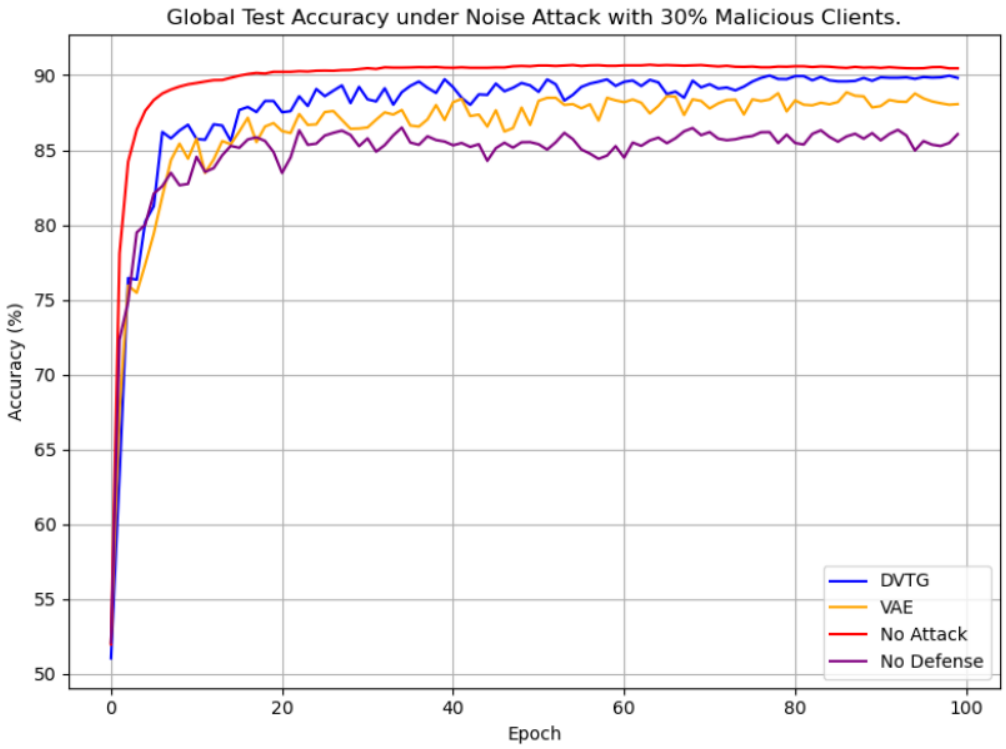

Figure 5: Global test accuracy under additive Gaussian noise attack with 30% malicious clients.

tack scenarios. The following analysis provides a detailed evaluation of DVTG's detection effectiveness and its contribution to robust FL.

5.2.1 Anomaly Detection Results

DVTG is compared with the baseline Single-VAE method under two representative model-poisoning attacks: sign-flipping and additive Gaussian noise, each evaluated at attack ratios of 10%, 20%, and 30%. The results are summarized in Table 1 and Table 2, respectively.

For the **sign-flipping attack** (Table 1), DVTG achieves consistently higher detection performance across all ratios, maintaining balanced precision and recall with an average F1-score of 0.92 compared to 0.81 for Single-VAE. Although Single-VAE attains slightly higher precision at the 30% attack ratio (0.923), its recall drops below 0.80, indicating that many malicious clients remain undetected. In contrast, DVTG's truncated Gaussian latent modeling helps preserve a clearer boundary between benign and poisoned updates, leading to more accurate anomaly separation even under severe contamination.

For the **additive Gaussian noise attack** (Table 2), DVTG demonstrates near-perfect

Table 1: Performance under Different Sign-Flipping Attack Ratios

Model	Attack Ratio	Metrics			
		Precision	**Recall**	**F1 Score**	**Accuracy**
Single-VAE	10%	0.845	0.804	0.824	0.903
	20%	0.850	0.771	0.800	0.844
	30%	0.923	0.782	0.804	0.803
DVTG	10%	0.943	0.899	0.920	0.971
	20%	0.922	0.903	0.912	0.938
	30%	0.919	0.906	0.912	0.912

detection results, with F1-scores exceeding 0.97 and overall accuracy above 0.98 across all ratios. Compared with Single-VAE, DVTG improves recall by approximately 10%–15% at higher contamination levels, confirming its effectiveness in identifying subtle stochastic perturbations. These results verify that the proposed three-stage structure effectively filters clean updates in Stage 1 and refines latent representation learning in Stage 2, resulting in robust and reliable anomaly identification.

Table 2: Performance under Different Additive Gaussian Noise Attack Ratios

Model	Attack Ratio	Metrics			
		Precision	**Recall**	**F1 Score**	**Accuracy**
Single-VAE	10%	0.899	0.826	0.764	0.922
	20%	0.872	0.842	0.814	0.875
	30%	0.856	0.800	0.827	0.907
DVTG	10%	0.991	0.953	0.976	0.991
	20%	0.995	0.951	0.974	0.982
	30%	0.996	0.966	0.986	0.983

5.2.2 Global Model Accuracy

To evaluate the macro-level impact of DVTG on federated training, we track global test accuracy under each attack scenario and compare it with the Single-VAE, No-Defense, and No-Attack baselines. As shown in Fig. 4, under a 30% sign-flipping attack, the No-Defense

model collapses to around 50% accuracy, while Single-VAE partially mitigates degradation but fluctuates during convergence. In contrast, DVTG rapidly approaches the No-Attack upper bound and stabilizes near 90% accuracy within the first 30 epochs, indicating that most poisoned gradients are successfully removed.

A similar pattern is observed in Fig. 5 for additive noise attacks. DVTG achieves faster convergence and maintains accuracy above 88% throughout training, whereas the Single-VAE model stabilizes around 80%. This performance improvement of about 8 to 10 percentage points at high attack ratios demonstrates DVTG's robustness to both adversarial and random perturbations. By filtering anomalous updates before aggregation, DVTG enhances gradient consistency across clients, resulting in smoother optimization and improved generalization of the global model. In summary, the results confirm that DVTG not only excels at detecting malicious clients but also preserves model convergence and accuracy under various adversarial intensities, demonstrating its practicality as a defense framework for real-world FL systems.

5.3 Summary

This chapter conducted a comprehensive experimental evaluation to validate the effectiveness of the proposed DVTG framework. Under both sign-flipping and additive Gaussian noise attacks, DVTG consistently achieved superior AD performance and maintained robust global model accuracy compared with the Single-VAE baseline. The three-stage design, enhanced by truncated Gaussian latent modeling, effectively distinguished benign from malicious client updates and improved resilience against both strong and subtle poisoning attacks. Overall, the experimental findings confirm the practical reliability and generalization capability of DVTG as an unsupervised defense mechanism in FL.

Chapter 6

6 Conclusion

6.1 Summary of the Thesis

This thesis explored AD and attack mitigation in FL, aiming to enhance the security and robustness of decentralized learning systems against adversarial threats, e.g., model poisoning. The research combined analytical analysis, methodological development, and experimental validation to provide a comprehensive understanding of how AD can be effectively applied in FL environments.

Specifically, the main outcomes of this research are summarized as follows:

- The thesis reviewed the fundamental concepts of FL and analyzed its critical security vulnerabilities that distinguish it from centralized learning frameworks. It also provided a systematic overview of existing AD methods, including reconstruction-based and prediction-based approaches, and discussed their limitations in distributed and privacy-preserving settings.

- Various defense mechanisms against model poisoning were examined and categorized into passive and proactive strategies. Their respective advantages and limitations were analyzed to establish a clearer understanding of current challenges in FL security.

- A formal problem formulation was introduced to define adversarial behavior modeling and the defense objectives in FL, forming the theoretical basis for the development of a new defense strategy.

- Building upon these foundations, we propose an unsupervised defense framework **DVTG**. The framework employs a three-stage structure: the first stage filters benign client updates, while the second stage, equipped with a truncated Gaussian latent prior, captures the distribution of normal updates and identifies malicious clients through anomaly scoring and client filtering.

- Comprehensive experiments under sign-flipping and additive Gaussian noise attacks were conducted to evaluate the proposed method. The results demonstrated that DVTG consistently achieved higher detection precision and recall, maintained stable global model convergence, and exhibited strong resilience compared with baseline methods.

In summary, this thesis contributes to the understanding of AD and adversarial defense in FL by providing both a conceptual framework and a practical solution. Through systematic analysis and extensive experiments, the proposed DVTG framework has proven to be a robust and effective unsupervised defense mechanism that improves the reliability, scalability, and trustworthiness of FL systems.

6.2 Future Works

The present study primarily focuses on a two-layer FL architecture consisting of a central server and distributed clients. However, in large-scale applications, such a structure often suffers from high communication overhead and limited aggregation efficiency. Future research could explore the incorporation of intermediate aggregation layers (such as cohorts or edge servers) to form hierarchical or multi-tier FL architectures, which can significantly improve communication efficiency and scalability. In such architectures, different layers can possess their own detection and defense capabilities. The local aggregation layer can independently perform AD to identify and filter out malicious client updates at an early stage, while the global server can adjust aggregation weights or update strategies based on the detection results reported by lower layers. This multi-level mechanism enables the system to respond rapidly to local attacks while maintaining overall stability, coordination, and robustness of the federated training process.

Although the current dual-VAE framework effectively models the distribution of benign client updates, future research could explore detection models that capture the relationships among clients rather than treating them independently. Attention-based mechanisms provide a promising direction for this purpose. By assigning importance weights based on the interactions among client updates, attention enables the system to identify patterns of synchronized deviations or group level similarities that may indicate coordinated adversarial behavior. It can also reveal clients whose updates appear normal when viewed individually but become suspicious in the context of their peers. This relational modeling makes attention particularly suitable for detecting stealthy or low intensity attacks that are difficult to identify through independent anomaly detection methods. Furthermore, in safety-critical domains such as healthcare and finance, interpretability plays a vital role in ensuring the transparency and reliability of AD models. In healthcare, anomalies may be associated with medical diagnoses, while in finance, fraud detection can have direct legal implications. Transparent model decisions allow domain experts to validate results and make informed judgments, which is essential for building trust and ensuring accountability. Future research could focus on developing explainable defense models that reveal the rationale behind AD, such as highlighting influential features or visualizing decision boundaries in the latent space. Enhancing interpretability in this way would not only improve model transparency and safety but also facilitate the broader adoption of FL in sensitive and regulated environments.

Although the proposed DVTG framework focuses on accurately identifying and filtering malicious clients, future research could further improve the post-detection defense process to create a more adaptive and closed-loop protection mechanism. Current defense strategies often remove detected anomalies directly from aggregation, which can lead to the loss of potentially useful information and reduced model generalization. A more effective approach would involve adaptive responses based on the severity and confidence of detection results. For example, the server could adjust aggregation weights according to each client's trust score, gradually restoring or excluding clients depending on their behavior in subsequent training rounds. Such a trust-aware or reputation-based mechanism would allow the system to mitigate attacks while maintaining participation diversity and learning continuity. Such improvements would transform the defense process from a static filtering scheme into a

dynamic, self-evolving mechanism that enhances the overall robustness of FL.

Bibliography

[1] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.

[2] A. El Ouadrhiri and A. Abdelhadi, "Differential privacy for deep and federated learning: A survey," *IEEE Access*, vol. 10, pp. 22 359–22 380, 2022.

[3] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, 2021.

[4] 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.

[5] T. D. Nguyen, P. Rieger, M. Miettinen, A.-R. Sadeghi *et al.*, "Poisoning attacks on federated learning-based iot intrusion detection system," in *Proceedings of the Workshop on Decentralized IoT Systems and Security (DISS)*, vol. 79, 2020, pp. 1–7.

[6] D. Cao, S. Chang, Z. Lin, G. Liu, and D. Sun, "Understanding distributed poisoning attack in federated learning," in *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2019, pp. 233–239.

[7] L. Lyu, H. Yu, X. Ma, C. Chen, L. Sun, J. Zhao, Q. Yang, and P. S. Yu, "Privacy and robustness in federated learning: Attacks and defenses," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 7, pp. 8726–8746, 2022.

[8] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, 2021.

[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1273–1282.

[10] R. S. Antunes, C. André da Costa, A. Küderle, I. A. Yari, and B. Eskofier, "Federated learning for healthcare: Systematic review and architecture proposal," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 4, pp. 1–23, 2022.

[11] G. Long, Y. Tan, J. Jiang, and C. Zhang, "Federated learning for open banking," in *Federated Learning: Privacy and Incentive*. Springer, 2020, pp. 240–254.

[12] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 72–80, 2020.

[13] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," *arXiv preprint arXiv:1907.02189*, 2019.

[14] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[15] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," vol. 33, pp. 7611–7623, 2020.

[16] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," *arXiv preprint arXiv:2111.04263*, 2021.

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

[18] X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou, "Fedbn: Federated learning on non-iid features via local batch normalization," *arXiv preprint arXiv:2102.07623*, 2021.

[19] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, "Federated learning with personalization layers," *arXiv preprint arXiv:1912.00818*, 2019.

[20] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5132–5143.

[21] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.

[22] H. Wu and P. Wang, "Fast-convergent federated learning with adaptive weighting," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1078–1088, 2021.

[23] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.

[24] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1175–1191.

[25] S. Liu, B. Zhou, Q. Ding, B. Hooi, Z. Zhang, H. Shen, and X. Cheng, "Time series anomaly detection with adversarial reconstruction networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 4293–4306, 2022.

[26] F. Dong, Y. Zhang, and X. Nie, "Dual discriminator generative adversarial network for video anomaly detection," *IEEE Access*, vol. 8, pp. 88 170–88 176, 2020.

[27] D. Guo, Z. Liu, and R. Li, "Regraphgan: A graph generative adversarial network model for dynamic network anomaly detection," *Neural Networks*, vol. 166, pp. 273–285, 2023.

[28] Y. Liu, Z. Li, C. Zhou, Y. Jiang, J. Sun, M. Wang, and X. He, "Generative adversarial active learning for unsupervised outlier detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1517–1528, 2019.

[29] J. Miao, H. Tao, H. Xie, J. Sun, and J. Cao, "Reconstruction-based anomaly detection for multivariate time series using contrastive generative adversarial networks," *Information Processing & Management*, vol. 61, no. 1, p. 103569, 2024.

[30] R. Liu, W. Liu, Z. Zheng, L. Wang, L. Mao, Q. Qiu, and G. Ling, "Anomaly-gan: A data augmentation method for train surface anomaly detection," *Expert Systems with Applications*, vol. 228, p. 120284, 2023.

[31] Y. Li, Z. Shi, C. Liu, W. Tian, Z. Kong, and C. B. Williams, "Augmented time regularized generative adversarial network (atr-gan) for data augmentation in online process anomaly detection," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3338–3355, 2021.

[32] G. Fan, Y. Ma, X. Mei, F. Fan, J. Huang, and J. Ma, "Hyperspectral anomaly detection with robust graph autoencoders," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2021.

[33] S. Wang, X. Wang, L. Zhang, and Y. Zhong, "Auto-ad: Autonomous hyperspectral anomaly detection network based on fully convolutional autoencoder," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2021.

[34] H. Liu, X. Su, X. Shen, and X. Zhou, "Msnet: Self-supervised multi-scale network with enhanced separation training for hyperspectral anomaly detection," *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[35] X. Lin, Z. Li, H. Fan, Y. Fu, and X. Chen, "Exploiting negative correlation for unsupervised anomaly detection in contaminated time series," *Expert Systems with Applications*, p. 123535, 2024.

[36] C. Huang, Z. Yang, J. Wen, Y. Xu, Q. Jiang, J. Yang, and Y. Wang, "Self-supervision-augmented deep autoencoder for unsupervised visual anomaly detection," *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 13 834–13 847, 2021.

[37] C. Yin, S. Zhang, J. Wang, and N. N. Xiong, "Anomaly detection based on convolutional recurrent autoencoder for iot time series," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 1, pp. 112–122, 2020.

[38] W. Zhang, C. Zhang, and F. Tsung, "Grelen: Multivariate time series anomaly detection from the perspective of graph relational learning." in *International Joint Conference on Artificial Intelligence*, 2022, pp. 2390–2397.

[39] X. Zhou, Y. Hu, W. Liang, J. Ma, and Q. Jin, "Variational lstm enhanced anomaly detection for industrial big data," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3469–3477, 2020.

[40] J. Wu, Z. Zhao, C. Sun, R. Yan, and X. Chen, "Fault-attention generative probabilistic adversarial autoencoder for machine anomaly detection," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7479–7488, 2020.

[41] Q. Su, B. Tian, H. Wan, and J. Yin, "Anomaly detection under contaminated data with contamination-immune bidirectional gans," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[42] J. Yu, X. Gao, F. Zhai, B. Li, B. Xue, S. Fu, L. Chen, and Z. Meng, "An adversarial contrastive autoencoder for robust multivariate time series anomaly detection," *Expert Systems with Applications*, vol. 245, p. 123010, 2024.

[43] X. Zhang, N. Li, J. Li, T. Dai, Y. Jiang, and S.-T. Xia, "Unsupervised surface anomaly detection with diffusion probabilistic model," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 6782–6791.

[44] S. Li, J. Yu, Y. Lu, G. Yang, X. Du, and S. Liu, "Self-supervised enhanced denoising diffusion for anomaly detection," *Information Sciences*, vol. 669, p. 120612, 2024.

[45] J. Zeng, X. Liu, and Z. Li, "Radio anomaly detection based on improved denoising diffusion probabilistic models," *IEEE Communications Letters*, 2023.

[46] X. Li, C. Xiao, Z. Feng, S. Pang, W. Tai, and F. Zhou, "Controlled graph neural networks with denoising diffusion for anomaly detection," *Expert Systems with Applications*, vol. 237, p. 121533, 2024.

[47] H. He, J. Zhang, H. Chen, X. Chen, Z. Li, X. Chen, Y. Wang, C. Wang, and L. Xie, "A diffusion-based framework for multi-class anomaly detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8472–8480.

[48] J. Pei, J. Wang, D. Shi, and P. Wang, "Detection and imputation-based two-stage denoising diffusion power system measurement recovery under cyber-physical uncertainties," *IEEE Transactions on Smart Grid*, vol. 15, no. 6, pp. 5965–5980, 2024.

[49] Y. Wang, X. Du, Z. Lu, Q. Duan, and J. Wu, "Improved lstm-based time-series anomaly detection in rail transit operation environments," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 9027–9036, 2022.

[50] H. Chen, H. Liu, X. Chu, Q. Liu, and D. Xue, "Anomaly detection and critical scada parameters identification for wind turbines based on lstm-ae neural network," *Renewable Energy*, vol. 172, pp. 829–840, 2021.

[51] P. Liu, X. Sun, Y. Han, Z. He, W. Zhang, and C. Wu, "Arrhythmia classification of lstm autoencoder based on time series anomaly detection," *Biomedical Signal Processing and Control*, vol. 71, p. 103228, 2022.

[52] Y. Yao, J. Ma, S. Feng, and Y. Ye, "Svd-ae: An asymmetric autoencoder with svd regularization for multivariate time series anomaly detection," *Neural Networks*, vol. 170, pp. 535–547, 2024.

[53] S. Longari, D. H. N. Valcarcel, M. Zago, M. Carminati, and S. Zanero, "Cannolo: An anomaly detection system based on lstm autoencoders for controller area network," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1913–1924, 2020.

[54] L. Zhang, W. Bai, X. Xie, L. Chen, and P. Dong, "Tmanomaly: Time-series mutual adversarial networks for industrial anomaly detection," *IEEE Transactions on Industrial Informatics*, 2023.

[55] B. Du, X. Sun, J. Ye, K. Cheng, J. Wang, and L. Sun, "Gan-based anomaly detection for multivariate time series using polluted training set," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 12, pp. 12 208–12 219, 2021.

[56] J. Pei, J. Wang, and D. Shi, "Data-driven measurement tampering detection considering spatial-temporal correlations," in *Proceedings of the 2019 IEEE 3rd Conference on Energy Internet and Energy Systems Integration (EI2)*, 2019, pp. 2641–2646.

[57] T. Lei, C. Gong, G. Chen, M. Ou, K. Yang, and J. Li, "A novel unsupervised framework for time series data anomaly detection via spectrum decomposition," *Knowledge-Based Systems*, vol. 280, p. 111002, 2023.

[58] D. Hu, S. Wu, J. Wang, and D. Shi, "Training a dynamic neural network to detect false data injection attacks under multiple unforeseen operating conditions," *IEEE Trans. Smart Grid*, 2023.

[59] C. Tang, L. Xu, B. Yang, Y. Tang, and D. Zhao, "Gru-based interpretable multivariate time series anomaly detection in industrial control system," *Computer Security*, vol. 127, p. 103094, 2023.

[60] J. Yu, X. Gao, B. Li, F. Zhai, J. Lu, B. Xue, S. Fu, and C. Xiao, "A filter-augmented auto-encoder with learnable normalization for robust multivariate time series anomaly detection," *Neural Networks*, vol. 170, pp. 478–493, 2024.

[61] L. Li, J. Yan, H. Wang, and Y. Jin, "Anomaly detection of time series with smoothness-inducing sequential variational auto-encoder," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 3, pp. 1177–1191, 2020.

[62] J. Kim, H. Kang, and P. Kang, "Time-series anomaly detection with stacked transformer representations and 1d convolutional network," *Engineering Applications of Artificial Intelligence*, vol. 120, p. 105964, 2023.

[63] S. Tuli, G. Casale, and N. R. Jennings, "Tranad: Deep transformer networks for anomaly detection in multivariate time series data," *arXiv preprint arXiv:2201.07284*, 2022.

[64] W. Zhu, W. Li, E. R. Dorsey, and J. Luo, "Unsupervised anomaly detection by densely contrastive learning for time series data," *Neural Networks*, vol. 168, pp. 450–458, 2023.

[65] H. Sun, M. Chen, J. Weng, Z. Liu, and G. Geng, "Anomaly detection for in-vehicle network using cnn-lstm with attention mechanism," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10 880–10 893, 2021.

[66] T. Le, H. C. Vu, A. Ponchet-Durupt, N. Boudaoud, Z. Cherfi-Boulanger, and T. Nguyen-Trang, "Unsupervised detecting anomalies in multivariate time series by robust convolutional lstm encoder–decoder (rcled)," *Neurocomputing*, vol. 592, p. 127791, 2024.

[67] C. Wang and G. Liu, "From anomaly detection to classification with graph attention and transformer for multivariate time series," *Advanced Engineering Informatics*, vol. 60, p. 102357, 2024.

[68] J. Fan, Z. Wang, H. Wu, D. Sun, J. Wu, and X. Lu, "An adversarial time–frequency reconstruction network for unsupervised anomaly detection," *Neural Networks*, vol. 168, pp. 44–56, 2023.

[69] Y. Shi, B. Wang, Y. Yu, X. Tang, C. Huang, and J. Dong, "Robust anomaly detection for multivariate time series through temporal gcns and attention-based vae," *Knowledge-Based Systems*, vol. 275, p. 110725, 2023.

[70] C. Ding, S. Sun, and J. Zhao, "Mst-gat: A multimodal spatial–temporal graph attention network for time series anomaly detection," *Information Fusion*, vol. 89, pp. 527–536, 2023.

[71] M. Jin, H. Y. Koh, Q. Wen, D. Zambon, C. Alippi, G. I. Webb, I. King, and S. Pan, "A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[72] Y. Wu, H.-N. Dai, and H. Tang, "Graph neural networks for anomaly detection in industrial internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9214–9231, 2022.

[73] X. Dong, C. J. Taylor, and T. F. Cootes, "Defect classification and detection using a multitask deep one-class CNN," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1719–1730, 2021.

[74] Z. Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, "Learning graph structures with transformer for multivariate time-series anomaly detection in IoT," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9179–9189, 2021.

[75] Y. Zheng, H. Y. Koh, M. Jin, L. Chi, K. T. Phan, S. Pan, Y.-P. P. Chen, and W. Xiang, "Correlation-aware spatial–temporal graph learning for multivariate time-series anomaly detection," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[76] Y. Liu, Z. Li, S. Pan, C. Gong, C. Zhou, and G. Karypis, "Anomaly detection on attributed networks via contrastive self-supervised learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 6, pp. 2378–2392, 2022.

[77] A. Ghosh, J. Hong, D. Yin, and K. Ramchandran, "Robust federated learning in a heterogeneous environment," *arXiv preprint arXiv:1906.06629*, 2019.

[78] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[79] Y. Chen, L. Su, and J. Xu, "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent," in *Abstracts of the 2018 ACM International Conference on Measurement and Modeling of Computer Systems*, 2018, pp. 96–96.

[80] Z. Chen, P. Tian, W. Liao, and W. Yu, "Zero knowledge clustering based adversarial mitigation in heterogeneous federated learning," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1070–1083, 2020.

[81] W. Wan, S. Hu, J. Lu, L. Y. Zhang, H. Jin, and Y. He, "Shielding federated learning: Robust aggregation with adaptive client selection," *arXiv preprint arXiv:2204.13256*, 2022.

[82] M. J. Idrissi, H. Alami, A. El Mahdaouy, A. El Mekki, S. Oualil, Z. Yartaoui, and I. Berrada, "Fed-anids: Federated learning for anomaly-based network intrusion detection systems," *Expert Systems with Applications*, vol. 234, p. 121000, 2023.

[83] C. Ma, J. Li, M. Ding, K. Wei, W. Chen, and H. V. Poor, "Federated learning with unreliable clients: Performance analysis and mechanism design," *IEEE Internet of Things Journal*, vol. 8, no. 24, pp. 17 308–17 319, 2021.

[84] Z. Chen, N. Lv, P. Liu, Y. Fang, K. Chen, and W. Pan, "Intrusion detection for wireless edge networks based on federated learning," *IEEE Access*, vol. 8, pp. 217 463–217 472, 2020.

[85] X. Lyu, N. Wang, Y. Xiao, S. Li, T. Li, D. Chen, and Y. Chen, "Two heads are better than one: Model-weight and latent-space analysis for federated learning on non-iid data against poisoning attacks," *arXiv preprint arXiv:2503.23288*, 2025.

[86] X. Du, J. Chen, J. Yu, S. Li, and Q. Tan, "Generative adversarial nets for unsupervised outlier detection," *Expert Systems with Applications*, vol. 236, p. 121161, 2024.

[87] X. Peng, H. Li, F. Yuan, S. G. Razul, Z. Chen, and Z. Lin, "An extreme learning machine for unsupervised online anomaly detection in multivariate time series," *Neurocomputing*, vol. 501, pp. 596–608, 2022.

[88] T. Ergen and S. S. Kozat, "Unsupervised anomaly detection with lstm neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 8, pp. 3127–3141, 2019.