Universal Backdoor Defense via Label Consistency in Vertical Federated Learning

Peng Chen[1], **Haolong Xiang**[1*], **Xin Du**[2], **Xiaolong Xu**[1], **Xuhao Jiang**[3], **Zhihui Lu**[4], **Jirui Yang**[4], **Qiang Duan**[5] and **Wanchun Dou**[6]

[1]Nanjing University of Information Science and Technology
[2]Zhejiang University
[3]Huawei
[4]Fudan University
[5]Pennsylvania State University
[6]Nanjing University
{pengchen, hlxiang}@nuist.edu.cn

Abstract

Backdoor attacks in vertical federated learning (VFL) are particularly concerning as they can covertly compromise VFL decision-making, posing a severe threat to critical applications of VFL. Existing defense mechanisms typically involve either label obfuscation during training or model pruning during inference. However, the inherent limitations on the defender's access to the global model and complete training data in VFL environments fundamentally constrain the effectiveness of these conventional methods. To address these limitations, we propose the Universal Backdoor Defense (UBD) framework. UBD leverages Label Consistent Clustering (LCC) to synthesize plausible latent triggers associated with the backdoor class. This synthesized information is then utilized for mitigating backdoor threats through Linear Probing (LP), guided by a constraint on Batch Normalization (BN) statistics. Positioned within a unified VFL backdoor defense paradigm, UBD offers a generalized framework for both detection and mitigation that critically does not necessitate access to the entire model or dataset. Extensive experiments across multiple datasets rigorously demonstrate the efficacy of the UBD framework, achieving state-of-the-art performance against diverse backdoor attack types in VFL, including both dirty-label and clean-label variants.

1 Introduction

With the rise of deep learning, data privacy and security have gained significant attention. Federated Learning (FL), a collaborative training paradigm that maintains data locally, has garnered significant attention. FL are principally categorized into Horizontal Federated Learning (HFL) and Vertical Federated Learning (VFL) based on the data distribution across participating parties [Yang *et al.*, 2019]. VFL is applicable when datasets share the same sample space (e.g., the same users) but possess different feature spaces. For example, a bank could enhance its credit assessment model by integrating more features from local retailers (the same users) and maintaining data privacy [Fu *et al.*, 2022; Qiu *et al.*, 2024; Zheng *et al.*, 2024; Zheng *et al.*, 2023; Xu *et al.*, 2025; Xiang *et al.*, 2024].

While FL is designed to enhance data security, it is nonetheless vulnerable to attacks such as backdoor injection, which leverages the exchange of intermediate training information [Jin *et al.*, 2023]. A backdoor attack involves an adversary embedding a specific trigger pattern into a subset of poisoned data, aiming to establish a connection between the trigger and a chosen target class during model training. The result is a compromised model that, during inference, consistently classifies any input containing the trigger as the backdoor target class. Because these attacks generally do not impact the model's performance on clean, trigger-free data, they are inherently stealthy, seriously threatening the integrity of model decision-making, especially in VFL with model and data splitting structure [Li *et al.*, 2024b].

Backdoor attacks in VFL can be classified as dirty-label and clean-label variants. In dirty-label attacks, the adversary, acting as a passive participant party, substitutes local feature embeddings and gradients of poisoned and clean samples, linking them to the clean target label by tampering with intermediate feature embeddings [Zou *et al.*, 2022]. In contrast, clean-label attacks leverage prior knowledge or label inference to identify target samples and embed triggers into their local inputs, associating them with the target class during the VFL training process without altering the labels or feature embeddings. [Naseri *et al.*, 2024; Bai *et al.*, 2023; Chen *et al.*, 2023; Chen *et al.*, 2024].

Existing VFL backdoor defense methods fall into two main categories: label protection and backdoor purification methods. Label protection methods [Li *et al.*, 2022; Fu *et al.*, 2022] attempt to prevent attacks by limiting information leakage, particularly ground truth labels, which the adversary exploits. Purification methods [Liu *et al.*, 2022; Bai *et al.*, 2023; Wu and Wang, 2021] focus on pruning poisoned model neurons or filtering malicious samples to sup-

[*]Corresponding Author

press backdoor effects. However, both approaches are significantly hampered by the information asymmetry inherent in VFL: the active party acting as the defender lacks access to the passive parties' data and local bottom models. This prevents tailored label protection against varied poisoning strategies targeting unseen data and limits the scope of purification methods that require visibility into the fully distributed model. Consequently, these methods often result in suboptimal backdoor defensive performance while simultaneously degrading the utility of the VFL model.

To address these challenges, this paper proposes a Universal Backdoor Defense (UBD) framework designed to defend against both dirty-label and clean-label backdoor attacks, which consists of Label Consistent Clustering (LCC) and Linear Probing (LP) mitigation modules. The LCC module identifies the backdoor target class and synthesizes triggers within the latent feature space controlled by the defender. It predicts potential poisoned samples by analyzing the consistency between ground truths and prediction results, combined with a clustering method. Backdoor triggers are then generated in the latent feature space by examining discrepancies between the predicted poisoned samples and a small set of clean validation data. LP mitigates attacks by fine-tuning the defender's top model using the identified backdoor class and triggers, with Batch Normalization (BN) statistics constraint to preserve the utility of the VFL model.

In summary, the main contributions of this paper can be summarized as:

- We propose a backdoor defense framework, UBD, for VFL systems that efficiently defends against generic backdoor attacks, including dirty-label and clean-label variants, without accessing models and data of passive parties, requiring only 5% of clean validation data.
- The proposed LCC module features a backdoor trigger inversion strategy that enables the identification of latent triggers and backdoor target class using minimal clean validation data without requiring model training.
- We design an efficient backdoor mitigation module LP with BN statistics constraint to achieve state-of-the-art defense performance, which does not need to fine-tune the whole VFL model.
- We perform extensive experiments on multiple datasets to validate the effectiveness, robustness, and efficiency of the proposed defense method in VFL.

2 Related Work

2.1 Backdoor Attacks in VFL

Backdoor attacks in VFL can be classified as dirty-label and clean-label variants, distinguished by whether or not the labels are tampered with during poisoning. As a typical dirty-label backdoor attack, LRB [Zou *et al.*, 2022] assumes the adversary has a few clean samples of the target class. Then, the adversary exchanges the intermediate information of poisoned and target samples, establishing an implicit connection between the target class and poisoned samples. Although labels in the active party cannot be directly modified, the replacement operation in the intermediate layer realizes a dirty-label backdoor attack [Chen *et al.*, 2017; Doan *et al.*, 2021; Li *et al.*, 2024b]. In contrast, Methods [Chen *et al.*, 2024; Bai *et al.*, 2023] utilize label inference to identify samples belonging to the target class, subsequently injecting the backdoor by directly embedding triggers into the inputs of these samples. Similarly, approaches described in [Naseri *et al.*, 2024; He *et al.*, 2023; Chen *et al.*, 2023] assume the adversary has access to a limited number of target samples and conducts clean-label backdoor attacks by directly associating triggers with these target samples.

2.2 Backdoor Defenses in VFL

Existing VFL backdoor defense methods fall into two main categories: label protection and backdoor purification methods. Label protection methods [Li *et al.*, 2022; Fu *et al.*, 2022; Wu *et al.*, 2022; Bai *et al.*, 2023] attempt to disrupt the malicious mapping between poisoned samples and the target label by interfering with label-derived information, often by perturbing transmitted gradients. Backdoor purification methods [Wu and Wang, 2021; Bai *et al.*, 2023; Xu *et al.*, 2024] aim to eliminate backdoor effects within the VFL model, for instance, by devising masks to identify and suppress poisoned features or neurons.

However, both approaches are significantly limited by the VFL setting. Label protection often leads to substantial utility decreases due to indiscriminate application to all training data, while purification is hampered by the defender's lack of access to passive parties' bottom models and data, preventing comprehensive fine-tuning or pruning. To address these limitations, this paper aims to design a generic backdoor defense framework capable of addressing both dirty-label and clean-label attacks. This framework conducts solely on the defender's accessible top model and requires only a minimal validation dataset, enabling efficient and practical defense against VFL backdoors.

3 Preliminaries

3.1 Backdoor Attacks in Vertical Federated Learning

VFL is a collaborative distributed machine learning system in which data is vertically partitioned into different participants. The system consists of K passive parties and one active party. Passive parties only possess local feature data, while the active party has access to and controls the labels. For a classification task with the training dataset $\mathcal{D} \triangleq \left\{ \left(\mathbf{x}_i^k, y_i \right)_{k=1}^{K} \right\}_{i=1}^{N}$ in VFL, where $\mathbf{x}_i^k$ denotes the i-th samples of k-th passive participant. The objective of VFL can be formulated as [Liu *et al.*, 2024; Ye *et al.*, 2024]:

$$\min_{\Theta} \ell(\Theta; \mathcal{D}) \triangleq \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\Theta; \mathbf{x}_i, y_i) + \lambda \sum_{k=1}^{K+1} \gamma(\Theta) \qquad (1)$$

where $\Theta = \{\theta_1, \cdots, \theta_K; \theta_{top}\}$ and $\mathcal{L}(\cdot)$ denote the whole model parameters and loss function of VFL respectively; $\gamma(\cdot)$ and λ represent the regularizer and corresponding hyperparameter. We omit the k-th notation in $\mathbf{x}$ for simplicity.

During the training stage of VFL, each passive party computes the local feature embedding $H_i^k = \Phi_k(x_i^k, \theta_k)$ with its local data and bottom model, and then, the active party receives all feature embeddings from passive parties and utilizes the top model and labels to compute the objective function in Eq.(1). Next, the active party computes the gradients $\frac{\partial \ell}{\partial \theta_{top}}$ to update its top model θ_{top}. Afterwards, the gradients of the loss function with respect to feature embeddings $\frac{\partial \ell}{\partial H_i^k}$ are transmitted to each passive party. Finally, each passive party k updates its local bottom model θ_k with the gradients. In the inference phase, the active party outputs predictions based on feature embeddings sent by the bottom model of each passive party, where input samples are locally split.

Backdoor attacks aim to implant a hidden task in the VFL model during training, causing it to misclassify inputs with the specific trigger while maintaining main task performance on clean inputs during inference [Li *et al.*, 2024b]. The objective of backdoor attacks in VFL can be formulated as:

$$\min_{\Theta} \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}_c}\mathcal{L}(\Theta;\mathbf{x},y)}_{\text{Main Task}} + \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}_p}\mathcal{L}(\Theta;\mathbf{x},\delta,\tau)}_{\text{Backdoor Task}} \quad (2)$$

where $\mathcal{D}_c$ and $\mathcal{D}_p$ denote the clean and poisoned data respectively, $\mathcal{D} = \mathcal{D}_c \cup \mathcal{D}_p$; δ and τ represent backdoor triggers and target class; $\mathbf{x}$ and y refer to the set of inputs and labels, respectively. For simplicity, the regularizer in Eq.(1) is omitted.

3.2 Threat Model

Adversary's capacity. Since the active party controls the top model and labels, it can easily modify labels or manipulate the predictions of the top model. Therefore, this paper investigates scenarios where the adversary is a malicious passive party. This adversary strictly adheres to the VFL protocol, limiting its actions to exchanging local feature embeddings and gradients with the active party.

Defender's capacity. The defender, acting as the active party in VFL, also adheres to the VFL protocol. The defender's goal is to detect the backdoor target class and mitigate backdoor attacks during the inference stage of VFL. In accordance with the VFL protocol, the defender has access to the feature embeddings from all passive parties, as well as the labels and prediction results for the training dataset. In addition, the defender has access to an extra validation dataset $\mathcal{D}_{val}$, which consists of a small number of clean samples from each class. This is a reasonable assumption, as model performance must be validated during the VFL training stage and aligns with existing backdoor defense efforts [Xu *et al.*, 2024; Zhu *et al.*, 2024; Guo *et al.*, 2023; Ma *et al.*, 2023].

4 Universal Backdoor Defense Framework

The goal of the Universal Backdoor Defense (UBD) framework is to mitigate backdoor attacks in VFL systems. As depicted in Fig.1, UBD achieves this through the integration of two core modules: the Label Consistent Clustering (LCC) module and the Linear Probing (LP) module. The LCC module identifies the backdoor target class and generates latent backdoor triggers, while the LP module utilizes the identified backdoor class and triggers to mitigate the backdoor effects in the VFL model.

4.1 LCC Backdoor Detection

The LCC method comprises two modules for generating triggers of the backdoor target class in VFL: Label Consistency Purification (LCP) and Feature Clustering Detection (FCD). The LCP module identifies suspicious poisoned training samples by analyzing the consistency between predictions and ground truth labels. Subsequently, the FCD module leverages the clustering of these suspicious samples to directly infer the backdoor target class and corresponding triggers.

Label Consistency Purification. Considering a scenario where the VFL model has already completed the training process, the backdoor has already been injected into the model. Consequently, the backdoor triggers are strongly associated with the target class, leading the VFL model to classify poisoned samples containing these triggers as the backdoor target class. In this context, the defender retains all feature embeddings and labels of the training data obtained during training and controls the VFL outputs through ownership of the top model in the active party.

For both clean-label and dirty-label backdoor attacks, any poisoned sample $\mathbf{x} \in \mathcal{D}_p$ is directly mapped to the backdoor target τ:

$$\Phi(\mathbf{x}) \mapsto \tau \quad \text{for all } \mathbf{x} \in \mathcal{D}_\text{p}. \quad (3)$$

where $\Phi(\cdot)$ denotes the feature mapping function of VFL, respectively.

In clean-label attacks, the adversary injects triggers directly into samples whose true label is already τ, further tightening the connection $\Phi(\mathbf{x}) \mapsto \tau$. By contrast, dirty-label attacks substitute $\mathcal{D}_p$ with target samples in ways that induce $\Phi(\mathbf{x}) \mapsto \tau$. From the defender's perspective, regardless of the specific backdoor injection method, the observed outcome is consistently the same: all poisoned feature embeddings concentrate in a region identified with the target label τ.

Definition 1 (Label Consistency Property)**.** *Let $\mathcal{G}$ be the label mapping function, and suppose $\mathcal{F}^{\text{poi}} = \mathcal{G} \circ \Phi$ represents the victim VFL model. For each training example $(\mathbf{x}, y) \in \mathcal{D}$ with $y = \tau$, define a trigger function $\eta(\cdot)$ that injects the backdoor triggers (if any) into $\mathbf{x}$. We say $(\mathbf{x}, y)$ satisfies* label consistency *with respect to τ if*

$$\begin{aligned} \mathcal{F}^{\text{poi}}(\mathbf{x}) &= \tau, \\ \mathcal{F}^{\text{poi}}\big(\eta(\mathbf{x})\big) &= \tau. \end{aligned} \quad (4)$$

Then the set of all such samples that satisfy the label consistency property is

$$\mathcal{D}_{\text{LCP}}^{\tau} = \Big\{ (\mathbf{x}, y) \in \mathcal{D} \;\Big|\; y = \tau,\ \mathcal{F}^{\text{poi}}(\mathbf{x}) = \tau, \\ \mathcal{F}^{\text{poi}}\big(\eta(\mathbf{x})\big) = \tau \Big\}. \quad (5)$$

In essence, every sample in $\mathcal{D}_{\text{LCP}}^{\tau}$ is one whose true label is τ but also predicted as τ by the poisoned model whether or not the backdoor $\eta(\cdot)$ is applied. Such label consistency may indicate suspicious poisoned samples.

According to Eq.(5), the defender can identify suspicious poisoned samples for the target class by selecting those where the predictions match their true labels. Utilizing label consistency, the process of identifying poisoned samples within the target class can be framed as a binary classification task [Xiang *et al.*, 2023b; Xiang *et al.*, 2023a].

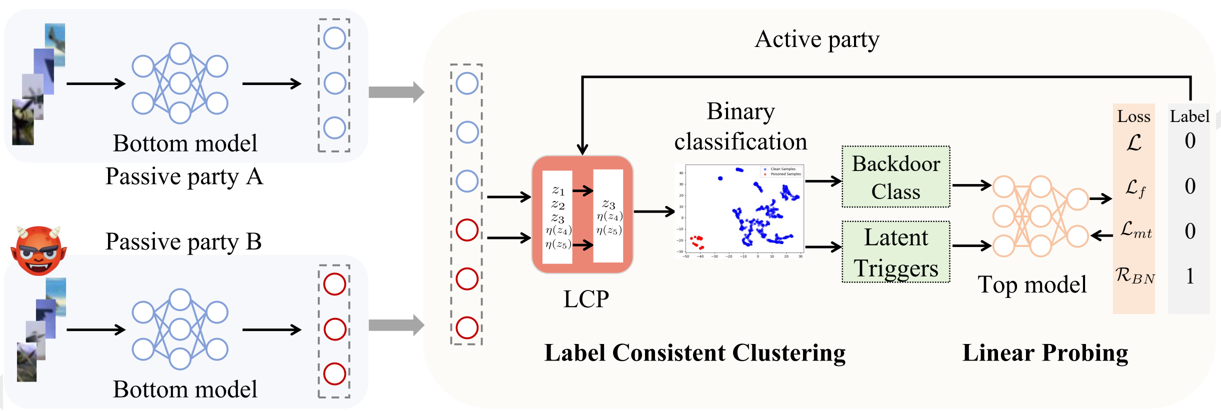

Figure 1: A sketch of our proposed UBD framework for VFL.

Feature Clustering Detection. Furthermore, as a binary classification task, the suspicious poisoned samples $\mathcal{D}_{LCP}$ can be decomposed by the clustering algorithm. This is because the identified samples now contain at most two categories: clean samples and poisoned samples with triggers for the same ground truth category.

Specifically, we adopt the K-means clustering method to locate the poisoned samples with triggers. For the concatenated feature embeddings sent from passive parties $H = [H^1, H^2, \cdots, H^K]$, where H refers to the set of feature embeddings of suspicious poisoned data $\mathcal{D}_{LCP}$, the FCD method leverages the K-means clustering algorithm to obtain two clusters $\mathcal{C}_\tau^m, m \in \{0, 1\}$. The poisoning rate of backdoor attacks maintains a low level because they are typically constrained by limited adversary's knowledge. This means the number of poisoned samples is significantly smaller than that of clean samples. As a result, the FCD method identifies the cluster with fewer samples as the poisoned cluster, denoted as $\mathcal{C}_\tau^S$. Using the available validation dataset, the defender can then infer the backdoor triggers in the latent feature space. [Guo *et al.*, 2023].

$$\hat{\delta} = \mathbb{E}[H(\mathbf{x}_i) \mid \mathbf{x}_i \in \mathcal{C}_\tau^S] - \mathbb{E}[H(x_j) \mid \mathbf{x}_j \in \mathcal{D}_{val}^\tau], \qquad (6)$$

where $\hat{\delta}$ denotes latent backdoor triggers with respect to $\eta(\mathbf{x})$; $\mathcal{D}_{val}^\tau$ refers to the subset of the validation dataset $\mathcal{D}_{val}$ with class τ; $\mathbb{E}[\cdot]$ represents the element-wise average operation.

Furthermore, the latent backdoor triggers $\hat{\delta}$ can cause the benign samples to be incorrectly classified as the backdoor target class:

$$ASR_\tau^S = \frac{1}{N_{\neg\tau}} \sum_{i=1}^{N_{\neg\tau}} \mathbb{I}(\mathcal{F}^{\text{poi}}\left(\mathbf{x}_i; \hat{\delta}; \Theta\right) \equiv \tau) \qquad (7)$$

where ASR_τ^S represents the attack success rate for the category τ; $\mathbb{I}(\cdot)$ denotes the indicator function; the data used in Eq.(7) consists of all samples in the validation set $\mathcal{D}_{val}$, excluding those belonging to category τ, resulting in a total size of $N\neg\tau$; latent backdoor triggers $\hat{\delta}$ is added to the concatenated feature embeddings H in an element-wise operation.

Remark 1. *For the sake of illustration, the LCC method described above assumes the ground truth is the backdoor target τ. However, in practice, the defender does not know the backdoor target class in advance. Without loss of generality, LCC is suitable for each category in the VFL classification task. More importantly, the method evaluates all categories individually to calculate the ASR for each category using Eq. (7). The category with the highest backdoor ASR is identified as the backdoor target class, and the corresponding latent triggers $\hat{\delta}$ are obtained simultaneously.*

Algorithm 1 LCC backdoor detection on the VFL system.

```
Require:
    Pretrained model parameters $\Theta$; Training and validation dataset $\mathcal{D}$, $\mathcal{D}_{val}$.
Ensure:
    Backdoor target class $\tau$ and latent triggers $\hat{\delta}$
for each batch $B$ in $\mathcal{D}$ do
    for each party $k = 1, 2, \ldots, K$ in parallel do
        $k$ computes feature embeddings $\{H_i^k\}_{i \in B}$ according to its bottom model $f_k$;
        % Feature embeddings are actually retained from the VFL training process
    end for
end for
Active party (Defender):
for each category $m = 1, 2, \cdots, M$ do
    % Step 1: Label Consistency Purification
    computes Eq. (5), and then purify the training data $\mathcal{D}$ to get the $\mathcal{D}_{LCP}^m$ for class $m$;
    % Step 2: Feature Clustering Detection
    $\mathcal{C}_m^S = clustering(\mathcal{D}_{LCP}^m)$
    candidate backdoor triggers $\hat{\delta}^m$ are computed according to Eq. (6) using $\mathcal{D}_{val}^m$ and $\mathcal{C}_m^S$;
    computes $ASR_m^S$ for category $m$ with Eq. (7);
end for
identifies the backdoor class as:
$S = \arg\max_{m=1}^{M} ASR_m^S$;
Output backdoor target class $S$; latent triggers $\hat{\delta}^S$
```

Analysis of LCC for dirty-label and clean-label attacks. With the LCP method iteratively performed on each category, the data obtained by the defender consisted of two parts:

clean samples correctly classified in each category and poisoned samples misclassified into the target class. Then, the FCD module utilizes K-means clustering to identify poisoned samples of the target class by iteratively evaluating all categories. The intuition is to treat the backdoor target class as ground truth and frame the problem as a binary classification task, using the clustering method to distinguish poisoned from clean samples. The feature embeddings of poisoned and clean samples for the backdoor target class inherently exhibit distinguishable inter-cluster distances, which K-means clustering exploits for binary classification.

In the case of clean-label backdoor attacks, poisoned and clean samples from the target class differ only in the backdoor triggers, which enlarge inter-cluster distances for clustering. Similarly, in dirty-label backdoor attacks, while the poisoned feature embeddings received by the defender exhibit a strong connection to the target class, they also retain intrinsic non-target features. These non-target features, combined with backdoor triggers, further amplify the inter-cluster distances between poisoned and clean samples. In essence, the inter-cluster distances can be interpreted as the backdoor triggers themselves. Using these generated backdoor triggers, along with the known validation set, the defender can reliably identify the backdoor target class. In fact, there are various clustering methods that can be employed in the LCC module, such as DBSCAN, etc., and we use K-means only because of its easy applicability to the binary classification task.

As outlined in Algorithm 1, after the VFL training process, the defender retains the labels and feature embeddings of the training dataset. The defender locally executes the LCC backdoor detection to identify the backdoor target class and its associated triggers. Specifically, the defender iteratively applies LCP and FCD modules to each category. By evaluating the ASR and the suspicious latent triggers for each category, the defender selects the category with the highest ASR as the backdoor target class and assigns the corresponding generated trigger as its associated triggers.

4.2 LP Backdoor Mitigation

The LCC detection method allows the defender to identify the backdoor target class and triggers. However, detection is insufficient as the backdoor has already been implanted into the model, threatening the VFL system. To address this issue, the UBD framework adopts a linear probing (LP) backdoor mitigation method to eliminate the backdoor in VFL.

We observed that for the backdoor target class, the clustering method in the embedded feature space could effectively separate poisoned samples from clean ones. However, in the output space, the top model predicts poisoned and clean samples as the target class, rendering them indistinguishable. In other words, top model parameters have a remarkable impact on the backdoor manipulation. Therefore, LP aims to fine-tune the top model parameters to remove the triggers' impact on poisoned samples.

With the identified backdoor target class and triggers, the defender can directly formulate the loss function to fine-tune the top model. In particular, the objective function can be formalized as:

$$\min_{\theta_{top}} \quad \frac{1}{N_v}\sum_{i=1}^{N_v}\Bigg[\lambda_1\mathcal{L}(H_i;\theta_{top}) + \lambda_2\mathcal{L}_{\text{filter}}(H_i, y_i, \hat{\delta}, S;\theta_{top}) + \lambda_3\mathcal{L}_{\text{maintain}}(H_i, \hat{\delta};\theta_{top}) + \lambda_4\mathcal{R}_{BN}(H_i)\Bigg]. \tag{8}$$

where N_v denotes the number of samples in the validation dataset $\mathcal{D}_{val}$; S refers to the identified backdoor target class of LCC backdoor detection; H_i represents concatenated feature embeddings of the i-th sample in the validation dataset $\mathcal{D}_{val}$; λ_1, λ_2, λ_3 and λ_4 are hyperparameters to balance the loss function.

In Eq.(8), the first part of the loss function aims to constrain the top model parameters with the cross-entropy loss on clean samples to prevent deviation from the main task of VFL. The second part aims to eliminate the connection between backdoor triggers and the target class in the top model, while at the same time, the third part wants to re-establish the link between poisoned samples and their ground truth. The specific loss functions can be expressed respectively as [Zhu *et al.*, 2024]:

$$\mathcal{L}_f = -\log\left(1 - \mathcal{G}_S(H;\hat{\delta};\theta_{top})\right), \tag{9}$$

$$\begin{aligned}\mathcal{L}_{mt} = &-\log\left(\mathcal{G}_y(H;\hat{\delta};\theta_{top})\right)\\ &-\log\left(1 - \max_{q\neq y}\mathcal{G}_k(H;\hat{\delta};\theta_{top})\right).\end{aligned} \tag{10}$$

where $\mathcal{G}_S$, $\mathcal{G}_y$ and $\mathcal{G}_k$ denote the probability of predicting concatenated feature embeddings H to label S, y and q by the active party $\mathcal{G}$ respectively. The latent triggers $\hat{\delta}$ are added to concatenated feature embeddings H, etc.$H \bigoplus \hat{\delta}$.

It is worth noting that Eq.(8) includes a regularization term R_{BN}, which indicates that LP is based on the running statistics of the BN layer as prior constraints. This aims to reduce sensitivity to poisoned samples while enhancing the generalization of the main task by leveraging the stable statistics obtained during the BN training process, as expressed below:

$$\begin{aligned}\mathcal{R}_{BN}(H) = &\sum_l \left\|\mu_l(H) - \text{BN}_l(\text{mean})\right\|_2 +\\ &\sum_l \left\|\sigma_l^2(H) - \text{BN}_l(\text{variance})\right\|_2.\end{aligned} \tag{11}$$

where $\mu_l(H)$and $\sigma_l^2(H)$ are the batch-wise mean and variance estimates of feature maps corresponding to the l-th layer of the top model. BN_l(mean) and BN_l(variance) represent the running mean and variance of the l-th batch normalization layer during training.

5 Experiments

In this section, we conduct comprehensive experiments to evaluate the effectiveness of the proposed UBD framework. Specifically, we first briefly illustrate the experiment setup. Then, we assess the UBD framework on the state-of-the-art dirty-label and clean-label backdoor attacks. In addition, we

Dataset↓	Defense→ Attack↓	No Defense		DP-SGD		BTI-DBF		UBD	
		BA	ASR	BA	ASR	BA	ASR	BA	ASR
CIFAR-10	LRB	73.50±4.19	94.10±3.38	67.68±15.51	16.76±25.50	72.96±2.99	40.76±30.23	**76.51±3.57**	**1.71±3.42**
	VILLAIN	**82.50±0.14**	99.82±0.35	77.81±0.68	22.60±38.70	75.77±0.65	17.85±15.18	82.46±0.24	**1.35±0.21**
NUSWIDE	LRB	**87.10±0.73**	95.88±2.00	73.93±1.05	**2.75±0.96**	86.36±0.34	40.59±17.21	84.19±0.48	4.90±5.22
	VILLAIN	85.82±0.11	99.99±0.01	**87.39±0.73**	99.99±0.01	82.01±0.51	22.10±27.11	85.57±0.20	**2.22±0.59**
CINIC-10	LRB	61.12±3.92	88.20±5.70	60.66±7.67	47.53±18.80	61.17±4.19	17.76±12.34	**65.55±3.96**	**10.84±11.60**
	VILLAIN	**75.16±0.50**	99.93±0.09	57.90±0.72	99.95±0.09	70.37±0.61	41.22±35.96	71.68±0.83	**5.74±11.08**
ImageNette	LRB	59.69±1.98	93.85±5.02	61.34±2.75	87.18±5.85	61.17±4.19	17.76±12.34	**61.38±1.62**	**0.00±0.00**
	VILLAIN	**73.24±1.07**	99.98±0.02	69.31±0.95	99.82±0.26	70.37±0.61	41.22±35.96	72.23±0.97	**0.00±0.00**

Table 1: The performance (%) of backdoor defenses on CIFAR-10, NUSWIDE, CINIC-10, and ImageNette datasets. A larger BA metric indicates better performance, while a smaller ASR metric reflects better defense effectiveness. The optimal results are highlighted in bold.

also introduce the defensive baseline of VFL to validate the superiority of the proposed backdoor defense approach. Finally, we implement an overall ablation study to analyze the UBD framework.

5.1 Experiment Setup

Datasets and Models. This section conducts extensive experiments on four datasets to evaluate the performance of the UBD framework: three image datasets, i.e. CIFAR-10 [Krizhevsky *et al.*, 2009; Li *et al.*, 2024a], Imagenette [Howard and Gugger, 2020], CINIC-10 [Darlow *et al.*, 2018] and one image-text multimodal dataset i.e. NUS-WIDE [Chua *et al.*, 2009]. For CIFAR-10, CINIC-10 and Imagenette, we employed ResNet-18 as the model structure. For NUSWIDE, we adopt a 3-layer MLP as the VFL model. The feature extractor of ResNet-18 and the first two fully connected layers of MLP serve as the bottom models of passive parties in VFL. While the UBD framework supports multiple participants in VFL, we conducted experiments using a two-participant setting for simplicity, consistent with related works [Bai *et al.*, 2023; Fu *et al.*, 2022; Qiu *et al.*, 2024]. The experiments randomly divide 5% from the training set as the validation set for the defender.

Evaluation Metrics. For a fair comparison, we evaluate the effectiveness of the backdoor defense in the UBD framework using Benign Accuracy (BA) and Attack Success Rate (ASR), following prior studies [Li *et al.*, 2024b]. Additionally, we report TPR, FPR, and AUC metrics to assess the performance of the LCC method, consistent with related works [Guo *et al.*, 2023; Mo *et al.*, 2024].

Attack Baselines. We exploit two representative and advanced backdoor attacks to evaluate the UBD framework, including (1) a dirty-label backdoor attack: LRB [Zou *et al.*, 2022], (2) a clean-label backdoor attack: VILLAIN [Bai *et al.*, 2023]. To verify the defense performance of UBD, we adjusted the hyperparameters of the related work [Zou *et al.*, 2022; Bai *et al.*, 2023] to achieve the best attack performance. Specifically, we set the poisoning rate as 1% and 5% for LRB and VILLAIN respectively.

Defense Baselines. To evaluate the effectiveness of UBD, we compare it with two representative and advanced defense methods: DP-SGD [Wu *et al.*, 2022; Bai *et al.*, 2023] and BTI-DBF [Xu *et al.*, 2024]. These methods represent the two primary defense categories outlined in Section 2: label protection and backdoor mitigation. The ratio of DP-SGD is set to 1.0 and the hyperparameters of BTI-DBF are configured following the original paper [Xu *et al.*, 2024].

Training Details. The experiments are conducted with PyTorch with two NVIDIA 3090 GPU cards. Each experiment is repeated 5 times with random seeds. The hyperparameters λ_1, λ_2, λ_3 and λ_4 are set to 1, 0.1, 1, 1. We used Adam optimizer to train LP, with learning rates ranging from 0.001 to 0.1 for different datasets.

5.2 Experimental Results

Performance of UBD. We systematically compare the performance of the UBD framework with current mainstream defense methods under dirty-label and clean-label backdoor attacks. Tab.1 illustrates the results of comparison. In terms of ASR metric, the UBD framework almost achieves the best performance compared with DP-SGD and BTI-DBF, especially for CIFAR10, CINIC-10 and Imagenette datasets. For the NUS-WIDE dataset, the UBD method achieved the best performance under the VILLAIN attack. In contrast, under the LRB attack, the DP-SGD method outperformed UBD on the ASR metric, achieving a 2.15% lower ASR and a 4.26% variance advantage. However, considering the BA metric, DP-SGD's lower ASR comes at the cost of a significant BA loss, with a 13.17% decrease compared to the LRB attack baseline. This characteristic is consistent across other datasets, where reductions in ASR are accompanied by substantial losses in BA. This trade-off highlights the limitations of label protection methods such as DP-SGD. For BTI-DBF, the trigger generator can only be added at the intermediate layer of the VFL, as the defender lacks access to the passive party's input data and bottom model. The method's inability to synchronize fine-tuning with bottom models not only hinders its effectiveness in eliminating ASR but also adversely affects the BA metric.

Impact of LCC. To evaluate the effectiveness of the LCC module, the comparison results for the use of label consistency in the LCC module are presented in Tab. 2. The experimental results indicate that incorporating prediction ground truth consistency provides a notable advantage in detecting poisoned samples, particularly for the VILLAIN method. For

Dataset	LCC	Attack	TPR	FPR	AUC
CIFAR10	w/ lc	LRB	89.33	9.48	89.93
		VILLAIN	100	0.00	100
	w/o lc	LRB	93.19	9.26	91.96
		VILLAIN	100	0	100
NUSWIDE	w/ lc	LRB	69.59	45.71	61.94
		VILLAIN	100	0.00	100
	w/o lc	LRB	40.34	42.34	49.00
		VILLAIN	100	0.00	100
CINIC-10	w/ lc	LRB	60.34	39.81	60.26
		VILLAIN	53.10	44.49	54.31
	w/o lc	LRB	57.44	36.28	60.58
		VILLAIN	30.14	40.04	45.05
ImageNette	w/ lc	LRB	91.01	1.56	94.72
		VILLAIN	100	0.00	100.0
	w/o lc	LRB	94.83	1.02	96.91
		VILLAIN	100	8.7	95.65

Table 2: Comparison of the label consistency in the LCC module (w/ lc denotes "with label consistency" and w/o lc indicates "without label consistency).

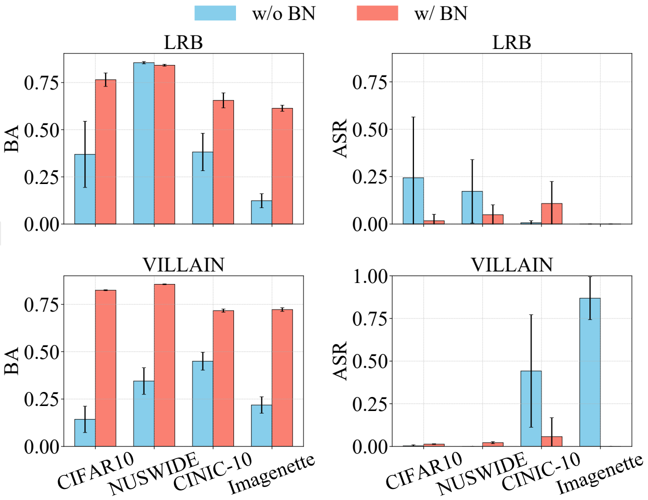

Figure 2: Comparison of LP effectiveness under the UBD framework (Standard Deviation Included.)

instance, on the CINIC-10 dataset, the use of label consistency improves AUC performance for backdoor detection by 9.26%. It should be noted that, for the LRB approach, label consistency does not appear to provide a significant advantage, particularly on the CIFAR-10 and Imagenette datasets. This limitation may stem from a performance bottleneck in the BA metric. For instance, on the Imagenette dataset, LRB shows a 13.55% lower BA compared to VILLAIN. This disparity reduces the label consistency strategy's ability to identify sufficient suspicious poisoned samples, thereby impacting the backdoor detection performance of the LCC module. Nevertheless, label consistency successfully achieved effective backdoor detection across all four datasets.

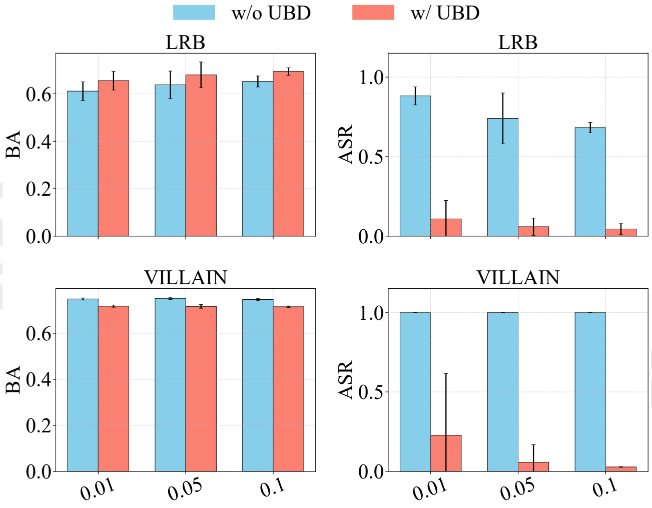

Figure 3: Comparison of different poisoning rates under the UBD framework (Standard Deviation Included.)

Impact of BN. To demonstrate the effectiveness of the LP module, we present a comparison of results with and without BN statistics in Fig.2. The experimental results clearly show that incorporating BN parameters significantly mitigates or even eliminates backdoor effects. For instance, against the VILLAIN attack, the LP method effectively mitigates backdoor effects across all four datasets, achieving an ASR of no more than 6% on the CINIC-10 dataset while limiting BA performance degradation to within 4%.

Impact of poison rate. In Fig.3, we illustrate the impact of varying proportions of poisoned samples on the UBD framework, using the CINIC-10 dataset as an example. The results demonstrate that even with only 1% poisoned samples, UBD remains effective in defending against both dirty-label and clean-label backdoor attacks. For the VILLAIN attack, some variance in UBD performance is observed with 1% poisoned samples. Nonetheless, clean-label backdoor attacks like VILLAIN, which rely on label inference, typically can obtain a large number of poisoned pseudo samples to inject backdoors. Therefore, UBD remains effective and valuable in mitigating clean-label backdoor attacks.

6 Conclusion

This paper proposes a novel backdoor defense VFL framework called UBD, which integrates the LCC and LP modules to detect and mitigate backdoor attacks simultaneously. Specifically, the LCC module synthesizes latent triggers of the backdoor class with the label consistency strategy. Furthermore, the LP module combines the identified backdoor class and generated triggers to efficiently mitigate backdoors in the VFL system by leveraging the statistics of the BN layer, requiring only linear probing of the classifier header. Extensive experiments across multiple datasets demonstrate that UBD significantly outperforms state-of-the-art backdoor defense methods in terms of both model utility and backdoor elimination.

Acknowledgements

This work was supported by the Jiangsu Provincial Major Project on Basic Research of Cutting-edge and Leading Technologies, under grant no. BK20232032, and National Key Research and Development Program of China under grant (2022YFC3302300), Yangtze River Delta Science and Technology Innovation Community Joint Research Project under grant (YDZX20233100004031), National Natural Science Foundation of China under Grant (No. 61873309, No. 92046024, No. 92146002) and Shanghai Science and Technology Project under Grant (No. 22510761000).

References

[Bai *et al.*, 2023] Yijie Bai, Yanjiao Chen, Hanlei Zhang, Wenyuan Xu, Haiqin Weng, and Dou Goodman. VILLAIN: Backdoor attacks against vertical split learning. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2743–2760, Anaheim, CA, August 2023. USENIX Association.

[Chen *et al.*, 2017] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[Chen *et al.*, 2023] Peng Chen, Jirui Yang, Junxiong Lin, Zhihui Lu, Qiang Duan, and Hongfeng Chai. A Practical Clean-Label Backdoor Attack with Limited Information in Vertical Federated Learning . In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 41–50, Los Alamitos, CA, USA, December 2023. IEEE Computer Society.

[Chen *et al.*, 2024] Peng Chen, Xin Du, Zhihui Lu, and Hongfeng Chai. Universal adversarial backdoor attacks to fool vertical federated learning. *Computers & Security*, 137:103601, 2024.

[Chua *et al.*, 2009] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pages 1–9, 2009.

[Darlow *et al.*, 2018] Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*, 2018.

[Doan *et al.*, 2021] Khoa Doan, Yingjie Lao, and Ping Li. Backdoor attack with imperceptible input and latent modification. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 18944–18957. Curran Associates, Inc., 2021.

[Fu *et al.*, 2022] Chong Fu, Xuhong Zhang, Shouling Ji, Jinyin Chen, Jingzheng Wu, Shanqing Guo, Jun Zhou, Alex X. Liu, and Ting Wang. Label inference attacks against vertical federated learning. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1397–1414, Boston, MA, August 2022. USENIX Association.

[Guo *et al.*, 2023] Wei Guo, Benedetta Tondi, and Mauro Barni. Universal detection of backdoor attacks via density-based clustering and centroids analysis. *IEEE Transactions on Information Forensics and Security*, 2023.

[He *et al.*, 2023] Ying He, Zhili Shen, Jingyu Hua, Qixuan Dong, Jiacheng Niu, Wei Tong, Xu Huang, Chen Li, and Sheng Zhong. Backdoor attack against split neural network-based vertical federated learning. *IEEE Transactions on Information Forensics and Security*, 2023.

[Howard and Gugger, 2020] Jeremy Howard and Sylvain Gugger. Fastai: a layered api for deep learning. *Information*, 11(2):108, 2020.

[Jin *et al.*, 2023] Di Jin, Bingdao Feng, Siqi Guo, Xiaobao Wang, Jianguo Wei, and Zhen Wang. Local-global defense against unsupervised adversarial attacks on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8105–8113, 2023.

[Krizhevsky *et al.*, 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[Li *et al.*, 2022] Oscar Li, Jiankai Sun, Xin Yang, Weihao Gao, Hongyi Zhang, Junyuan Xie, Virginia Smith, and Chong Wang. Label leakage and protection in two-party split learning. In *International Conference on Learning Representations*, 2022.

[Li *et al.*, 2024a] Minghui Li, Wei Wan, Yuxuan Ning, Shengshan Hu, Lulu Xue, Leo Yu Zhang, and Yichen Wang. Darkfed: a data-free backdoor attack in federated learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024.

[Li *et al.*, 2024b] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(1):5–22, 2024.

[Liu *et al.*, 2022] Jing Liu, Chulin Xie, Sanmi Koyejo, and Bo Li. Copur: Certifiably robust collaborative inference via feature purification. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 26645–26657. Curran Associates, Inc., 2022.

[Liu *et al.*, 2024] Yang Liu, Yan Kang, Tianyuan Zou, Yanhong Pu, Yuanqin He, Xiaozhou Ye, Ye Ouyang, Ya-Qin Zhang, and Qiang Yang. Vertical federated learning: Concepts, advances, and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[Ma *et al.*, 2023] Wanlun Ma, Derui Wang, Ruoxi Sun, Minhui Xue, Sheng Wen, and Yang Xiang. The "beatrix" resurrections: Robust backdoor detection via gram matrices. In *30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3, 2023*. The Internet Society, 2023.

[Mo *et al.*, 2024] Xiaoxing Mo, Yechao Zhang, Leo Yu Zhang, Wei Luo, Nan Sun, Shengshan Hu, Shang Gao, and

Yang Xiang. Robust backdoor detection for deep learning via topological evolution dynamics. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 171–171. IEEE Computer Society, 2024.

[Naseri *et al.*, 2024] Mohammad Naseri, Yufei Han, and Emiliano De Cristofaro. BadVFL: Backdoor Attacks in Vertical Federated Learning . In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2013–2028, Los Alamitos, CA, USA, May 2024. IEEE Computer Society.

[Qiu *et al.*, 2024] Pengyu Qiu, Xuhong Zhang, Shouling Ji, Changjiang Li, Yuwen Pu, Xing Yang, and Ting Wang. Hijack vertical federated learning models as one party. *IEEE Transactions on Dependable and Secure Computing*, pages 1–18, 2024.

[Wu and Wang, 2021] Dongxian Wu and Yisen Wang. Adversarial neuron pruning purifies backdoored deep models. *Advances in Neural Information Processing Systems*, 34:16913–16925, 2021.

[Wu *et al.*, 2022] Ruihan Wu, Jin Peng Zhou, Kilian Q Weinberger, and Chuan Guo. Does label differential privacy prevent label inference attacks? *arXiv preprint arXiv:2202.12968*, 2022.

[Xiang *et al.*, 2023a] Haolong Xiang, Xuyun Zhang, Mark Dras, Amin Beheshti, Wanchun Dou, and Xiaolong Xu. Deep optimal isolation forest with genetic algorithm for anomaly detection. In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 678–687, 2023.

[Xiang *et al.*, 2023b] Haolong Xiang, Xuyun Zhang, Hongsheng Hu, Lianyong Qi, Wanchun Dou, Mark Dras, Amin Beheshti, and Xiaolong Xu. Optiforest: optimal isolation forest for anomaly detection. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23, 2023.

[Xiang *et al.*, 2024] Haolong Xiang, Xuyun Zhang, Xiaolong Xu, Amin Beheshti, Lianyong Qi, Yujie Hong, and Wanchun Dou. Federated learning-based anomaly detection with isolation forest in the iot-edge continuum. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2024.

[Xu *et al.*, 2024] Xiong Xu, Kunzhe Huang, Yiming Li, Zhan Qin, and Kui Ren. Towards reliable and efficient backdoor trigger inversion via decoupling benign features. In *The Twelfth International Conference on Learning Representations*, 2024.

[Xu *et al.*, 2025] Xiaolong Xu, Hongsheng Dong, Haolong Xiang, Xiyuan Hu, Xiaoyong Li, Xiaoyu Xia, Xuyun Zhang, Lianyong Qi, and Wanchun Dou. C2lrec: Causal contrastive learning for user cold-start recommendation with social variable. *ACM Transactions on Information Systems*, 2025.

[Yang *et al.*, 2019] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.*, 10(2), January 2019.

[Ye *et al.*, 2024] Mang Ye, Wei Shen, Bo Du, Eduard Snezhko, Vassili Kovalev, and Pong C Yuen. Vertical federated learning for effectiveness, security, applicability: A survey. *arXiv preprint arXiv:2405.17495*, 2024.

[Zheng *et al.*, 2023] Fei Zheng, Chaochao Chen, Lingjuan Lyu, and Binhui Yao. Reducing communication for split learning by randomized top-k sparsification. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23, 2023.

[Zheng *et al.*, 2024] Fei Zheng, Chaochao Chen, Lingjuan Lyu, Xinyi Fu, Xing Fu, Weiqiang Wang, Xiaolin Zheng, and Jianwei Yin. Protecting split learning by potential energy loss. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024.

[Zhu *et al.*, 2024] Mingli Zhu, Shaokui Wei, Hongyuan Zha, and Baoyuan Wu. Neural polarizer: A lightweight and effective backdoor defense via purifying poisoned features. *Advances in Neural Information Processing Systems*, 36, 2024.

[Zou *et al.*, 2022] Tianyuan Zou, Yang Liu, Yan Kang, Wenhan Liu, Yuanqin He, Zhihao Yi, Qiang Yang, and Ya-Qin Zhang. Defending batch-level label inference and replacement attacks in vertical federated learning. *IEEE Transactions on Big Data*, pages 1–12, 2022.