RESSUB: DUAL-LEVEL BACKDOOR DEFENSE VIA RESIDUAL ALIGNMENT AND SUBSPACE PERTURBATION IN VERTICAL FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Vertical Federated Learning (VFL) enables collaborative learning over vertically partitioned data without sharing raw features, but remains vulnerable to backdoor attacks launched by malicious passive parties. Existing defenses primarily rely on attack detection or indiscriminate perturbation, while overlooking that backdoor behaviors fundamentally arise from the supervision and gradient feedback exposed during collaborative optimization. In this paper, we propose ResSub, a dual-level backdoor defense framework that simultaneously restructures supervision and gradient feedback in VFL. Specifically, Residual-Transformed Alignment reformulates the optimization objective of passive parties using residual-transformed targets, encouraging them to learn only the information missing from the active party's local prediction rather than complete label semantics. Meanwhile, Subspace-Structured Gradient Perturbation injects structure-aware perturbations into gradient subspaces, suppressing exploitable class-discriminative optimization signals while preserving model utility. Extensive experiments across multiple datasets and attack settings demonstrate that ResSub reduces the average attack success rates from 89.7% to 8.3%, while maintaining competitive clean-task performance.

1 INTRODUCTION

Vertical Federated Learning (VFL) Yang et al. (2019); Liu et al. (2024) enables multiple parties to collaboratively train machine learning models over vertically partitioned features without sharing raw data, making it particularly attractive for privacy-sensitive applications such as finance, healthcare, and recommendation systems Cheng et al. (2020). During collaborative training, participating parties exchange intermediate representations and gradient information instead of raw features, allowing joint model optimization while preserving data locality. However, these interactions inevitably expose task-related learning signals Qiu et al. (2022); Zhu et al. (2024), creating new opportunities for malicious participants to manipulate the collaborative learning process.

Among security threats, backdoor attacks have emerged as one of the most critical threats to VFL He et al. (2023); Bai et al. (2024). Unlike centralized learning or horizontal federated learning, malicious passive parties cannot access ground-truth labels or complete training samples. Nevertheless, they can establish trigger-target associations by exploiting the learning feedback received from the active party during collaborative training. Existing attacks implant triggers into local features or intermediate representations and progressively strengthen their association with target classes through iterative model updates Zou et al. (2022); Bai et al. (2023); Shen et al. (2025). Since the active party cannot directly inspect passive parties' local features or internal behaviors, such attacks remain difficult to detect and defend.

Existing VFL backdoor defenses mainly improve robustness through three directions, including protecting interaction information Li et al. (2022), detecting anomalous behaviors Hu et al. (2026), and mitigating implanted backdoors Chen et al. (2025). Although effective under specific settings, **they overlook that supervision and gradient feedback continuously expose exploitable task-discriminative information during collaborative training**. Some recent studies reconstruct supervision objectives to improve label privacy and model robustness Gao et al. (2024a;b). However,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

they are designed for privacy preservation rather than suppressing backdoor formation and therefore cannot effectively eliminate the information exploited by backdoor attackers.

We observe that passive-party learning in VFL is fundamentally guided by two types of feedback provided by the active party. The first is the supervision objective, which determines what information passive parties are encouraged to learn, while the second is the embedding gradients, which continuously convey task-discriminative signals during collaborative optimization. Together, these two feedback channels enable malicious passive parties to progressively associate trigger patterns with target classes. Therefore, defending against VFL backdoor attacks requires simultaneously reducing the exploitable information contained in both supervision and gradient feedback.

Motivated by this observation, we propose **ResSub, a dual-level backdoor defense framework for Vertical Federated Learning**. ResSub protects both supervision and gradient feedback during collaborative training. Specifically, we propose Residual-Transformed Alignment (RTA), which reformulates the supervision received by passive parties using residual-transformed optimization targets. Instead of directly learning complete label semantics, passive-party outputs are aligned with residual-transformed targets derived from the active party's local prediction, allowing them to contribute only the residual information required for collaborative prediction. Furthermore, we introduce Subspace-Structured Gradient Perturbation (SSGP), which injects structured perturbations into embedding gradient subspaces to suppress exploitable class-discriminative signals while preserving task utility. By jointly protecting supervision and gradient feedback, ResSub effectively suppresses trigger-target association learning without modifying the standard VFL training protocol or requiring access to passive parties' local features or model parameters.

Extensive experiments on multiple benchmark datasets and representative VFL backdoor attacks demonstrate that ResSub consistently reduces attack success rates (ASR) while maintaining competitive main-task accuracy, significantly outperforming existing defense methods. Ablation studies further verify that RTA and SSGP complement each other in defending against VFL backdoor attacks. Our main contributions are summarized as follows:

- We revisit VFL backdoor attacks from the perspective of collaborative learning feedback, revealing that supervision and gradient feedback jointly constitute the key information exploited by malicious passive parties to establish trigger-target associations.
- We propose ResSub, a dual-level backdoor defense framework consisting of RTA and SSGP, which respectively protect supervision and gradient feedback to suppress backdoor formation during collaborative learning.
- Extensive experiments on multiple benchmark datasets and representative VFL backdoor attacks demonstrate that ResSub consistently achieves a better trade-off between defense effectiveness and clean-task performance than existing methods.

2 Related Work

2.1 Backdoor Attacks in VFL

VFL backdoor attacks differ fundamentally from those in centralized and horizontal federated learning because malicious passive parties cannot directly access ground-truth labels or complete training samples. Instead, they progressively establish trigger-target associations by exploiting the learning feedback exchanged during training. Existing VFL backdoor attacks can be broadly categorized into two groups.

Gradient-driven attacks exploit label-related information contained in embedding gradients to identify target samples or optimize backdoor injection. LRB reveals batch-level label leakage from embedding gradients and performs label replacement attacks through gradient manipulation Zou et al. (2022). LFBA further constructs poisoned samples by measuring embedding-gradient consistency with a small set of anchor samples and reinforces trigger learning through selective sample switching Shen et al. (2025).

Label-recovery attacks first recover or estimate target labels and subsequently inject backdoors into local features or intermediate representations. Representative methods include VILLAIN and BadVFL, which identify target-class samples through label inference or auxiliary data Bai et al.

(2023); Naseri et al. (2024). LR-BA further reconstructs label information via model completion and generates malicious embeddings that progressively drive triggered samples toward the target class Gu & Bai (2023).

Although these attacks adopt different strategies, they all rely on the supervision or gradient feedback received during collaborative training to progressively establish trigger-target associations. This observation indicates that learning feedback constitutes the fundamental medium through which backdoor attacks are formed in VFL.

2.2 Backdoor Defenses in VFL

Existing VFL backdoor defenses mainly improve robustness through three directions: protecting interaction information during collaborative training, detecting malicious behaviors, and mitigating implanted backdoors.

Label information protection mainly reduces the exploitable information available during collaborative learning. Marvell perturbs gradients according to inter-class gradient distributions to reduce label separability Li et al. (2022), while DiscreteSGD quantizes embedding gradients into discrete levels Fu et al. (2022). Sun et al. suppress label leakage by minimizing the statistical dependence between embeddings and labels Sun et al. (2022). Another related direction redesigns the supervision received by passive parties. CKD introduces Complementary Label Coding to reconstruct the supervision target of passive parties, thereby reducing label semantic leakage Gao et al. (2024a). Although effective for privacy preservation, these methods are not designed for backdoor defense and do not explicitly limit the learning feedback exploited for trigger-target association learning.

Detection- and purification-based defenses mitigate backdoor threats by identifying abnormal behaviors or mitigating implanted backdoors. Representative methods include VFLIP, which detects and reconstructs anomalous embeddings during inference Cho et al. (2024), FILTER, which combines suspicious embedding filtering with backdoor unlearning Hu et al. (2026), and purification-based approaches such as UBD, BTI-DBF, and ANP, which eliminate implanted backdoors through trigger inversion, fine-tuning, or neuron pruning Wu & Wang (2021); Xu et al. (2024); Chen et al. (2025).

Unlike existing methods that independently modify either supervision or interaction information, ResSub jointly redesigns supervision and gradient feedback, providing a unified defense against backdoor formation.

3 Preliminaries

3.1 Problem Setup

VFL Setting. We consider a multi-party VFL classification task. The training dataset is denoted by $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i is the complete feature vector of sample i, $y_i \in \{1, \dots, C\}$ is its class label, and C is the number of classes. In VFL, all parties share aligned sample identities while holding different feature subspaces. Accordingly, each sample is vertically partitioned as $x_i = (x_i^0, x_i^1, \dots, x_i^K)$, where the active party P_0 owns local features x_i^0 together with labels y_i, and passive party P_k owns local features x_i^k $(k = 1, \dots, K)$. Each party first extracts a local representation through its bottom model. Specifically, the active party computes $u_i = f_\theta(x_i^0)$, while passive party P_k computes $v_i^k = h_{\psi_k}(x_i^k)$. The active party collects representations from all available passive parties and fuses them together with its own representation through an aggregation module $r_i = (u_i + g_\lambda(\{v_i^k\}_{k\in\mathcal{K}}))$, where $\mathcal{K}$ denotes the set of participating passive parties, and r_i denotes the aggregated representation. The aggregated representation is then fed into the active party's top model,$z_i = t_\phi(r_i)$, where $z_i \in \mathbb{R}^C$ denotes the prediction logits. The prediction distribution is $\hat{\mathbf{y}}_i = \mathrm{softmax}(z_i)$. The overall optimization objective of VFL is $\min_\Theta \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\mathrm{CE}}(z_i, y_i)$, where $\Theta = \{\theta, \psi_1, \dots, \psi_K, \lambda, \phi\}$ contains the parameters of the active-party bottom model, passive-party bottom models, aggregation module, and top model, respectively.

Task Definition of Backdoor Attacks. The objective of a training-time backdoor attack is to implant a hidden malicious behavior into the VFL model while preserving normal prediction perfor-

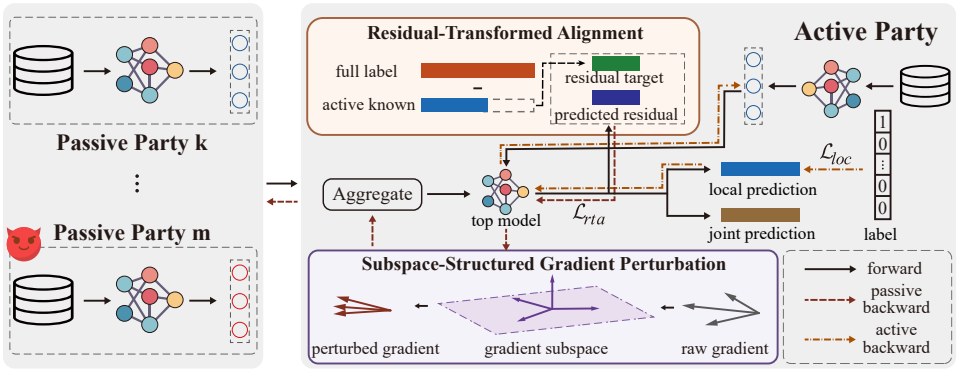

Figure 1: Overview of the proposed ResSub framework.

mance on clean samples. Specifically, given a predefined target class τ, the attacker injects a trigger into selected samples during training such that the trained model predicts the target label whenever the trigger is present. Let $\mathcal{T}_\delta(\cdot)$ denote the trigger injection function with trigger pattern δ. For a clean sample x_i, the corresponding triggered sample is $\tilde{x}_i = \mathcal{T}_\delta(x_i)$. The optimization objective of a backdoor attack can be formulated as:

$$\min_{\Theta} \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}_c}\left[\mathcal{L}(F_\Theta(x), y)\right]}_{\text{Main Task}} + \lambda_b \underbrace{\mathbb{E}_{\tilde{x}\sim\mathcal{D}_p}\left[\mathcal{L}(F_\Theta(\tilde{x}), \tau)\right]}_{\text{Backdoor Task}},$$

where $F_\Theta(\cdot)$ denotes the complete VFL model, $\mathcal{D}_c$ and $\mathcal{D}_p$ denote the clean and poisoned training sets, respectively, and λ_b controls the relative importance of the backdoor objective.

3.2 Threat Model

Attacker's Capability. We consider the standard VFL threat model where one or more passive parties behave maliciously while following the prescribed training protocol. A malicious passive party can arbitrarily manipulate its local data, bottom model, and generated embeddings to implant backdoors, but cannot access the active party's labels, local features, aggregation module, prediction head, or the private data and models of other parties. Therefore, the attacker can only exploit the supervision and gradient feedback exchanged during collaboration to establish trigger-target associations.

Defender's Capability. The defender is deployed at the trusted active party. Besides its local features and labels, it has access to the aggregated representations, prediction logits, and embedding gradients returned during backpropagation. Without accessing passive parties' private data or modifying the standard VFL workflow, the defender is allowed to regulate the supervision and gradient feedback exchanged during collaboration to suppress backdoor formation while preserving clean-task utility.

4 Methodology

In this section, we present ResSub, a dual-level defense framework against backdoor attacks in VFL. As illustrated in Fig. 1, ResSub consists of two complementary components. RTA reconstructs supervision for passive parties, while SSGP protects gradient feedback during collaborative training. Together, they reduce the exploitable learning feedback that enables trigger-target association learning.

4.1 Residual-Transformed Alignment

Although passive parties do not possess ground-truth labels, the supervision still determines what information they learn during collaborative training. To mitigate this risk, we reformulate the supervision received by passive parties using residual-transformed targets. Instead of directly learning

complete label semantics, passive parties are encouraged to align their outputs with residual information derived from the active party's local prediction, thereby contributing only the information required for collaborative prediction.

Residual-Transformed Target Construction. For a training sample x_i, the active party first produces its local prediction $u_i = f_\theta(x_i^0)$, followed by $z_i^{loc} = t_\phi(u_i)$, where z_i^{loc} denotes the local prediction logits. The corresponding prediction distribution is $\hat{\mathbf{y}}_i^{loc} = \mathrm{softmax}(z_i^{loc})$. Let $\mathbf{e}_{y_i}$ denote the one-hot label vector. The prediction residual $\mathbf{e}_{y_i} - \hat{\mathbf{y}}_i^{loc}$ captures the information that has not yet been explained by the active party's local prediction. However, different classes exhibit different prediction sensitivities under softmax cross-entropy loss. Directly using the prediction residual treats all class directions equally and ignores the optimization geometry of the loss landscape. To obtain a residual target consistent with the local optimization objective, we employ the Hessian diagonal approximation of the softmax cross-entropy loss to normalize the prediction residual,

$$\bar{\mathbf{c}}_i = (\mathbf{e}_{y_i} - \hat{\mathbf{y}}_i^{loc}) \oslash (\hat{\mathbf{y}}_i^{loc} \odot (1 - \hat{\mathbf{y}}_i^{loc}) + \epsilon)$$

where $\odot$ and $\oslash$ denote element-wise multiplication and division respectively, and $\epsilon > 0$ is a numerical stability constant. The obtained vector is a Newton-style residual representation indicating how the current prediction should be adjusted. Since softmax predictions are invariant to adding the same constant to every logit, only relative class differences contribute to classification. Therefore, we remove the global offset by subtracting the mean value, $\mathbf{c}_i = \bar{\mathbf{c}}_i - \frac{1}{C}(\mathbf{1}^\top \bar{\mathbf{c}}_i)\mathbf{1}$. After centering, the residual-transformed target satisfies $\mathbf{1}^\top \mathbf{c}_i = 0$ indicating that only the relative residual information across classes is preserved.

Sample-aware Residual Alignment. Samples with higher prediction uncertainty generally require more complementary information from passive parties to improve collaborative learning. Therefore, we assign larger optimization weights to uncertain samples according to the entropy of the active party's prediction. The sample weight is defined as

$$w_i = \frac{-\sum_{c=1}^{C} \hat{y}_{i,c}^{loc} \log \hat{y}_{i,c}^{loc}}{\sum_{j \in \mathcal{B}} \left(-\sum_{c=1}^{C} \hat{y}_{j,c}^{loc} \log \hat{y}_{j,c}^{loc} \right)}$$

where $\hat{y}_i^{loc}$ denotes the prediction distribution produced by the active party. Samples with higher prediction entropy receive larger optimization weights, encouraging passive parties to provide more residual information for samples that remain difficult for the active party to classify.

For each passive party, local representations are extracted by the bottom model, $v_i^k = h_{\psi_k}(x_i^k)$, and aggregated by

$$r_i^{rta} = \frac{\sum_{k \in \mathcal{K}} \lambda_k v_i^k}{\sum_{k \in \mathcal{K}} \lambda_k}, \qquad \lambda_k \geq 0, \quad \sum_{k=1}^{K} \lambda_k = 1$$

The aggregated representation is then fed into the shared prediction model, $z_i^{rta} = t_\phi(u_i + \eta r_i^{rta}) - z_i^{loc}$, which represents the information contributed by passive parties to the collaborative prediction. Rather than directly approximating the original supervision, passive parties are optimized to match the residual supervision target through the weighted residual alignment loss $\mathcal{L}_{rta} = \sum_{i \in \mathcal{B}} w_i \| z_i^{rta} - \mathbf{c}_i \|_2^2 / \sum_{i \in \mathcal{B}} w_i$. Meanwhile, the active party updates its local model using the standard cross-entropy objective $\mathcal{L}_{loc} = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \mathcal{L}_{CE}(z_i^{loc}, y_i)$. The active party is optimized by the local classification objective, while passive parties are guided by the residual supervision objective. The overall objective is $\mathcal{L} = \mathcal{L}_{loc} + \gamma \mathcal{L}_{rta}$, where $\gamma > 0$ balances the two objectives. During training, the shared top model is jointly optimized by both objectives. The local classification objective updates the active bottom model together with the shared top model, while the RTA objective updates the passive bottom models, the aggregation module, and the shared top model. Consequently, the active party maintains strong local discriminative capability, whereas passive parties are guided to align their outputs with residual-transformed targets instead of directly learning complete label semantics.

4.2 Subspace-Structured Gradient Perturbation

While RTA restructures passive parties' supervision, the embedding gradients exchanged during training may still reveal class-discriminative information that facilitates backdoor learning. To ad-

dress this issue, we propose SSGP, which selectively suppresses class-discriminative gradient components while preserving task-related gradient information. Specifically, SSGP identifies the dominant subspace capturing class-wise gradient variations and injects structured perturbations within a limited perturbation budget. By perturbing only these discriminative directions, SSGP reduces exploitable gradient feedback while largely preserving collaborative learning performance.

Class-wise Gradient Statistics and Subspace Construction. For the k-th passive party, the active party computes the gradient with respect to the embedding v_i^k based on the objective: $g_i^k = \frac{\partial \mathcal{L}_{rta}}{\partial v_i^k} \in \mathbb{R}^{d_k}$, where d_k is the dimension of the embedding generated by the k-th passive party. Given a mini-batch $\mathcal{B}$, samples are divided according to their labels. For class c, the corresponding subset is $\mathcal{B}_c = \{i \in \mathcal{B} : y_i = c\}$ and the class occurrence probability is $\pi_c = \frac{|\mathcal{B}_c|}{|\mathcal{B}|}$. The set of classes appearing in the current batch is $\mathcal{C}_\mathcal{B} = \{c : |\mathcal{B}_c| > 0\}$. The class-wise mean gradient and the batch mean gradient are computed as $\mu_c^k = \frac{1}{|\mathcal{B}_c|} \sum_{i \in \mathcal{B}_c} g_i^k, \bar{\mu}^k = \sum_{c \in \mathcal{C}_\mathcal{B}} \pi_c \mu_c^k$. The difference between μ_c^k and $\bar{\mu}^k$ reflects the class-dependent component of embedding gradients. Therefore, we construct the class variation matrix $M_k = [\sqrt{\pi_c}(\mu_c^k - \bar{\mu}^k)]_{c \in \mathcal{C}_\mathcal{B}}$. Where $M_k \in \mathbb{R}^{d_k \times |\mathcal{C}_\mathcal{B}|}$. The weighting factor $\sqrt{\pi_c}$ transforms the contribution of each class into a covariance-like form, so that frequent classes contribute proportionally when measuring the overall gradient variation. We then perform singular value decomposition on M_k: $M_k = U_k S_k V_k^T$, where U_k contains orthogonal directions of gradient variation. The left singular vectors corresponding to singular values larger than the threshold τ_{svd} form the class-dependent gradient subspace $U_k^r \in \mathbb{R}^{d_k \times r_k}$, which captures the principal directions of class-wise gradient variation and serves as the primary perturbation region.

Subspace-Structured Gradient Perturbation. Instead of injecting isotropic Gaussian noise, we construct an adaptive perturbation covariance according to the extracted class-dependent gradient subspace:

$$\Sigma_{k,c} = U_k^r \operatorname{diag}(\alpha_{k,c})(U_k^r)^T + \beta_{k,c}(I - U_k^r (U_k^r)^T)$$

Where I is the identity matrix, $\alpha_{k,c} \in \mathbb{R}_+^{r_k}$ controls the perturbation strength along the class-dependent subspace, while $\beta_{k,c} \geq 0$ controls the orthogonal complement. If perturbation is restricted to the class-dependent subspace, we set $\beta_{k,c} = 0$. For each class, the original embedding gradients are modeled as a Gaussian distribution $g_i^k | y_i = c \sim \mathcal{N}(\mu_c^k, s_{k,c}^2 I)$. Where $s_{k,c}^2$ represents the intra-class gradient variance: $s_{k,c}^2 = \frac{1}{d_k |\mathcal{B}_c|} \sum_{i \in \mathcal{B}_c} \|g_i^k - \mu_c^k\|_2^2$. After adding Gaussian perturbation $\xi_i^k \sim \mathcal{N}(0, \Sigma_{k,c})$, the perturbed gradient distribution $\widetilde{g}_i^k \sim \mathcal{N}(\mu_c^k, C_{k,c})$. Where $C_{k,c} = (s_{k,c}^2 + \epsilon) I + \Sigma_{k,c}$ and $\epsilon > 0$ is a numerical stability term. The perturbation covariance is optimized by minimizing inter-class distribution discrepancy. Specifically, we use the weighted pairwise symmetric KL divergence:

$$\mathcal{J}_k = \sum_{a<b} \pi_a \pi_b D_{SKL}(\mathcal{N}(\mu_a^k, C_{k,a}), \mathcal{N}(\mu_b^k, C_{k,b}))$$

Where $D_{\text{SKL}}(P, Q) = \frac{1}{2}[D_{\text{KL}}(P\|Q) + D_{\text{KL}}(Q\|P)]$. A smaller $\mathcal{J}_k$ indicates that gradients from different classes become harder to distinguish, thereby reducing the class-specific information exploitable by malicious participants. To avoid excessive perturbation that damages the main task, we introduce an adaptive perturbation budget $\Gamma_k = \rho \operatorname{tr}(M_k M_k^T)$. Where $\rho > 0$ is a budget scale hyperparameter, and $\operatorname{tr}(M_k M_k^\top) = \sum_{c \in \mathcal{C}_\mathcal{B}} \pi_c \left\|\mu_c^k - \bar{\mu}^k\right\|_2^2$ measures the overall magnitude of inter-class gradient variation. When class gradients are highly separated, a larger perturbation budget is allowed; otherwise, the mechanism applies weaker perturbations. The optimal perturbation strength is obtained by solving:

$$\{\alpha_{k,c}^\star, \beta_{k,c}^\star\}_{c \in \mathcal{C}_\mathcal{B}} = \arg \min_{\{\alpha_{k,c}, \beta_{k,c}\}} \mathcal{J}_k \qquad \text{s.t.} \quad \sum_{c \in \mathcal{C}_\mathcal{B}} \pi_c \operatorname{tr}(\Sigma_{k,c}) \leq \Gamma_k, \alpha_{k,c} \geq 0, \beta_{k,c} \geq 0$$

Here, $\alpha_{k,c}^\star$ and $\beta_{k,c}^\star$ are the optimal perturbation strengths. The optimization determines how much perturbation should be allocated to each class and each gradient direction. Finally, the active party samples an independent Gaussian perturbation with zero mean, $\xi_i^k \sim \mathcal{N}(0, \Sigma_{k,y_i}^\star)$ and replaces the original embedding gradient: $\widetilde{g}_i^k = g_i^k + \xi_i^k$. Passive parties update their bottom models using the perturbed gradients. The perturbation is periodically updated according to gradient statistics, allowing it to adapt to evolving optimization dynamics without requiring assumptions about attackers or access to private features and local models.

5 EXPERIMENTS

We evaluate ResSub in terms of defense performance, module effectiveness, and robustness through the following research questions. **RQ1**: Can ResSub effectively defend against representative VFL backdoor attacks while preserving clean-task performance across different datasets and party settings? **RQ2**: How do RTA and SSGP contribute to the overall defense, and does SSGP outperform unstructured gradient perturbation? **RQ3**: How do the poisoning rate and perturbation budget affect the performance of ResSub?

5.1 EXPERIMENTAL SETUP

Datasets. We evaluate ResSub on four datasets covering multimodal classification, human activity recognition, phishing website detection, and image classification. 1) *NUSWIDE* Chua et al. (2009): a multimodal dataset with five classes; in the two-party setting, 1,000-dim text features and 634-dim image features are held by separate parties. 2) *UCIHAR* Anguita et al. (2013): a human activity recognition dataset with six activity classes. 3) *PHISHING* Mohammad & McCluskey (2012): a binary phishing website detection dataset with 30-dim web features. 4) *CIFAR10* Krizhevsky et al. (2009): an image classification dataset with ten natural image classes. All datasets follow the VFL setting, where labels are held by the active party. Except for the two-party NUSWIDE setting, features are evenly partitioned by dimension across two or four parties.

Attacks. We evaluate ResSub against two representative VFL backdoor attacks targeting different learning feedback. LRB Zou et al. (2022) exploits supervision feedback through label replacement and gradient manipulation, whereas LFBA Shen et al. (2025) exploits class-discriminative embedding gradients. In the four-party setting, LFBA is launched by one passive party, while LRB is jointly launched by three passive parties.

Defenses. We compare ResSub with No Defense, GC, DiscreteSGD Fu et al. (2022), FILTER Hu et al. (2026), BTI-DBF Xu et al. (2024), ANP Wu & Wang (2021), and LapNoise Fu et al. (2022). LapNoise serves as an unstructured perturbation baseline. RTA and SSGP denote the corresponding ablation variants.

Metrics. We evaluate clean-task utility using benign accuracy (BA) and backdoor effectiveness using attack success rate (ASR) together with the peak ASR during training (Peak ASR). To jointly measure utility and defense effectiveness, we define the Backdoor Defense Score (BDS) as the harmonic mean of normalized BA and attack resistance:

$$\mathrm{BDS} = \frac{2\min\left(1, \frac{\mathrm{BA}}{\mathrm{BA}_{\mathrm{benign}}}\right)(1-\mathrm{ASR})}{\min\left(1, \frac{\mathrm{BA}}{\mathrm{BA}_{\mathrm{benign}}}\right) + (1-\mathrm{ASR})}$$

Here, $\mathrm{BA}_{\mathrm{benign}}$ denotes the BA without attack or defense. The normalized BA is clipped at 1 to avoid rewarding performance beyond the benign model. The harmonic mean encourages a balanced trade-off between clean-task utility and backdoor resistance. All metrics are reported as percentages and averaged over five runs.

5.2 EXPERIMENTAL RESULTS

Overall defense performance. Tables 1 and 4 summarize defense performance across different datasets, attacks, and party settings. Overall, ResSub achieves the highest BDS (92.3%) while maintaining competitive clean-task performance. Compared with No Defense, ResSub reduces the average ASR from 89.7% to 8.3% while maintaining comparable BA (84% vs. 83.7%), demonstrating that it effectively suppresses backdoor attacks without sacrificing model utility. Across all 16 settings, ResSub achieves the best or second-best BDS, demonstrating consistently strong performance rather than superiority limited to a few specific scenarios. Existing defenses exhibit much larger performance variations across datasets. GC performs well on low-dimensional PHISHING but degrades on CIFAR10, while DiscreteSGD performs best on UCIHAR but sacrifices BA on NUSWIDE. Such observations suggest that defenses relying on a single mechanism are more sensitive to data characteristics and collaboration patterns. By simultaneously regulating supervision feedback and gradient

Table 1: Defense performance (%). Best and second-best BDS are in bold and underline.

Dataset	Attack → / Defense ↓	LFBA						LRB					
		2 parties			4 parties			2 parties			4 parties		
		BA	ASR	BDS	BA	ASR	BDS	BA	ASR	BDS	BA	ASR	BDS
CIFAR10	No Defense	79.0	99.5	1.0	64.0	80.7	31.4	65.1	95.4	8.7	53.3	99.1	1.8
	GC	70.5	78.1	35.1	62.2	89.8	18.2	73.1	6.3	**92.9**	57.5	89.0	19.2
	DiscreteSGD	72.9	2.7	**94.5**	57.6	5.6	84.4	72.6	5.6	**92.9**	57.5	6.3	**84.0**
	BTI-DBF	79.0	99.5	1.0	64.0	80.7	31.4	65.1	95.4	8.7	53.3	99.1	1.8
	ANP	78.2	99.5	0.9	65.1	66.8	47.9	69.9	51.8	62.3	53.2	38.1	65.9
	FILTER	71.7	57.7	57.6	68.6	7.1	**91.9**	72.7	94.8	9.8	57.5	97.4	5.0
	LapNoise	64.6	4.2	88.0	47.6	6.7	75.3	62.5	7.1	85.3	13.5	68.5	22.8
	ResSub	72.2	3.5	93.7	64.4	5.0	89.9	70.1	5.7	91.3	55.7	6.1	82.6
NUSWIDE	No Defense	88.6	99.9	0.2	87.3	83.7	28.0	88.3	85.8	24.9	87.9	99.6	0.8
	GC	88.1	26.8	84.5	87.5	5.4	**96.7**	88.2	82.9	29.2	87.7	98.8	2.3
	DiscreteSGD	75.4	35.9	73.5	63.0	28.2	71.5	86.1	64.4	52.3	65.9	19.7	77.3
	BTI-DBF	88.3	99.6	0.8	87.3	83.7	28.0	88.3	85.8	24.9	87.9	99.6	0.8
	ANP	88.6	99.8	0.4	86.9	81.1	31.8	88.5	97.0	5.8	87.1	99.8	0.3
	FILTER	86.3	94.9	9.6	71.2	72.7	40.8	86.2	82.4	29.9	71.1	95.5	8.5
	LapNoise	81.6	18.6	86.9	74.8	11.8	86.4	82.0	49.4	65.7	79.5	25.1	81.7
	ResSub	86.5	4.8	**97.0**	84.8	5.9	95.0	86.6	17.1	**90.2**	83.8	17.2	**88.4**
PHISHING	No Defense	96.3	95.4	8.8	95.7	87.7	21.8	92.0	76.3	37.8	93.8	88.5	20.5
	GC	98.1	1.9	**99.0**	98.0	1.7	**99.0**	96.7	37.8	76.3	92.9	85.5	25.2
	DiscreteSGD	92.8	7.7	93.4	60.7	68.8	41.5	93.5	12.0	**91.5**	70.2	44.8	62.3
	BTI-DBF	96.3	95.4	8.8	95.3	52.1	64.1	92.0	76.3	37.8	93.8	88.5	20.5
	ANP	97.1	98.9	2.1	95.7	81.3	31.4	93.0	22.0	85.6	94.2	95.4	8.7
	FILTER	96.5	90.7	17.0	62.1	38.5	62.4	90.2	16.4	87.5	63.3	5.5	76.7
	LapNoise	95.1	4.6	96.1	57.4	73.4	36.6	91.7	20.8	85.7	84.0	73.4	40.6
	ResSub	94.5	2.9	96.7	95.2	2.8	97.1	93.0	14.5	89.9	91.9	34.2	**77.3**
UCIHAR	No Defense	87.3	98.9	2.2	92.9	89.7	18.7	91.7	72.6	42.8	76.5	83.1	28.0
	GC	93.9	9.0	95.1	89.5	8.6	93.5	82.2	67.4	47.5	83.0	74.6	39.4
	DiscreteSGD	93.0	4.1	**97.3**	88.7	4.9	94.9	95.2	0.3	**99.8**	90.0	1.0	**97.5**
	BTI-DBF	87.3	98.9	2.2	92.9	89.7	18.7	91.7	72.6	42.8	76.5	83.1	28.0
	ANP	85.6	44.7	68.8	91.3	47.6	68.2	72.0	57.3	54.8	59.7	0.0	77.8
	FILTER	92.3	4.2	96.9	92.1	18.8	89.0	87.9	4.2	94.5	75.9	10.6	85.0
	LapNoise	88.5	5.5	94.3	87.9	6.3	93.7	93.1	2.7	98.1	88.9	2.1	96.4
	ResSub	93.4	4.7	**97.3**	89.2	5.2	**95.0**	93.0	0.7	99.0	89.5	2.7	96.4

Table 2: Ablation study under LFBA on NUSWIDE with two parties (%).

Method	BA↑	ASR↓	Peak ASR↓	BDS↑
No Defense	88.6	99.9	100.0	0.2
LapNoise	81.6	18.6	18.6	86.9
RTA	88.2	26.9	26.9	84.5
SSGP	84.4	10.8	10.8	92.7
ResSub	86.5	4.8	5.3	**97.0**

Table 3: Sensitivity analysis of ρ under LFBA on NUSWIDE with two parties (%).

ρ	BA↑	ASR↓	BDS↑
0.1	86.1	67.3	49.2
1	86.5	4.8	97.0
10	86.2	4.2	**97.2**
100	84.5	12.6	91.8
1000	78.5	20.1	84.5

feedback, ResSub maintains a more stable balance between utility and robustness across multimodal, image, activity recognition, and tabular datasets.

More participating parties make defense more challenging because attackers collectively control a larger proportion of feature information and collaborative learning becomes increasingly dependent on cross-party interactions. This trend is evident for most baselines, whose BDS drops substantially under four-party settings. In contrast, ResSub consistently maintains high BDS and low ASR across both two-party and four-party scenarios. A particularly challenging case is PHISHING under LRB

Table 4: Average defense performance (%) over 16 settings.

Defense	BA↑	ASR↓	BDS↑
No Defense	83.7	89.7	17.3
GC	84.3	47.7	59.6
DiscreteSGD	77.2	19.5	81.8
BTI-DBF	83.7	87.5	20.0
ANP	81.6	67.6	38.3
FILTER	77.9	49.5	53.9
LapNoise	74.5	23.8	77.1
ResSub	84.0	8.3	**92.3**

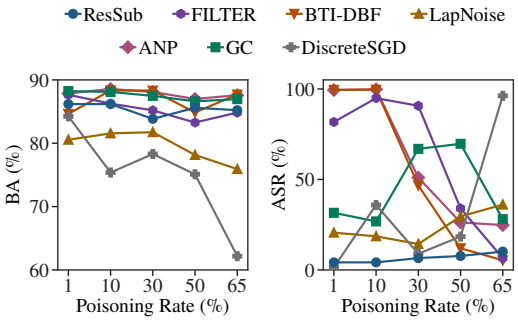

Figure 2: Impact of poisoning rate.

with four parties, where 23 of the 30 feature dimensions are controlled by malicious parties. Even in this extreme setting, ResSub achieves the highest BDS by preserving higher BA and reducing ASR, demonstrating a superior trade-off between model utility and robustness.

Module ablation. Table 2 evaluates the individual contributions of RTA and SSGP. Using RTA alone reduces ASR from 99.9% to 26.9% with only a 0.4 percentage point reduction in BA, demonstrating that restructuring supervision effectively weakens the learning of backdoor-related signals while preserving collaborative prediction. SSGP alone further reduces ASR to 10.8%. Compared with unstructured LapNoise, SSGP achieves both higher BA (+2.8 percentage points) and lower ASR (-7.8 percentage points), indicating that structured perturbation is substantially more effective than indiscriminate noise injection. Combining RTA and SSGP further reduces ASR to 4.8%, limits its Peak ASR to 5.3%, and achieves the highest BDS of 97%, confirming that the two components provide complementary benefits.

Impact of poisoning rate. Fig. 2 evaluates the robustness of ResSub under different poisoning rates. As the poisoning rate increases from 1% to 65%, ResSub consistently maintains ASR below 10% while preserving BA around 87%, indicating stable defense performance across different attack strengths. In comparison, FILTER leaves relatively high residual ASR under low and moderate poisoning rates, whereas DiscreteSGD and BTI-DBF exhibit larger fluctuations in either BA or ASR. These results demonstrate that ResSub provides a more stable trade-off between utility and backdoor defense across a wide range of poisoning rates.

Sensitivity analysis on ρ. Table 3 evaluates the influence of the perturbation budget ρ. As ρ increases from 0.1 to 10, ASR decreases substantially from 67.3% to 4.2%, while BA remains nearly unchanged (86.1%–86.2%), resulting in the highest BDS of 97.2%. This indicates that a moderate perturbation budget effectively suppresses exploitable gradient feedback without compromising clean-task performance. However, excessively large perturbation budgets interfere with normal optimization. When $\rho = 1000$, BA decreases to 78.5% and ASR rises to 20.1%. These results suggest that an appropriate perturbation budget is essential for balancing clean-task performance and defense effectiveness.

6 Conclusion

This paper proposes ResSub, which jointly regulates supervision feedback through RTA and gradient feedback through SSGP to suppress backdoor formation in VFL. Extensive experiments demonstrate that ResSub consistently improves defense effectiveness while maintaining competitive model utility. Ablation studies further validate the effectiveness of each component and their complementary benefits.

Limitations. ResSub has two main limitations: our evaluation does not consider adaptive attackers aware of the defense, and the theoretical robustness and convergence properties of SSGP remain to be investigated.

AI USE STATEMENT

Generative AI tools were used to assist with language polishing, LaTeX formatting and limited code development. All AI-assisted content was carefully reviewed and revised by the authors. AI-assisted code was manually reviewed, tested, and validated by the authors before being used in the experiments. The technical claims, methodology, experimental results, and conclusions presented in this work were independently verified by the authors. We take full responsibility for the final content of this paper, including any text or artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work studies backdoor attacks and defenses in Vertical Federated Learning for the purpose of improving the security and robustness of collaborative learning systems. All experiments are conducted in controlled research environments using publicly available benchmark datasets, and no real-world systems are attacked. The attack techniques considered in this work are existing research methods and are used solely for evaluating the effectiveness of the proposed defense. We acknowledge the dual-use nature of security research and encourage responsible use of the methods and findings presented in this paper.

REPRODUCIBILITY STATEMENT

We provide the methodological details of ResSub in Section 4 and describe the datasets, attack settings, defense baselines, and evaluation metrics in Section 5.1. Additional implementation details, hyperparameter settings, data preprocessing procedures, and attack configurations are provided in the appendix and supplementary material. An anonymized implementation will also be provided as supplementary material. These materials are intended to support independent reproduction of the reported experimental results.

REFERENCES

D. Anguita, Alessandro Ghio, L. Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *The European Symposium on Artificial Neural Networks*, 2013. URL https://api.semanticscholar.org/CorpusID:6975432.

Yang Bai, Gaojie Xing, Hongyan Wu, Zhihong Rao, Chuan Ma, Shiping Wang, Xiaolei Liu, Yimin Zhou, Jiajia Tang, Kaijun Huang, et al. Backdoor attack and defense on deep learning: A survey. *IEEE Transactions on Computational Social Systems*, 12(1):404–434, 2024.

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)*, pp. 2743–2760, Anaheim, CA, August 2023. USENIX Association. ISBN 978-1-939133-37-3. URL https://www.usenix.org/conference/usenixsecurity23/presentation/bai.

Peng Chen, Haolong Xiang, Xin Du, Xiaolong Xu, Xuhao Jiang, Zhihui Lu, Jirui Yang, Qiang Duan, and Wanchun Dou. Universal backdoor defense via label consistency in vertical federated learning. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 4743–4751, 2025.

Yong Cheng, Yang Liu, Tianjian Chen, and Qiang Yang. Federated learning for privacy-preserving ai. *Communications of the ACM*, 63(12):33–36, 2020.

Yungi Cho, Woorim Han, Miseon Yu, Younghan Lee, Ho Bae, and Yunheung Paek. Vflip: A backdoor defense for vertical federated learning via identification and purification. In *European Symposium on Research in Computer Security*, pp. 291–312. Springer, 2024.

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*, pp. 1–9, 2009.

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)*, pp. 1397–1414, 2022.

Dashan Gao, Sheng Wan, Lixin Fan, Xin Yao, and Qiang Yang. Complementary knowledge distillation for robust and privacy-preserving model serving in vertical federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19832–19839, 2024a.

Dashan Gao, Sheng Wan, Hanlin Gu, Lixin Fan, Xin Yao, and Qiang Yang. Label privacy source coding in vertical federated learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 313–331. Springer, 2024b.

Yuhao Gu and Yuebin Bai. Lr-ba: Backdoor attack against vertical federated learning using local latent representations. *Computers & Security*, 129:103193, 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*, 19:748–763, 2023.

Zhanyi Hu, Cen Chen, and Yanhao Wang. Filter: A framework for defending against backdoor attacks in vertical federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 35490–35499, 2026.

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

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. URL https://openreview.net/forum?id=cOtBRgsf2fO.

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*, 36(7):3615–3634, 2024.

Rami Mohammad and Lee McCluskey. Phishing Websites. UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C51W2X.

Mohammad Naseri, Yufei Han, and Emiliano De Cristofaro. Badvfl: Backdoor attacks in vertical federated learning. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 2013–2028. IEEE, 2024.

Pengyu Qiu, Xuhong Zhang, Shouling Ji, Tianyu Du, Yuwen Pu, Jun Zhou, and Ting Wang. Your labels are selling you out: Relation leaks in vertical federated learning. *IEEE Transactions on Dependable and Secure Computing*, 20(5):3653–3668, 2022.

Wei Shen, Wenke Huang, Guancheng Wan, and Mang Ye. Label-free backdoor attacks in vertical federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 20389–20397, 2025.

Jiankai Sun, Xin Yang, Yuanshun Yao, and Chong Wang. Label leakage and protection from forward embedding in vertical federated learning, 2022. URL https://arxiv.org/abs/2203.01451.

Dongxian Wu and Yisen Wang. Adversarial neuron pruning purifies backdoored deep models. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 16913–16925. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/8cbe9ce23f42628c98f80fa0fac8b19a-Paper.pdf.

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.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

Derui Zhu, Jinfu Chen, Xuebing Zhou, Weiyi Shang, Ahmed E Hassan, and Jens Grossklags. Vulnerabilities of data protection in vertical federated learning training and countermeasures. *IEEE Transactions on Information Forensics and Security*, 19:3674–3689, 2024.

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*, 10(6):1016–1027, 2022.

SUPPLEMENTARY MATERIAL FOR RESSUB: DUAL-LEVEL BACKDOOR DEFENSE VIA RESIDUAL ALIGNMENT AND SUBSPACE PERTURBATION IN VERTICAL FEDERATED LEARNING

A DETAILED ALGORITHMS OF RESSUB

Here, $\mathbf{\Sigma}_k = \{\mathbf{\Sigma}_{k,c}\}_{c\in\mathcal{C}_\mathcal{B}}$ denotes the collection of class-dependent perturbation covariances for passive party p_k. The operator $\mathrm{Perturb}$ applies $\mathbf{\Sigma}_{k,y_i}$ to the gradient of each sample $i \in \mathcal{B}$.

Algorithm 1 ResSub Training

Require: Aligned data $\mathcal{D} = \{(\boldsymbol{x}_i^0, \ldots, \boldsymbol{x}_i^K, y_i)\}$

▷ **Cold Start**

Active party P_0 trains f_θ and t_ϕ on $\{x_i^0, y_i\}$ via $\mathcal{L}_{loc}$

▷ **Federated Learning**

Passive parties $\{P_k\}_{k=1}^K$ initialize $\{\psi_k\}_{k=1}^K$

while stopping epoch not met **do**

  for each batch $\mathcal{B}$ **do**

    $\boldsymbol{u}_\mathcal{B} \leftarrow f_\theta(\boldsymbol{x}_\mathcal{B}^0)$

    P_0 computes $\boldsymbol{c}_\mathcal{B}$ and $\boldsymbol{w}_\mathcal{B}$

    for all passive party P_k **in parallel do**

      $\boldsymbol{v}_\mathcal{B}^k \leftarrow h_{\psi_k}(\boldsymbol{x}_\mathcal{B}^k)$

      P_k sends $\boldsymbol{v}_\mathcal{B}^k$ to P_0

    end for

    P_0 computes $\mathcal{L}_{loc}$, and $\mathcal{L}_{rta}$

    P_0 computes $\{\boldsymbol{g}_k\}_{k=1}^K$

    if SSGP is scheduled for update **then**

      $\{\mathbf{\Sigma}_k\}_{k=1}^K \leftarrow \mathrm{SSGP}(\{\boldsymbol{g}_k\}_{k=1}^K, \boldsymbol{y}_\mathcal{B})$

    end if

    for all passive party P_k **in parallel do**

      $\widetilde{\boldsymbol{g}}_k \leftarrow \mathrm{Perturb}(\boldsymbol{g}_k, \mathbf{\Sigma}_k)$

      P_0 sends $\widetilde{\boldsymbol{g}}_k$ to P_k

      $\psi_k \leftarrow \mathrm{Update}(\psi_k, \widetilde{\boldsymbol{g}}_k)$

    end for

    $\theta \leftarrow \mathrm{Update}(\theta, \mathcal{L}_{loc})$

    $\phi \leftarrow \mathrm{Update}(\phi, \mathcal{L}_{loc}, \mathcal{L}_{rta})$

    $\boldsymbol{\lambda} \leftarrow \mathrm{Update}(\boldsymbol{\lambda}, \mathcal{L}_{rta})$

  end for

end while

Ensure: θ, ϕ, $\{\psi_k\}_{k=1}^K$, and $\boldsymbol{\lambda}$.

B More Experimental Details

In this section, we present experimental details omitted from the main paper due to space limitations, including additional dataset information, attack configurations, model architectures, training settings, and parameter settings used for the compared defenses and ResSub.

B.1 Datasets

We evaluate ResSub on four public datasets whose data characteristics differ substantially. The active party holds the labels in every setting. Unless stated otherwise, all parties share aligned sample identifiers and hold disjoint feature subspaces.

NUSWIDE NUSWIDE is a real-world web image collection containing visual content and user-provided text tags. We retain five categories (buildings, grass, animal, water, and person) and use the 634-dimensional low-level image descriptor together with the 1,000-dimensional text-tag representation. This dataset provides a natural two-party VFL setting because the two modalities can be held by different parties. For the four-party setting, we concatenate the two modality-specific feature representations and divide the resulting features as evenly as possible among the four parties.

PHISHING PHISHING is a binary classification dataset that provides 30 features for determining whether a website is a phishing website, with 8,844 training samples and 2,211 test samples. Its low-dimensional tabular features complement the image and high-dimensional datasets used in our evaluation.

CIFAR10 CIFAR10 is an image dataset for 10 classification tasks with 50000 training samples and 10000 testing samples. It therefore tests ResSub under non-tabular inputs and a convolutional architecture, complementing the three datasets.

UCIHAR UCIHAR is a human activity recognition dataset with six classes: 'walking', 'walking upstairs', 'walking downstairs', 'sitting', 'standing', and 'laying', with 7352 training samples and 2947 testing samples.

Why these four datasets We select four datasets to evaluate whether ResSub remains effective across different data types, classification tasks, and model architectures. NUSWIDE provides a multimodal classification task involving image and text features. UCIHAR represents a high-dimensional multi-class human activity recognition task, whereas PHISHING provides a low-dimensional binary website classification task. CIFAR10 allows us to evaluate ResSub in an image classification setting using convolutional bottom models.

Together, these datasets cover multimodal data, high- and low-dimensional feature vectors, and image inputs. They further span binary and multi-class classification, multilayer perceptron and convolutional architectures. This selection allows us to examine whether ResSub maintains consistent defense effectiveness and clean-task utility across different learning settings. We additionally consider two-party and four-party configurations to evaluate its effectiveness under different collaboration scales and attacker configurations.

B.2 Defenses

We use fixed defense parameters across all experimental settings. Specifically, the retained gradient fraction of GC Fu et al. (2022) is set to 0.5, the number of discretization intervals in DiscreteSGD Fu et al. (2022) is set to $N = 6$, and the noise scale of LapNoise Fu et al. (2022) is set to 10^{-2}. The pruning threshold of ANP Wu & Wang (2021) is set to 0.9. BTI-DBF Xu et al. (2024) uses the default parameters recommended by its authors. For ResSub, we set $\rho = 1$, $\gamma = 1$, $\eta = 1.5$, and the SVD threshold $\tau_{\text{svd}} = 10^{-8}$.

For FILTER Hu et al. (2026), we adapt its training procedure to our early-stopping protocol. After early stopping, we continue training the selected model with FILTER for 10 additional epochs and perform label-shuffling backdoor unlearning during the final 2 epochs. All other FILTER hyperparameters follow the default settings in the original paper.

Table 5: Sensitivity to active-party informativeness under LFBA on two-party NUS-WIDE. "Retained" denotes the fraction of the active party's original 634 features, and "Dim." is the resulting feature dimension. Active-only Acc. is measured by a separately trained active-only classifier. All metrics other than Dim. are percentages. Lower ASR and Peak ASR are better; boldface marks the lower attack metric between RTA and ResSub.

			RTA			ResSub (RTA+SSGP)		
Retained	Dim.	Active-only Acc.↑	BA↑	ASR↓	Peak ASR↓	BA↑	ASR↓	Peak ASR↓
0.5%	3	40.8	83.2	3.3	3.3	83.1	**1.9**	**1.9**
2%	13	55.4	84.4	45.3	45.3	84.1	**4.4**	**4.4**
5%	32	67.1	84.8	19.0	19.0	85.7	**13.5**	**13.5**
10%	63	70.1	85.6	28.5	28.5	86.2	**17.2**	**17.2**
25%	158	73.6	85.8	24.8	24.8	85.9	**4.5**	**5.3**
100%	634	76.0	88.2	26.9	26.9	86.5	**4.8**	**5.3**
Average	–	–	85.3	24.6	24.6	85.2	**7.7**	**7.9**

B.3 Attacks

For LFBA Shen et al. (2025), we follow the parameter settings used in the original paper. For LRB Zou et al. (2022), we fix the gradient amplification factor at 20, while all other attack parameters follow the settings in the original paper. Within each experimental setting, the same attack configuration is used for all defense methods.

B.4 Model Architectures and Training Settings

For CIFAR10, each party uses ResNet-18 as its bottom model. For NUSWIDE, UCIHAR, and PHISHING, each party uses a four-layer MLP as its bottom model. The top model is a three-layer MLP in all settings. All models are trained using Adam with a batch size of 256 for up to 100 epochs. The learning rate is set to 0.001 for CIFAR10 and NUSWIDE and 0.003 for UCIHAR and PHISHING. We apply early stopping with a patience of 20 epochs. Within each experimental setting, the target class, trigger pattern, and other attack parameters are kept identical across all defense methods to ensure a fair comparison.

C Sensitivity to Active-Party Informativeness

Motivation. Residual-Transformed Alignment (RTA) constructs the supervision target for the passive party from the information not captured by the active party's local prediction. Its effectiveness may therefore depend on how informative the active party's local features are. We test this dependence by systematically varying the fraction of features available to the active party.

Experimental setup. We use the two-party NUS-WIDE setting under LFBA and retain $\{0.5, 2, 5, 10, 25, 100\}\%$ of the active party's 634 features, while keeping the passive party's features unchanged. We then compare RTA with the full ResSub method, which combines RTA and SSGP.

Results. As shown in Table 5, the active-only accuracy spans a broad range from 40.8% to 76.0%. RTA alone exhibits non-monotonic sensitivity to the active-party partition, with ASR ranging from 3.3% to 45.3%. In particular, when only 2% of the active features are retained and the active-only accuracy is 55.4%, RTA attains 45.3% ASR. Adding SSGP reduces ASR to 4.4%, while changing BA from 84.4% to 84.1%. Across all six retention levels, ResSub reduces ASR relative to RTA. Its average ASR is 7.7%, compared with 24.6% for RTA, whereas their average BA values remain comparable (85.2% versus 85.3%).

Interpretation and limitation. These results show that active-only accuracy is not a monotonic predictor of backdoor susceptibility: the retained feature subset and the resulting residual-learning

geometry also matter. Nevertheless, RTA can leave exploitable supervision under some weak active-party partitions, whereas SSGP consistently provides an additional reduction in ASR across all tested partitions. The remaining ASR of 13.5% and 17.2% at the 5% and 10% retention levels also indicates that the full method is not invariant to active-party informativeness. Finally, ASR measures attack effectiveness rather than label leakage directly; therefore, this study should be interpreted as a robustness stress test rather than an information-theoretic quantification of the residual target.