NDSS SYMPOSIUM/2025

Passive Inference Attacks on Split Learning via Adversarial Regularization

Xiaochen Zhu, Xinjian Luo, Yuncheng Wu, Yangfan Jiang,
Xiaokui Xiao, Beng Chin Ooi

NUS National University of Singapore

Massachusetts Institute of Technology

MOHAMED BIN ZAYED UNIVERSITY OF ARTIFICIAL INTELLIGENCE

RENMIN UNIVERSITY OF CHINA 1937 中国人民大学

Overview

- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: <u>**S**</u>imulator <u>**D**</u>ecoding with <u>**A**</u>dversarial <u>**R**</u>egularization
- Results and discussions
- Countermeasures and future work

Overview

- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: **S**imulator **D**ecoding with **A**dversarial **R**egularization
- Results and discussions
- Countermeasures and future work

Overview

- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: **S**imulator **D**ecoding with **A**dversarial **R**egularization
- Results and discussions
- Countermeasures and future work

Overview

- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: **S**imulator **D**ecoding with **A**dversarial **R**egularization
- Results and discussions
- Countermeasures and future work

Overview

- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: **S**imulator **D**ecoding with **A**dversarial **R**egularization
- Results and discussions
- Countermeasures and future work

Overview

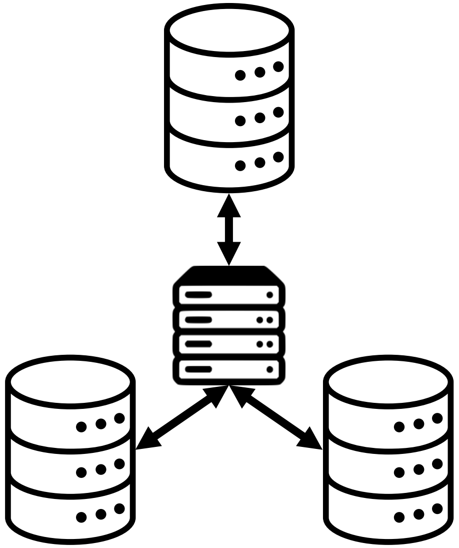
- Split learning (SL)
- Privacy vulnerabilities of split learning
- Existing attacks on SL and their limitations
- SDAR: **S**imulator **D**ecoding with **A**dversarial **R**egularization
- Results and discussions
- Countermeasures and future work

Background

Limited, biased and distributed data

Background

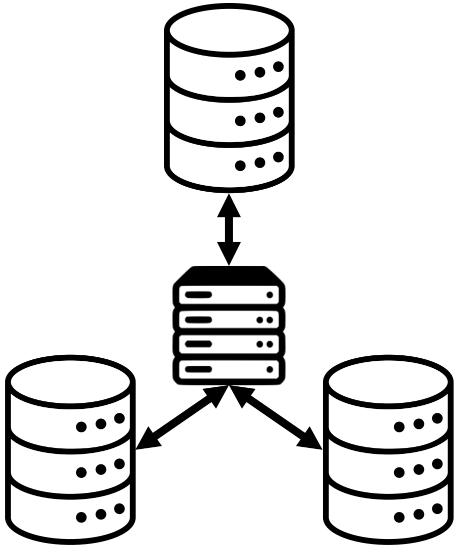
Limited, biased and distributed data



Federated Learning

Background

Limited, biased and distributed data

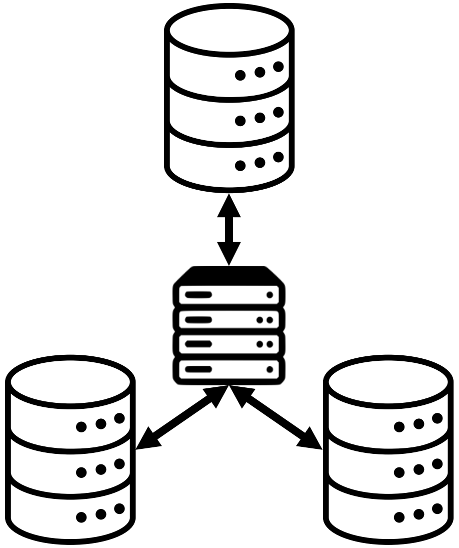


Federated Learning

Limited computational resources

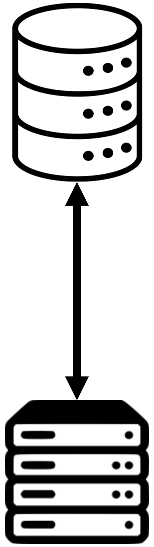
Background

Limited, biased and distributed data



Federated Learning

Limited computational resources

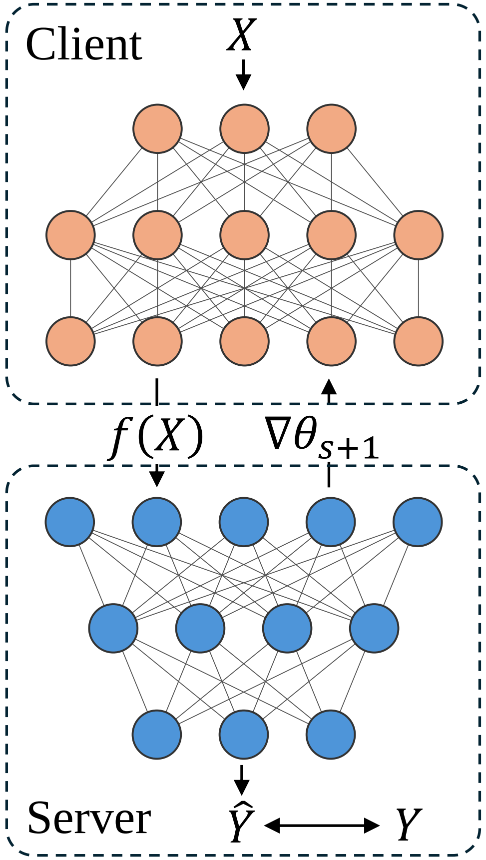


ML as a Service

Split learning

$f = L_s \circ \cdots \circ L_1$

$g = L_n \circ \cdots \circ L_{s+1}$

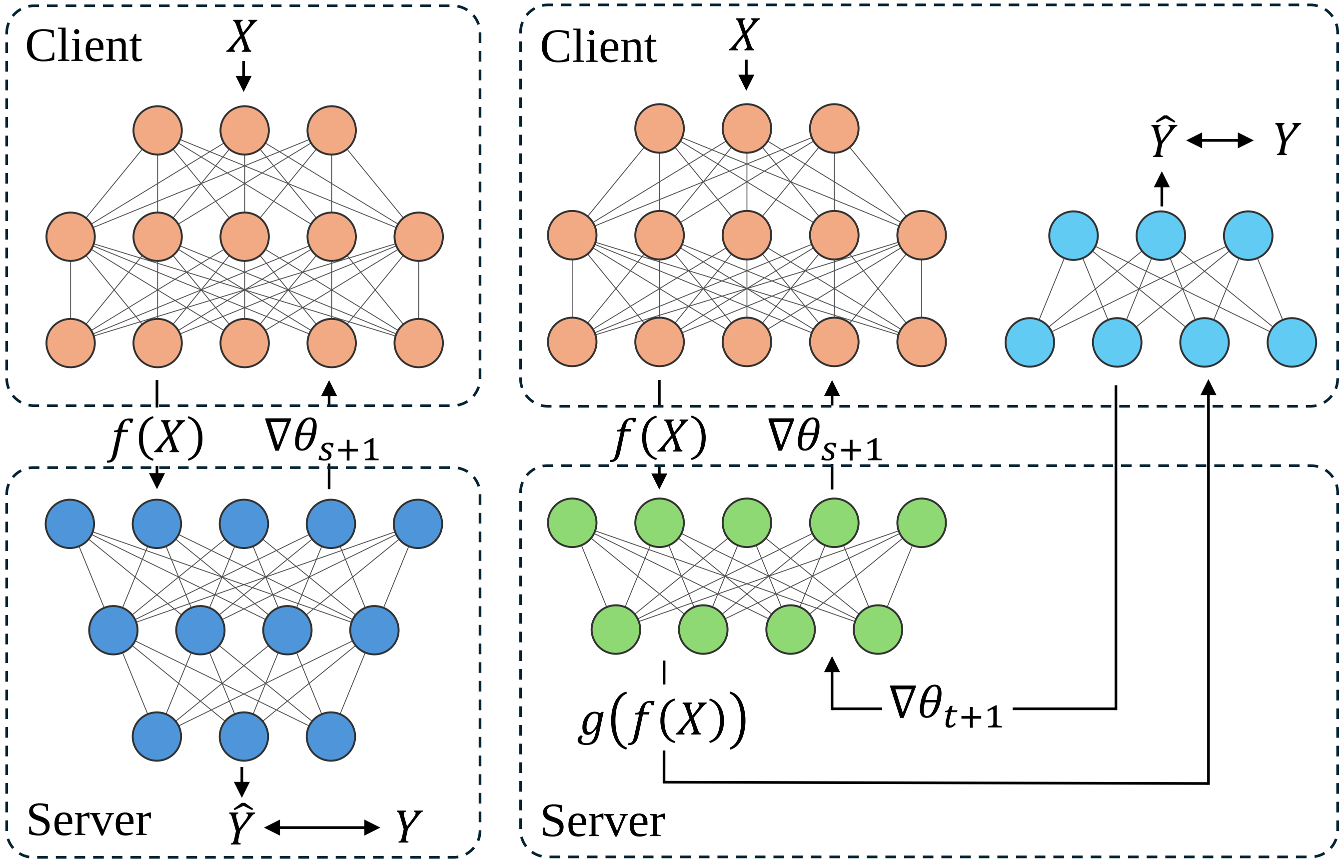


Vanilla SL
$H = g \circ f$

Split learning

$f = L_s \circ \cdots \circ L_1$

$g = L_n \circ \cdots \circ L_{s+1}$



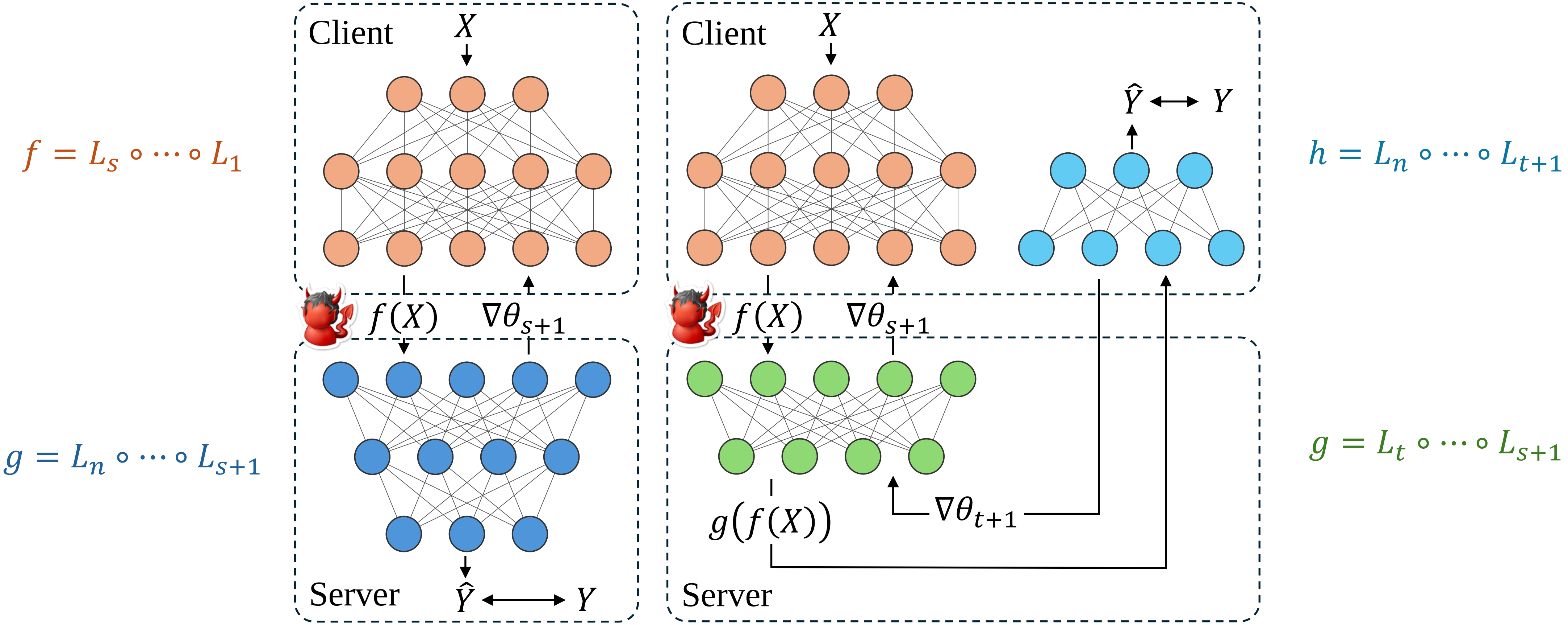
$h = L_n \circ \cdots \circ L_{t+1}$

$g = L_t \circ \cdots \circ L_{s+1}$

Vanilla SL
$H = g \circ f$

U-shaped SL
$H = h \circ g \circ f$

Split learning



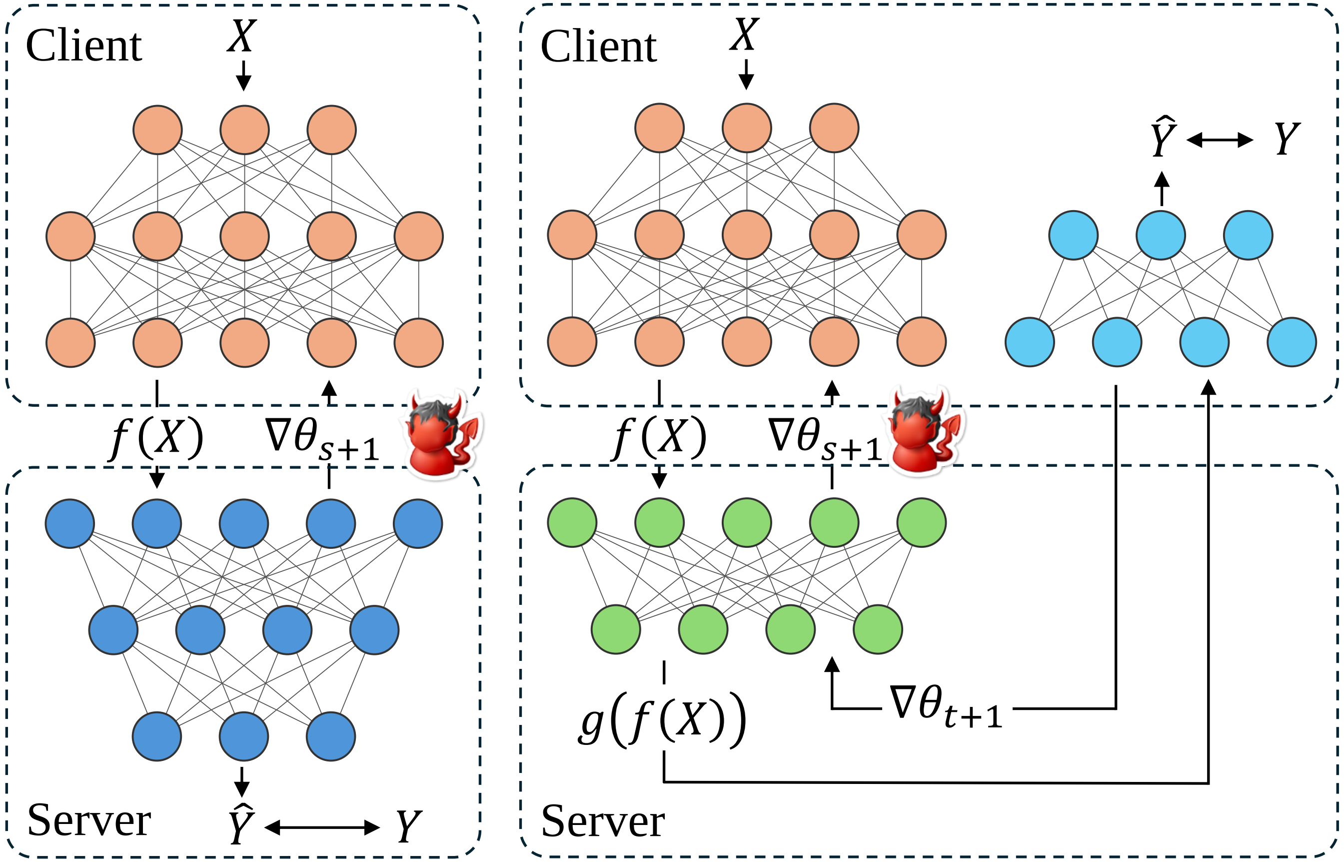
Vanilla SL
$H = g \circ f$

U-shaped SL
$H = h \circ g \circ f$

Split learning

$f = L_s \circ \cdots \circ L_1$

$g = L_n \circ \cdots \circ L_{s+1}$



$h = L_n \circ \cdots \circ L_{t+1}$

$g = L_t \circ \cdots \circ L_{s+1}$

Vanilla SL
$H = g \circ f$

U-shaped SL
$H = h \circ g \circ f$

Our contributions

Attack	Passive?	Attack features?	Attack labels?	Assume in-domain auxiliary data?	Assume knowledge of client's model?	Reconstruction quality
FSHA (CCS '21)	❌	✅	❌	Features	Not necessary	High
EXACT	✅	✅	✅	None	Architecture & weights	High
UnSplit	✅	✅	✅	None	Architecture	Low
PCAT (USENIX Sec '23)	✅	✅	✅	Features & labels	Not necessary	Medium

Our contributions

Attack	**Passive?**	**Attack features?**	**Attack labels?**	**Assume in-domain auxiliary data?**	**Assume knowledge of client's model?**	**Reconstruction quality**
FSHA (CCS '21)	❌	✅	❌	Features	Not necessary	High
EXACT	✅	✅	✅	None	Architecture & weights	High
UnSplit	✅	✅	✅	None	Architecture	Low
PCAT (USENIX Sec '23)	✅	✅	✅	Features & labels	Not necessary	Medium
SDAR (Ours)	✅	✅	✅	**Features & labels**	**Not necessary**	**High**

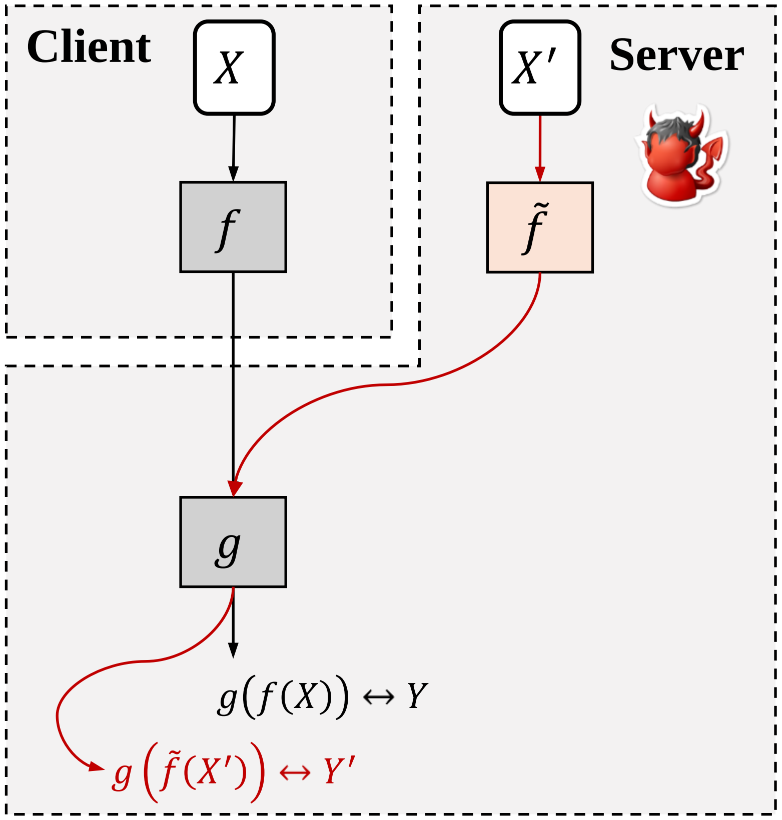
Our attack is passive (honest-but-curious server), requires no access to the client's model (white-box or black-box), and can attack both the client's features and labels with superior performance under challenging settings, with a labeled auxiliary dataset in the same domain

A naïve attempt: simulator decoding attack

The attacker (server) has labeled auxiliary data

- With extra data (X', Y'), server can train a simulator $\tilde{f}$ such that $g \circ \tilde{f}$ can classify X', i.e., minimize

$$\mathcal{L}_{\tilde{f}} = \mathrm{CrossEntropy}\left(g\left(\tilde{f}(X')\right), Y'\right)$$



A naïve attempt: simulator decoding attack

The attacker (server) has labeled auxiliary data

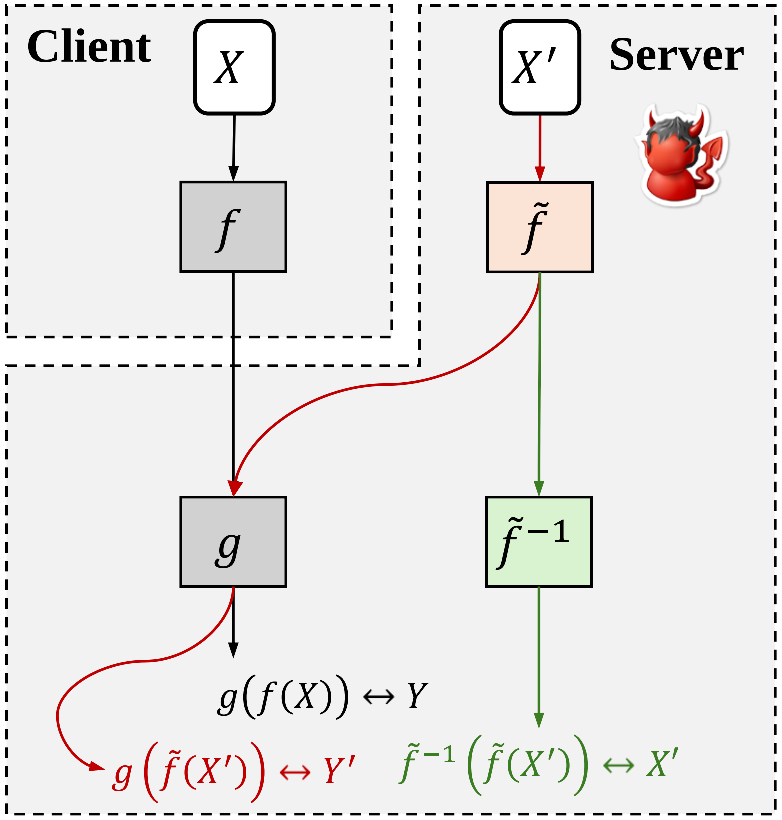
- With extra data (X', Y'), server can train a simulator $\tilde{f}$ such that $g \circ \tilde{f}$ can classify X', i.e., minimize

$$\mathcal{L}_{\tilde{f}} = \mathrm{CrossEntropy}\left(g\left(\tilde{f}(X')\right), Y'\right)$$

- With extra data (X', Y'), server can also train a decoder $\tilde{f}^{-1}$, such that $\tilde{f}^{-1}$ can decode $\tilde{f}(X')$, i.e., minimize

$$\mathcal{L}_{\tilde{f}^{-1}} = \mathrm{MSE}\left(\tilde{f}^{-1}\left(\tilde{f}(X')\right), X'\right)$$

Hopefully, $\tilde{f}$ behaves similarly to f and $\tilde{f}^{-1}$ can decode f as well.



A naïve attempt: simulator decoding attack

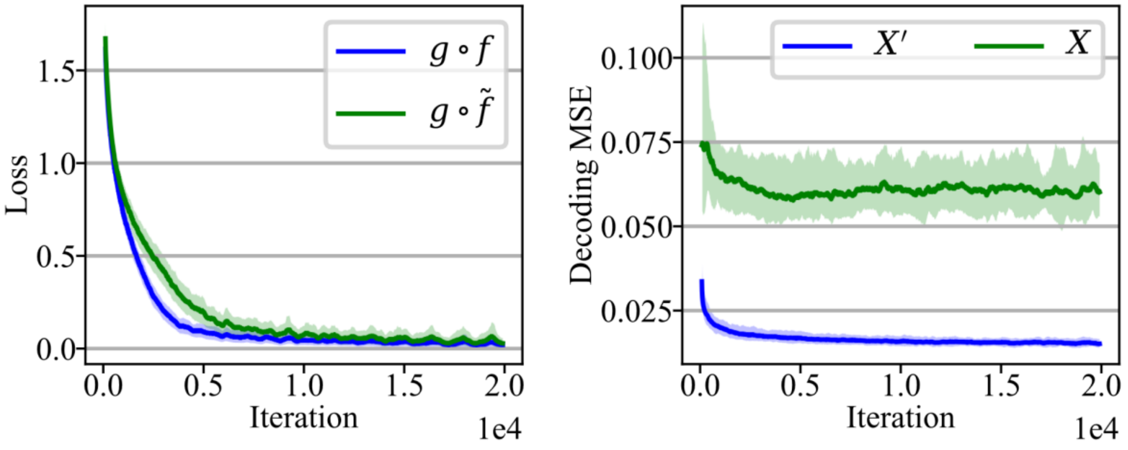
- Reconstruction results are bad

Original images

Reconstruction by naïve SDA

A naïve attempt: simulator decoding attack

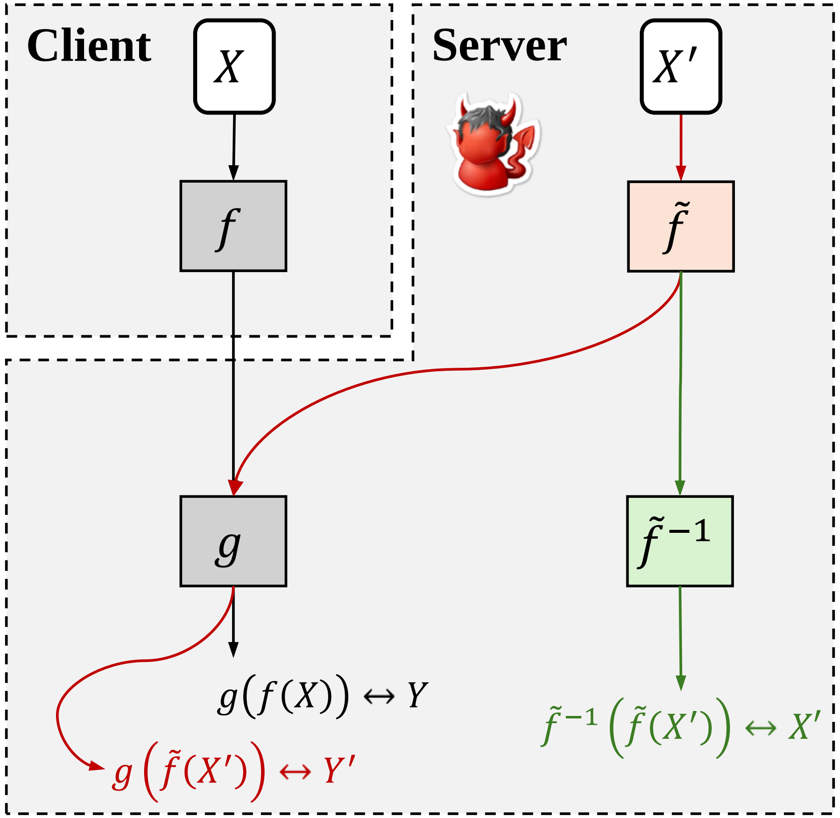
- Reconstruction results are bad
- Issue 1: The simulator $\tilde{f}$ can classify X' together with g doesn't mean it learns the same representations as client's model f.
- Issue 2: The decoder can decode $\tilde{f}(X')$ doesn't mean it can decode $f(X)$.



(b) Training loss of $g \circ f$ vs $g \circ \tilde{f}$ (c) Decoding MSE on X' vs X

Discriminator as regularizer

Issue 1: The simulator $\tilde{f}$ can classify X' doesn't mean it learns the same representations as client's model.

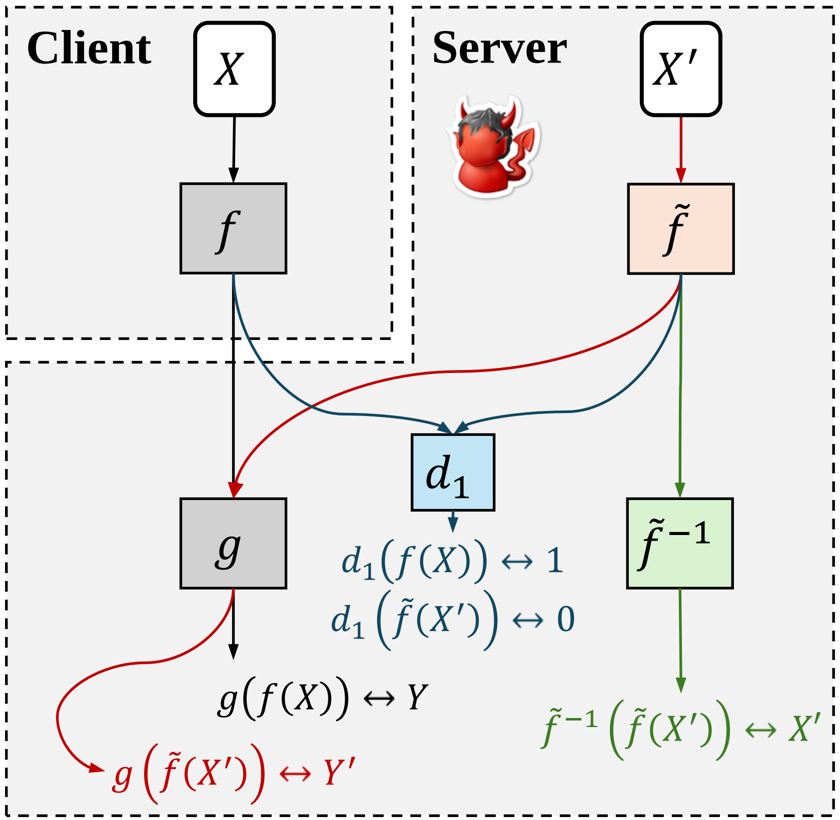


Discriminator as regularizer

Issue 1: The simulator $\tilde{f}$ can classify X' doesn't mean it learns the same representations as client's model.

- Introduce a discriminator d_1 to distinguish $f(X)$ and $\tilde{f}(X')$
- Add GAN generation loss as a regularization term to $\tilde{f}$'s loss so it is optimized to produce representations like f:

$$\mathrm{CrossEntropy}\left(g\left(\tilde{f}(X')\right), Y'\right) + \lambda_1 \mathrm{CrossEntropy}\left(d_1\left(\tilde{f}(X')\right), 1\right)$$

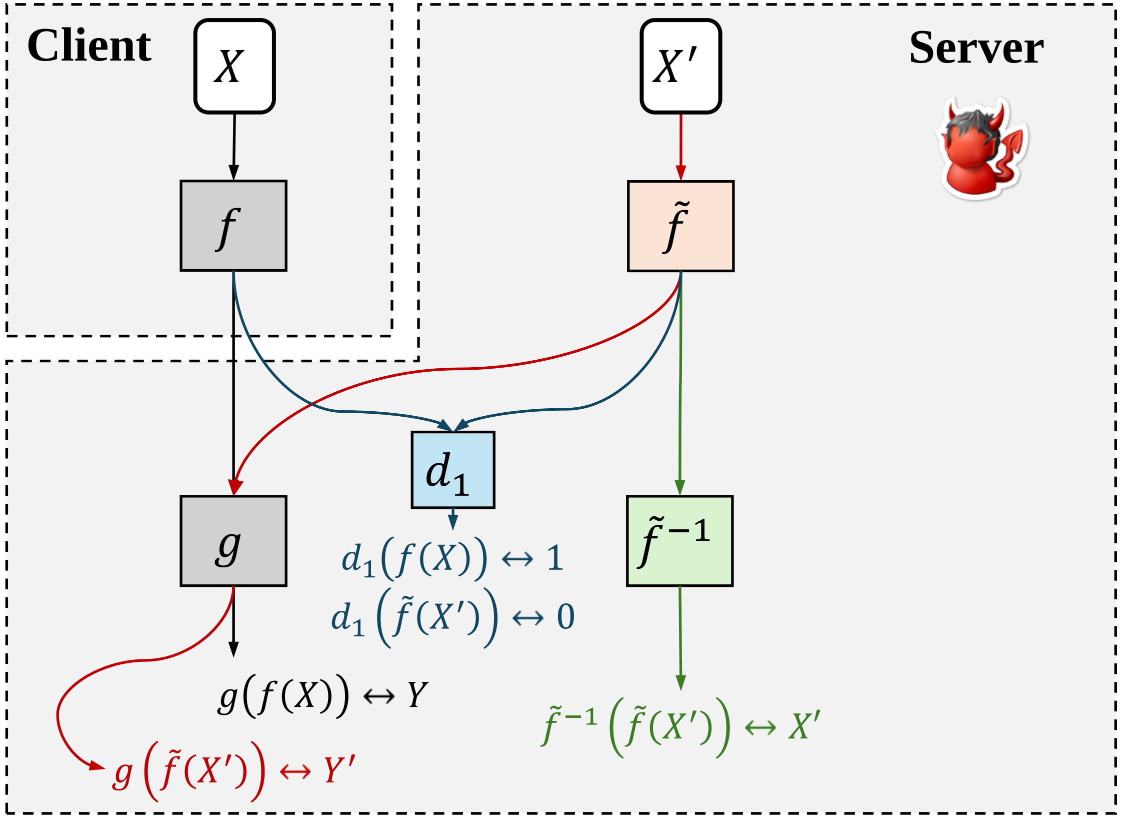


Discriminator as regularizer

Issue 2: The decoder can decode $\tilde{f}(X')$ doesn't mean it can decode $f(X)$.

Original images

Reconstruction by naïve SDA

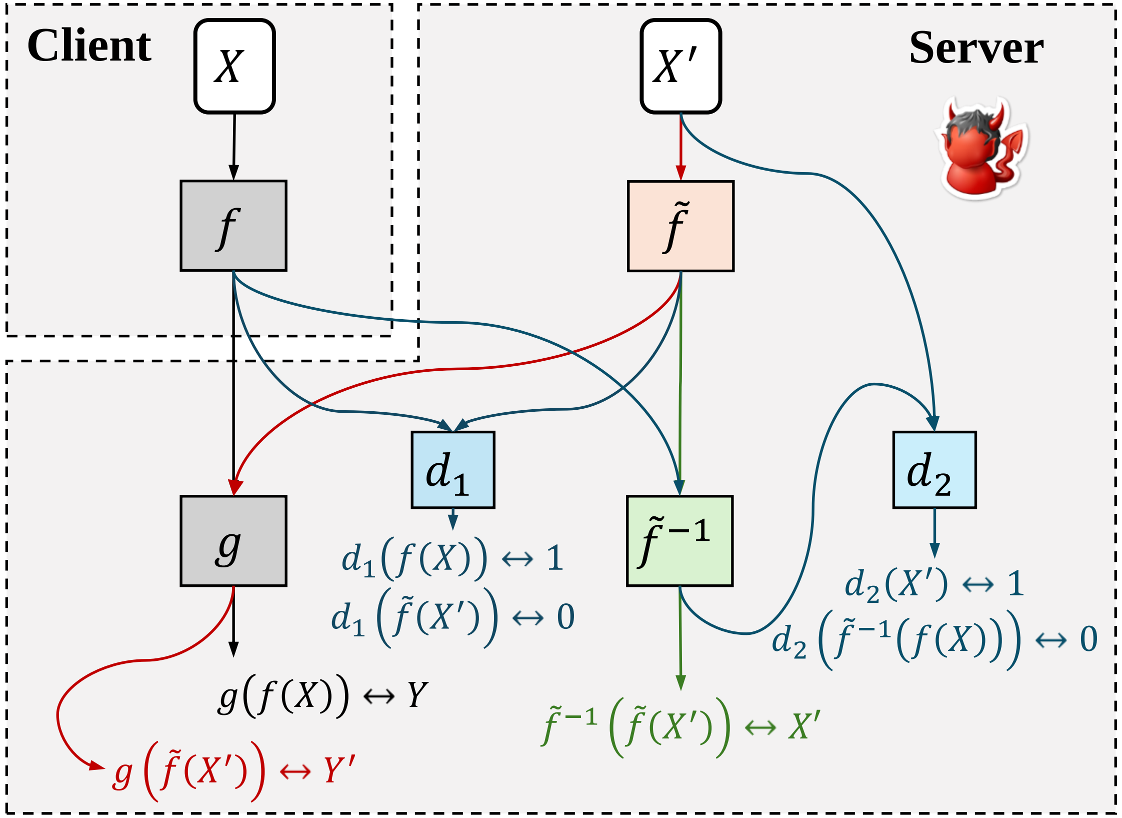


Discriminator as regularizer

Issue 2: The decoder can decode $\tilde{f}(X')$ doesn't mean it can decode $f(X)$.

- Discriminator d_2 to distinguish X' and $\tilde{f}^{-1}(f(X))$
- Add GAN generation loss as a regularization term to $\tilde{f}^{-1}$'s loss, such that it is optimized to produce plausible images on private data:

$$\mathrm{MSE}\left(\tilde{f}^{-1}\left(\tilde{f}(X')\right), X'\right) + \lambda_2 \mathrm{CrossEntropy}\left(d_2\left(\tilde{f}^{-1}(f(X))\right), 1\right)$$



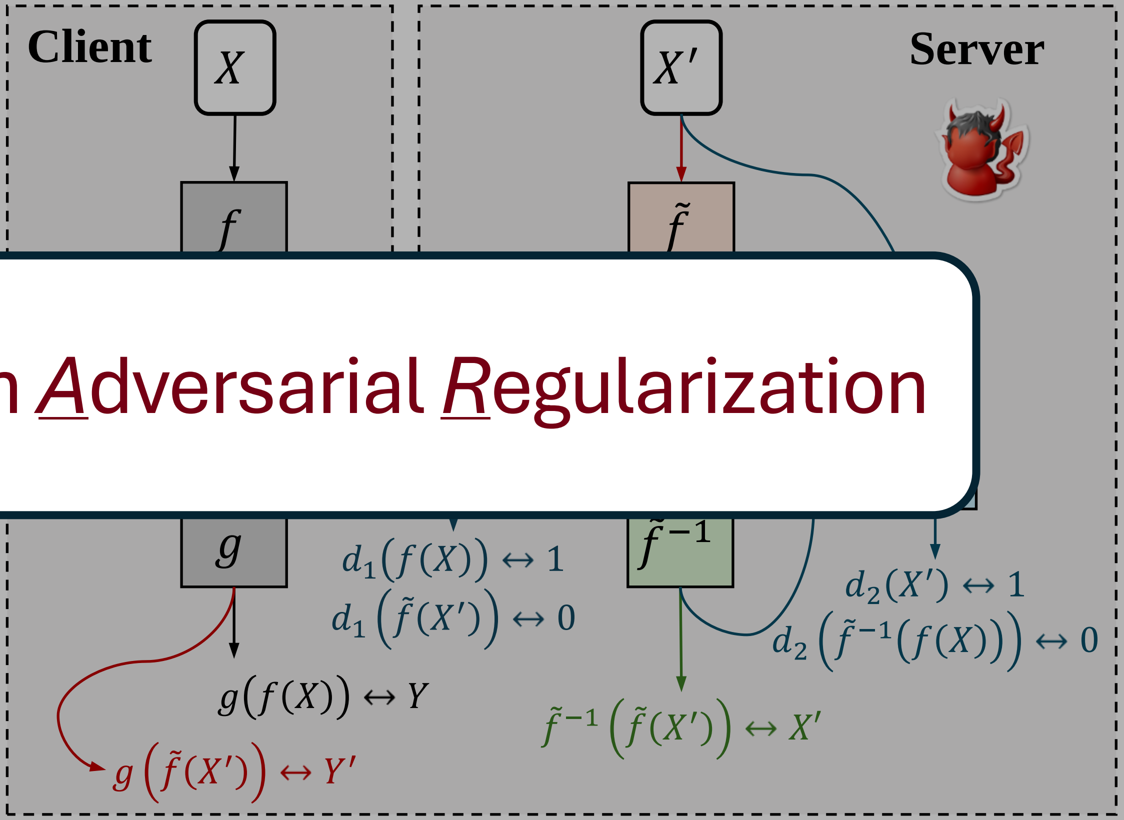
Discriminator as regularizer

Issue 2: The decoder can decode $\tilde{f}(X')$ doesn't mean it can decode $f(X)$.

- Dis

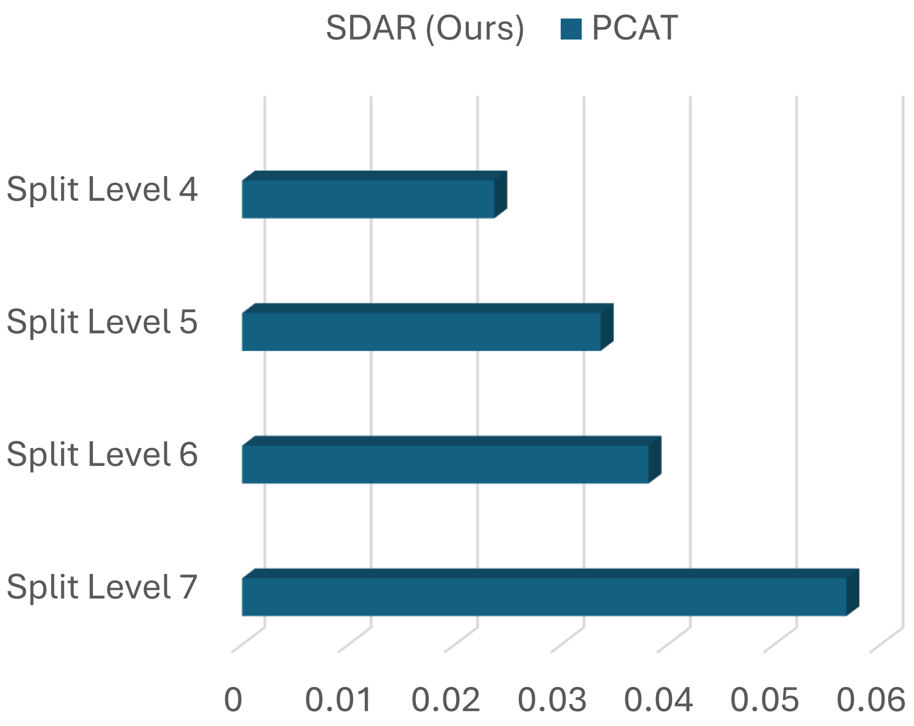
- Ad
to
pla

$$\mathrm{MSE}\left(\tilde{f}^{-1}\left(\tilde{f}(X')\right), X'\right)$$
$$+\lambda_2 \mathrm{CrossEntropy}\left(d_2\left(\tilde{f}^{-1}(f(X))\right), 1\right)$$

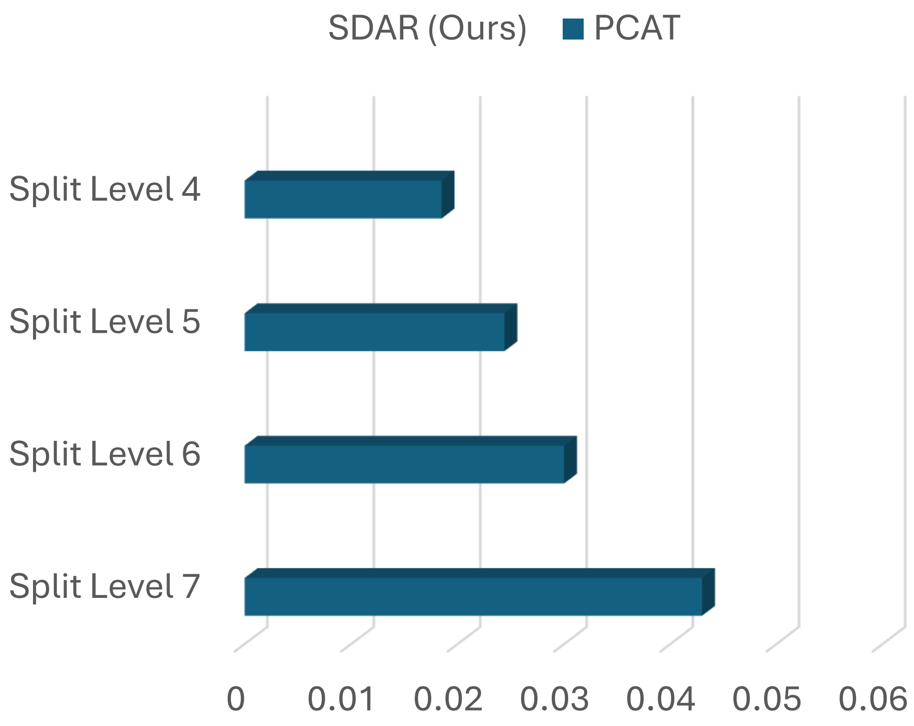


SDAR: Simulator Decoding with Adversarial Regularization

Attack results on vanilla SL



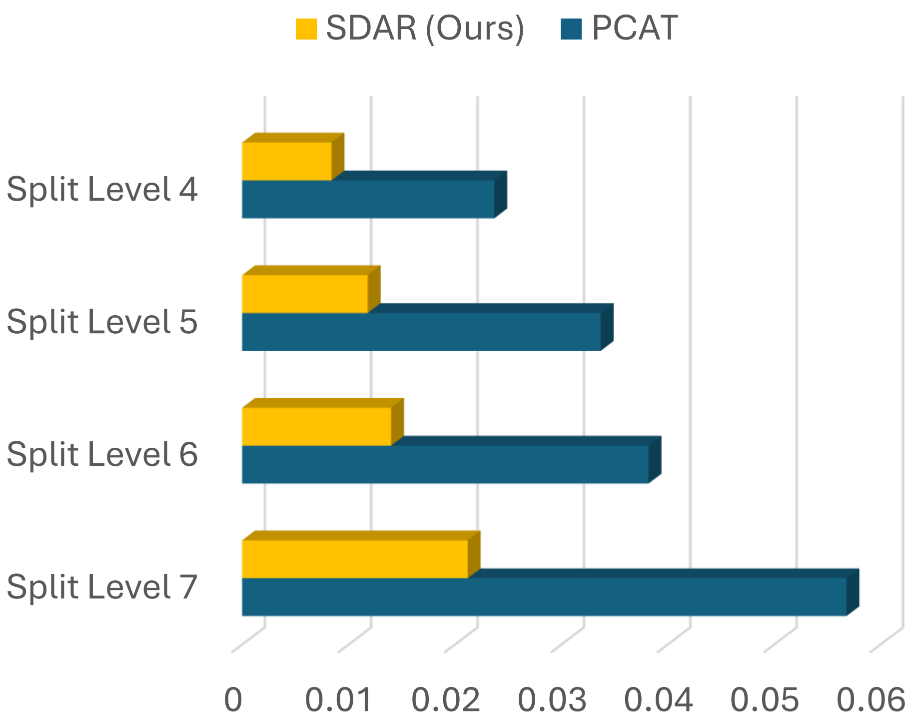
CIFAR-10



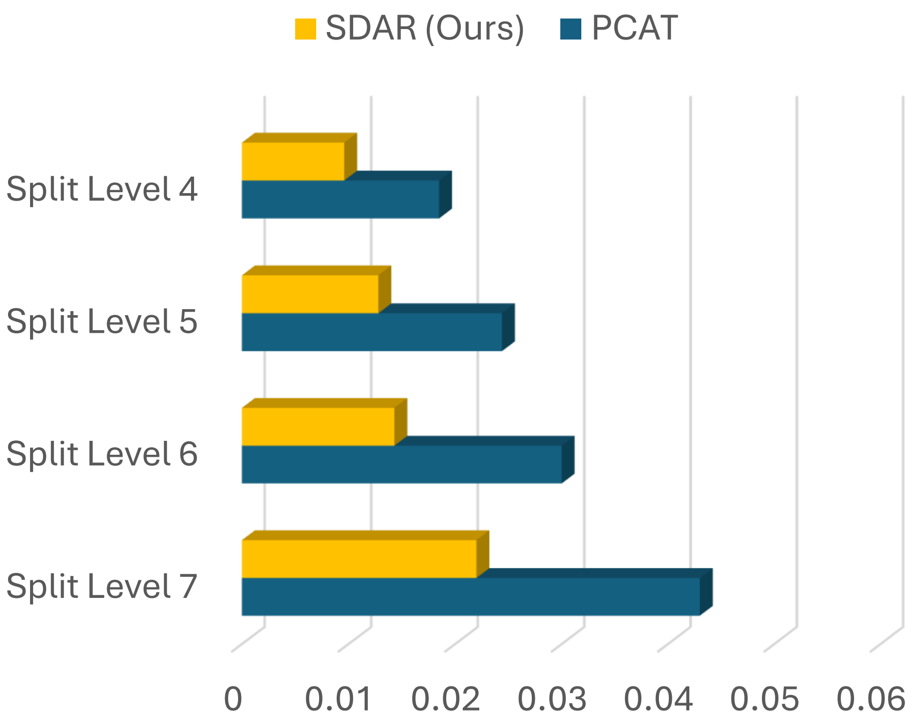
CIFAR-100

Feature inference attack mean squared error (MSE) on vanilla SL with ResNet-20
(lower is better)

Attack results on vanilla SL



CIFAR-10

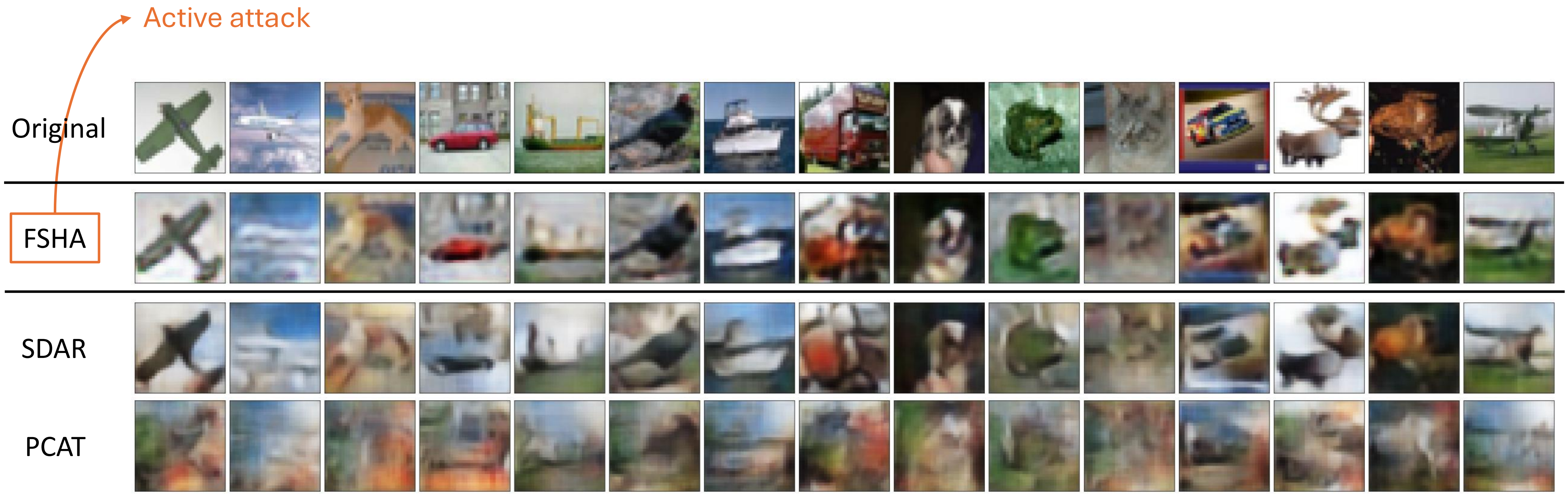


CIFAR-100

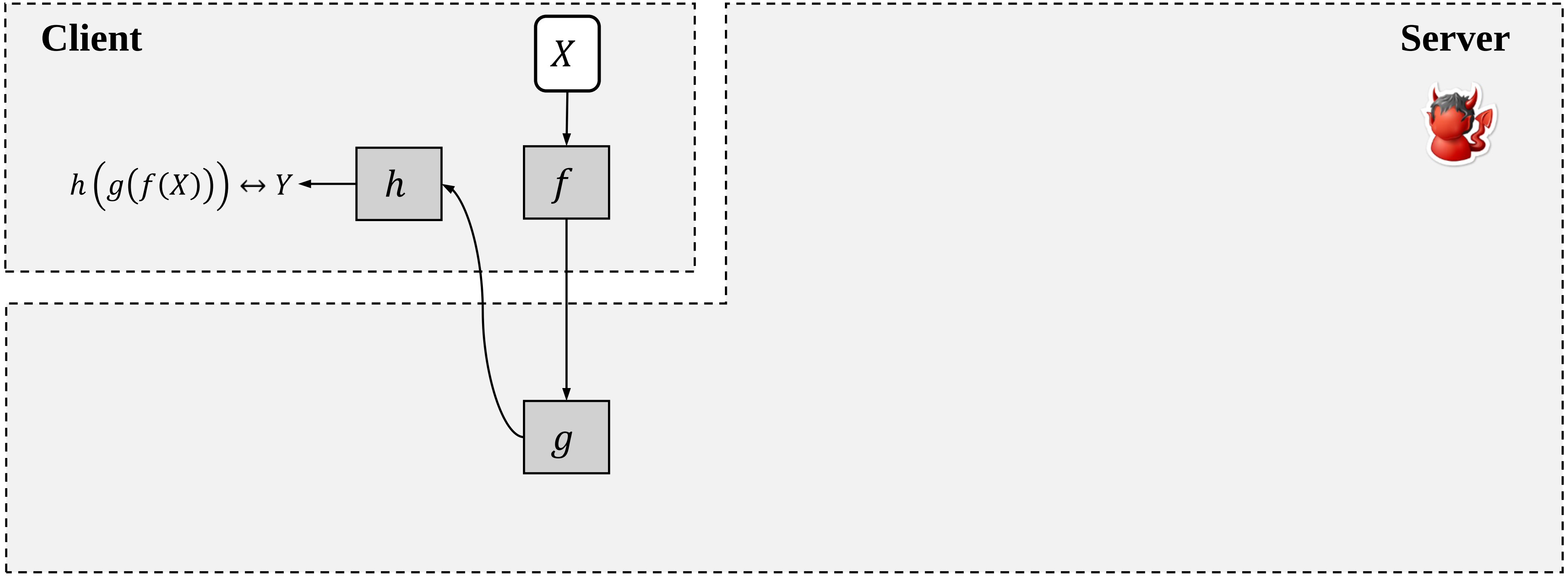
Feature inference attack mean squared error (MSE) on vanilla SL with ResNet-20
(lower is better)

Attack results on vanilla SL

Attack results on vanilla SL

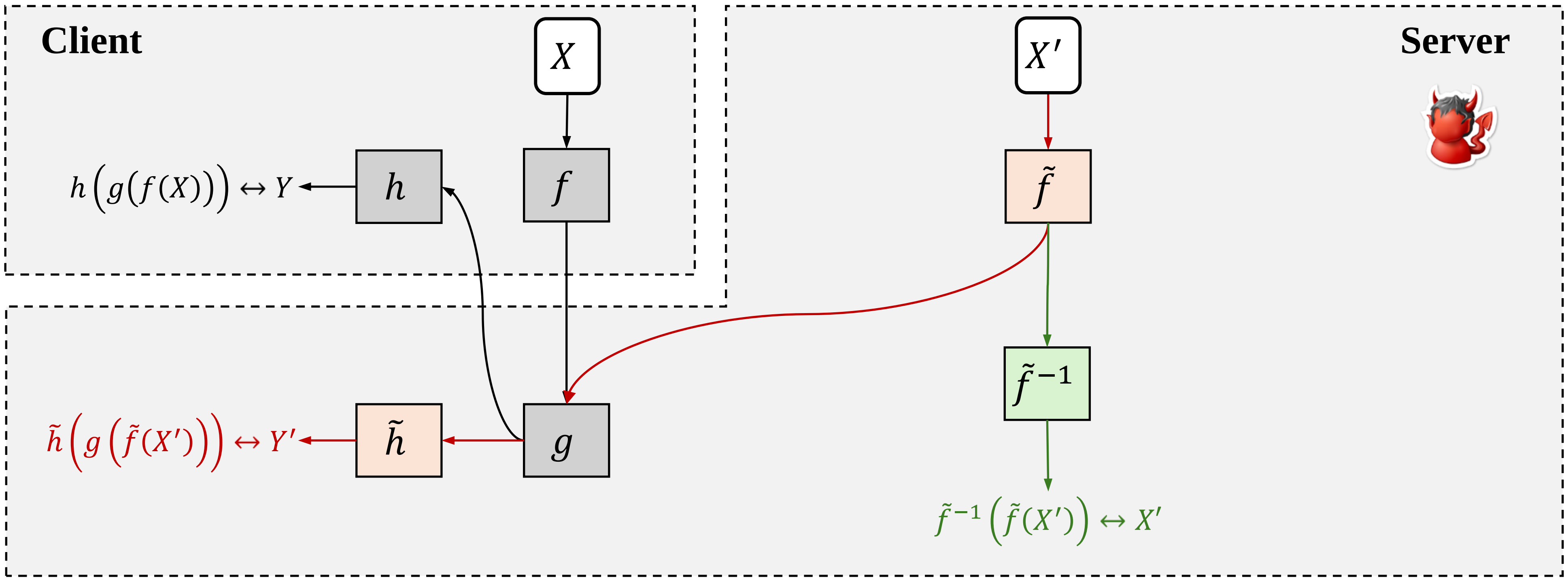


SDAR: on the U-shaped split learning



The server no longer has client's training examples' labels or the final layers.

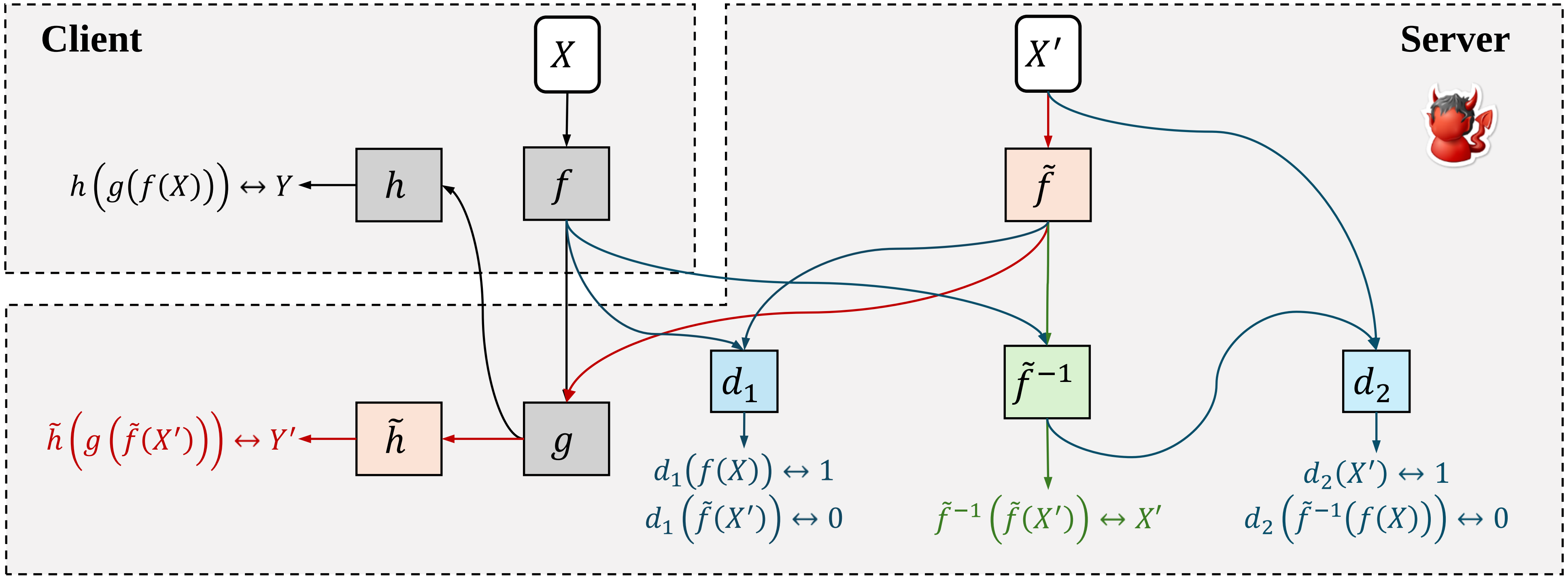
SDAR: on the U-shaped split learning



Like previous attacks, we have simulator $\tilde{f}$ and decoder $\tilde{f}^{-1}$.

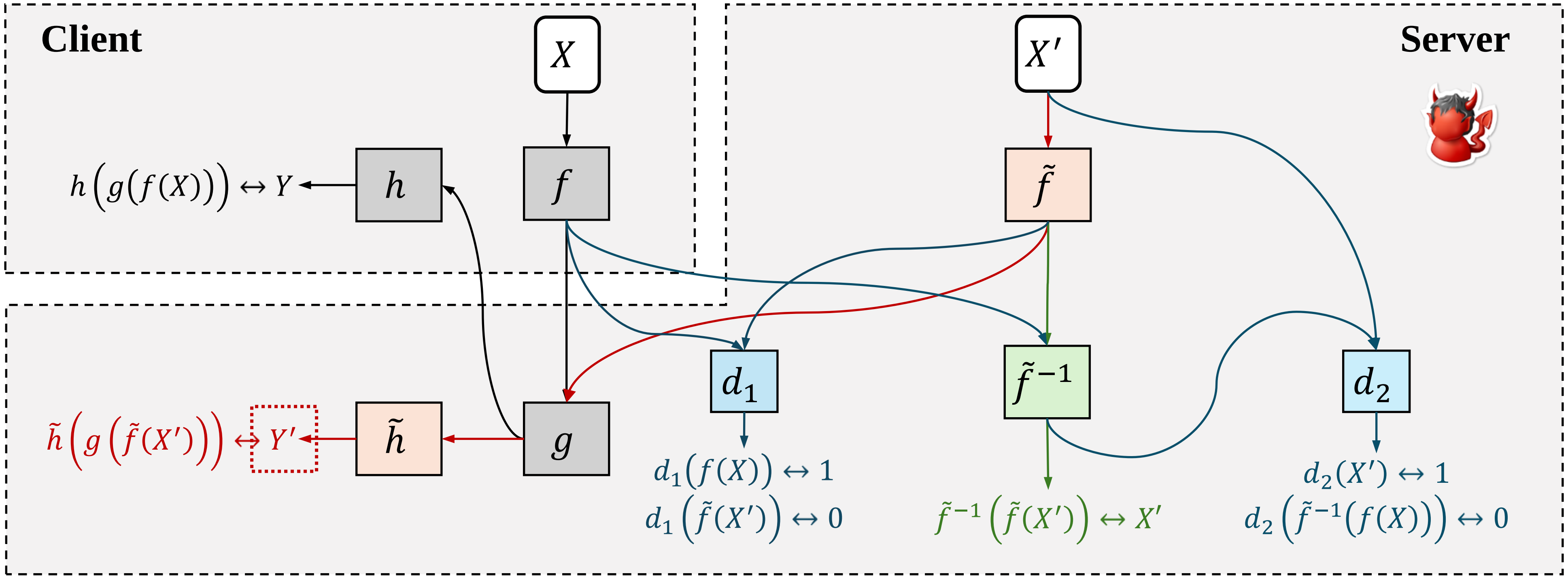
Additional simulator $\tilde{h}$: server trains $\tilde{h} \circ g \circ \tilde{f}$ on (X', Y').

SDAR: on the U-shaped split learning



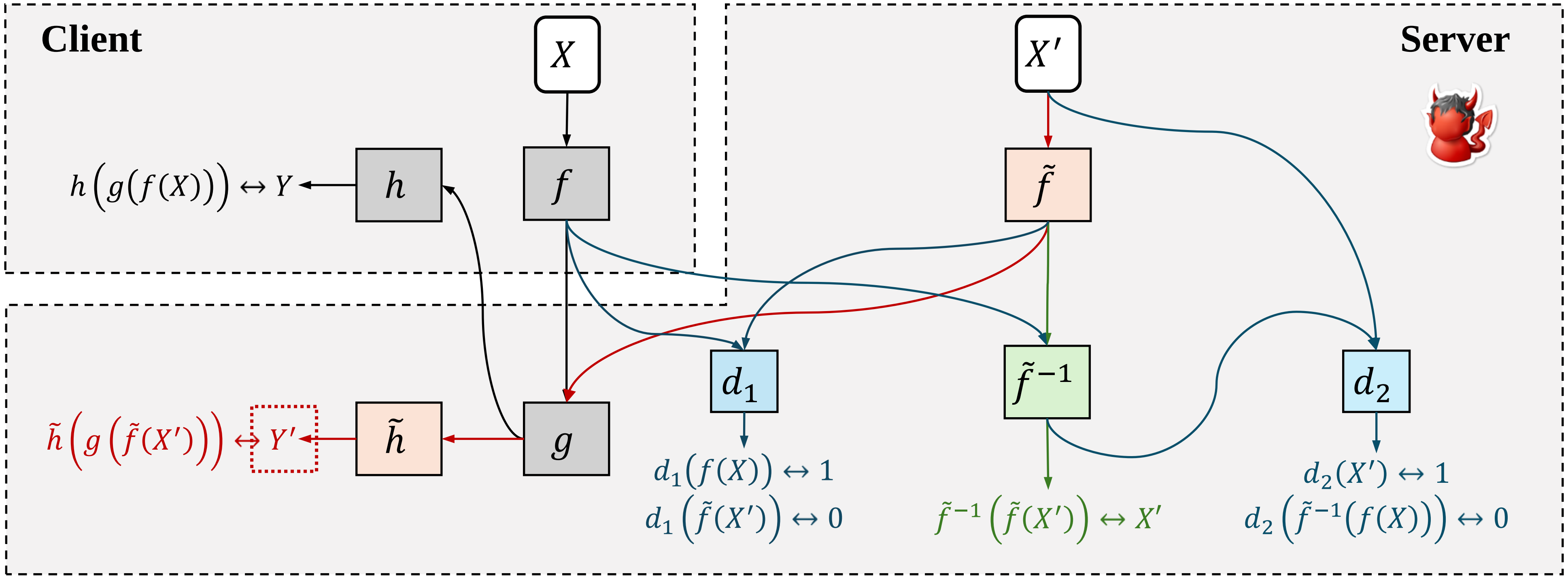
Like previous attacks, we have discriminators d_1, d_2.

SDAR: on the U-shaped split learning



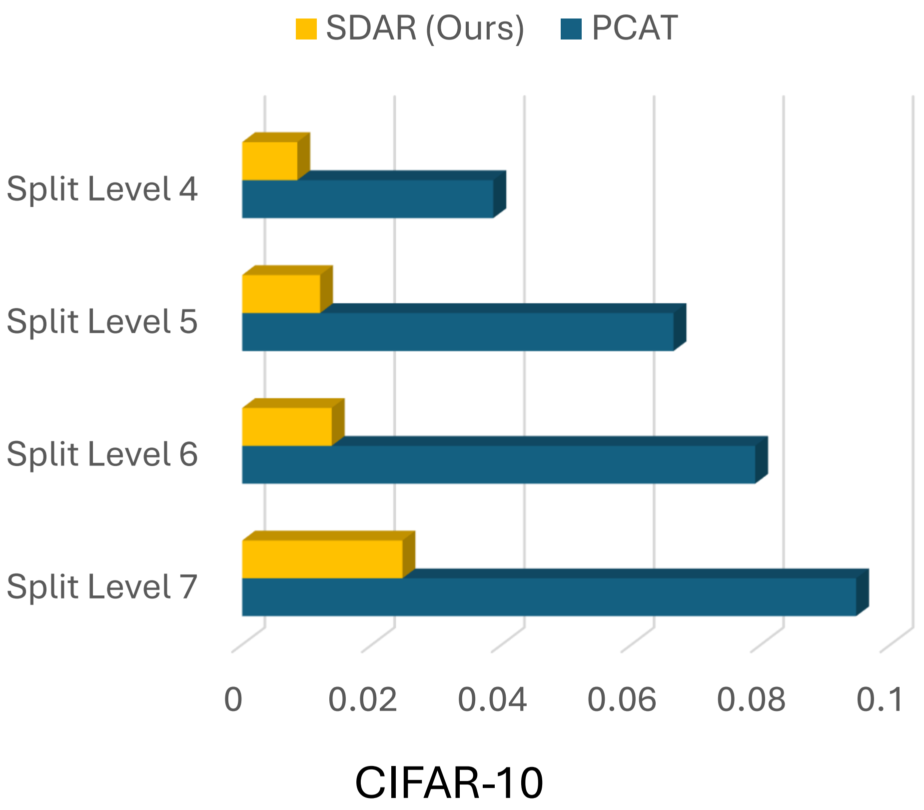
Prevent $\tilde{h}$ from overfitting to (X', Y'): random label flipping.

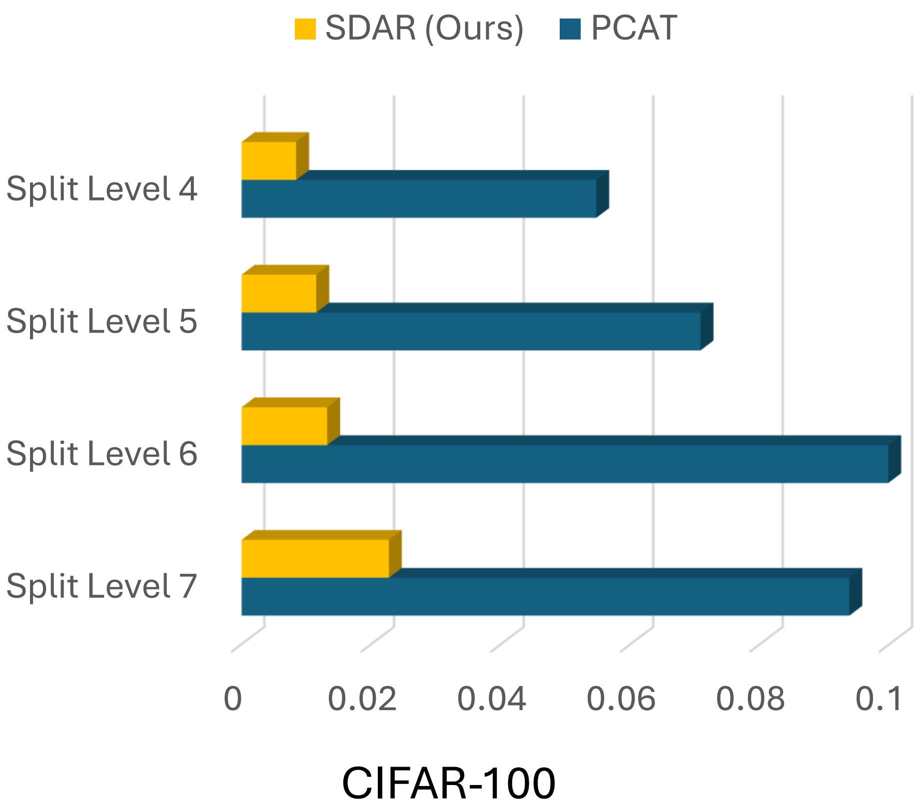
SDAR: on the U-shaped split learning



Label inference attack: feed $g\left(f(X)\right)$ to $\tilde{h}$.

Feature inference results on U-shaped SL





Feature inference attack mean squared error (MSE) on U-shaped SL with ResNet-20

(lower is better)

Feature inference results on U-shaped SL

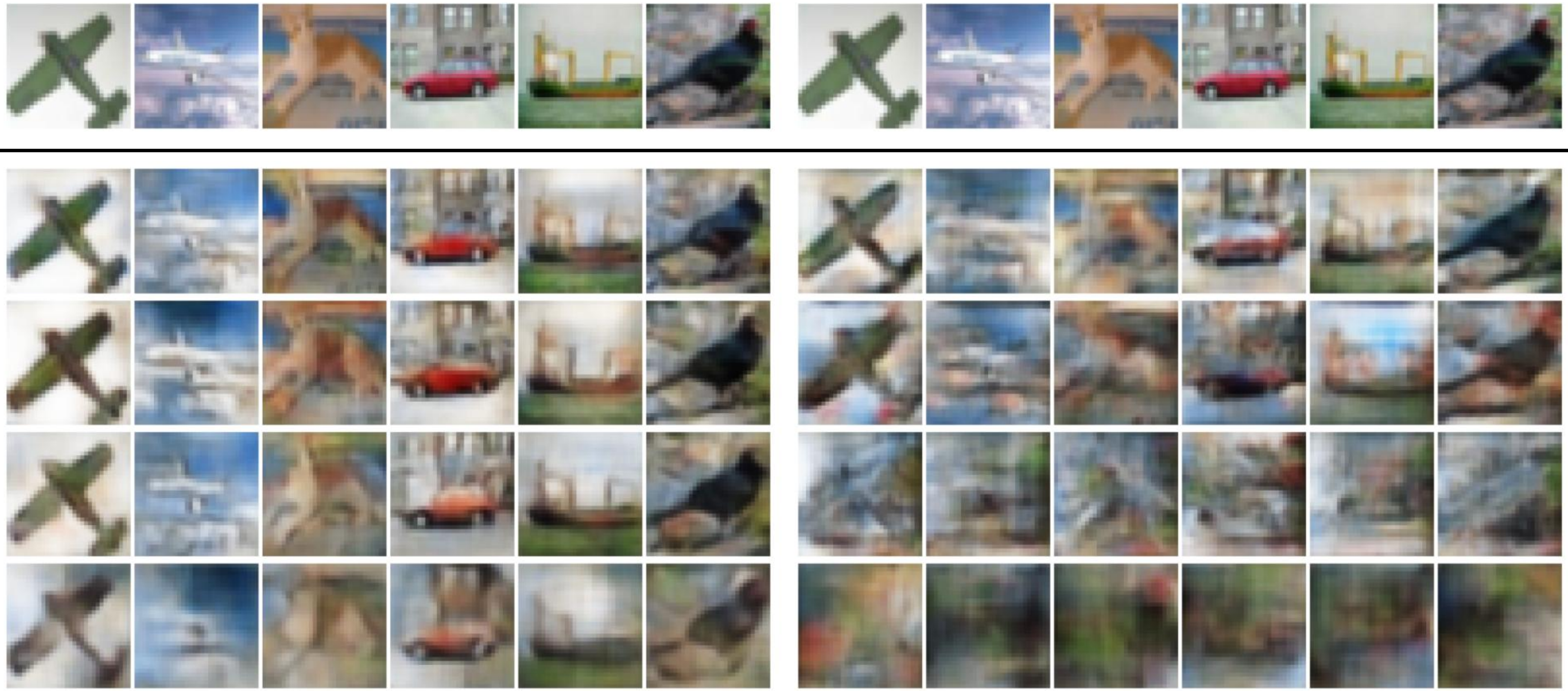
Original images

Split Level 4

Split Level 5

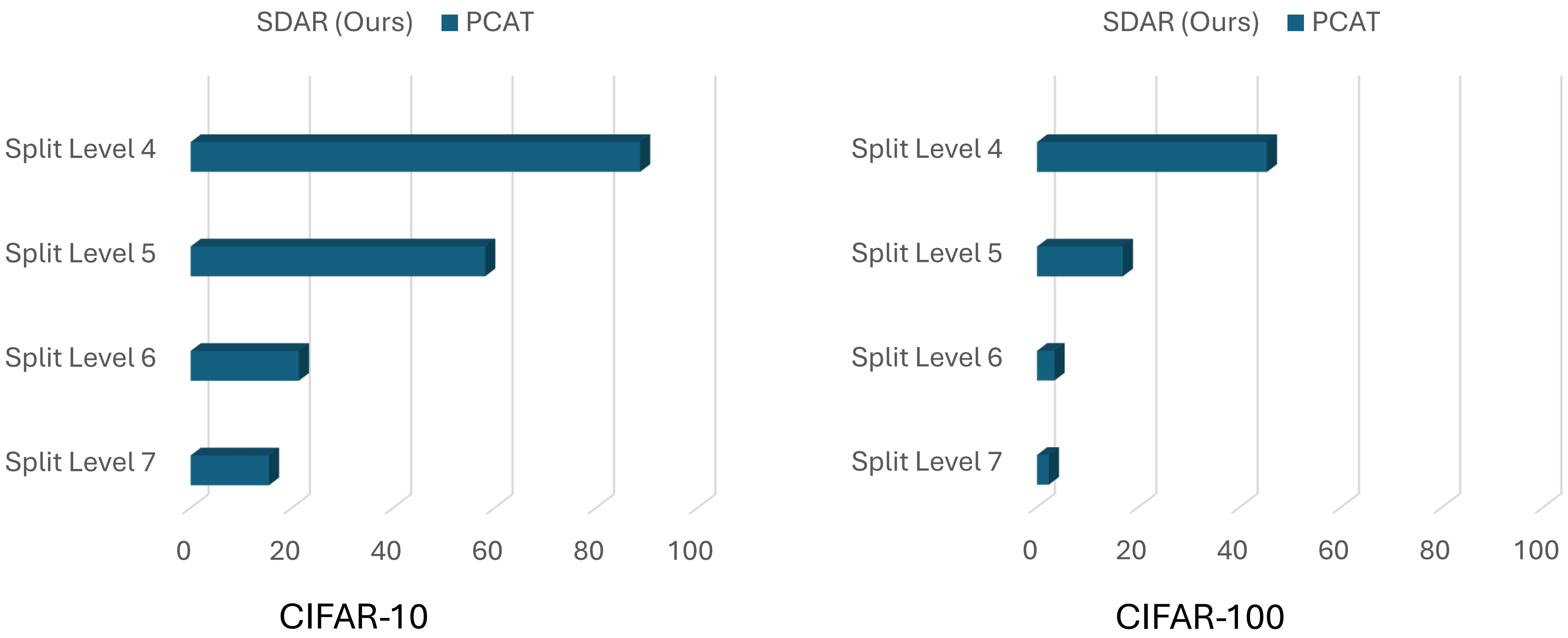
Split Level 6

Split Level 7



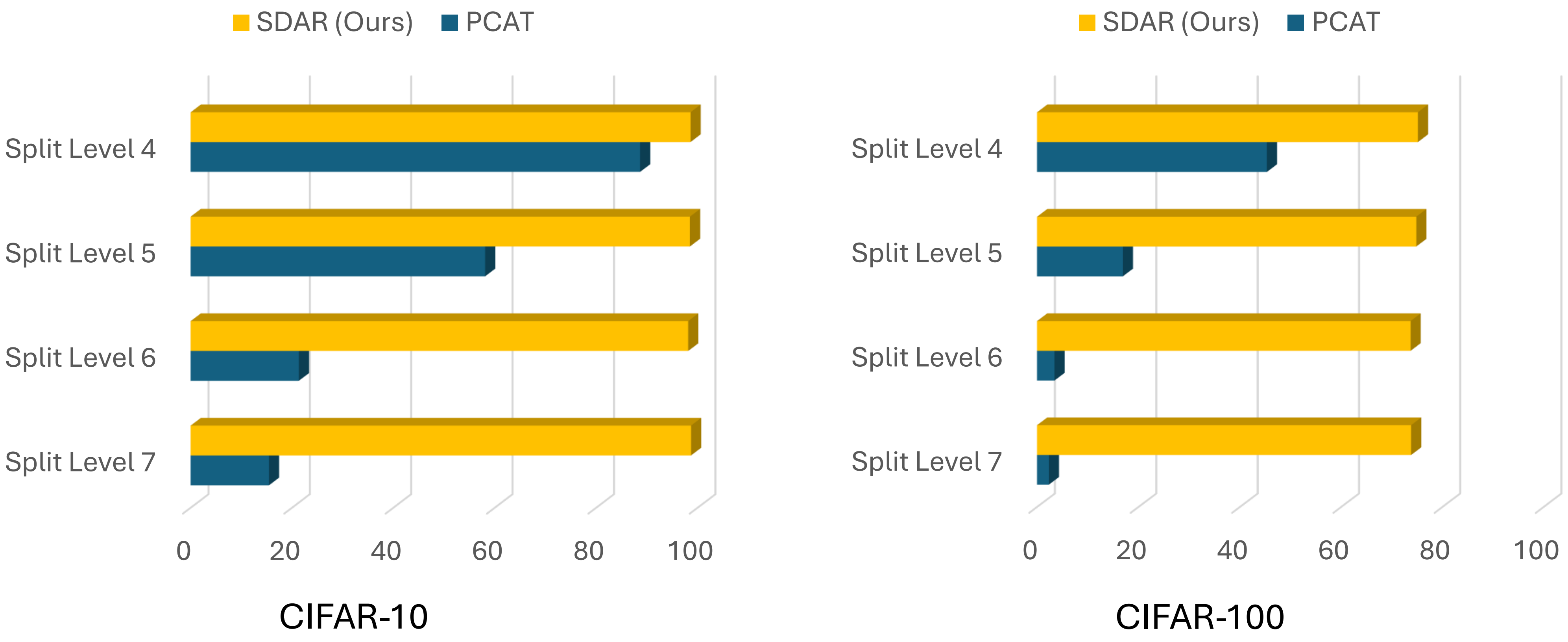
SDAR (Ours) **PCAT**

Label inference results on U-shaped SL



Label inference accuracy (%) on U-shaped SL with ResNet-20
(higher is better)

Label inference results on U-shaped SL



Label inference accuracy (%) on U-shaped SL with ResNet-20
(higher is better)

Further discussions

- Effects of auxiliary data distribution
  - SDAR is still effective when auxiliary dataset is much smaller than target dataset (5%)
  - SDAR is still effective when auxiliary dataset is o.o.d. of the target dataset
- Effects of target model architecture
  - ResNet is more prone to attacks than PlainNet
  - A shallower and wider client's model is more prone to inference attacks
- Effects of the server's knowledge of the client's model architecture
  - It helps if the server knows the client's model architecture, but SDAR remains effective when it does not
- Ablation studies

Further discussions

- Effects of auxiliary data distribution
  - SDAR is still effective when auxiliary dataset is much smaller than target dataset (5%)
  - SDAR is still effective when auxiliary dataset is o.o.d. of the target dataset
- Effects of target model architecture
  - ResNet is more prone to attacks than PlainNet
  - A shallower and wider client's model is more prone to inference attacks
- Effects of the server's knowledge of the client's model architecture
  - It helps if the server knows the client's model architecture, but SDAR remains effective when it does not
- Ablation studies

Further discussions

- Effects of auxiliary data distribution
  - SDAR is still effective when auxiliary dataset is much smaller than target dataset (5%)
  - SDAR is still effective when auxiliary dataset is o.o.d. of the target dataset
- Effects of target model architecture
  - ResNet is more prone to attacks than PlainNet
  - A shallower and wider client's model is more prone to inference attacks
- Effects of the server's knowledge of the client's model architecture
  - It helps if the server knows the client's model architecture, but SDAR remains effective when it does not
- Ablation studies

Further discussions

- Effects of auxiliary data distribution
  - SDAR is still effective when auxiliary dataset is much smaller than target dataset (5%)
  - SDAR is still effective when auxiliary dataset is o.o.d. of the target dataset
- Effects of target model architecture
  - ResNet is more prone to attacks than PlainNet
  - A shallower and wider client's model is more prone to inference attacks
- Effects of the server's knowledge of the client's model architecture
  - It helps if the server knows the client's model architecture, but SDAR remains effective when it does not
- Ablation studies

Potential countermeasures

- Deeper split levels or narrower models
- Regularization (dropout, l1, l2)
- Decorrelation

Potential countermeasures

- Deeper split levels or narrower models
- Regularization (dropout, l1, l2)
- Decorrelation
- Homomorphic encryption
- Multi-party computation
- Differential privacy

Thank you!