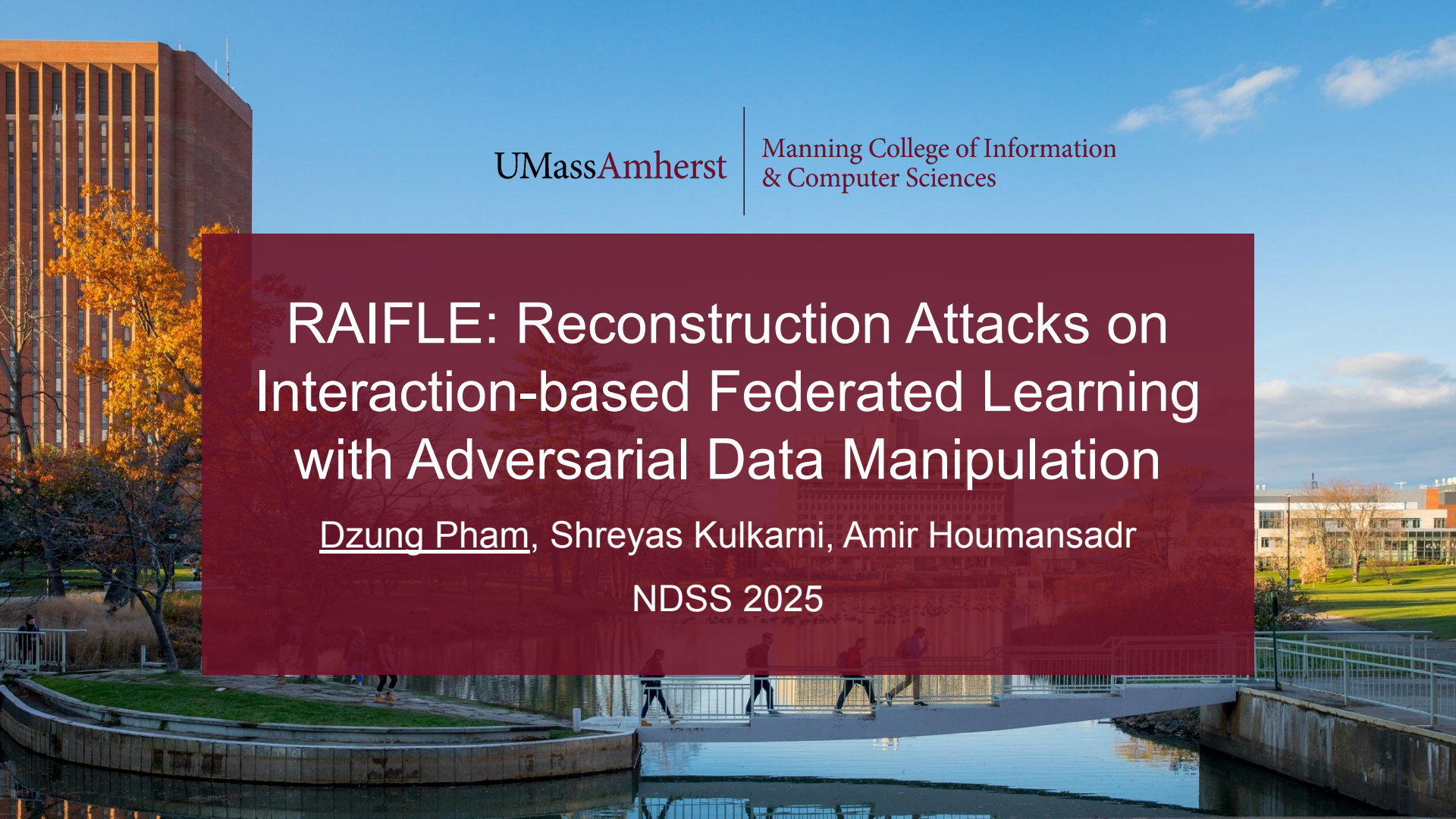


UMassAmherst | Manning College of Information & Computer Sciences

RAIFLE: Reconstruction Attacks on Interaction-based Federated Learning with Adversarial Data Manipulation

Dzung Pham, Shreyas Kulkarni, Amir Houmansadr

NDSS 2025

Motivation

- Ranking/Recommendation systems (RS) are *everywhere*:
  - Social network, e-commerce, entertainment, search engine, etc.

Motivation

- Ranking/Recommendation systems (RS) are *everywhere*:
  - Social network, e-commerce, entertainment, search engine, etc.
- Federated RS has been researched…
  - … but **no real-world** deployment

Motivation

- Ranking/Recommendation systems (RS) are *everywhere*:
  - Social network, e-commerce, entertainment, search engine, etc.
- Federated RS has been researched…
  - … but **no real-world** deployment
- What are the privacy vulnerabilities in federated RS?

What is Interaction-based FL (IFL)?

- Generalization of federated RS/learning-to-rank systems

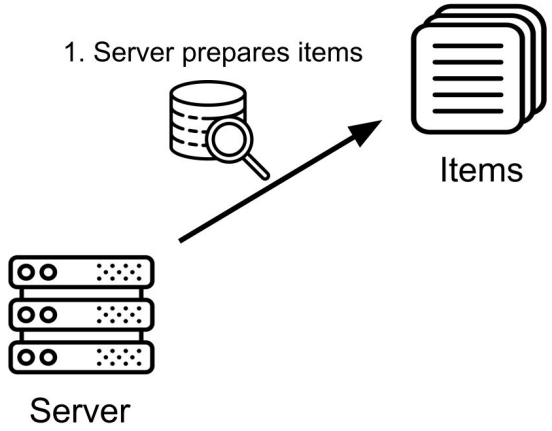
What is Interaction-based FL (IFL)?

- Generalization of federated RS/learning-to-rank systems
- Typical FL settings:
  - FL servers don’t have any influence on users’ private data

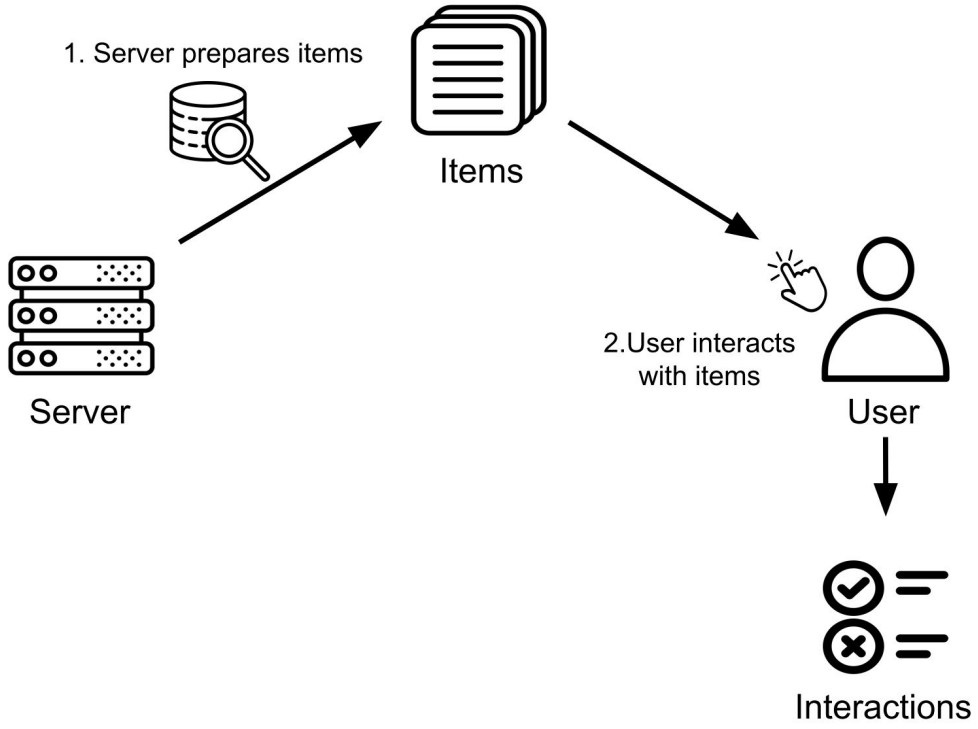
What is Interaction-based FL (IFL)?

- Generalization of federated RS/learning-to-rank systems
- Typical FL settings:
  - FL servers don't have any influence on users' private data
- In IFL:
  - FL servers **present** "items" to users for them to interact with

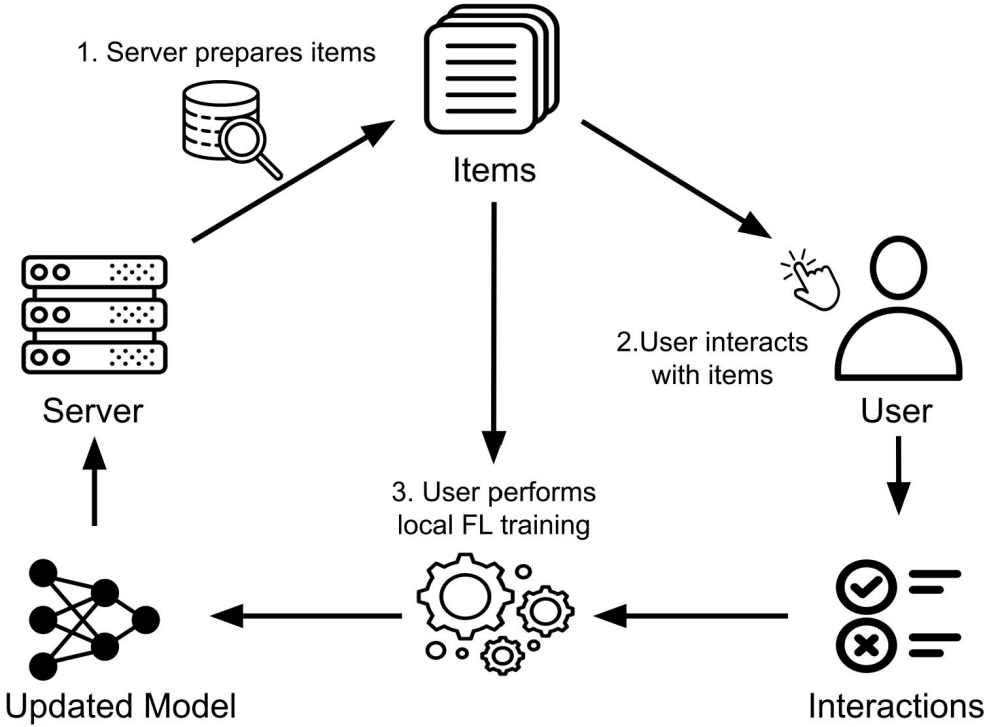
Interaction-based FL



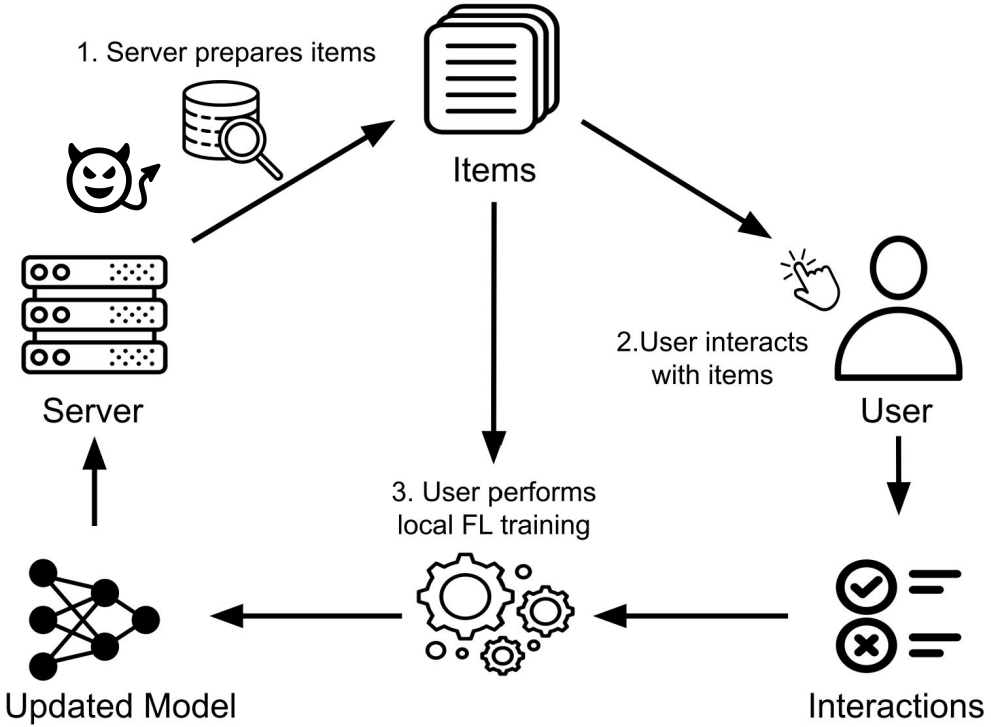
Interaction-based FL



Interaction-based FL



Interaction-based FL



Honest-but-curious IFL server

- Reconstruct user data with gradient inversion:
  - Given original model weights W and updated weights W*
  - Initialize some random interactions I’
  - Simulate the local training with W and I’ => W’
  - Calculate loss between W’ and W*
  - Optimize w.r.t I’

Honest-but-curious IFL server

- Reconstruct user data with gradient inversion:
  - Given original model weights W and updated weights W*
  - Initialize some random interactions I’
  - Simulate the local training with W and I’ => W’
  - Calculate loss between W’ and W*
  - Optimize w.r.t I’
- Defenses:
  - Apply differential privacy (i.e. add noise to W*)
  - Secure aggregation (i.e. break links between users and W*)

Vanilla Gradient Inversion Evaluation

- Dataset: MovieLens-100K, Steam-200K
- Algorithm: Federated Neural Collaborative Filtering
- Baseline: Interaction Membership Inference Attack (WWW 2023)
- Metrics: F1. Label distribution: Pos : Neg = 1 : 4

Vanilla Gradient Inversion Evaluation

- Dataset: MovieLens-100K, Steam-200K
- Algorithm: Federated Neural Collaborative Filtering
- Baseline: Interaction Membership Inference Attack (WWW 2023)
- Metrics: F1. Label distribution: Pos : Neg = 1 : 4

Method	IMIA Defense	MovieLens-100K	Steam-200K
IMIA	No	0.593	0.671
Grad Inv	No	**0.983**	**0.923**
IMIA	Yes	0.215	0.206
Grad Inv	Yes	**0.382**	**0.316**

Risks of malicious IFL server

- Question: What can a malicious server do?

Risks of malicious IFL server

- Question: What can a malicious server do?
- Exploit control of the model weights?
  - Already researched in traditional FL

Risks of malicious IFL server

- Question: What can a malicious server do?
- Exploit control of the model weights?
  - Already researched in traditional FL
- Exploit control of the **presented items**?
  - Unique to IFL
  - This had not been explored

Attack Idea: Adversarial Data Manipulation

- “Uniquify” the contribution of each item to the gradients

Attack Idea: Adversarial Data Manipulation

- “Uniquify” the contribution of each item to the gradients
- One approach: Strategically zero out features for each item
  - Intuition: Each item influences a unique parameter

Item	x1	x2	x3
Item 1	0.123	0	0
Item 2	0	0.456	0
Item 3	0	0	0.789

Attack Idea: Adversarial Data Manipulation

- “Uniquify” the contribution of each item to the gradients
- One approach: Strategically zero out features for each item
  - Intuition: Each item influences a unique parameter
  - Drawbacks:
    - A bit too obvious
    - What if there are more items than parameters (N > D)?

Attack Idea: Adversarial Data Manipulation

- “Uniquify” the contribution of each item to the gradients
- One approach: Strategically zero out features for each item
  - Intuition: Each item influences a unique parameter
  - Drawbacks:
    - A bit too obvious
    - What if there are more items than parameters (N > D)?
- Another approach: Replace feature values with random noise
  - Not as obvious
  - Can (somewhat) overcome the N > D scenario

ADM Evaluation

- Dataset: MQ2007, MSLR-WEB10K
- Algorithm: Federated Pairwise Differentiable Gradient Descent
- Metrics: AUC. Baseline: Vanilla gradient inversion

ADM Evaluation

- Dataset: MQ2007, MSLR-WEB10K
- Algorithm: Federated Pairwise Differentiable Gradient Descent
- Metrics: AUC. Baseline: Vanilla gradient inversion

MQ2007

Model	ADM	Informational			Navigational		
		4	8	16	4	8	16
Linear	None	0.87	0.76	0.66	0.95	0.84	0.71
	RAIFLE	**1.00**	**0.98**	**0.82**	**1.00**	**0.98**	**0.88**
Neural (4 hid. units)	None	0.78	0.66	0.57	0.86	0.70	0.60
	RAIFLE	**1.00**	**0.99**	**0.95**	**1.00**	**1.00**	**0.98**
Neural (8 hid. units)	None	0.77	0.64	0.56	0.86	0.68	0.59
	RAIFLE	**1.00**	**1.00**	**0.99**	**1.00**	**1.00**	**1.00**
Neural (16 hid. units)	None	0.75	0.60	0.53	0.82	0.61	0.55
	RAIFLE	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

MSLR-WEB10K

Model	ADM	Informational			Navigational		
		12	24	48	12	24	48
Linear	None	0.54	0.51	0.50	0.55	0.51	0.51
	RAIFLE	**1.00**	**0.98**	**0.80**	**1.00**	**0.97**	**0.86**
Neural (4 hid. units)	None	0.52	0.50	0.50	0.52	0.51	0.50
	RAIFLE	**0.98**	**0.95**	**0.87**	**0.99**	**0.97**	**0.93**
Neural (8 hid. units)	None	0.51	0.50	0.50	0.51	0.50	0.50
	RAIFLE	**0.99**	**0.96**	**0.91**	**1.00**	**0.99**	**0.96**
Neural (16 hid. units)	None	0.51	0.50	0.50	0.51	0.50	0.51
	RAIFLE	**0.97**	**0.94**	**0.91**	**0.99**	**0.98**	**0.96**

Threat Model Too Strong?

Threat Model Too Strong?

npr DONATE

All Tech Considered

SOCIAL WEB

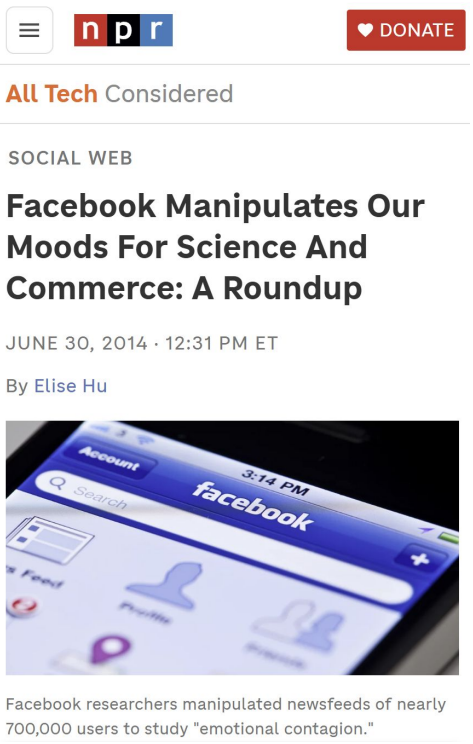
Facebook Manipulates Our Moods For Science And Commerce: A Roundup

JUNE 30, 2014 · 12:31 PM ET

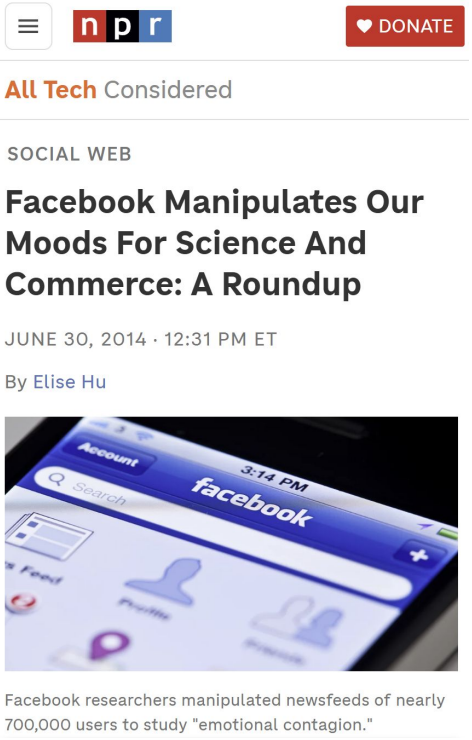
By Elise Hu

Facebook researchers manipulated newsfeeds of nearly 700,000 users to study "emotional contagion."

Threat Model Too Strong?



Threat Model Too Strong?



Indirect Adversarial Data Manipulation

- What if the server **cannot directly control** the training features?

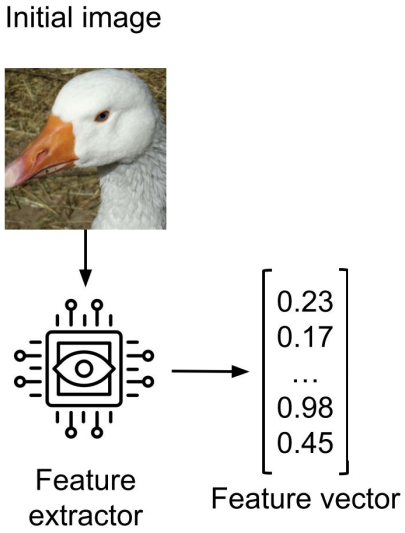
Indirect Adversarial Data Manipulation

- What if the server **cannot directly control** the training features?
- Example scenario: Ranking with images:
  - Server sends images to users
  - Users use a pre-trained model to extract features

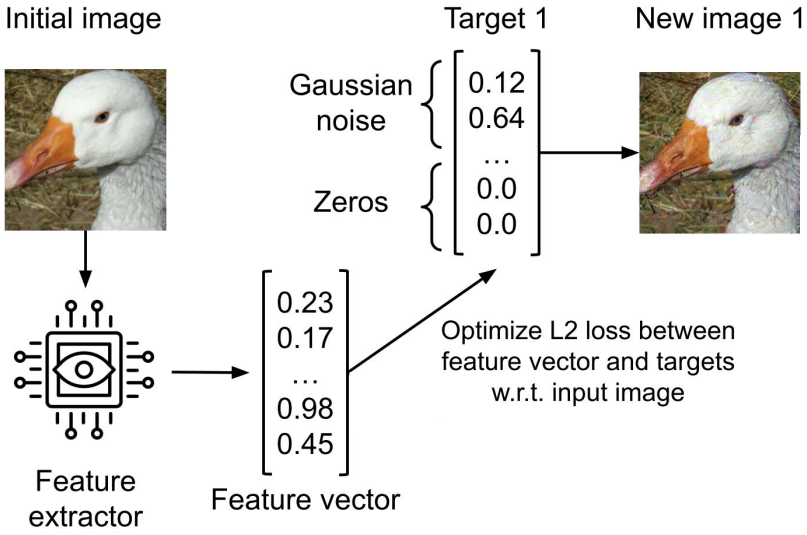
Indirect Adversarial Data Manipulation

- What if the server **cannot directly control** the training features?
- Example scenario: Ranking with images:
  - Server sends images to users
  - Users use a pre-trained model to extract features
- How to modify the images?
  - Can we make the extracted features resemble noise?

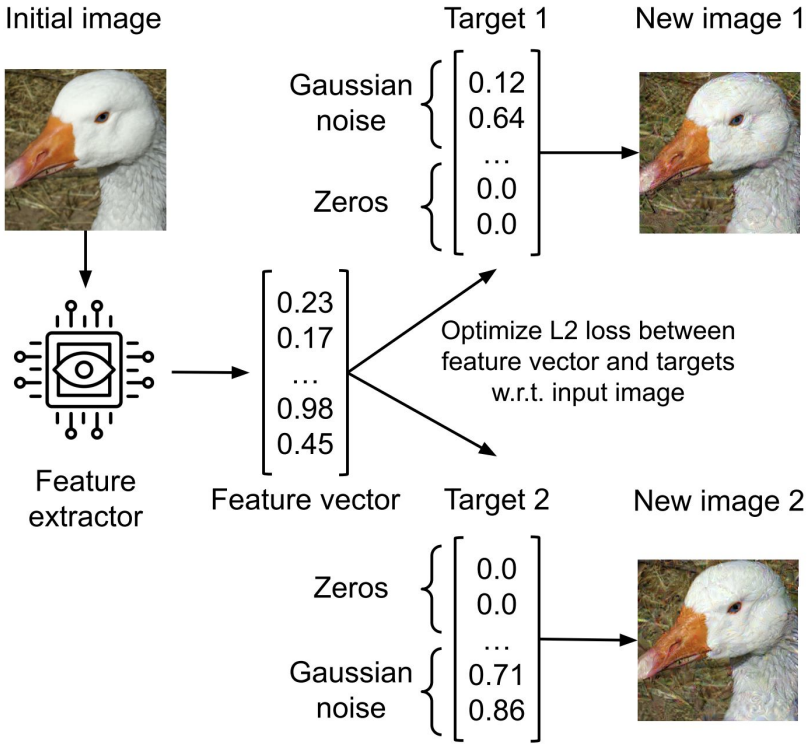
Indirect Adversarial Data Manipulation



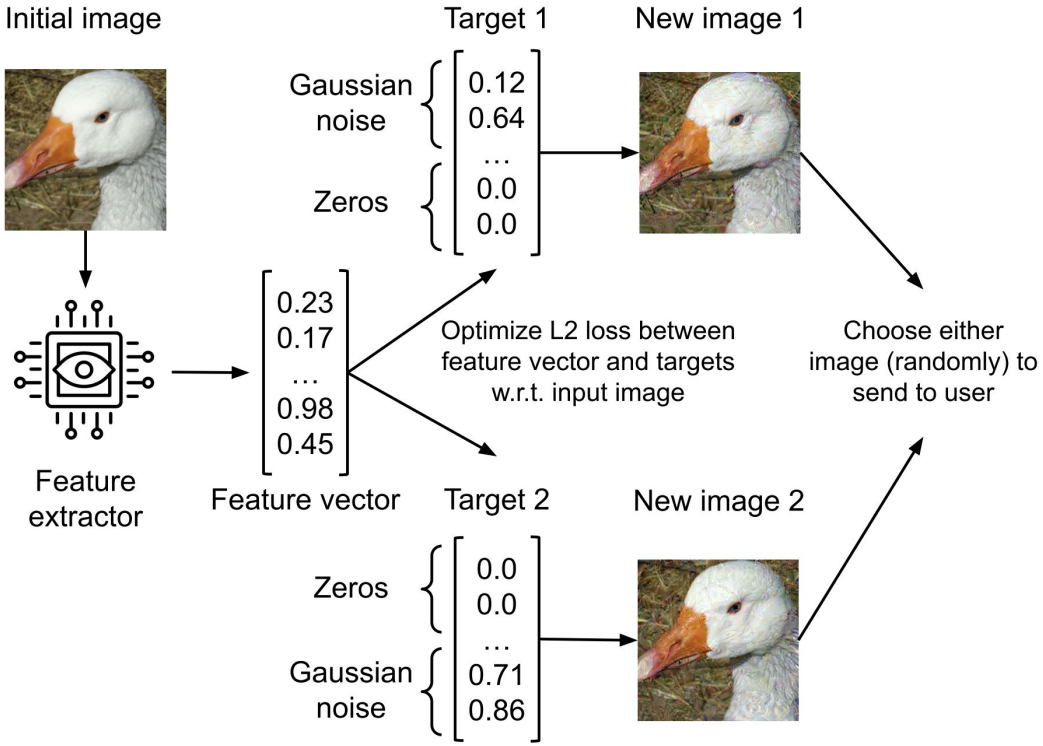
Indirect Adversarial Data Manipulation



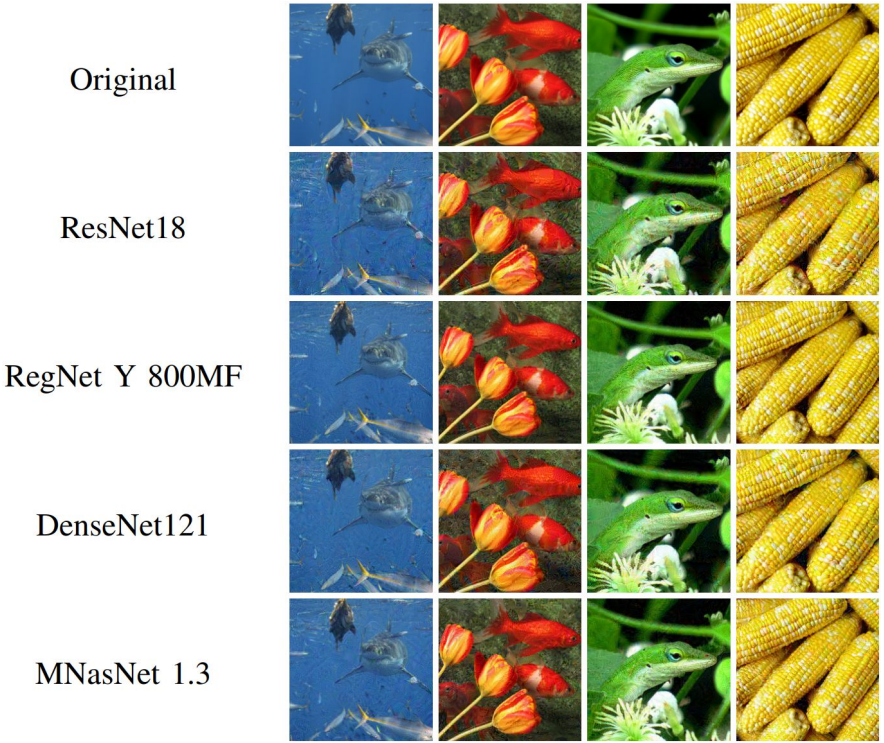
Indirect Adversarial Data Manipulation



Indirect Adversarial Data Manipulation



Indirect Adversarial Data Manipulation



Evaluation

- Dataset: ImageNet 2012 (Validation set)
- Feature extractor: ResNet, RegNet, DenseNet, MNasNet
- Metrics: AUC. Baseline: Vanilla gradient inversion, FGSM

Evaluation

- Dataset: ImageNet 2012 (Validation set)
- Feature extractor: ResNet, RegNet, DenseNet, MNasNet
- Metrics: AUC. Baseline: Vanilla gradient inversion, FGSM

Vision Model	ADM	Linear			Neural (2 hidden units)			Neural (4 hidden units)			Neural (8 hidden units)		
		1x	2x	4x	1x	2x	4x	1x	2x	4x	1x	2x	4x
ResNet18 (512 features)	None	1.000	0.916	0.767	0.921	0.868	0.771	0.985	0.945	0.857	0.999	0.985	0.924
	FGSM	1.000	0.915	0.766	0.914	0.857	0.759	0.981	0.934	0.841	0.998	0.977	0.906
	RAIFLE	**1.000**	**0.943**	**0.772**	**0.946**	**0.920**	**0.823**	**0.993**	**0.981**	**0.922**	**1.000**	**0.998**	**0.978**
RegNet Y 800MF (784 features)	None	0.999	0.918	**0.772**	0.948	0.913	0.801	0.991	0.977	0.891	1.000	0.994	0.925
	FGSM	0.999	0.913	0.771	0.948	0.908	0.796	0.990	0.974	0.884	1.000	0.993	0.922
	RAIFLE	**1.000**	**0.952**	0.767	**0.956**	**0.932**	**0.833**	**0.993**	**0.989**	**0.938**	**1.000**	**0.999**	**0.984**
DenseNet121 (1024 features)	None	**0.933**	**0.772**	**0.667**	0.902	0.805	0.700	0.970	0.896	0.770	0.995	0.946	0.827
	FGSM	0.933	0.771	0.666	0.891	0.794	0.691	0.961	0.884	0.759	0.992	0.932	0.810
	RAIFLE	0.919	0.765	0.664	**0.923**	**0.825**	**0.717**	**0.983**	**0.935**	**0.815**	**0.999**	**0.983**	**0.904**
MNasNet 1.3 (1280 features)	None	0.994	0.895	0.764	0.936	0.858	0.754	0.985	0.926	0.787	0.996	0.942	0.775
	FGSM	0.989	0.885	0.758	0.930	0.845	0.742	0.982	0.913	0.771	0.994	0.928	0.755
	RAIFLE	**1.000**	**0.939**	**0.775**	**0.940**	**0.882**	**0.791**	**0.991**	**0.965**	**0.875**	**0.999**	**0.992**	**0.924**

Defense: Local Differential Privacy

- Apply Gaussian noise to local update before sending to server

Defense: Local Differential Privacy

- Apply Gaussian noise to local update before sending to server

Scenario	ADM	$\varepsilon = 1$	$\varepsilon = 20$	$\varepsilon = 100$	$\varepsilon = 500$	No DP
FNCF w/ ML-100K	N/A	0.50	0.52	0.56	0.74	1.00
FNCF w/ Steam-200K	N/A	0.50	0.56	0.72	0.90	0.96
FPDGD w/ MQ 2007	None	0.50	0.54	0.57	0.58	0.66
	RAIFLE	0.50	0.56	0.66	0.75	0.82
FPDGD w/ MSLR10K	None	0.50	0.50	0.50	0.50	0.50
	RAIFLE	0.50	0.52	0.58	0.62	0.80
FOLTR w/ ResNet18	None	0.50	0.52	0.55	0.62	1.00
	RAIFLE	0.50	0.52	0.55	0.63	1.00
FOLTR w/ DenseNet121	None	0.50	0.51	0.54	0.59	1.00
	RAIFLE	0.50	0.51	0.53	0.59	1.00

Defense: Secure Aggregation

- Server only gets a single update without knowing user identities

Defense: Secure Aggregation

- Server only gets a single update without knowing user identities
- Modified attack: Single out specific user(s) by fingerprinting items

Defense: Secure Aggregation

- Server only gets a single update without knowing user identities
- Modified attack: Single out specific user(s) by fingerprinting items
- Example: FPDGD with MQ2007

User	d_1	d_2	d_3	d_4	d_5	d_6
⋮	⋮	⋮	⋮	⋮	⋮	⋮
u_{k-1}	0.1	0.2	0.3	**0.0**	**0.0**	**0.0**
$\boldsymbol{u_k}$	**0.0**	**0.0**	**0.0**	0.4	0.5	0.6
u_{k+1}	0.1	0.2	0.3	**0.0**	**0.0**	**0.0**
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Defense: Secure Aggregation

- Server only gets a single update without knowing user identities
- Modified attack: Single out specific user(s) by fingerprinting items
- Example: FPDGD with MQ2007

User	d_1	d_2	d_3	d_4	d_5	d_6
⋮	⋮	⋮	⋮	⋮	⋮	⋮
u_{k-1}	0.1	0.2	0.3	**0.0**	**0.0**	**0.0**
$\boldsymbol{u_k}$	**0.0**	**0.0**	**0.0**	0.4	0.5	0.6
u_{k+1}	0.1	0.2	0.3	**0.0**	**0.0**	**0.0**
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Model	ε	Number of participants			
		10	100	500	1000
Linear	∞	1.00	1.00	1.00	1.00
	700	0.79	0.64	0.56	0.54
	500	0.75	0.60	0.55	0.54
	300	0.71	0.57	0.54	0.53
	100	0.61	0.54	0.52	0.51
Neural (16 hidden units)	∞	1.00	1.00	1.00	1.00
	700	0.81	0.63	0.56	0.54
	500	0.77	0.60	0.54	0.53
	300	0.73	0.58	0.52	0.51
	100	0.61	0.54	0.51	0.51

Other defenses

- Check for data manipulation:
  - Cryptography: checksum
  - Heuristics: examine feature values

Other defenses

- Check for data manipulation:
  - Cryptography: checksum
  - Heuristics: examine feature values
- Minimize shared information:
  - Don’t share entire model updates

Other defenses

- Check for data manipulation:
  - Cryptography: checksum
  - Heuristics: examine feature values
- Minimize shared information:
  - Don’t share entire model updates
- Decentralized FL:
  - Peer-to-peer gossip

Conclusion

- RAIFLE: Proposes data manipulation – novel attack vector in interaction-based FL
  - Strong performance compared to existing baselines

Conclusion

- RAIFLE: Proposes data manipulation – novel attack vector in interaction-based FL
  - Strong performance compared to existing baselines
- Potential improvements:
  - Stealth
  - Other domains: Text
  - Combine with model manipulation

Conclusion

- RAIFLE: Proposes data manipulation – novel attack vector in interaction-based FL
  - Strong performance compared to existing baselines
- Potential improvements:
  - Stealth
  - Other domains: Text
  - Combine with model manipulation
- Code available at: https://github.com/dzungvpham/raifle