Not All Edges are Equally Robust: Evaluating the Robustness of Ranking-Based Federated Learning

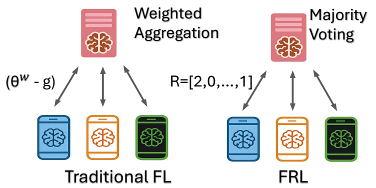
IEEE S&P

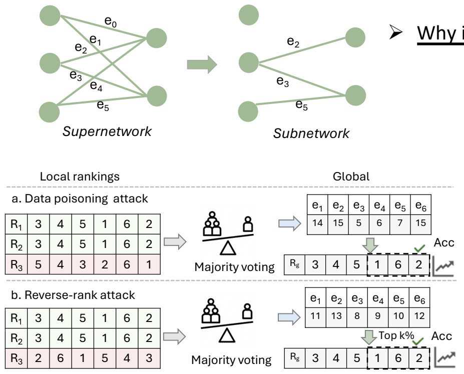
Zirui Gong[1], Yanjun Zhang[2], Leo Yu Zhang[1], Zhaoxi Zhang[2,1], Yong Xiang[3], and Shirui Pan[1]

[1] Griffith University, [2] University of Technology Sydney, [3] Deakin University

1. Problem statement

- FRL is a promising FL training framework, designed to address communication bottlenecks and improve system robustness against poisoning attacks.
- It shares edge ranking to server for aggregation.
- The server uses majority voting to get the global ranking.
- The global ranking is used to select the subnetwork.





Failure example of existing attacks.

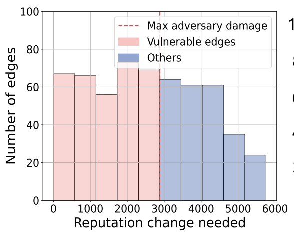
- **Why is it robust against client-side poisoning attacks?**
  - Discrete update narrows the potential space for malicious updates from an infinite range to n!, effectively bounding the adversary's damage within a defined budget.
  - Discrete updates also make existing optimization-based model poisoning attacks inapplicable directly.
  - It utilizes majority voting to get the global ranking. This approach prevents malicious clients from making significant adversarial modifications to the global model, as each client only has a single vote.

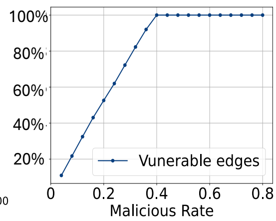
2. Our work

Main contributions:

- We conduct the first systematic analysis of FRL's robustness, uncovering a critical vulnerability within the framework.
- Based on the results of the analysis, we design and implement a new attack (VEM) that targets and effectively manipulates the vulnerable edges.
- Extensive experiments across different network architectures and datasets demonstrate that our VEM significantly outperforms SOTA attacks.

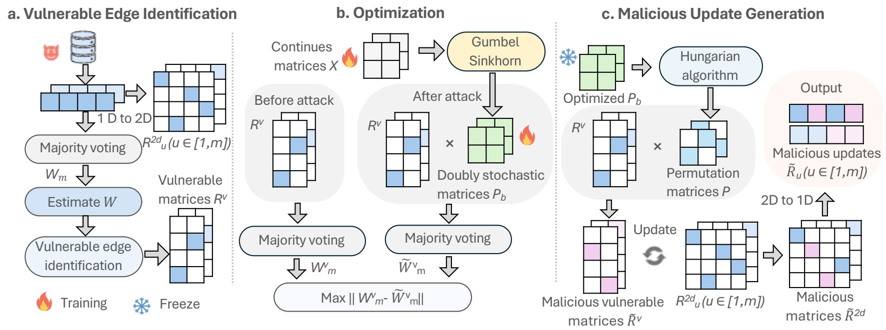
We observe that FRL is not inherently robust, and there are specific edges that are particularly vulnerable to poisoning attacks (vulnerable edges).





Vulnerable Edge Manipulation (VEM):

Our VEM unfolds in three main stages: vulnerable edge identification, optimization, and malicious update generation. In the first stage, the adversary aims to identify vulnerable edges within each layer using Theorem 1 . Once the vulnerable edges are identified, the adversary extracts the ranking of those vulnerable edges to form vulnerable matrices. In the optimization stage, the adversary aims to target those vulnerable edges and form the optimization function such that the global model's reputation of those vulnerable edges deviates significantly from their original values. To solve the optimization function, we use the Gumbel-Sinkhorn method to convert a discrete problem into a continuous problem. After the optimization process, we use the optimized parameter to generate malicious vulnerable matrices, which are then used to produce malicious updates.



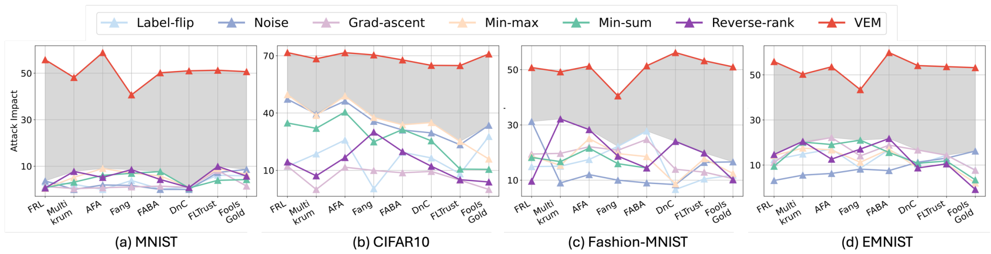
4. Results

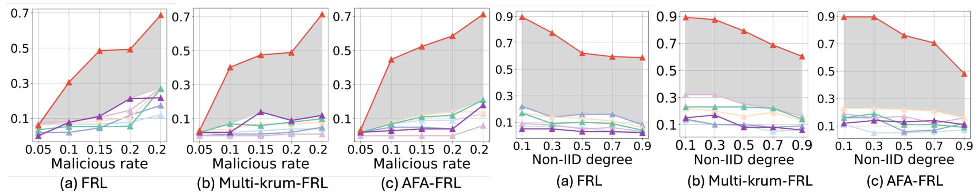
Defenses:

Category	AGRs	Applicability
Distance-based	Krum [31]	●
	Multi-Krum [31]	●
	AFA [26]	●
	FABA [28]	●
	DnC [11]	●
Dimension-wise	Median [27]	○
	Bulyan [24]	○
	TrMean [27]	○
Norm-bounded	Norm bound [32]	○
	Centered Clip [23]	○
Weighted-aggregation	FLTrust [19]	◒
	FLAIR [25]	◒
	FoolsGold [29]	◒
Validation-based	Fang [10]	◒
	FLDetector [21]	○

Comparison with the State-of-the-art attacks

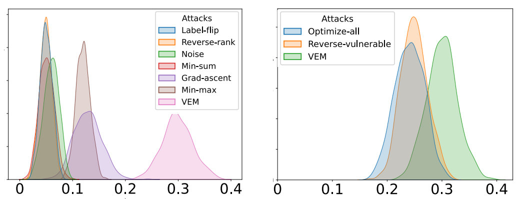
It achieves 53.23% attack impact and is 3.7x more impactful than others.





AGRs	Optimize-all	Reverse-vulnerable	VEM
FRL	23.96	24.79	**55.80**
Multi-krum	22.66	23.80	**48.17**
AFA	21.99	22.97	**58.88**
Fang	15.01	18.72	**47.70**
FABA	21.35	27.46	**50.19**
DnC	11.40	18.25	**51.03**
FLTrust	16.82	22.01	**46.67**
FoolsGold	20.21	24.05	**49.67**

Attack impact.



Edge cross rate.

4. Resources

Paper

Linkedin

Contact: zirui.gong@griffithuni.edu.au