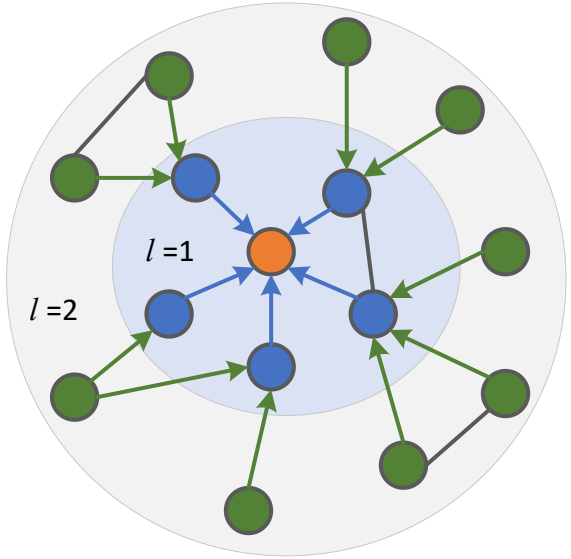
Distributed Private Aggregation in Graph Neural Networks

Huanhuan Jia †, Yuanbo Zhao †, Kai Dong*, Zhen Ling, Ming Yang, Junzhou Luo, Xinwen Fu

Background

- Graph Neural Networks (GNNs) can efficiently process graph-structured data.
- GNNs enhance node *representation* through neighborhood **aggregation**.



nodes: ● edges: →

◆ GNN **aggregation**

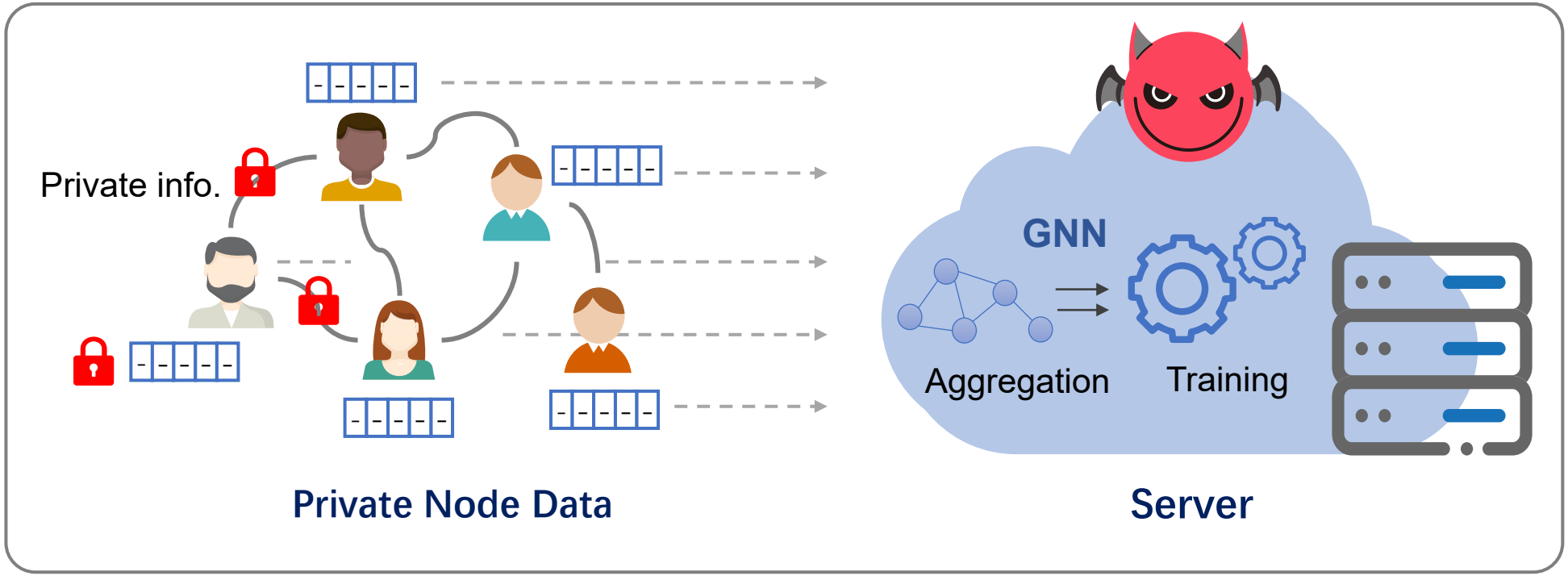
Social networks

Financial systems

Healthcare networks

Background (Cont'd)

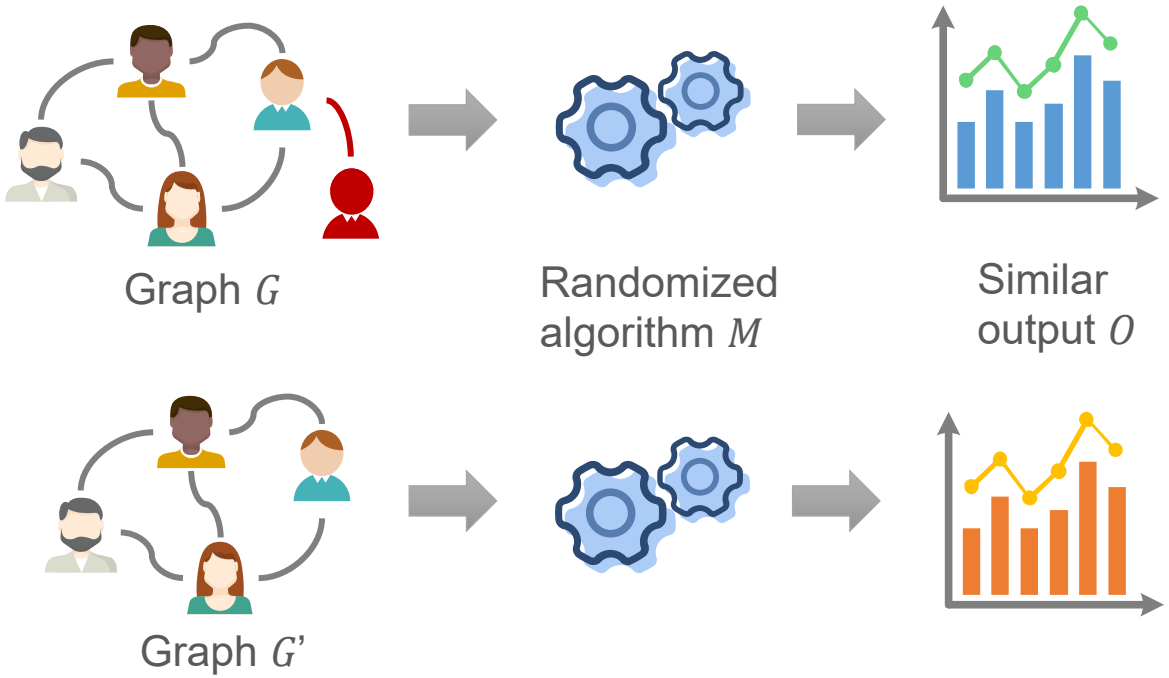
- Node attributes and edge information in graphs are inherently sensitive.
- Aggregation and training of GNNs on untrusted servers can pose severe privacy risks.



Node-level DP (or Edge-level DP)

A randomized algorithm M satisfies node-level (or edge-level) Differential Privacy (DP) if and only if, for any two adjacent graphs G and G' that only differ by one node (or one edge), and for any possible outcome O, we have:

$$\Pr[M(G) \in O] \leq e^{\varepsilon} \Pr[M(G') \in O] + \delta.$$



ε represents *privacy budget*.

$\varepsilon \downarrow$, privacy $\uparrow$, utility $\downarrow$

$\varepsilon \uparrow$, privacy $\downarrow$, utility $\uparrow$

- **Edge-level DP** protects only the **edges**.
- **Node-level DP** protects node **edges**, **features**, and **labels**.

Node-level DP in Distributed Settings

- Existing privacy-preserving GNN methods

	Related Work	Features	Edges	Labels
LDP GNN methods	Blink [CCS' 23]		√	
	LPGNN [CCS' 21]	√		√
	Solitude [TIFS' 22]	√		√
	RGNN [SDM' 24]	√	√	
	DPRR [Trans' 24]		√	
Centralized DP GNN methods	GAP-NDP [Security' 23]	√	√	√
	PNPiGNNs [S&P' 24]	√	√	√
	DPDGC [NeurIPS' 23]	√	√	√

- Locally perturbing all node information introduces substantial noise and degrades model performance.
- Node-level DP designed for centralized settings is not suitable for distributed scenarios.

Problem

- Target Scenario in Distributed Settings

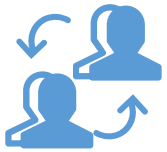
  An untrusted server trains a GNN across distributed users.

  All node-held features X, labels Y, and edges (adjacency matrix A) are considered private.

- Privacy Constraints

  Edge-level (ε, δ)-DP Node-level (ε, δ)-DP

- Trust Assumptions

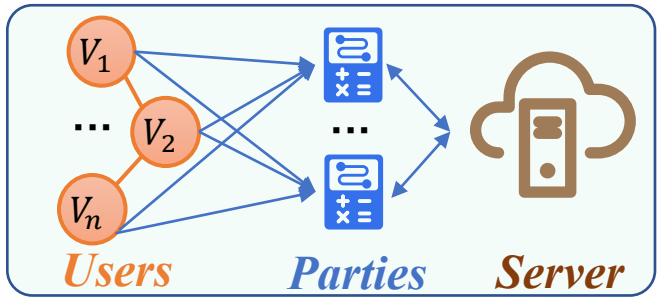
  - The users, parties, and server are semi-honest.
  - Users and parties can communicate anonymously with each other.

- Problem Definition

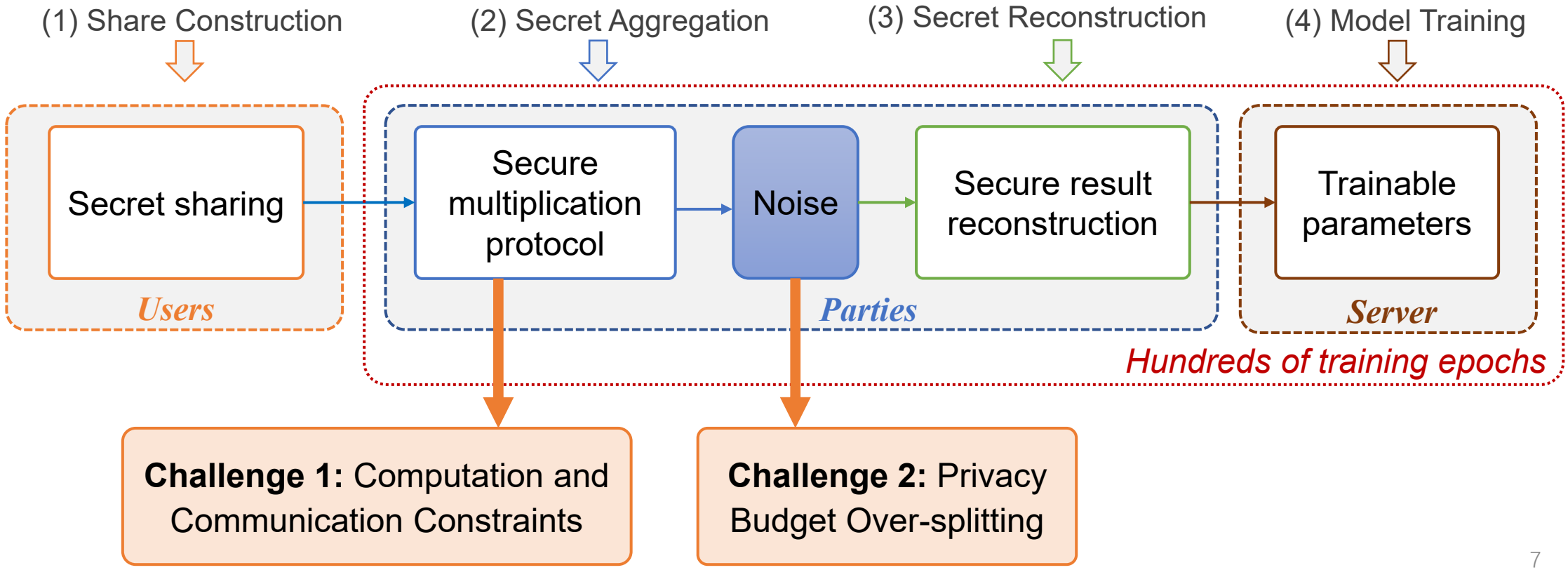
  

  Given the above settings, how can the untrustworthy server construct an effective GNN under DP constraints?

Basic Idea and Challenges



- Our basic idea is to employ Secure Multi-Party Computation (MPC) protocols to build a GNN in distributed settings under DP constraints. —— a new research problem



Challenges

Challenge 1: Computation and Communication Constraints.

- GNN aggregation involves heavy matrix multiplication, causing substantial computation and communication overhead under MPC protocols.

Challenge 2: Privacy Budget Over-splitting.

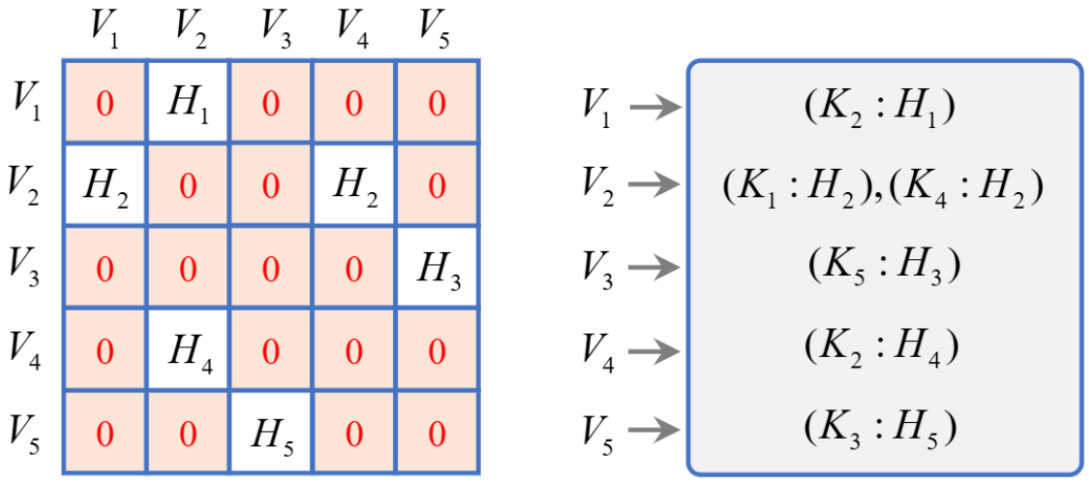
- Since GNN training runs for hundreds of epochs and repeatedly accesses private edges, the privacy budget is split hundreds of times, significantly degrading model performance.

Modifications Addressing Challenge 1

- Target: Reducing the computation and communication overhead.

  ✓ Modification 1. Matrix addition instead of matrix multiplication.

  ✓ Modification 2. Key-value instead of a sparse matrix.



- H_i: the feature vector of node V_i
- K_j: the index of node V_i's neighbor

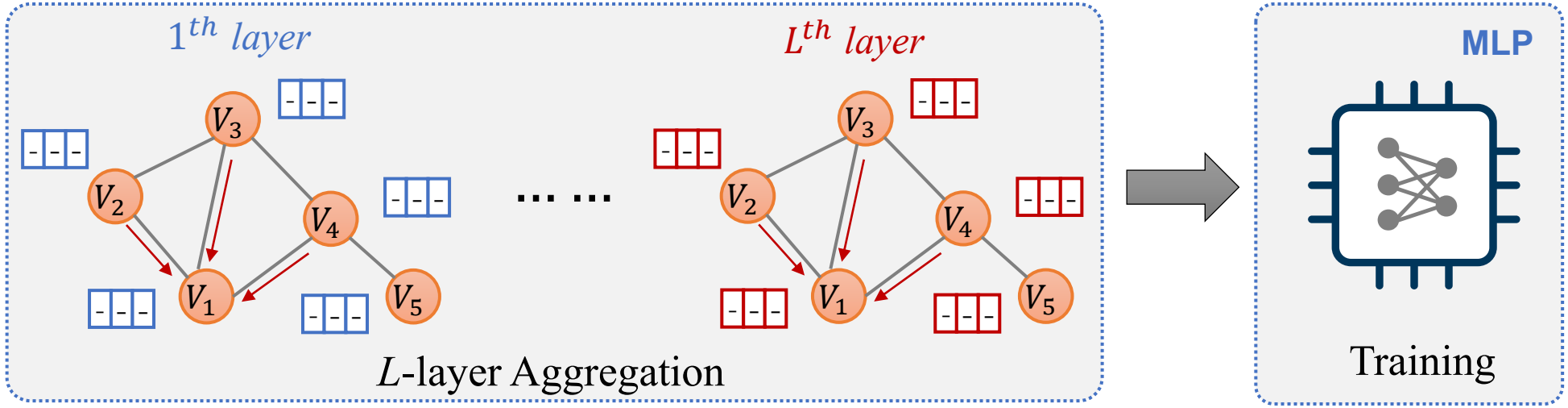
The feature passing matrix with 25 vectors is transformed into 6 key-value pairs.

✓ Modification 3. Achieving DP defined on edges via dummies.

Modifications Addressing Challenge 2

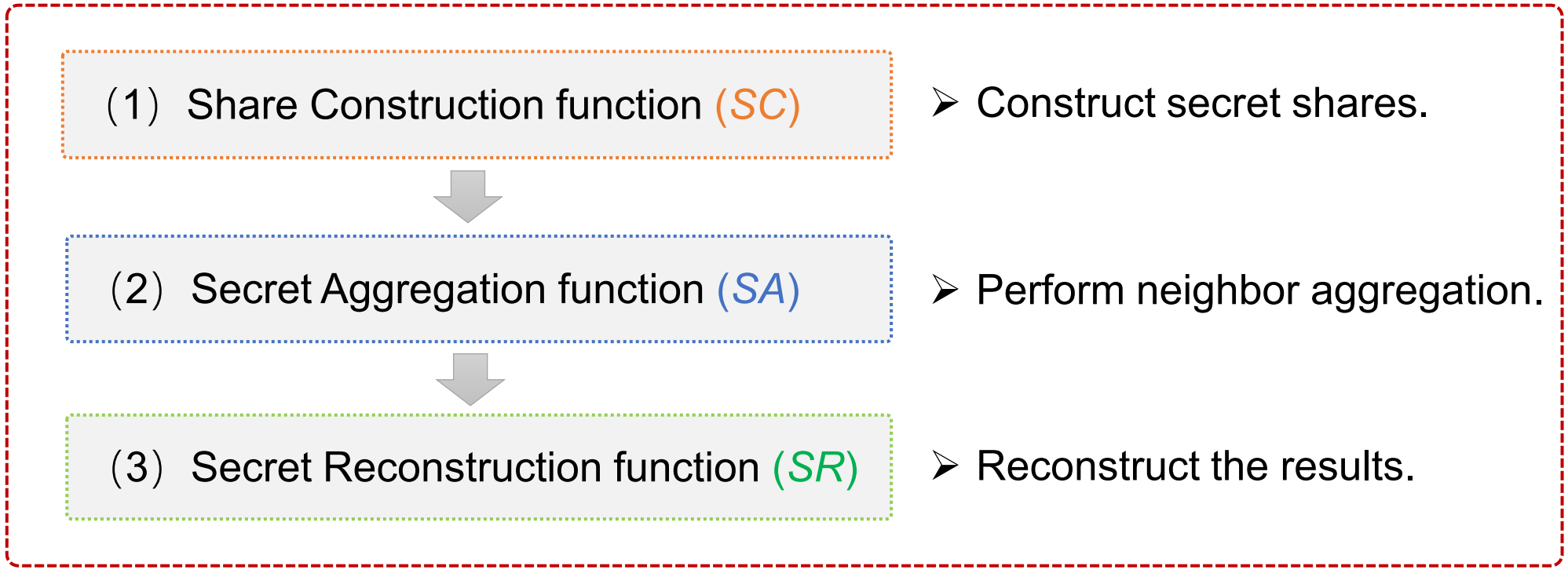
- Target: Reducing privacy budget over-splitting during training.

  - ✓ Modification 4. Separating the aggregation (satisfying DP) and training stages.
  - ✓ Modification 5. Secret reconstruction on the user side.
  - ✓ Modification 6. Aggregating the perturbed labels.



DPA: Distributed Private Aggregation

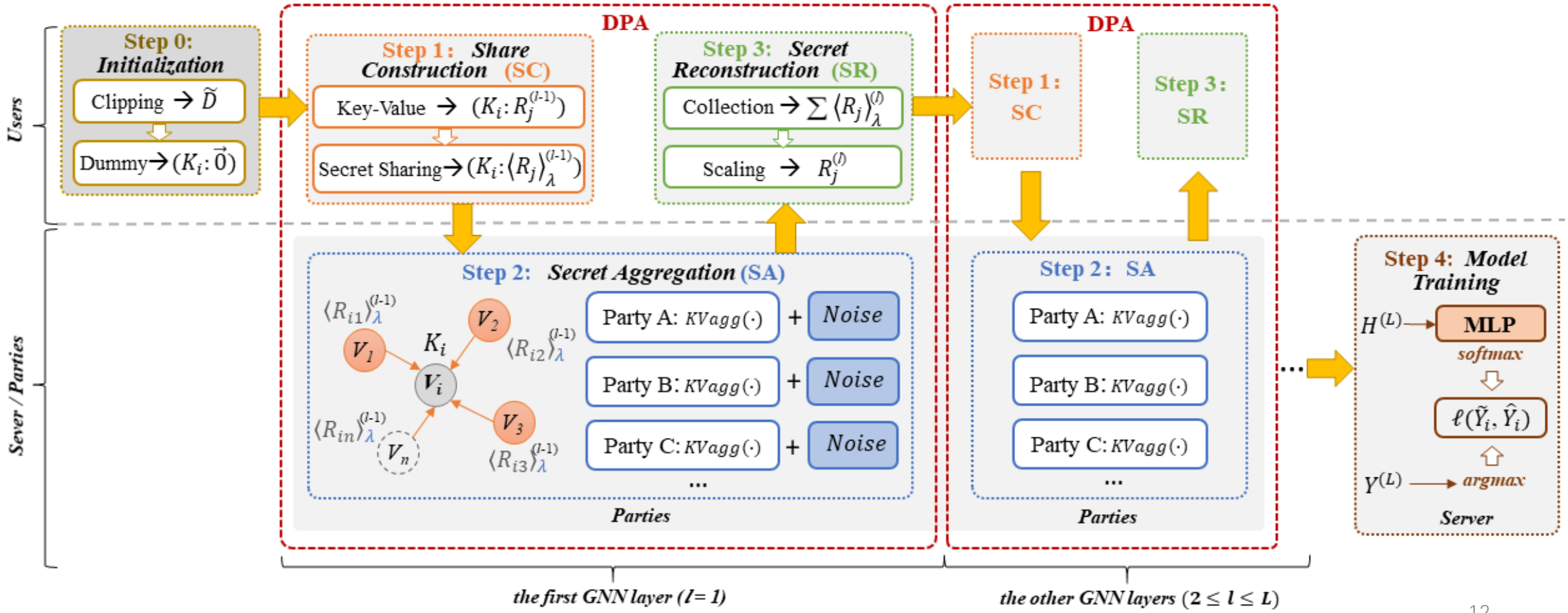
- We conclude these modifications and propose the DPA method, which can be abstracted as the fundamental aggregation function. The DPA function is composed of three core subfunctions.



DPA function: $\mathrm{DPA}\left(\mathcal{R}_{ij}^{(l-1)}\right) \triangleq SR\left(SA\left(\left\{SC\left(\mathcal{R}_{ij}^{(l-1)}\right)\right\}\right)\right)$ where R_{ij} represents key-value pairs.

Workflow of DPA-GNN

- The DPA-GNN consists of five main modules: Initialization, Share Construction (SC), Secret Aggregation (SA), Secret Reconstruction (SR), and Model Training.



Analysis

- Privacy Analysis
  - DPA-GNN satisfies edge-level ε_e -DP and node-level $(\varepsilon_e + \varepsilon_h + \varepsilon_y, 2\delta)$- DP. ε_e, ε_h and ε_y denote the privacy budgets for edges, features, and labels, respectively.

- Communication and Offline Setup Costs

	Communication		
	User-Party (per User)	Party-Party (Training)	Offline Computation
Initial	$O(m(N+C))$	$O(mkT(N^2+NC))$	$O(m(N+C))$
DPA-GNN	$O(kdtC)$	$O(mNC)$	$O(kdtC)$

C: the dimension of features and labels
k: the number of GNN layers
d: the number of key-value pairs
t: secret shares
m: the number of parties
N: the number of users
T: the number of epochs

Evaluation

- Datasets

Dataset	Nodes	Features	Edges	Classes
Cora [53]	2708	1433	10858	7
CiteSeer [53]	3312	3703	9430	6
LastFM [38]	7624	7842	55612	18
Facebook [37]	22470	4714	342004	4
Amazon [44]	13471	767	491772	10
Reddit [56]	140667	602	51605960	12

- Baselines

	Baselines	Features	Edges	Labels
Category 1.	LDP GNN methods			
Category 2. Enhanced variants of existing methods	Solitude+RR	√	√	√
	LPGNN+RR	√	√	√
	RGNN+RR	√	√	√
Category 3.	Centralized DP GNN methods			
DPA-GNN		√	√	√

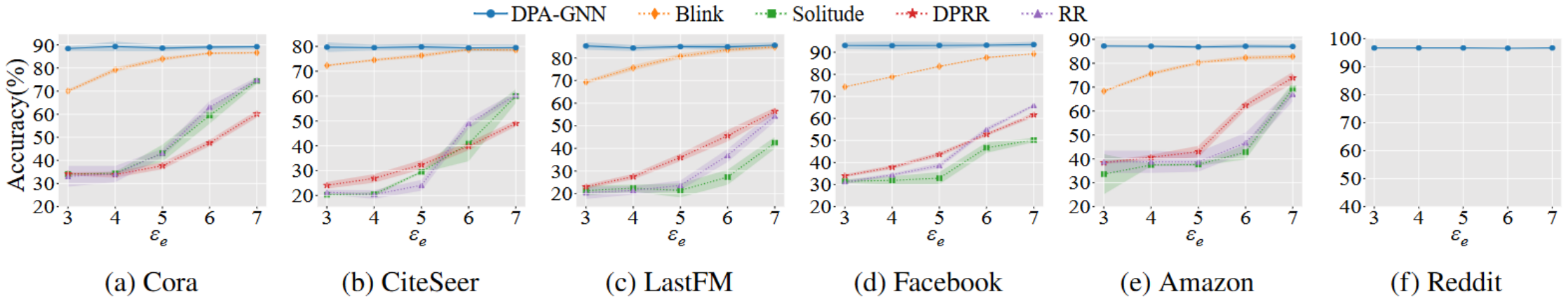
Evaluation (Cont'd)

- Experiments in utility-privacy trade-off

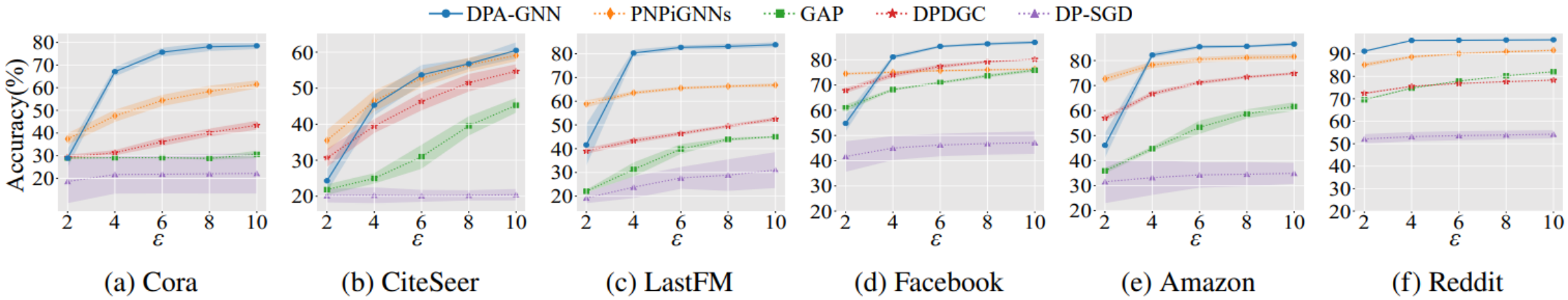
	Method	$\varepsilon=2$	$\varepsilon=4$	$\varepsilon=6$	$\varepsilon=8$	$\varepsilon=10$	$\varepsilon=2$	$\varepsilon=4$	$\varepsilon=6$	$\varepsilon=8$	$\varepsilon=10$
		Cora					CiteSeer				
ε_h=5%ε	Solitude+RR	29.0±4.40	30.3±2.21	30.9±1.57	30.5±0.99	62.8±3.00	20.7±1.48	20.7±1.67	20.2±1.19	22.6±3.31	40.8±1.11
	LPGNN+RR	21.5±7.53	16.7±9.66	18.8±9.09	30.5±6.54	59.7±5.26	15.9±2.81	17.1±4.06	17.2±2.78	19.6±1.64	38.0±3.46
	RGNN+RR	29.5±1.02	29.5±1.04	29.5±1.01	29.2±1.28	50.9±1.38	19.8±1.34	20.4±1.19	20.3±1.14	21.6±1.36	31.3±1.77
	DPA-GNN	**27.1**±3.97	**66.8**±2.07	**74.5**±1.95	**77.4**±1.94	**78.4**±1.42	**24.9**±3.05	**44.8**±2.88	**52.9**±2.73	**56.1**±1.43	**60.5**±2.23
		LastFM					Facebook				
ε_h=5%ε	Solitude+RR	13.9±7.32	11.7±6.37	14.0±6.67	12.9±7.43	44.1±5.18	30.1±0.81	30.6±1.13	29.9±0.95	35.5±6.50	72.0±0.98
	LPGNN+RR	8.7±7.73	9.0±7.65	11.6±6.76	17.8±4.47	38.6±12.81	25.9±5.89	28.4±2.30	28.1±2.52	38.2±7.94	60.0±7.51
	RGNN+RR	19.8±2.01	20.3±1.70	20.3±1.56	25.7±2.08	57.2±1.29	30.4±0.56	30.9±0.78	30.9±0.43	33.6±2.68	50.1±1.87
	DPA-GNN	**41.6**±8.73	**80.3**±1.46	**82.7**±0.86	**83.1**±1.06	**83.8**±1.01	**54.8**±3.28	**81.1**±0.85	**84.9**±0.46	**85.8**±0.59	**86.1**±0.59
		Amazon					Reddit				
ε_h=5%ε	Solitude+RR	18.7±13.20	37.1±0.71	37.3±0.44	55.0±11.72	73.2±1.13	–	–	–	–	–
	LPGNN+RR	12.0±12.66	21.7±12.67	29.7±11.67	52.6±16.47	71.7±1.60	–	–	–	–	–
	RGNN+RR	35.8±0.64	36.8±0.73	35.7±2.97	58.8±3.74	70.7±1.71	–	–	–	–	–
	DPA-GNN	**46.2**±4.45	**82.6**±0.91	**85.7**±0.52	**85.6**±0.43	**85.9**±0.74	**91.2**±0.38	**96.0**±0.15	**96.1**±0.05	**96.2**±0.07	**96.3**±0.10

Evaluation (Cont'd)

- Ablation Study: protecting edges only.



- Scenario Transition: Performance in centralized settings.



Conclusion

DPA-GNN: Distributed Private Aggregation in Graph Neural Networks

- We propose DPA, the first GNN aggregation method ensuring node-level DP in distributed settings.
- We implement DPA-GNN based on DPA, which surpasses existing privacy-preserving GNN methods.

Future Directions: Dynamic Graph Privacy

- Temporal updates in dynamic graphs increase privacy risks and budget consumption.
- Future work will develop more efficient DP mechanisms for dynamic GNNs.

THANK YOU!

Questions: hhjia@seu.edu.cn
zyb.seu@gmail.com