Verifiable Byzantine Robust Graph Neural Networks using Federated Learning

2005090 - Tawkir Aziz Rahman
2005074 - Dipanto Kumar Roy Nobo

CSE472: Machine Learning Sessional
Department of Computer Science and Engineering
Bangladesh Univerrsity of Engineering and Technology (BUET)

December 2025

Problem & Motivation

Problem Statement

Graph Neural Networks (GNNs) are vulnerable to adversarial attacks and fail catastrophically when deployed in federated learning settings with Byzantine (malicious) participants.

Challenges:

- **Privacy**: Multiple parties need to collaborate without sharing sensitive graph data
- **Security**: Malicious clients can poison both graph structure and model parameters
- **Robustness**: Existing defenses fail under adaptive attacks

Real-World Applications:

- Healthcare: Disease networks across hospitals
- Finance: Fraud detection across banks
- Social Networks: Privacy-preserving community detection
- IoT/Cybersecurity: Distributed attack detection

Background & Related Work

Base Paper: RUNG (NeurIPS 2024)

- **Problem**: ℓ_1-based robust GNNs suffer from estimation bias
- **Solution**: Minimax Concave Penalty (MCP) for unbiased aggregation
- **Key Innovation**: Quasi-Newton IRLS algorithm with convergence guarantees

RUNG Aggregation

$$F^{(k+1)} = (\text{diag}(q^{(k)})+\lambda I)^{-1}[(W^{(k)}\odot\tilde{A})F^{(k)}+\lambda F^{(0)}]$$

where $W_{ij}^{(k)} = \max(0, \frac{1}{y_{ij}^{(k)}} - \frac{1}{\gamma})$

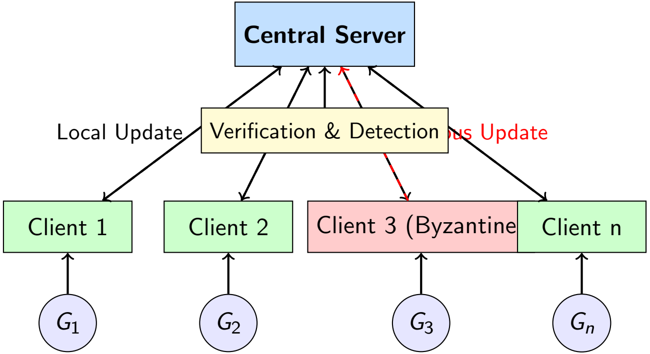
Federated Learning Challenges

- **Byzantine Attacks**: Up to f out of n clients are malicious
- **Data Heterogeneity**: Non-IID graph distributions
- **Communication Cost**: Iterative model exchanges

Byzantine-Robust FL Methods

- *Krum*: Distance-based filtering
- *Median/Trimmed Mean*: Coordinate-wise aggregation
- *BRIDGE*: Bucketing with averaging

Proposed Approach: System Architecture



Key Components

1. **Local Training**: Each client trains RUNG on local graph G_i
2. **Verifiable Aggregation**: Cryptographic proofs ensure correct local computation
3. **Byzantine Detection**: Statistical tests identify malicious clients

Proposed Approach: Technical Innovations

1. Verifiable Aggregation Protocol

- **Commitment Scheme**: Client commits to local edge weights $W_i^{(k)}$
- **Zero-Knowledge Proof**: Proves $W_i^{(k)}$ computed correctly without revealing graph structure
- **Lightweight Verification**: Server verifies proofs in $O(1)$ per client

2. Byzantine Detection Mechanism

- **Distance-Based Filtering**: Compute pairwise distances between client updates
- **Statistical Outlier Test**: Detect updates with abnormal magnitudes or directions
- **Adaptive Weighting**: $\alpha_i \propto \exp(-\text{dist}(\theta_i, \text{median}))$

Local Update with Proof

$$\theta_i^{(t+1)} = \theta_i^{(t)} - \eta \nabla \mathcal{L}(F_i^{RUNG}, y_i; \theta_i^{(t)})$$

$$\pi_i = \text{ZKP}(W_i^{(k)} \text{ satisfies MCP})$$

Global Aggregation

$$\theta^{(t+1)} = \frac{\sum_{i=1}^{n} \alpha_i \cdot \theta_i^{(t+1)}}{\sum_{i=1}^{n} \alpha_i}$$

where $\alpha_i = 0$ if client i fails verification

Experimental Plan

Datasets

Dataset	Nodes	Edges
Cora	2,708	5,429
CiteSeer	3,327	4,732
PubMed	19,717	44,338
ogbn-arxiv	169,343	1,166,243

Federated Setup

- **Clients**: $n = \{10, 20, 50\}$
- **Byzantine Ratio**: $f/n = \{0\%, 10\%, 20\%, 40\%\}$
- **Data Split**: Dirichlet distribution ($\alpha = 0.5$) for non-IID

Baselines

1. **FedAvg**: Standard federated averaging
2. **FedProx**: Proximal regularization
3. **Krum**: Distance-based Byzantine-robust FL
4. **Median/TrimmedMean**: Coordinate-wise aggregation
5. **RUNG (centralized)**: Upper bound performance

Evaluation Metrics

- **Accuracy**: Node classification accuracy vs Byzantine ratio
- **Attack Detection Rate**:

Timeline & Expected Contributions

Project Timeline (12 Weeks)

Week	Milestone
1-2	Literature review & problem formulation
3-4	Design verification protocol & detection
5-6	Implement Fed-RUNG framework
7-8	Implement baselines & datasets
9-10	Run experiments & collect results
11	Analyze results & ablation studies
12	Paper writing & final presentation

Team Responsibilities

- **Member 1**: Verification protocol, theoretical analysis
- **Member 2**: Implementation, experiments,

Expected Contributions

1. **Novel Algorithm**: First verifiable Byzantine-robust federated GNN
2. **Theoretical Analysis**: Convergence guarantees under Byzantine attacks
3. **Empirical Validation**: Comprehensive experiments on multiple datasets
4. **Open-Source**: Release code for reproducibility

Target Venue

NeurIPS 2026 or ICML 2026

References

Hou, Z., Feng, R., Derr, T., & Liu, X. (2024).
Robust Graph Neural Networks via Unbiased Aggregation.
Advances in Neural Information Processing Systems (NeurIPS), 37.

McMahan, B., et al. (2017).
Communication-Efficient Learning of Deep Networks from Decentralized Data.
AISTATS.

Blanchard, P., et al. (2017).
Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent.
NeurIPS.

Blanchard, P., et al. (2017).
Byzantine-Tolerant Machine Learning.
arXiv preprint.

Zügner, D., et al. (2018).
Adversarial Attacks on Neural Networks for Graph Data.
KDD.