# 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 University of Engineering and Technology (BUET)

December 2025

# Problem & Motivation

## Problem Statement

Training AI models on graph data across multiple organizations is challenging due to security threats and privacy concerns.

**Key Challenges:**

- **Privacy**: Organizations can't share sensitive data
- **Security**: Malicious participants can corrupt the model
- **Trust**: Hard to verify if participants are honest

**Real-World Applications:**

- Healthcare: Disease networks across hospitals
- Finance: Fraud detection across banks
- Social Networks: Privacy-preserving analysis
- Cybersecurity: Distributed threat detection

## Our Goal

Develop a secure and verifiable system for collaborative graph-based machine learning.

# Background & Related Work

## Foundation: RUNG (NeurIPS 2024)

- Robust method for handling graph data
- Reduces bias in predictions
- Strong performance guarantees

## Why RUNG?

- State-of-the-art robustness
- Handles noisy connections effectively
- Proven theoretical guarantees

## Collaborative Learning Challenges

- **Malicious Participants**: Some clients may be dishonest
- **Data Diversity**: Different data distributions across clients
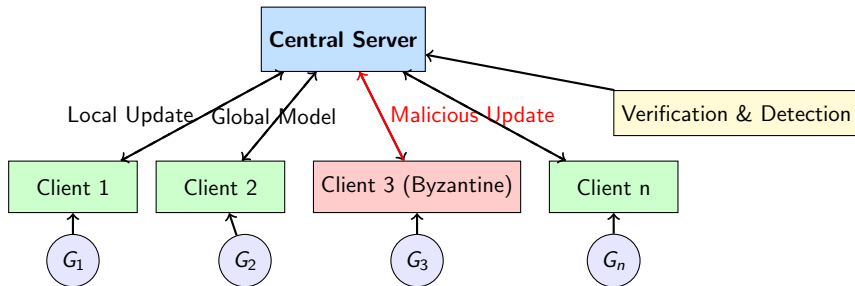- **Efficiency**: Need to minimize communication overhead

## Existing Defense Methods

- Distance-based filtering
- Statistical aggregation
- Averaging techniques

## Limitation

Current methods don't work well with graph data and lack verification.

# Proposed Approach: System Architecture



```
                          Central Server
            Local Update  Global Model    Malicious Update    Verification & Detection

     Client 1      Client 2      Client 3 (Byzantine)      Client n

       G₁            G₂              G₃                        Gₙ
```

## 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

4. **Robust Update**: Weighted aggregation down-weights suspicious updates

# Proposed Approach: Key Innovations

## 1. Verifiable Training

- Each client provides cryptographic proof of honest computation
- Server can verify correctness without accessing private data
- Fast and efficient verification process

### Benefits

- Ensures data privacy
- Detects dishonest behavior
- Low computational overhead

## 2. Malicious Client Detection

- Compare updates from different clients
- Identify suspicious or outlier behavior
- Automatically down-weight malicious contributions

### Aggregation Strategy

- Trusted clients get higher weight
- Suspicious clients are excluded
- Adaptive to attack patterns

### Guarantee

Our system remains accurate even when up to 30% of participants are malicious.

# Experimental Plan

**Benchmark Datasets**

- Cora - Small academic citation network
- CiteSeer - Medium citation network
- PubMed - Large biomedical network
- ogbn-arxiv - Very large academic network

**Testing Scenarios**

- Vary number of clients (10 to 50)
- Test with different malicious ratios (0% to 40%)
- Simulate realistic data distributions

**Comparison Methods**

- Standard federated learning
- Existing robust methods
- Centralized training (ideal case)

**What We'll Measure**

- Prediction accuracy under attacks
- Ability to detect malicious clients
- Communication efficiency
- Training speed and convergence

## Additional Analysis

We will study the impact of verification overhead and detection sensitivity.

# Timeline & Expected Contributions

**Expected Outcomes**

1. A novel secure training system
2. Strong performance guarantees
3. Comprehensive experimental validation
4. Open-source code release

## Publication Target

NeurIPS 2026

## Impact

Enables secure collaboration for sensitive graph data across organizations.