Gradient Rewiring for Editable Graph Neural Network Training

Zhimeng Jiang[1], Zirui Liu[2], Xiaotian Han*[3], Qizhang Feng[1], Hongye Jin[1], Qiaoyu Tan[4], Kaixiong Zhou[5], Na Zou[6], Xia Hu[7]

[1]*Texas A&M University*, [2]*Case Western Reserve University*, [3]*University of Minnesota*,
[4]*NYU Shanghai*, [5]*North Carolina State University*, [6]*University of Houston*, [7]*Rice University*

Research Motivation

- Deep neural networks, including GNNs, can suffer significant performance degradation due to prediction errors when real-world data changes, resulting in critical misclassifications.
- Current model editing techniques focus primarily on computer vision and NLP, with limited exploration of editable training for GNNs.
- **Key question:** Can we develop an effective method to edit GNNs that ensures corrections for erroneous predictions while maintaining model stability across unaffected nodes? If so, how?

Why Gradient Rewiring?

Preliminary experiments show that direct fine-tuning of GNNs for model editing can lead to a significant increase in training loss, indicating performance degradation.

- **Statement:** There is a considerable gradient discrepancy between the target and training data, causing higher degradation for GNNs compared to MLPs.
- **Insight:** A method is needed to maintain training performance during model editing, motivating the development of a gradient rewiring approach.

Gradient Rewiring Method

- **Problem Formulation:** Model editing aims to fix prediction errors at the target node while preserving performance on training nodes: (1) the training loss should not exceed its value prior to model editing (see Eq. (2)); and (2) the differences in model predictions after editing should remain within a predefined range (see Eq. (3)).

$$\min_{\theta} \mathcal{L}_{tg}(f_{\theta}(\mathbf{x}_{tg}), y_{tg}) \tag{1}$$

$$\text{s.t. } \mathcal{L}_{train}(f_{\theta'}, \mathcal{V}_{train}) \leq \mathcal{L}_{train}(f_{\theta_0}, \mathcal{V}_{train}) \tag{2}$$

$$\|\frac{1}{|\mathcal{V}_{train}|} \sum_{i \in \mathcal{V}_{train}} f_{\theta'}(\mathbf{x}_i) - f_{\theta_0}(\mathbf{x}_i)\|^2 \leq \delta', \tag{3}$$

- **Problem Solver:** (1) Approximation: Use Taylor expansion to estimate the influence of the model's parameters for both the target prediction and the training performance. (2) Transforming into Gradient Optimization (3) Solution via Dual Optimization: Solve the gradient adjustment problem more efficiently by converting it into a simpler form in the dual space.

Algorithm 1 Gradient Rewiring Editable (GRE) Graph Neural Networks Training

Input: Target samples $(\mathbf{x}_{\text{tg}}, \mathbf{y}_{\text{tg}})$, hyperparameter λ, well-trained GNN model $f_{\theta}(\cdot)$, and its corresponding gradient for the training subgraph.
Output: Updated GNN model $f_{\theta'}(\cdot)$.
while $f_{\theta}(\mathbf{x}_{\text{tg}}) \neq \mathbf{y}_{\text{tg}}$ **do**
  Compute the model gradient g_{tg} for the target loss $\mathcal{L}_{\text{tg}}$.
  Rewire the target loss gradient g_{tg} by reducing the projection component on g_{train}, then scale with $(1+\lambda)^{-1}$:
    $g^* = (1+\lambda)^{-1}(g_{\text{tg}} - v^* g_{\text{train}})$.
  Replace g_{tg} with g^* and update the model parameters using the optimizer to obtain θ'.
end while

Experiment Results

- **Experimental Results in the Independent Editing Setting** (a) Our proposed GRE and GRE+ notably surpass both GD and ENN in terms of test drawdown; (b) Our proposed GRE and GRE+ are compatible with EGNN and further improve the performance.

	Editor	Cora			A-computers			A-photo			Coauthor-CS		
		Acc↑	DD↓	SR↑	Acc↑	DD↓	SR↑	Acc↑	DD↓	SR↑	Acc↑	DD↓	SR↑
MLP	GD	68.15±0.33	3.85±0.33	0.98	**73.22**±0.48	**6.78**±0.48	1.00	**83.19**±0.91	**6.81**±0.91	1.00	93.59±0.05	0.41±0.05	1.00
	ENN	37.16±3.80	52.24±4.76	1.00	15.51±10.99	72.36±10.87	1.00	16.71±14.81	77.07±15.20	1.00	4.94±3.78	89.43±3.34	1.00
	GRE	69.41±0.44	2.59±0.44	0.96	61.21±1.26	18.79±1.26	1.00	73.56±1.41	16.44±1.41	1.00	93.27±0.09	0.73±0.09	1.00
	GRE+	**71.19**±0.28	**0.61**±0.28	0.96	61.27±1.15	18.73±1.15	1.00	78.26±1.15	11.74±1.15	1.00	**93.73**±0.07	**0.27**±0.07	1.00
GCN	GD	84.37±5.84	5.03±6.40	1.00	44.78±22.41	43.09±22.32	1.00	28.70±21.26	65.08±20.13	1.00	91.07±3.23	3.30±2.22	1.00
	ENN	37.16±3.80	52.24±4.76	1.00	15.51±10.99	72.36±10.87	1.00	16.71±14.81	77.07±15.20	1.00	4.94±3.78	89.43±3.34	1.00
	GRE	84.98±0.47	4.02±0.47	0.96	46.28±3.47	51.72±3.47	0.98	35.88±2.26	58.12±2.26	0.99	89.46±0.29	4.54±0.29	1.00
	GRE+	**88.84**±0.35	**0.56**±0.35	0.98	**47.75**±0.45	**40.25**±0.45	1.00	**50.13**±1.36	**43.87**±1.36	1.00	**91.99**±0.30	**2.01**±0.30	1.00
Graph-SAGE	GD	82.06±4.33	4.54±5.32	1.00	21.68±20.98	61.15±20.33	1.00	38.98±30.24	55.32±29.35	1.00	90.15±5.58	5.01±5.32	1.00
	ENN	33.16±1.45	53.44±2.23	1.00	16.89±16.98	65.94±16.75	1.00	15.06±11.92	79.24±11.25	1.00	13.71±2.73	81.45±2.11	1.00
	GRE	83.64±0.20	3.36±0.20	1.00	20.11±2.30	62.89±2.30	0.96	41.96±1.57	52.04±1.57	0.98	91.07±0.44	3.93±0.44	1.00
	GRE+	**86.59**±0.07	**0.41**±0.07	1.00	**22.23**±1.60	**60.77**±1.60	0.97	**44.05**±0.83	**50.32**±0.83	1.00	**91.75**±0.43	**3.25**±0.43	1.00
EGNN-GCN	GD	87.58±0.31	1.42±0.31	1.00	87.27±0.14	0.73±0.14	0.78	93.24±0.59	0.76±0.59	0.77	93.99±0.02	0.01±0.02	0.91
	GRE	87.47±0.41	1.53±0.41	1.00	83.38±1.20	4.62±1.20	0.87	88.01±1.20	5.99±1.20	0.86	93.92±0.07	0.08±0.07	0.94
	GRE+	**88.99**±0.21	**0.05**±0.21	1.00	**88.10**±1.21	**0.51**±1.21	1.00	**94.22**±0.98	−**0.21**±0.98	1.00	**94.32**±0.06	−**0.32**±0.06	1.00
EGNN-SAGE	GD	85.05±0.11	0.95±0.11	1.00	85.93±0.08	0.07±0.08	0.90	93.87±0.20	0.13±0.20	0.81	95.0±0.01	0.00±0.01	0.99
	GRE	84.79±0.19	1.21±0.19	1.00	81.94±1.71	4.06±1.71	0.96	88.55±1.19	5.45±1.19	0.95	94.85±0.05	0.15±0.05	1.00
	GRE+	**86.24**±1.43	−**0.24**±1.43	1.00	**85.97**±0.83	−**0.16**±0.83	1.00	**94.07**±0.03	−**0.07**±0.03	0.98	**95.07**±0.03	−**0.07**±0.03	1.00

- **Experimental Results in the Sequential Editing Setting.** (a) The proposed GRE and GRE+ consistently outperform GD in the sequential setting. (b) The improvement of GRE+ over GRE is quite limited in the sequential setting.

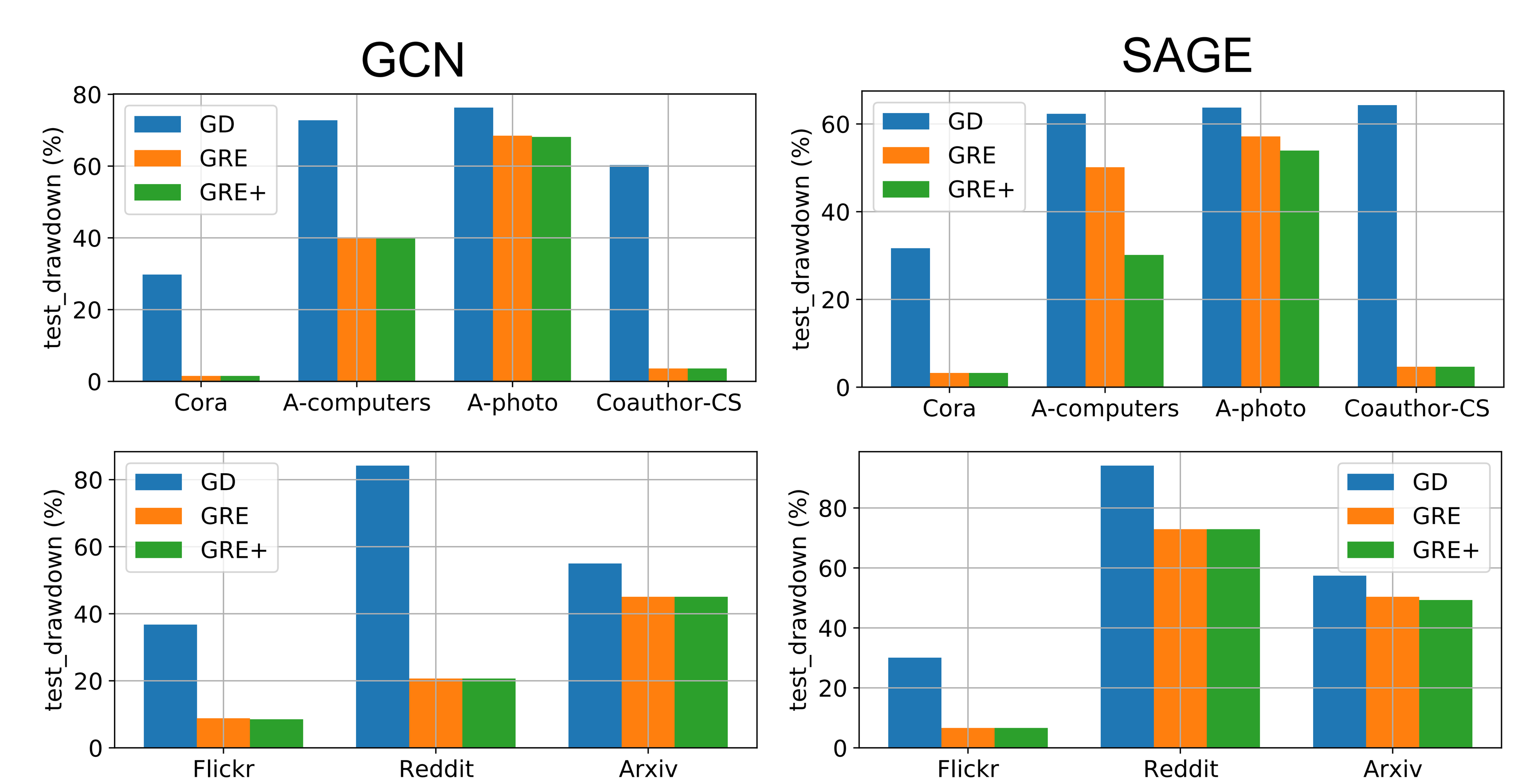

Figure: The test accuracy drawdown in sequential editing setting for GCN and GraphSAGE on various datasets. The units for y-axis are percentages (%).

Acknowledgement: The work is in part supported by NSF grants NSF IIS-2310260, IIS-2224843, IIS-2450662, IIS-2431515 and IIS-2239257.