Data Analytics and Machine Learning Group
Department of Informatics
Technical University of Munich

Certifiable Robustness to Graph Perturbations

Aleksandar Bojchevski, Stephan Günnemann
www.daml.in.tum.de/graph-cert/

TL;DR: First certificate w.r.t. graph perturbations for a general class of models including Label Prop. and GNNs.

GNNs are vulnerable to Adversarial Attacks

Semi-Supervised Node Classification: Given a few labelled nodes predict the classes of the remaining nodes in the graph
Targeted Attack: Perturb the graph to misclassify a target node

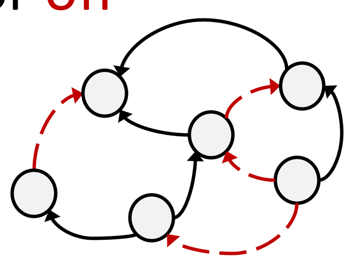
Research Questions

Certification: How to verify if a graph-based model is robust?
Robust Training: How can we improve certified robustness?

Flexible Threat Model

Attacker controls fragile edges they can turn on or off

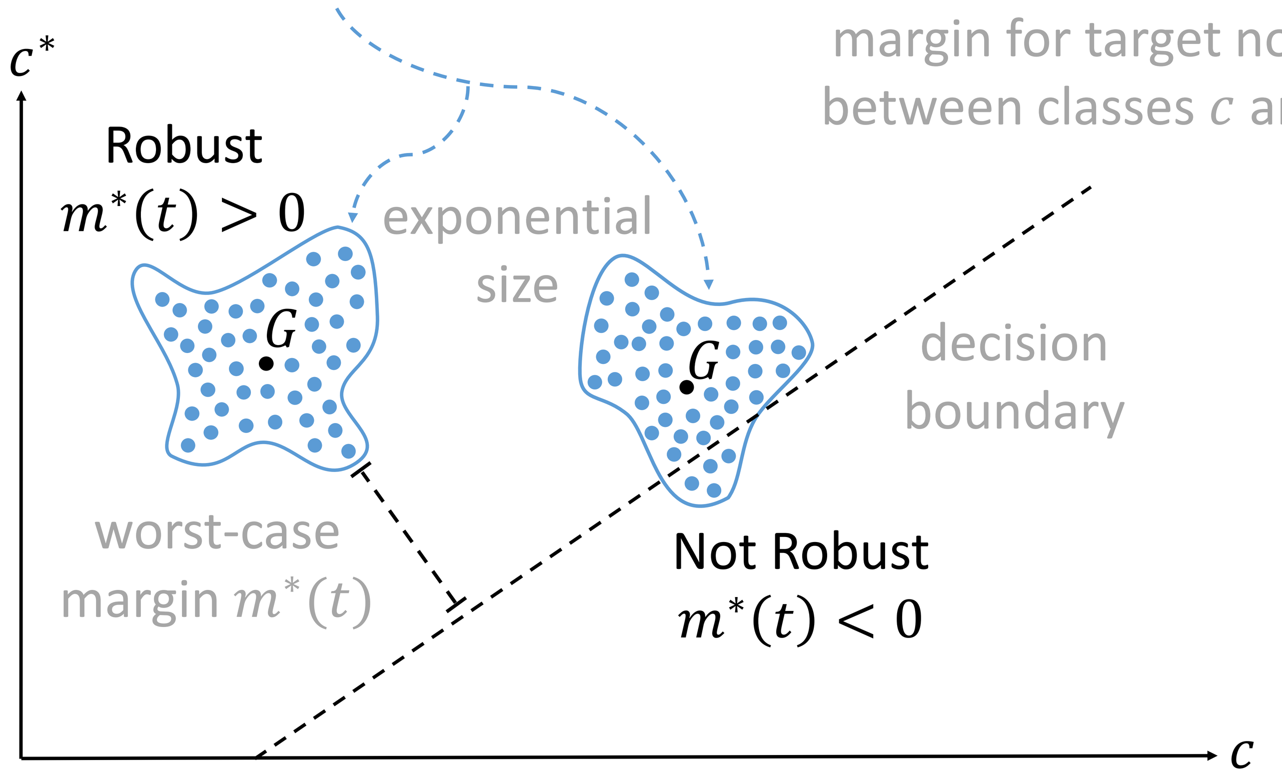
Global Budget: perturb at most B edges in total
Local Budget: at most b_v edges for each node v



Robustness Certificate

Guarantee that the prediction does not change under any admissible perturbation of the input graph

$$m^*(t) = \min_{\substack{G\,\in\,\text{all admissible}\\ \text{perturbed graphs}}} \; \min_{\text{class } c \neq c^*} \; \underbrace{\log p_G(t,c^*) - \log p_G(t,c)}_{\substack{\text{margin for target node } t\\ \text{between classes } c \text{ and } c^*}}$$



Family of Models based on PageRank

Predictions are a linear function of (Personalized) PageRank

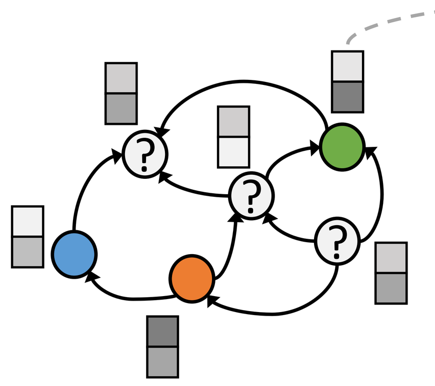
$$\log p_G(t,c) = \pi_G(t)^T h(c)$$

Personalized PageRank $\pi(t)$: Stationary distribution of a random walker teleporting back to node t with probability α

EX 1 - Label Propagation: repeatedly diffuse initial beliefs $H^{(0)}$

$$H^{(0)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ & \vdots & \end{bmatrix} \qquad \begin{aligned} H^{(t+1)} &= (1-\alpha)D^{-1}AH^{(t)} + \alpha\, H^{(0)} \\ \log p_G(t,c) &= H^{(\infty)}_{t,c} = \pi_G(t)^T H^{(0)}_{:,c} \end{aligned}$$

EX 2 - Graph Neural Network (PPNP): first map node features to initial beliefs with a NN f_θ then diffuse with $\pi_G(t)$

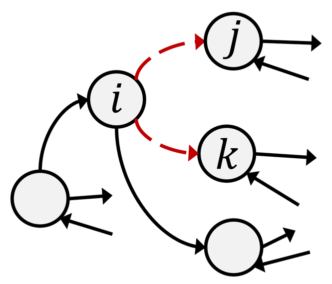


$$H^{(0)}_{t,:} = f_\theta(\cdot) \qquad [\,\cdots\,] = H^{(0)}$$

$$\log p_G(t,c) = \pi_G(t)^T H^{(0)}_{:,c}$$

Certificate ⇔ PageRank Optimization ⇔ MDP

Computing certificates amounts to finding optimal PageRank which can be done efficiently via a Markov Decision Process



node = state　　transition law

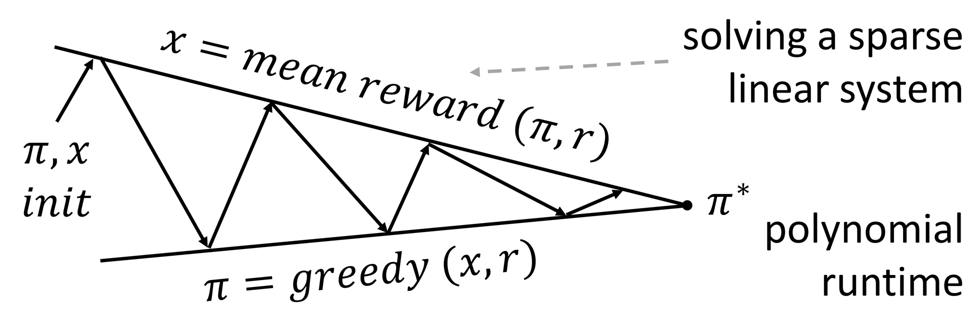
$(\mathcal{S}, (\mathcal{A}_i)_{i\in\mathcal{S}}, p, r)$　　reward

$\mathcal{A}_i = \{\emptyset, \{j\}, \{k\}, \{j,k\}\}$　　actions = subsets of fragile edges

Local budget: find optimal fragile edges with policy iteration

$$r = H^{(0)}_{c*} - H^{(0)}_{c}$$

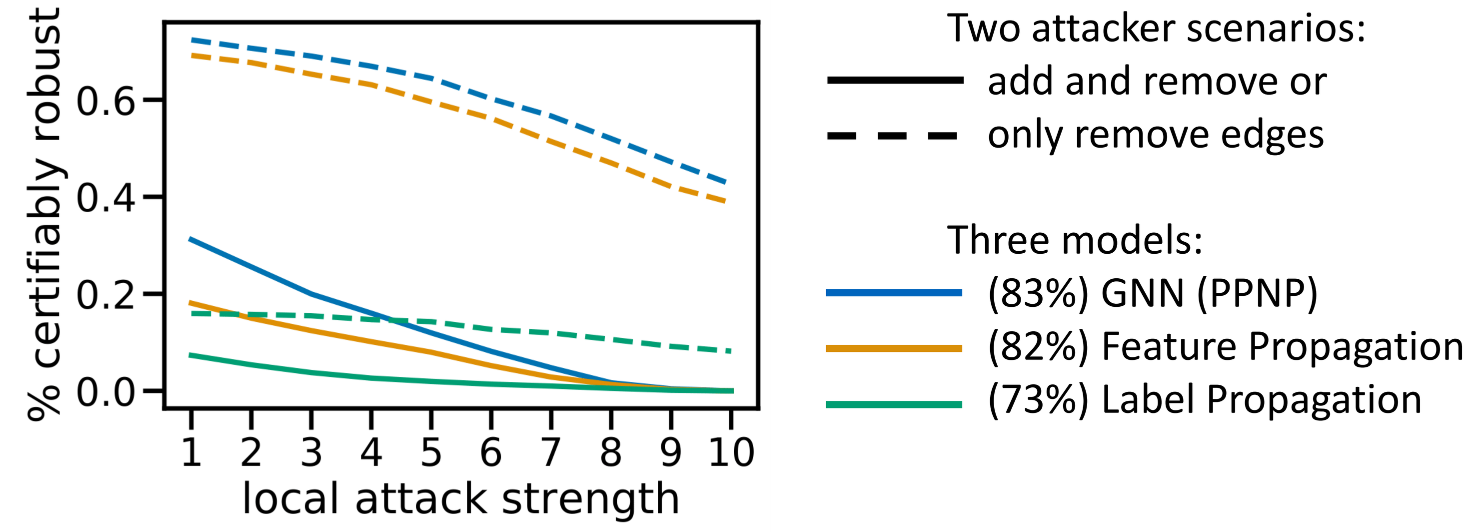
set reward to the logit difference



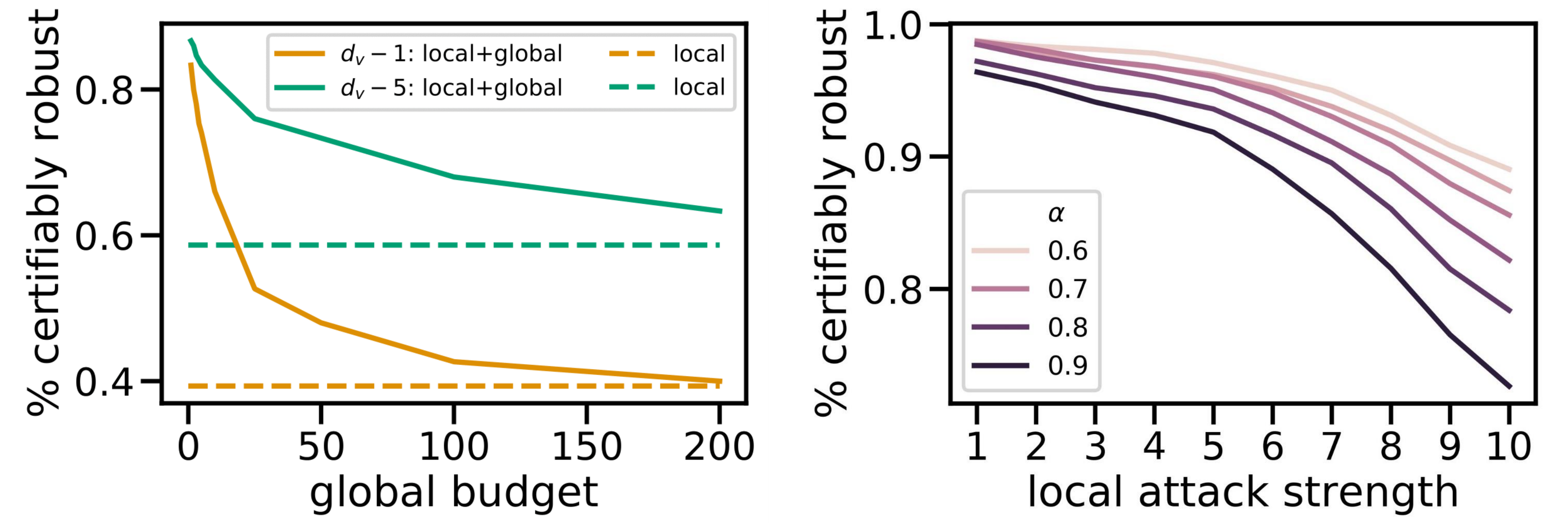
Local + Global budget: NP-Hard, augment graph & solve a QP

Certification Results

GNNs are more robust than Label/Feature Propagation



To increase ratio of certified nodes: decrease budget / lower α



Robust Training

Use worst-case margin during training to learn robust weights

Hinge-loss penalty: maximize the worst-case margin
Robust cross-entropy: worst-case instead of standard logits

$$\mathcal{L}_{CEM} = \mathcal{L}_{CE} + \textstyle\sum_{c\neq c^*} \max(0,\ M - m^*) \qquad \mathcal{L}_{RCE} = \mathcal{L}_{CE}(-m^*)$$

Robust training increases ratio of certified nodes and accuracy