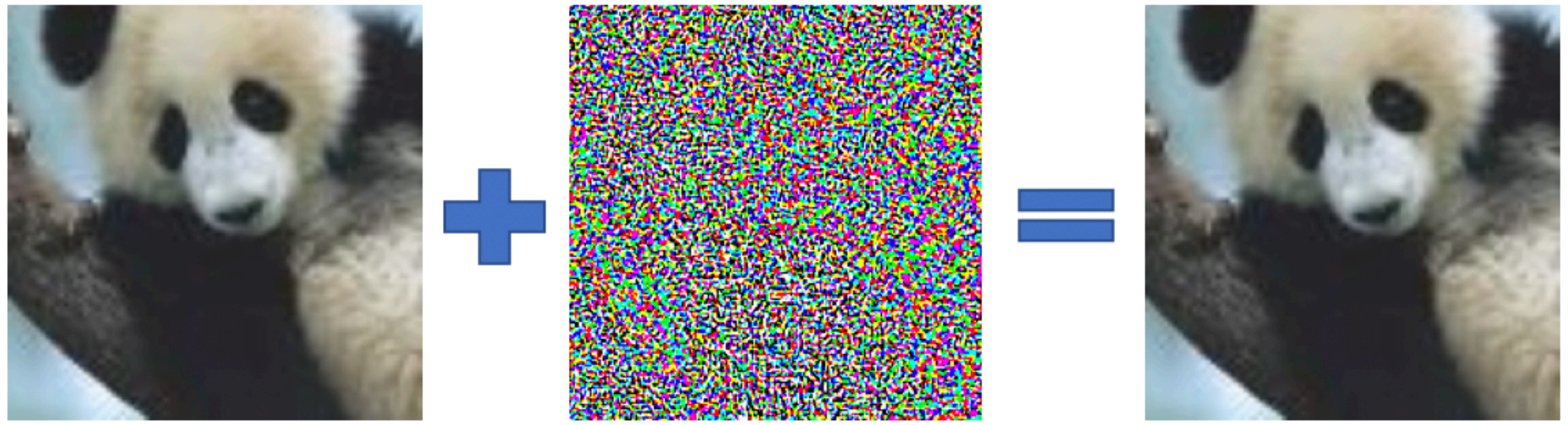
Robustness of GNN

刘闯
chuangliu@whu.edu.cn
2019.03.08

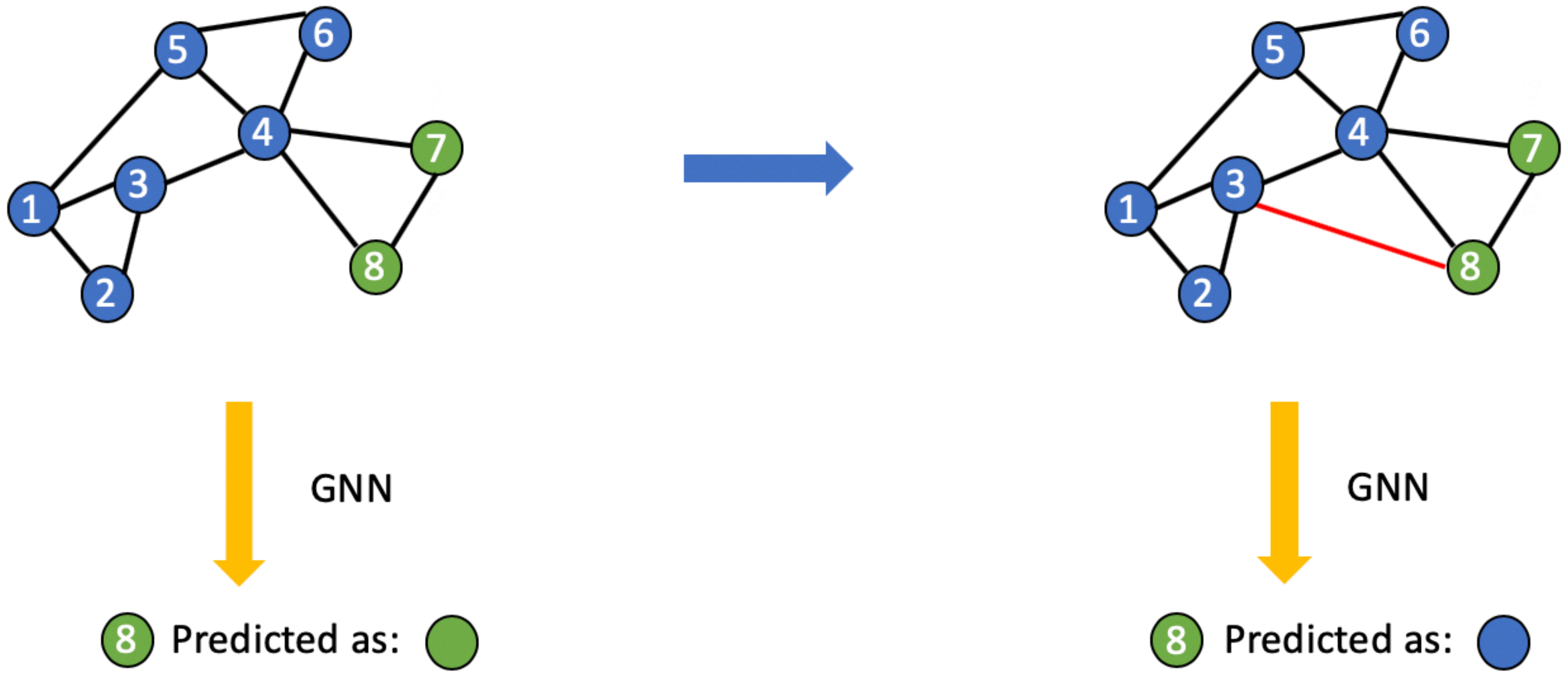
Adversarial Attacks



Classified as panda | Small adversarial noise | Classified as gibbon

x | ϵ | x'

Find x' satisfying $||x' - x|| \leq \Delta$
such that $C(x') \neq y$



Adversarial Attacks on DL

Adversarial Attacks on GNN

Adversarial Attacks on GNN

图的关联: 存在级联效应

方向：

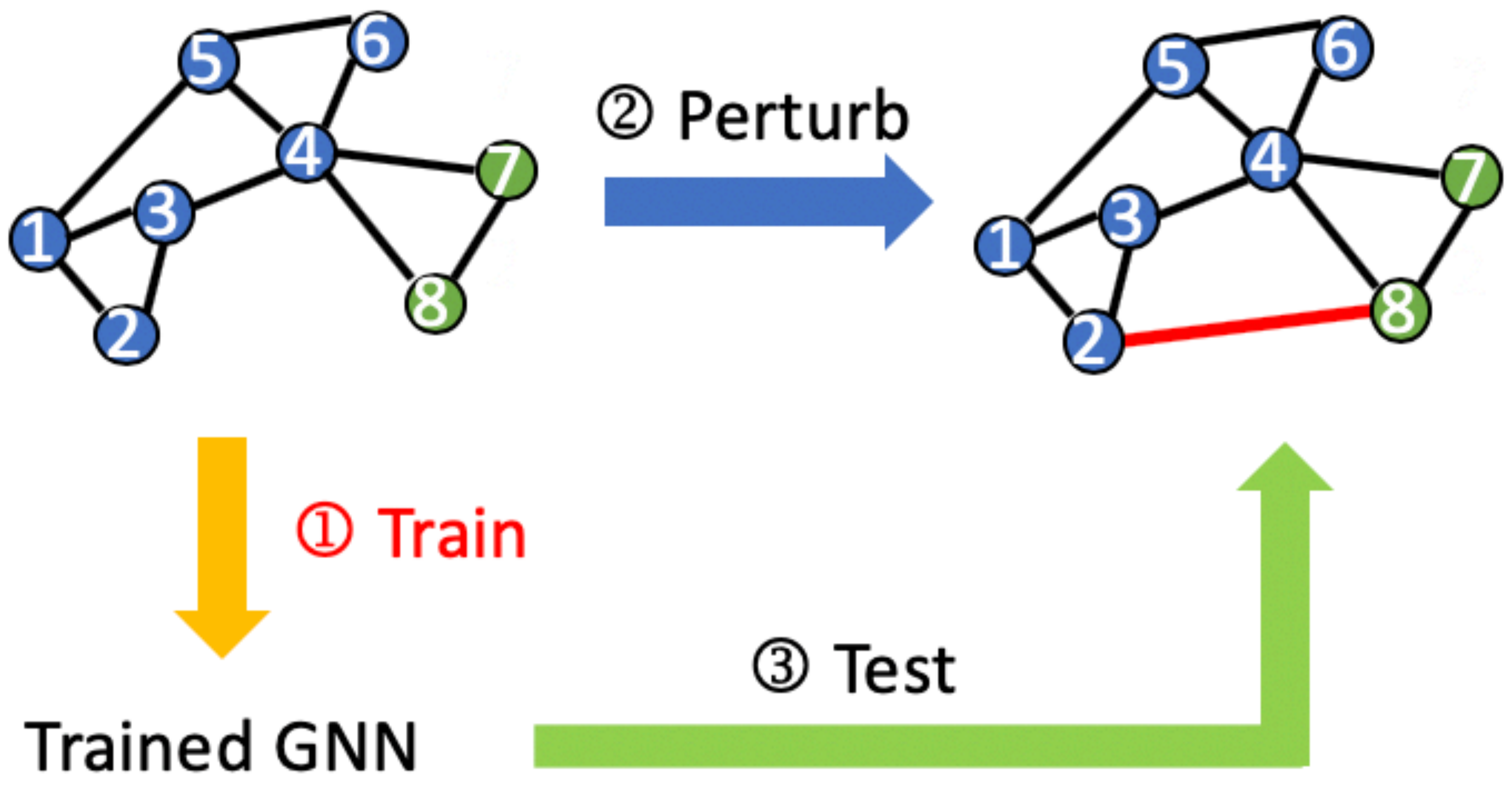
1. 模型是否鲁棒
2. 防范网络攻击的策略

应用：

1. 更改网站排名
2. 更改信用评级
3. 舆论控制

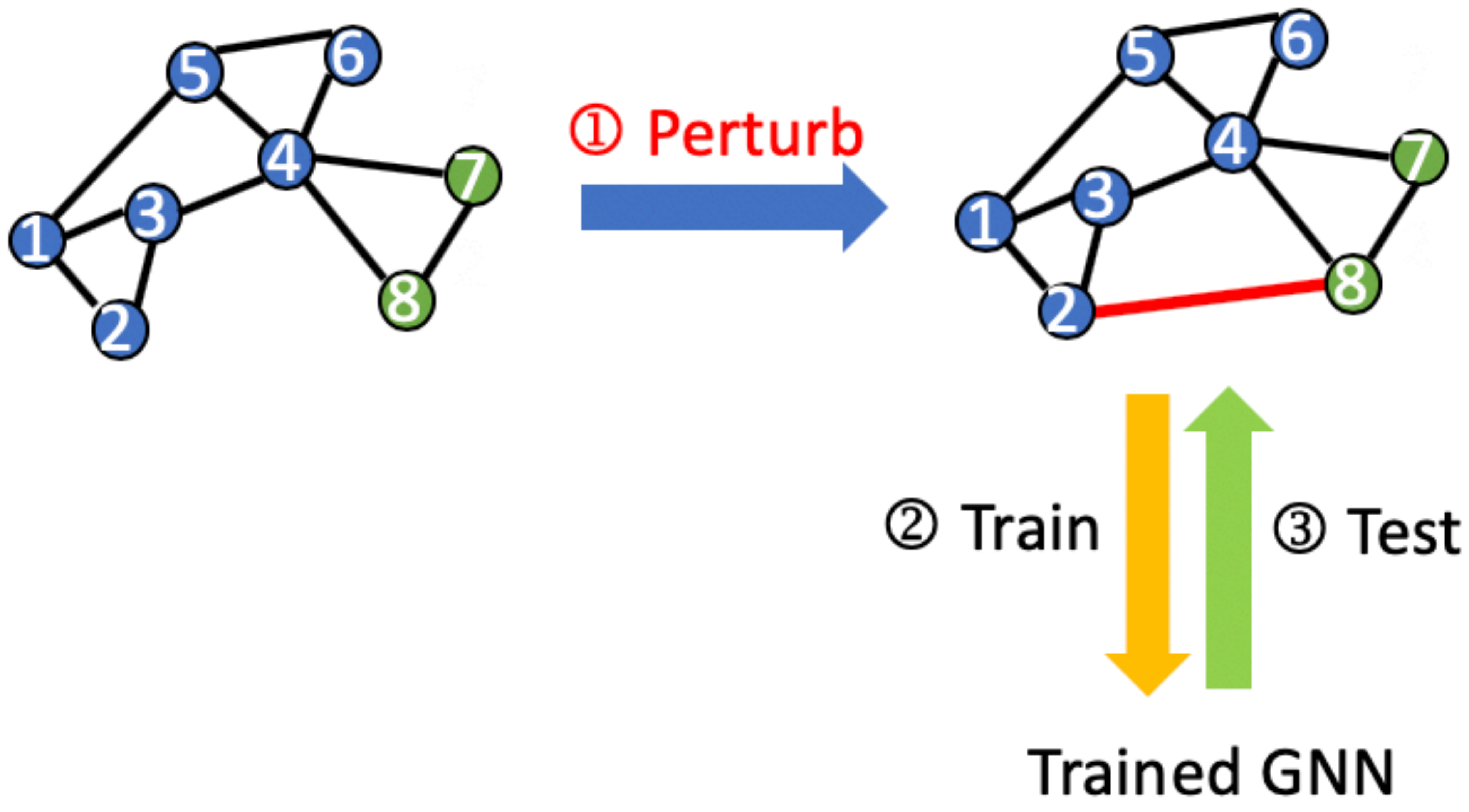
Adversarial Attacks Type

Evasion Attack



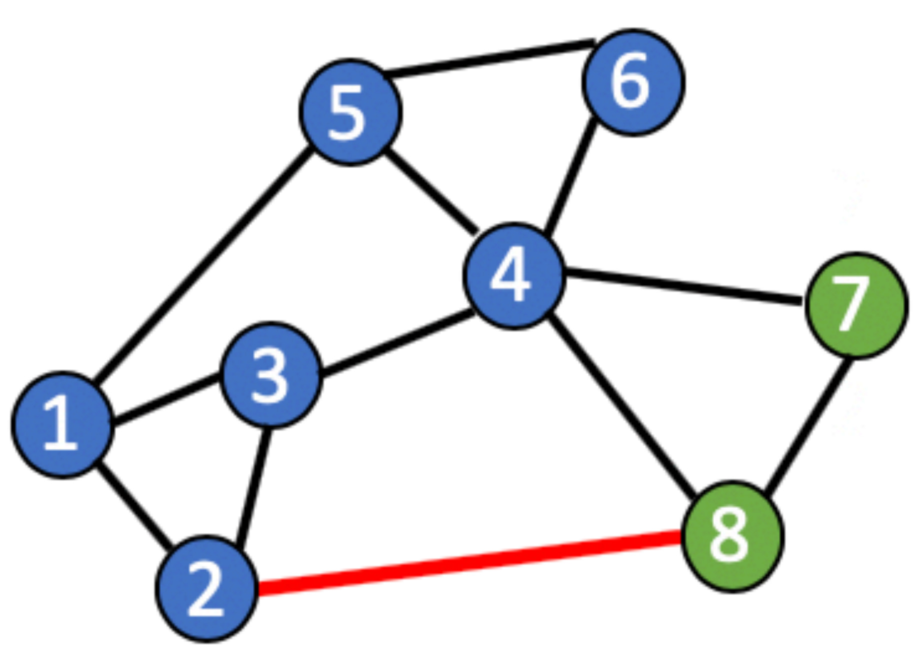
基于测试数据

Poisoning Attack

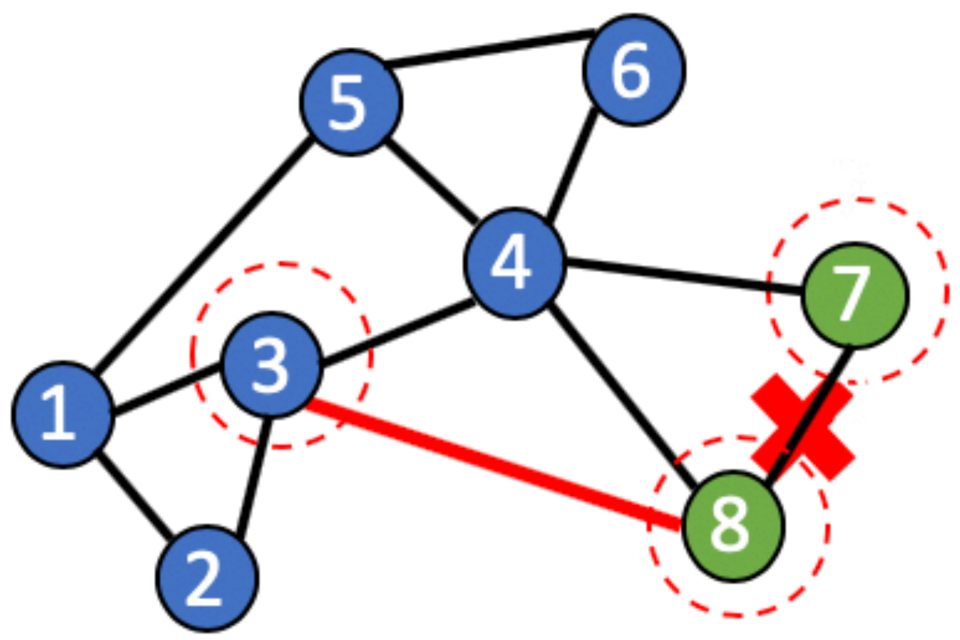


基于训练数据

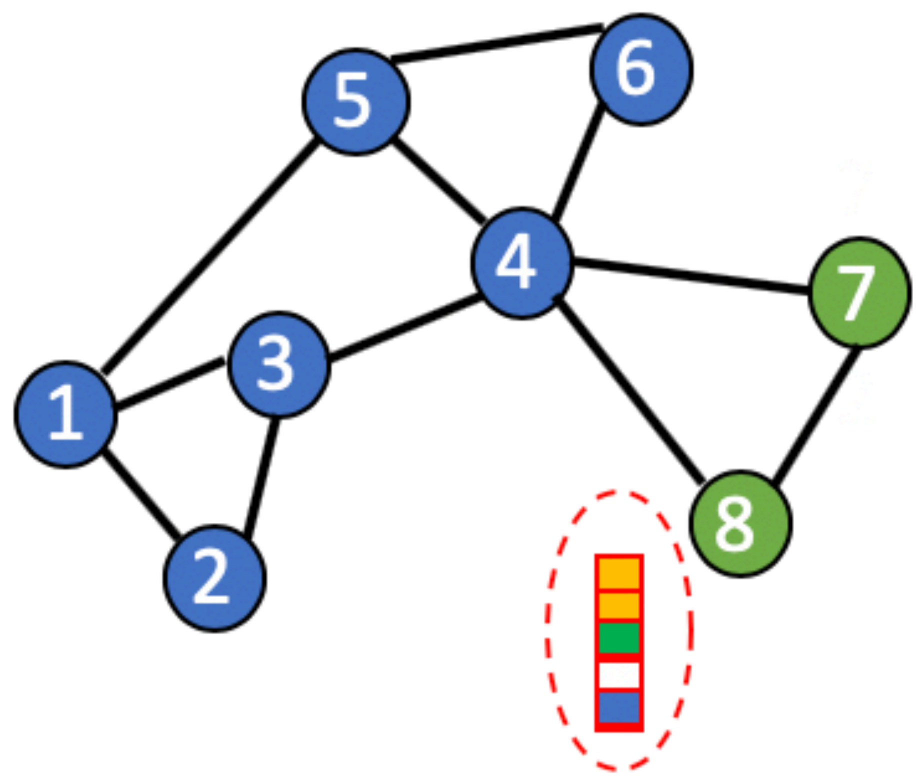
Perturbation Type



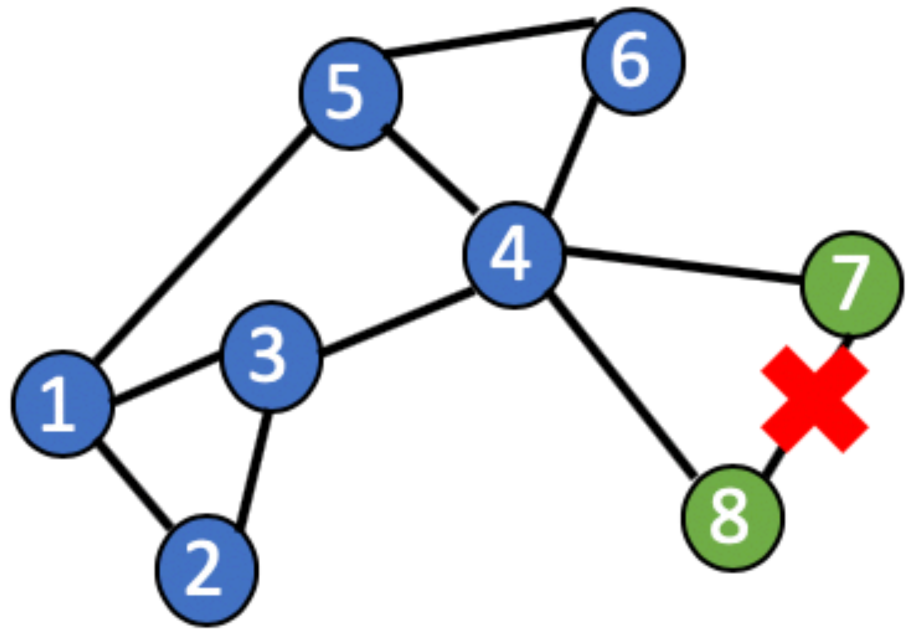
Adding an edge



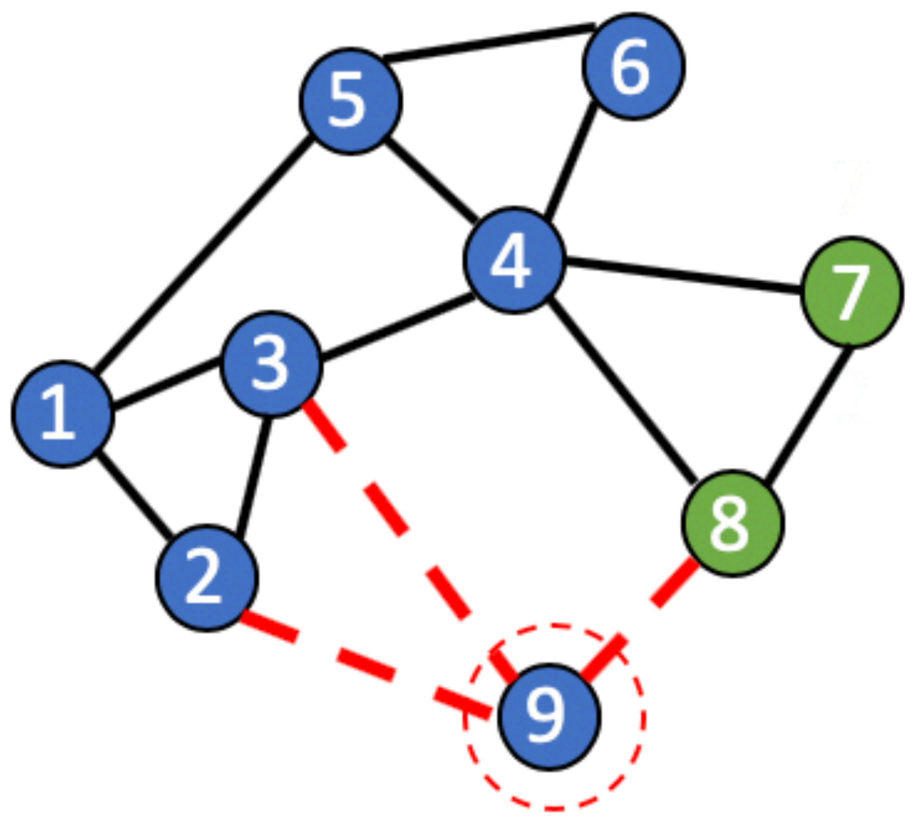
Rewiring



Modifying Features



Deleting an edge



Node Injection

Reference

1. Adversarial Attacks on Neural Networks for Graph Data. KDD 2018 (P7 - P16)
2. Adversarial Attack on Graph Structured Data. ICML 2018.
3. Adversarial Examples on Graph Data: Deep Insights into Attack and Defense. IJCAI 2019
4. Adversarial Attacks on Graph Neural Networks via Meta Learning. ICLR 2019
5. Robust Graph Convolutional Networks Against Adversarial Attacks. KDD 2019.
6. Certifiable Robustness and Robust Training for Graph Convolutional Networks. KDD 2019

Adversarial Attacks

Adversarial Attacks on Neural Networks for Graph Data. KDD 2018

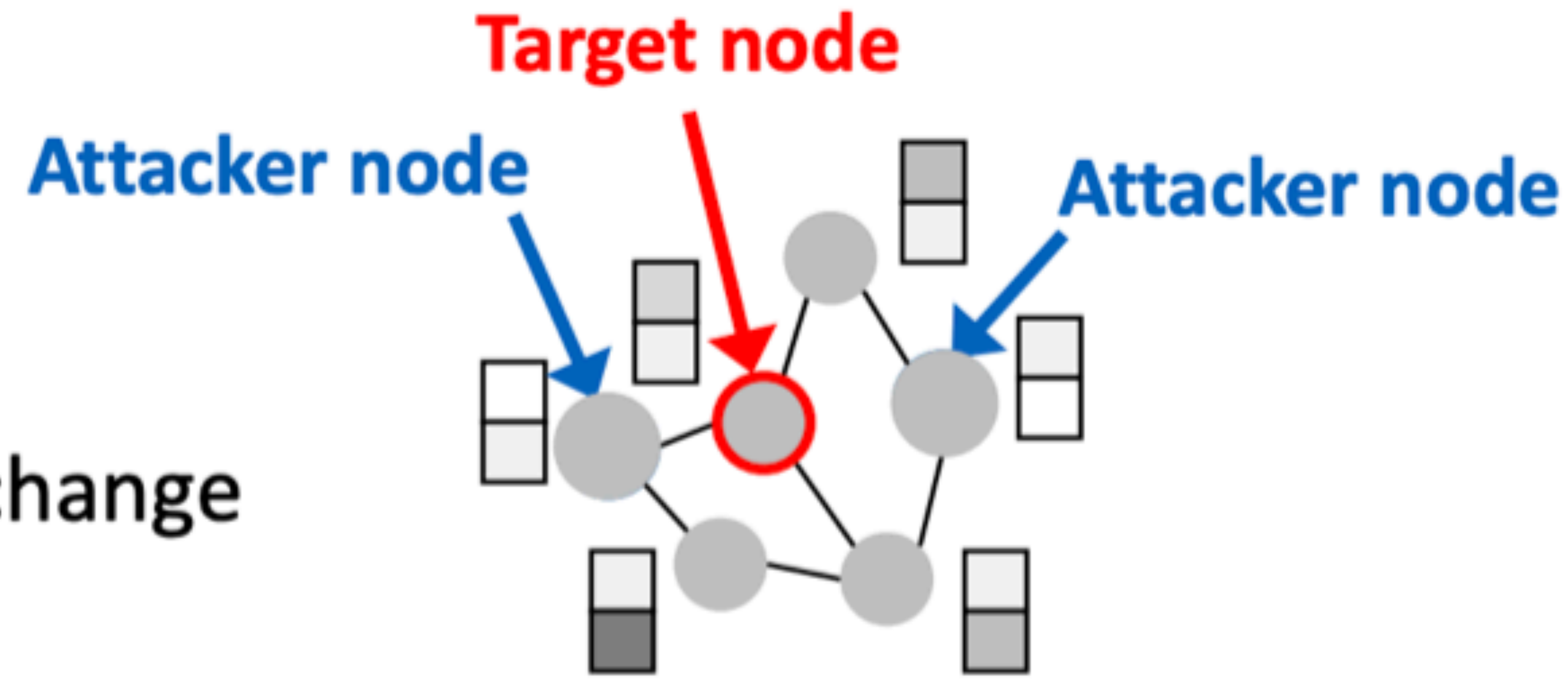
图网络对抗攻击开山文章 , Best paper

Challenge

1. Graph 不像图像这种由连续特征组成的数据，图的结构和以及大部分情况下的节点特征都是离散的，所以基于梯度构造干扰的方法不适用，设计有效的算法来在离散空间找对抗样本。
2. 对抗样本一个要求是对于人类的不可分辨性，例如图像，我们可以通过限制每个像素变化很小的值使得人类无法分辨图像的变化。对于大规模的 Graph 来说，可视化适不适合人肉观察的，如何定义“不可分辨性”。
3. 对于图节点任务，一般是 Transductive，意味着常用的 Evasion 攻击是不符合实际的，需要考虑 Poisoning 攻击。

Attacks possibilities

Direct VS Indirect



Target node $t \in V$: node whose classification label we want to change

Attacker nodes $S \subset V$: nodes the attacker can modify

Direct attack ($S = \{t\}$)

- Modify the **target**'s features

Example

Change website content

- Add connections to the **target**

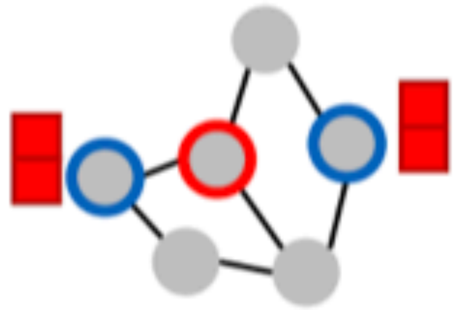
Buy likes/ followers

- Remove connections from the **target**

Unfollow untrusted users

Indirect attack ($t \notin S$)

- Modify the **attackers**' features

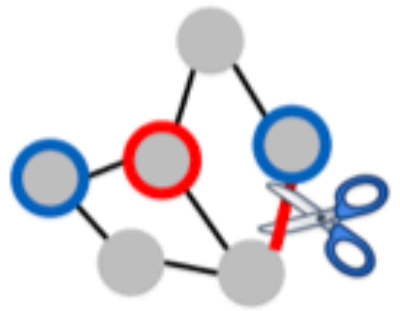


Example

Hijack friends of target

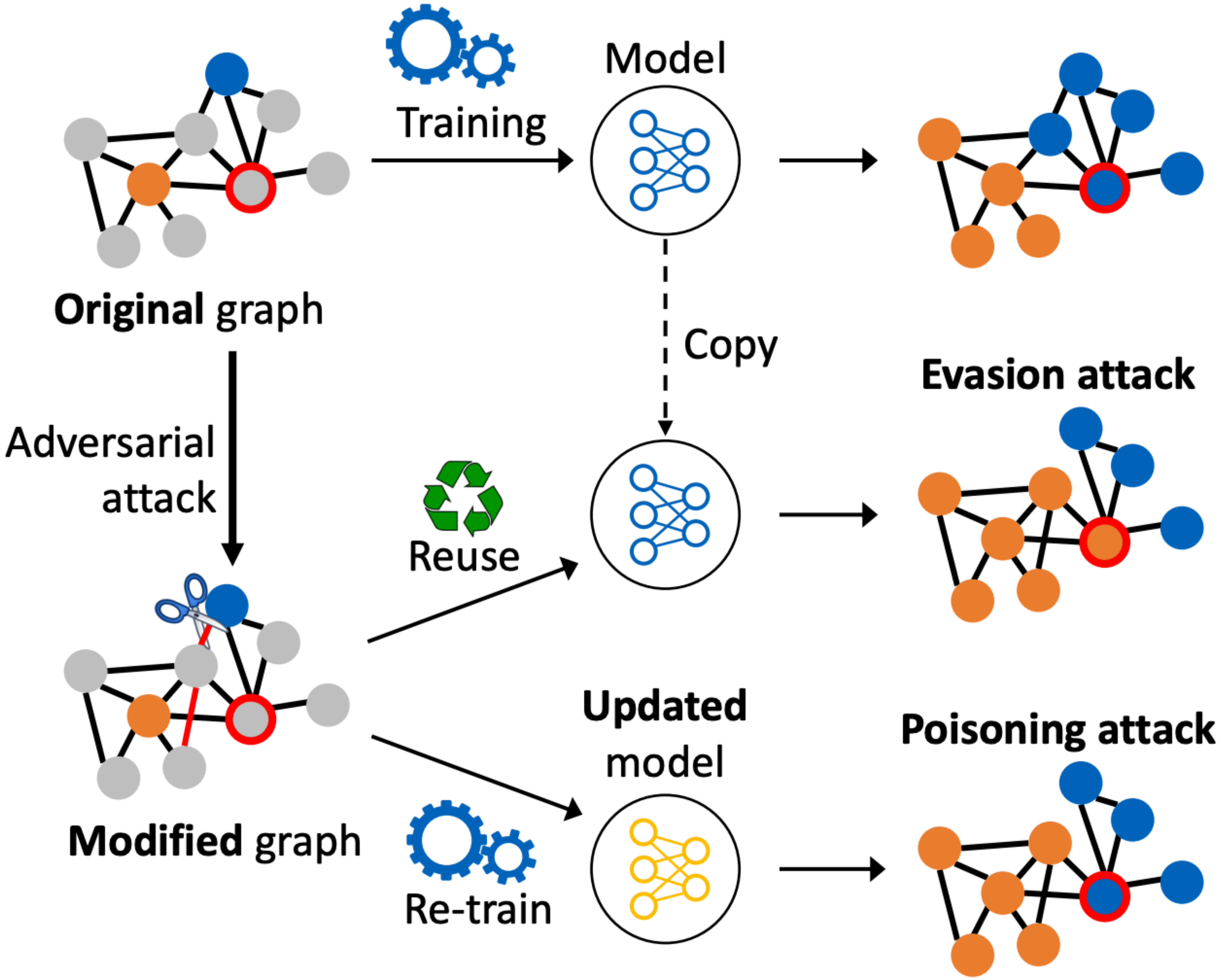
- Add connections to the **attackers**

- Remove connections from the **attackers**



Create a link/ spam farm

Adversarial Attacks on Graph

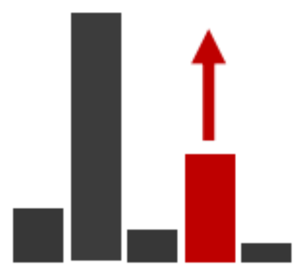
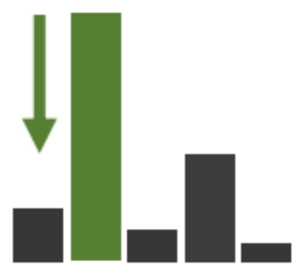


Transductive learning: data consists of labeled and unlabeled samples; all data used for training.

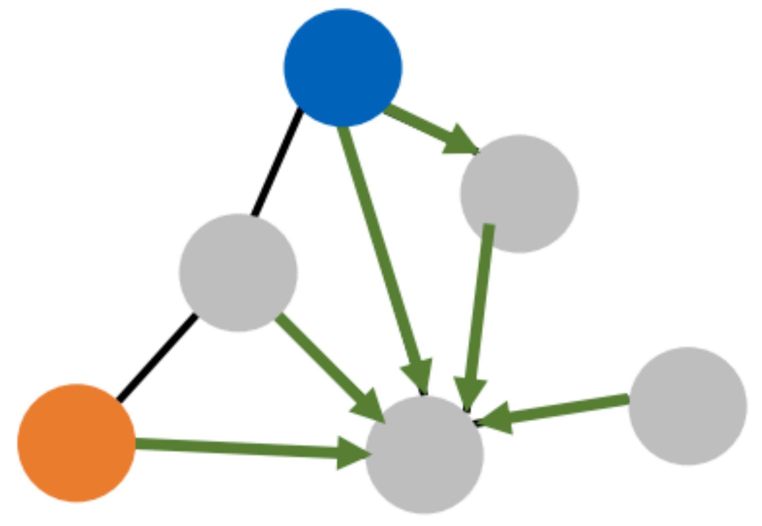
Evasion attack: Modify data to fool a static classifier.

But: modifications are on the **training data** (transductive setting).
Re-training can restore the predictions

Poisoning Attack on Node Classification

$$\arg\max_{A',X'} \max_{c \neq c_{old}} \log Z^*_{v,c} - \log Z^*_{v,c_{old}}$$



$$where\ Z^* = f_{\theta^*}(A',X') = softmax\left(\hat{A}'\ ReLU\left(\hat{A}'X'W^{(1)}\right)W^{(2)}\right),$$

$$with\ \ \theta^* = \arg\min_{\theta} \mathcal{L}\,(\theta; A', X')\ \text{(after re-train)}$$

c.f. $\mathcal{L}(\theta; A, X)$: evasion

$$\boxed{s.t.\,(A',X') \approx (A,X)}$$

"Unnoticeability"
constraint

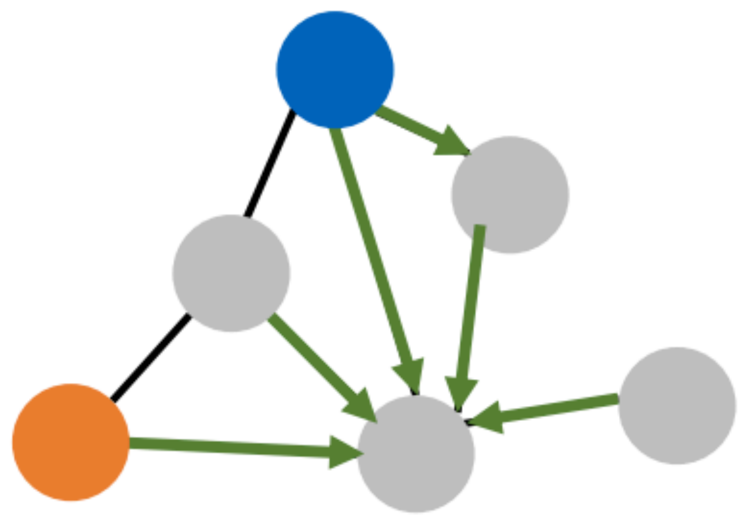
不可察觉的约束

$A \in \{0,1\}^{N\times N}$: original adjacency matrix
$X \in \{0,1\}^{N\times D}$: (binary) node attributes
A': modified structure
X': modified features
v : target node

Challenges

1. $\arg\max_{A',X'}$: optimization over **discrete variables** (gradient information less reliable)

2. Relational dependencies between the nodes: propagation effects



3. $(A', X') \approx (A, X)$:
   what is a sensible measure of ‘**closeness**’ for (attributed) graphs?

4. $\theta^* = \arg\min_{\theta} \mathcal{L}\,(\theta; A', X')$:
   minimize classification accuracy **after** (re-)**training** on the modified data (transductive setting)

Idea: Surrogate Model

Based on a two-layer Graph Convolutional Network (GCN):

$$Z = f_\theta(A, X) = \cancel{softmax}\left(\hat{A}\ \cancel{ReLU}\left(\hat{A}XW^{(1)}\right)W^{(2)}\right)$$ **Linearize** classifier

$$\log Z' \quad = \hat{A}^2 XW'$$

Structure perturbations: $\max_{\hat{A}} \mathcal{L}'\left(\log Z'_v\right)\ where\ \log Z'_v = [\hat{A}^2 C]_v$ Constants

Feature perturbations: $\max_{X} \mathcal{L}'\left(\log Z'_v\right)\ where\ \log Z'_v = [C_1 X C_2]_v$

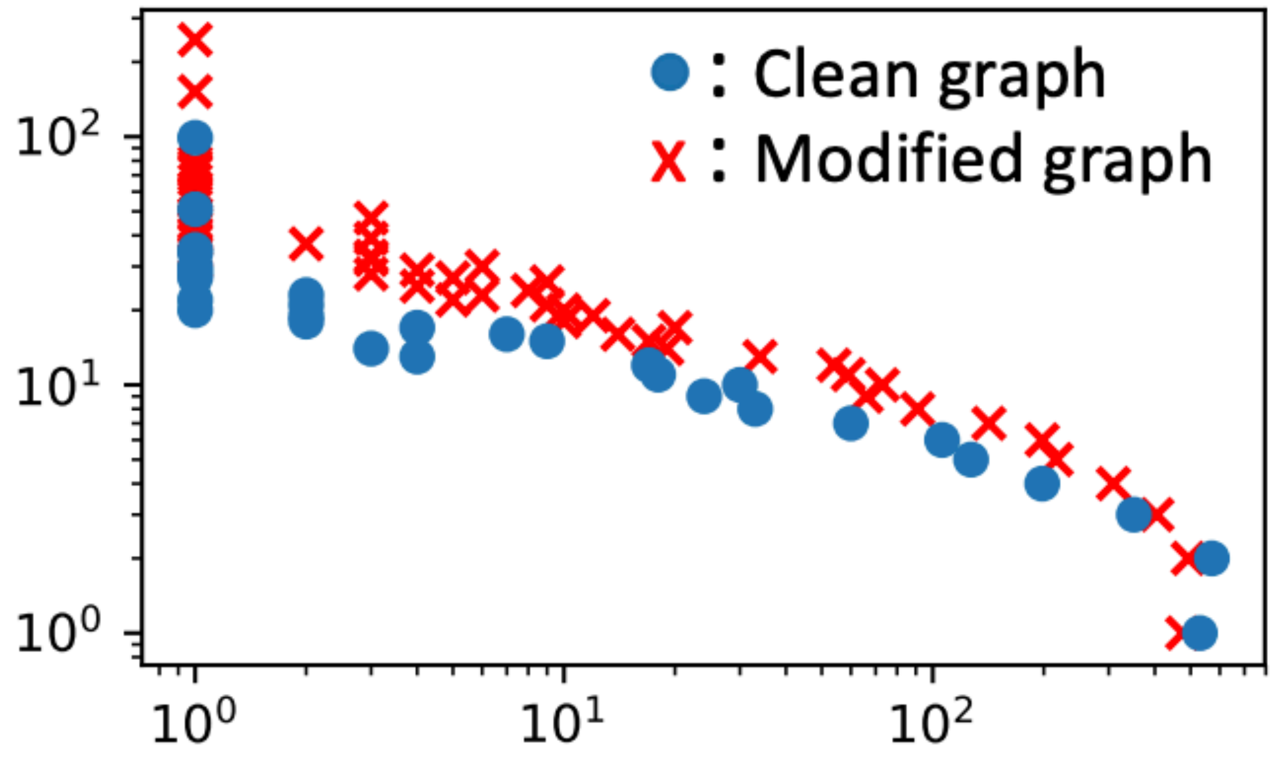
Unnoticeability Constraint

$(A', X') \approx (A, X)$: **Visual inspection** by a human **is not an option** for graphs.

What are sensible measures of '**closeness**' for graphs?

Structure perturbations: $A' \approx A$

Statistical test on the original and modified **degree distributions** to ensure structural similarity.



Degree distributions

幂律分布

Feature perturbations: $X' \approx X$

Co-occurrence constraint for features to prevent addition of unrealistic, easy to detect features

Adversarially inserted words
to ML paper abstracts:

with constraint	without constraint
probabilistic	efforts
bayesian	david
inference	family

经常共现

Adversarial Attacks : Algorithm

Algorithm 1: NETTACK: Adversarial attacks on graphs

Input: Graph $G^{(0)} \leftarrow (A^{(0)}, X^{(0)})$, target node v_0,
attacker nodes $\mathcal{A}$, modification budget Δ

Output: Modified Graph $G' = (A', X')$

Train surrogate model on $G^{(0)}$ to obtain W // Eq. (13);

$t \leftarrow 0$;

while $|A^{(t)} - A^{(0)}| + |X^{(t)} - X^{(0)}| < \Delta$ **do**

$C_{struct} \leftarrow \texttt{candidate\_edge\_perturbations}(A^{(t)}, \mathcal{A})$;

$e^* = (u^*, v^*) \leftarrow \underset{e \in C_{struct}}{\arg\max}\ s_{struct}\left(e; G^{(t)}, v_0\right)$;

$C_{feat} \leftarrow \texttt{candidate\_feature\_perturbations}(X^{(t)}, \mathcal{A})$;

$f^* = (u^*, i^*) \leftarrow \underset{f \in C_{feat}}{\arg\max}\ s_{feat}\left(f; G^{(t)}, v_0\right)$;

if $s_{struct}(e^*; G^{(t)}, v_0) > s_{feat}(f^*; G^{(t)}, v_0)$ **then**

$G^{(t+1)} \leftarrow G^{(t)} \pm e^*$;

else $G^{(t+1)} \leftarrow G^{(t)} \pm f^*$;

$t \leftarrow t + 1$;

return $:G^{(t)}$

// Train final graph model on the corrupted graph $G^{(t)}$;

修改一部分节点

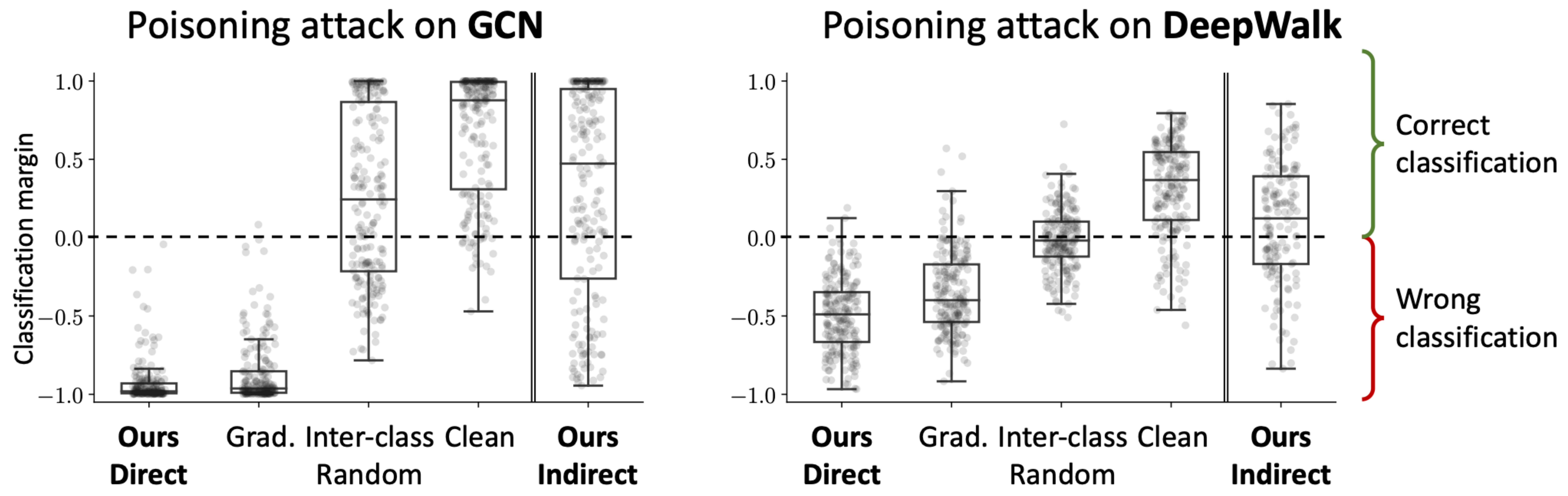
候选可改变的连边

哪个连边改变之后 分类结果变化最大

特征

连边和特征哪个影响更大

Results



	Ours Direct	Grad.	Inter-class Random	Clean	Ours Indirect
% Correct (GCN):	**1.0%**	2.7%	60.8%	90.3%	67.2%
% Correct (DeepWalk):	**2.1%**	9.8%	46.2%	83.8%	59.2%

Deep learning models for graphs are **not robust** to adversarial attacks.

Baselines:

- **Inter-class random** (direct; structure): insert edges randomly to nodes from different classes.
- **Gradient** (direct; structure): insert/remove edges based on the gradient.

Results: Limited knowledge

Setup: Only provide a **small part of the network** around the target node to the surrogate model to attack (evaluation with GCN on the complete graph).