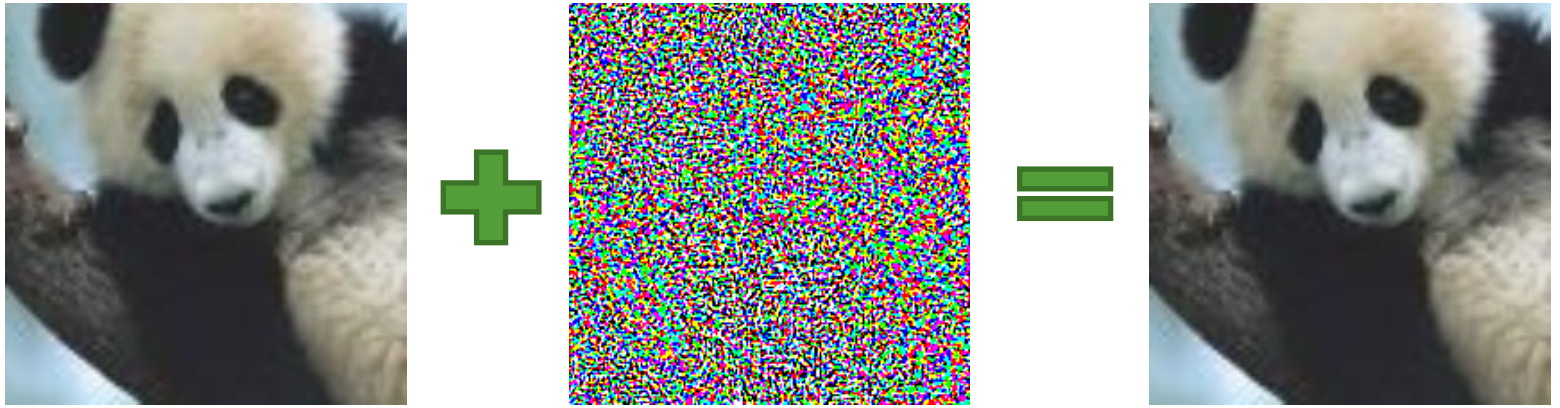
深蓝学院
shenlanxueyuan.com

图神经网络的鲁棒性

教材：图深度学习，电子工业出版社
https://baike.baidu.com/item/图深度学习

目录

CNN的鲁棒性



x ϵ x'

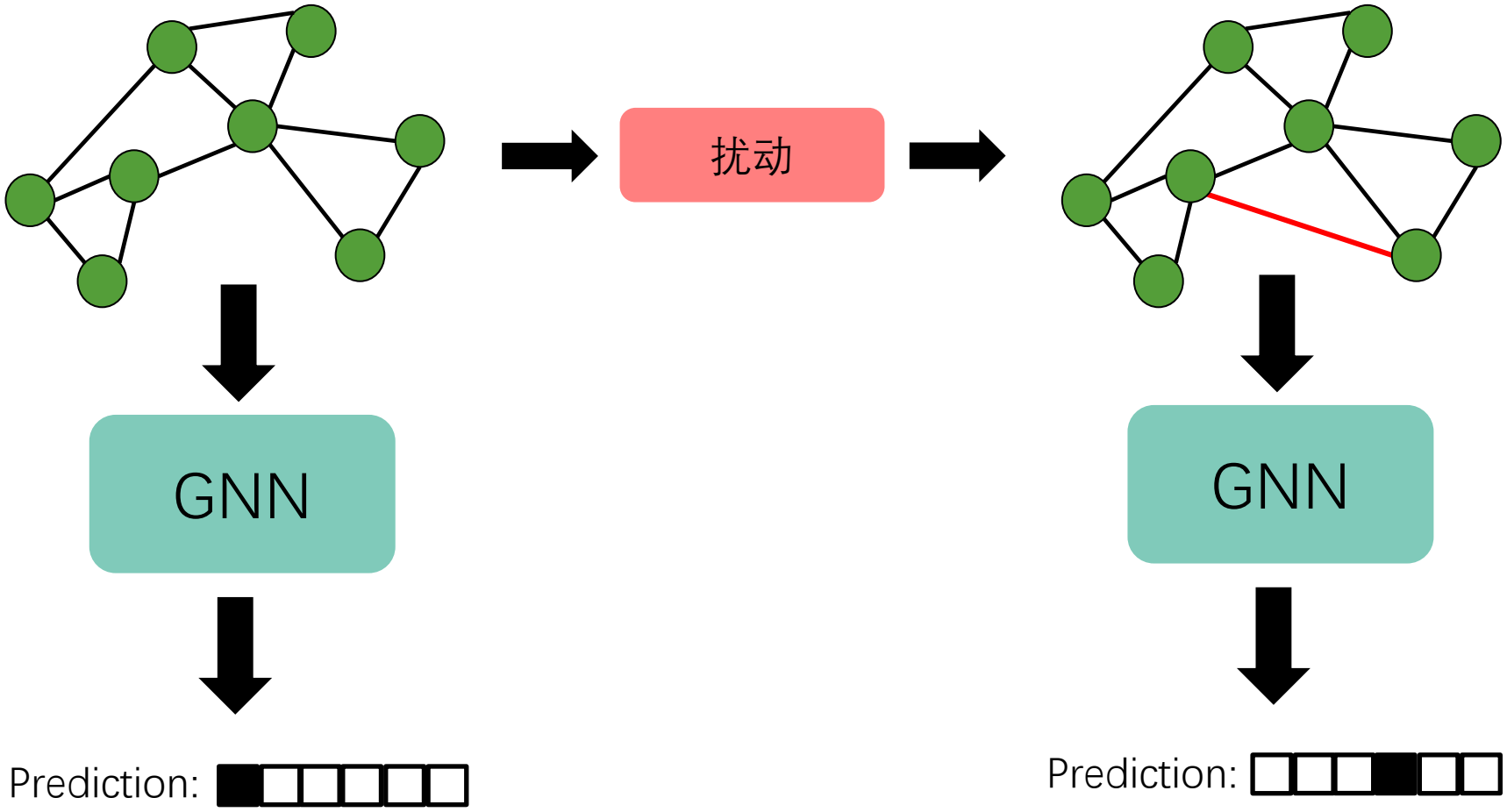
微小的扰动

CNN模型 CNN模型

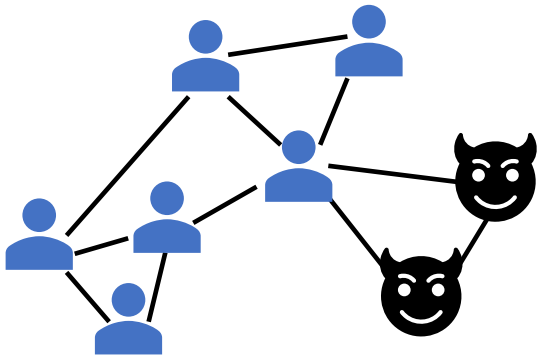
对抗攻击

“熊猫” “长臂猿”

CNN的鲁棒性



模型不鲁棒的后果

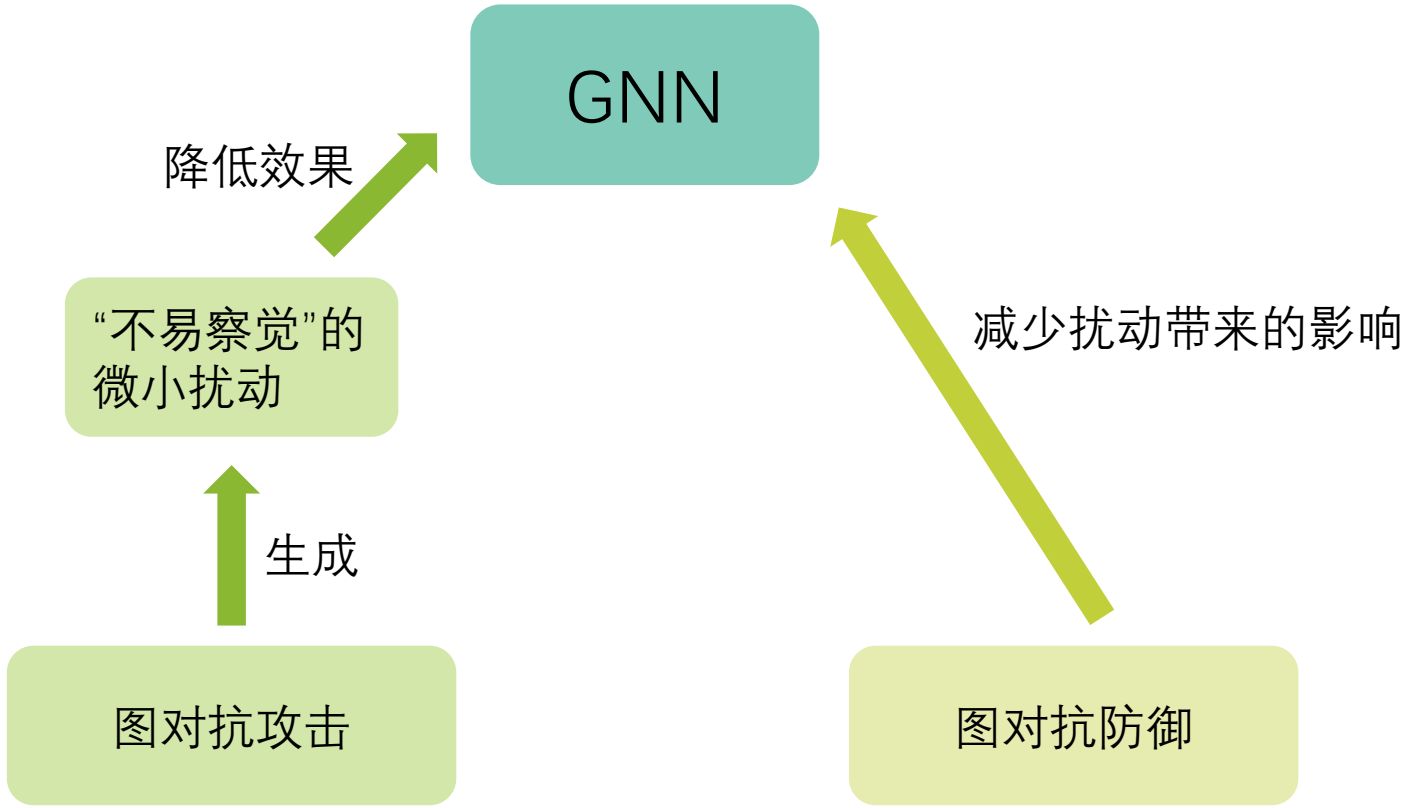


- 金融系统
  - 欺诈检测
  - 金融违约预测

...

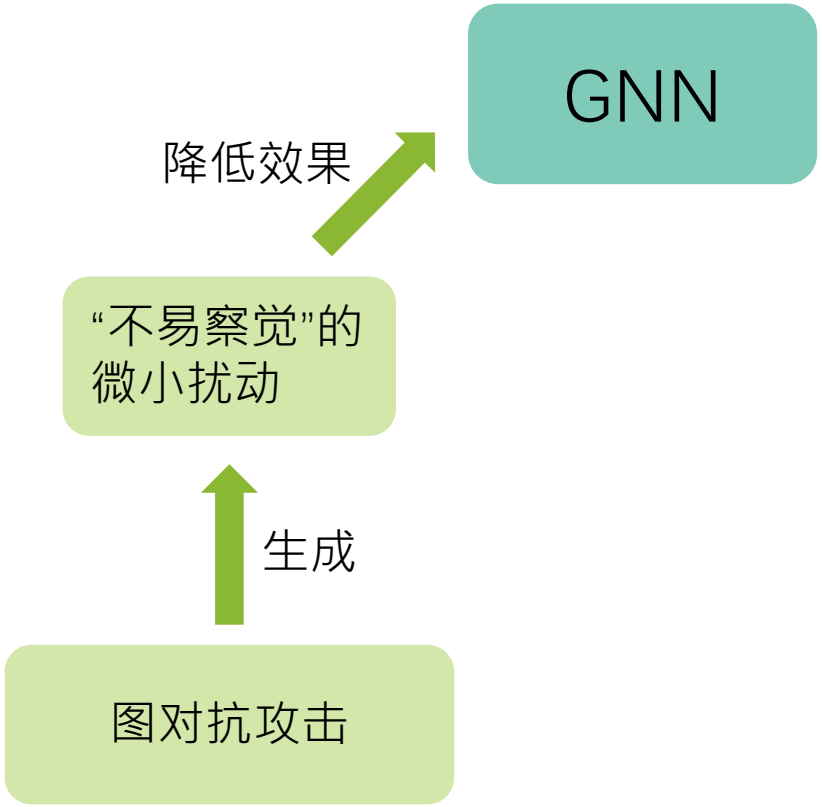
犯罪者能以极小的成本隐藏自己

图对抗攻击和防御

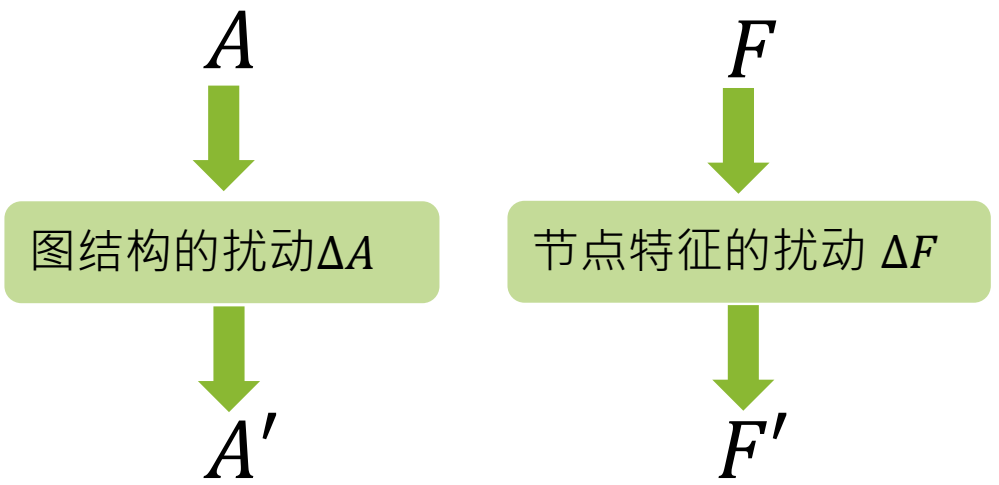


- 图神经网络鲁棒性简介
- 图对抗攻击
- 图对抗防御

图对抗攻击

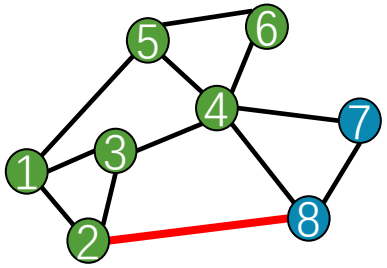


“不易察觉”的微小扰动

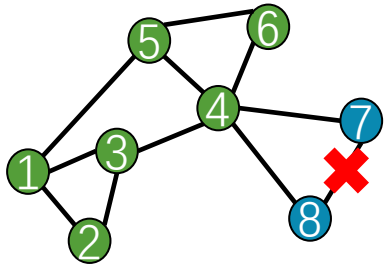


$$\|\boldsymbol{A}' - \boldsymbol{A}\|_0 + \|\boldsymbol{F}' - \boldsymbol{F}\|_0 \leqslant \Delta$$

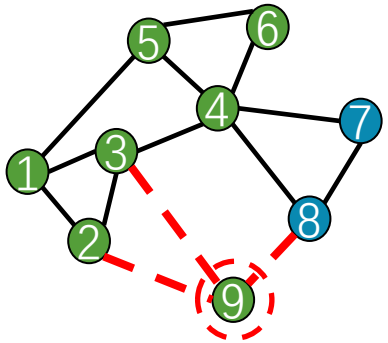
扰动的类型



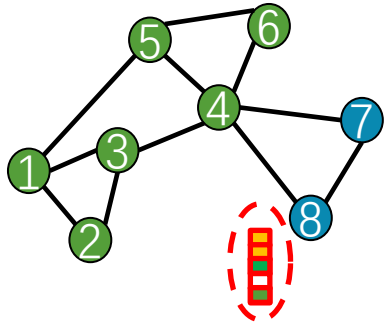
加边



减边



加节点

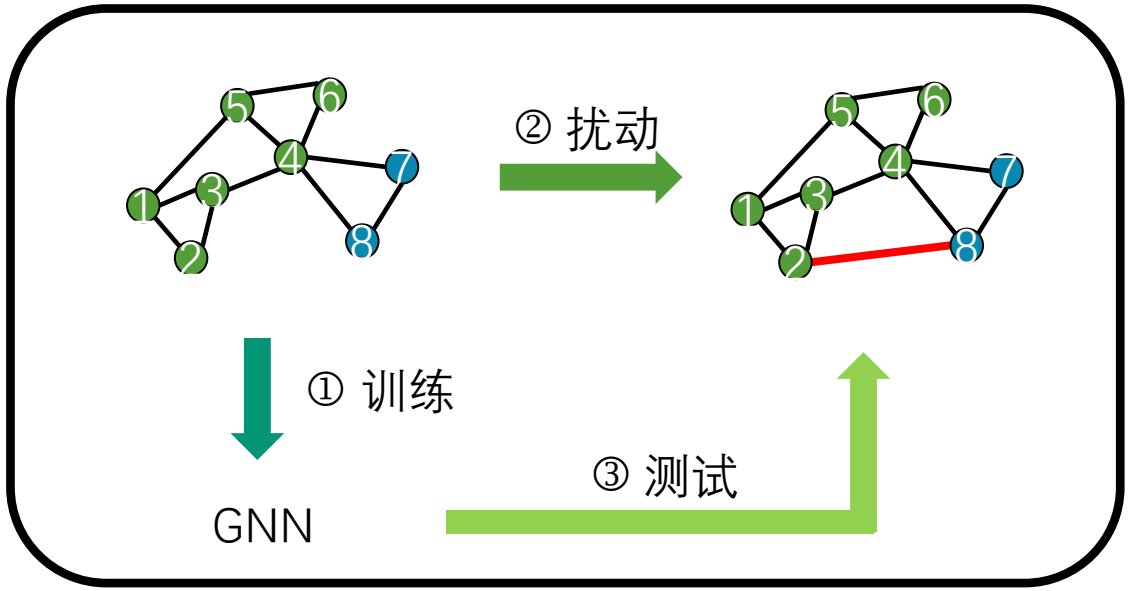


更改特征

攻击的类型

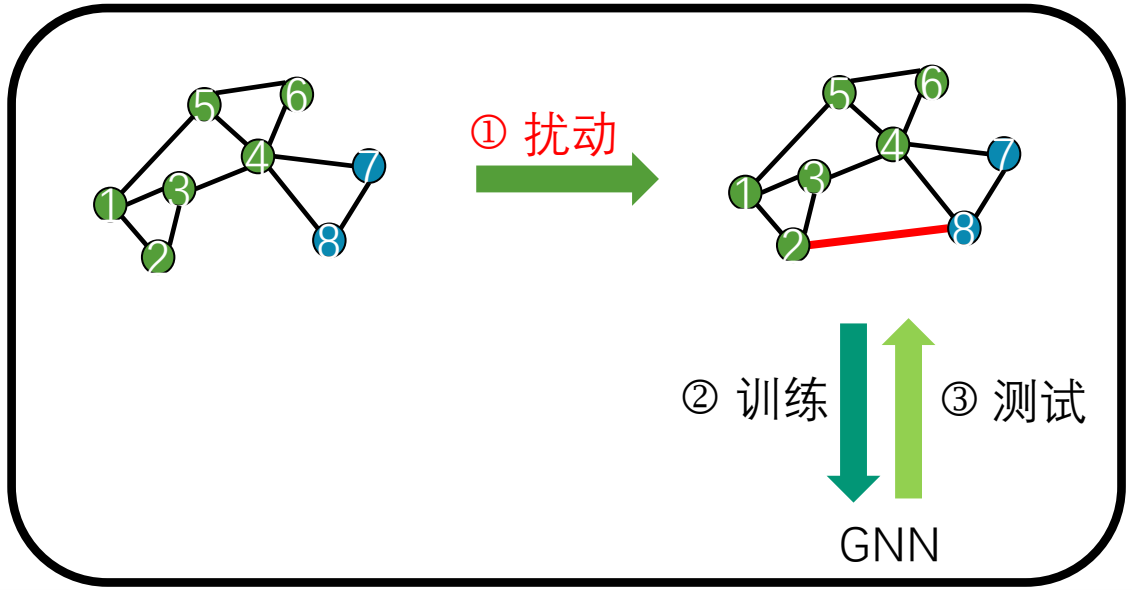
逃逸攻击

- GNN模型已经训练好且固定
- 进行图扰动
- 在被扰动的图上测试模型

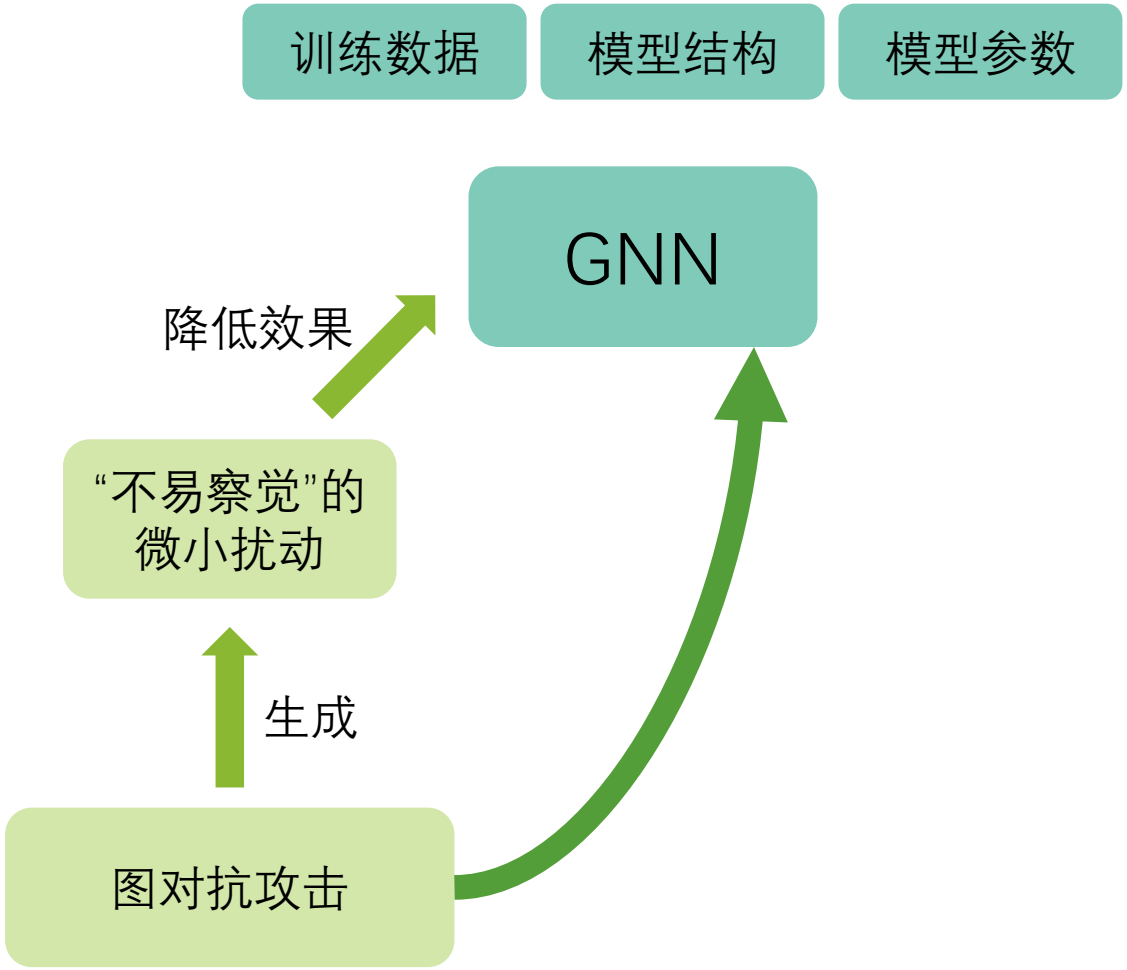


投毒攻击

- 进行图扰动
- 在被扰动的图上训练模型
- 在被扰动的图上测试模型



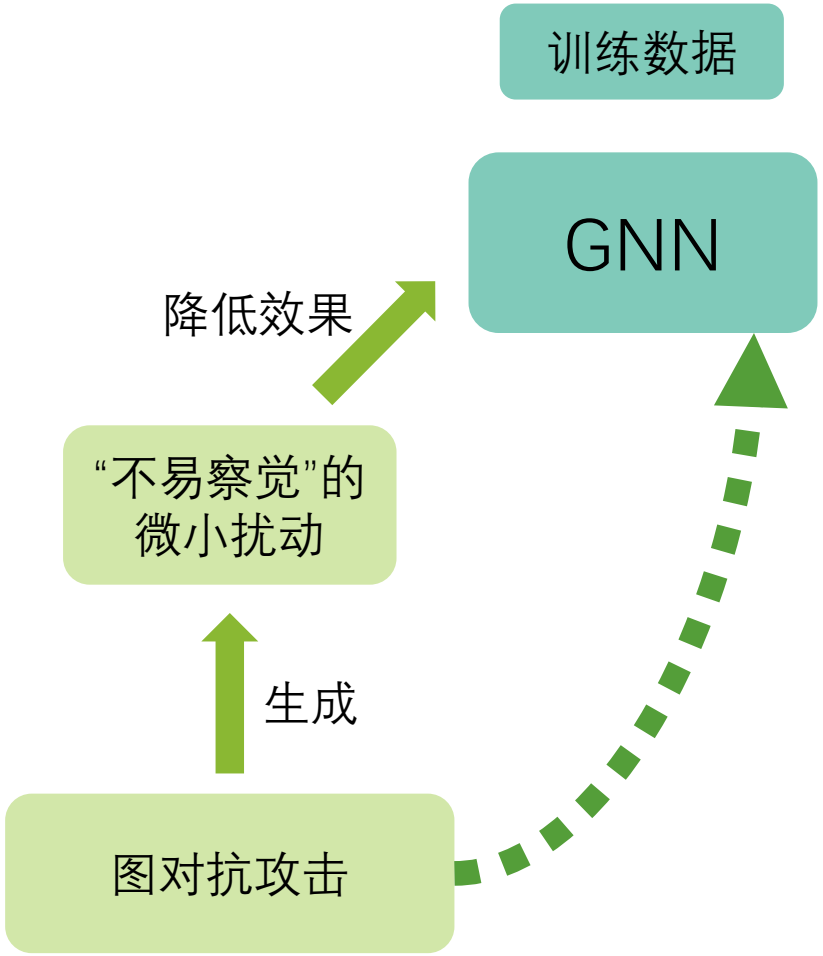
攻击者的关于GNN模型的知识



白盒攻击

攻击者可以获得被攻击的模型的完整信息

攻击者的关于GNN模型的知识



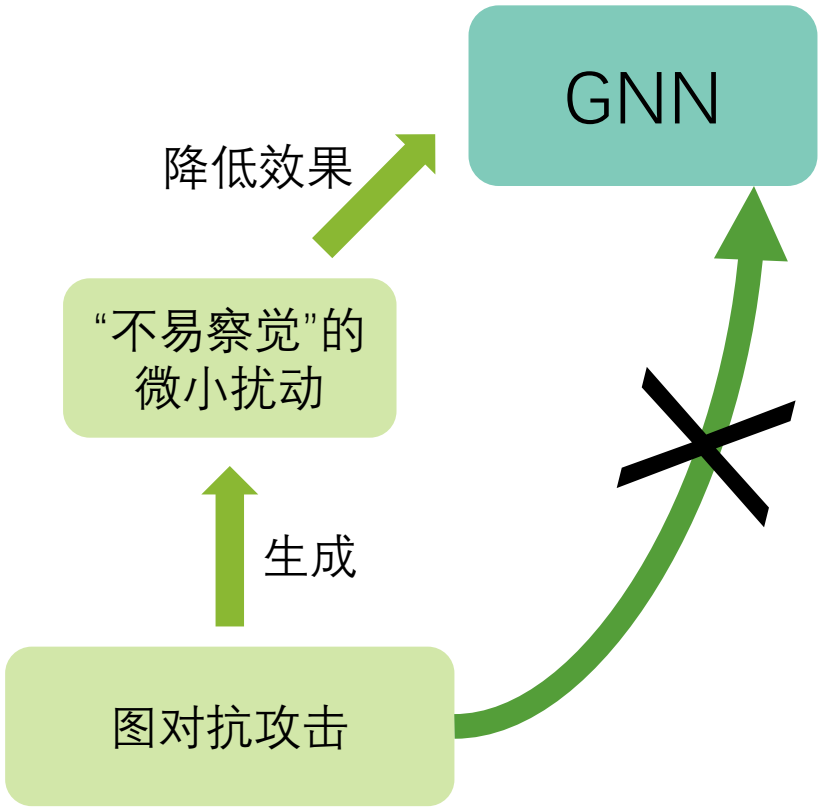
白盒攻击

攻击者可以获得被攻击的模型的完整信息

灰盒攻击

攻击者可以获得被攻击的模型的训练数据

攻击者的关于GNN模型的知识



白盒攻击

攻击者可以获得被攻击的模型的完整信息

灰盒攻击

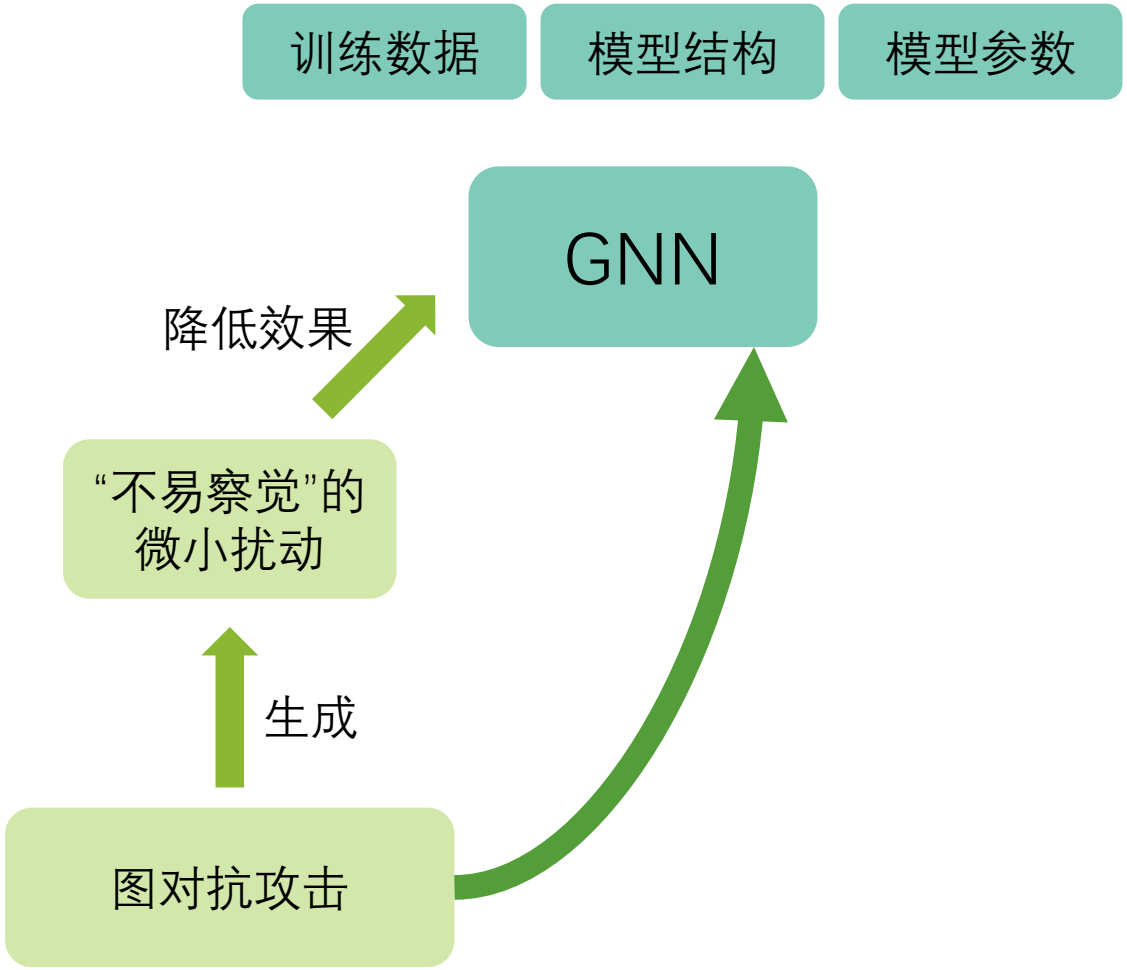
攻击者可以获得被攻击的模型的训练数据

黑盒攻击

攻击者不能可以获得被攻击的模型的任何信息

- 图对抗攻击
  - 白盒攻击
  - 灰盒攻击
  - 黑盒攻击

PGD拓扑攻击



白盒攻击

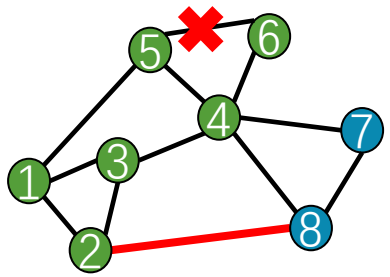
攻击者可以获得被攻击的模型的完整信息

攻击目标

- 攻击节点分类任务
- 对于某个标签为y_i的节点v_i，攻击者希望GNN能把它分类为其它标签

Topology Attack and Defense for Graph Neural Networks: An Optimization Perspective

PGD拓扑攻击



A

A'

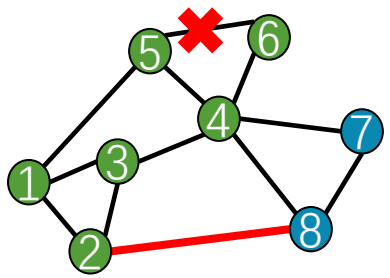
$\Delta A = (\bar{A} - A) \odot S$

如何改动
- 加边
- 减边

$S \in \{0,1\}^{N \times N}$
- 决定改动哪些“边”
- 攻击者需要学习

A的补
- 当$A_{ij} = 1$时，$\bar{A}_{ij} = 0$
- 当$A_{ij} = 0$时，$\bar{A}_{ij} = 1$

PGD拓扑攻击



A

$\Delta A = (\bar{A} - A) \odot S$

$S \in \{0,1\}^{N \times N}$
- 决定改动哪条边
- 攻击者需要学习

A'

GNN

参数固定

预测的节点v_i分类概率

采样

可以同时攻击多个节点

$$L(V_{at}) = \sum_{v \in V_{at}} l(v)$$

$z[y_i]$:分类为y_i的概率

攻击目标

使得这个概率不是最大

损失函数

$$l(v_i) = z[y_i] - \max_{c \neq y_i} z[c]$$

改进的损失函数

$$l(v_i) = \max\{z[y_i] - \max_{c \neq y_i} z[c], -\kappa\}$$

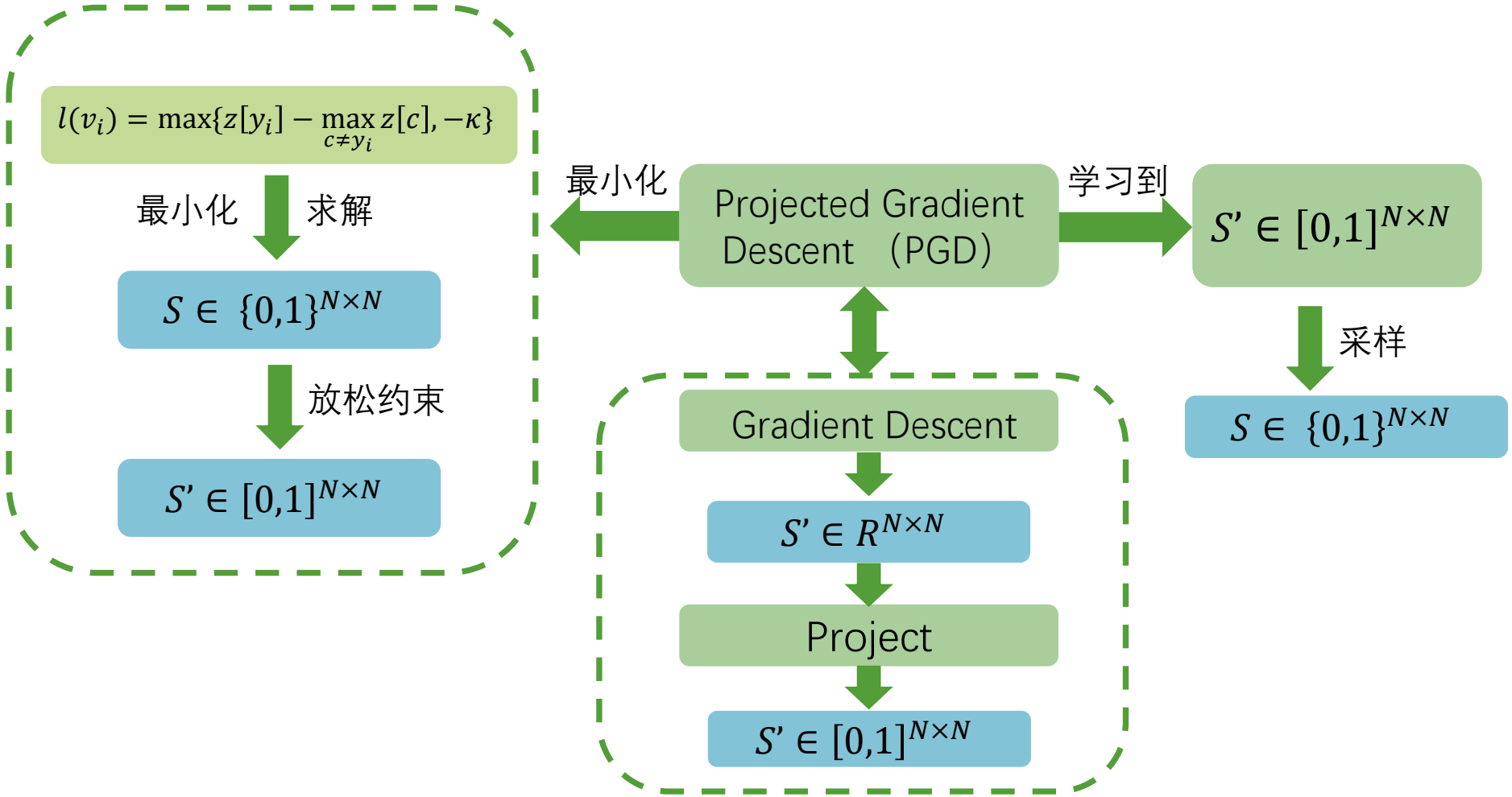
最小化

Projected Gradient Descent （PGD）

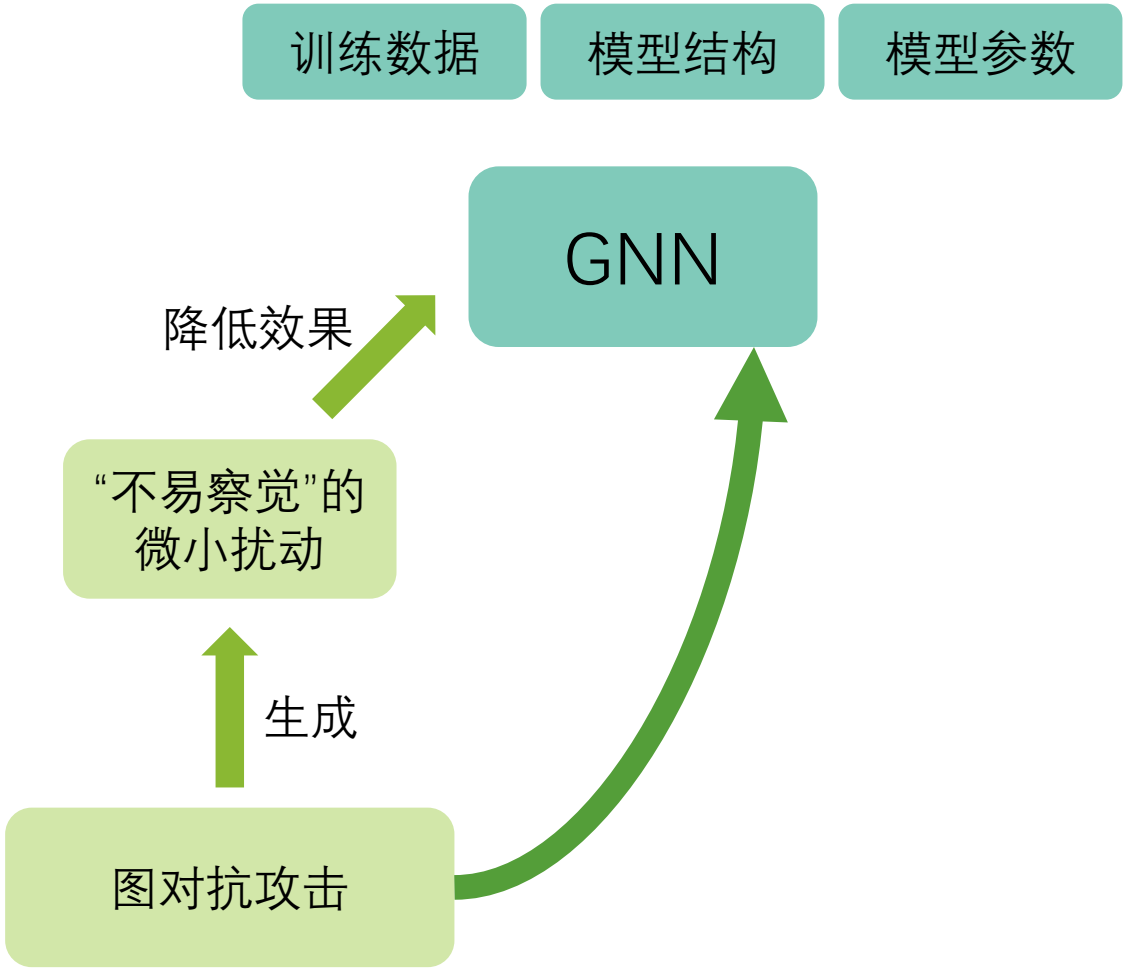
学习到

$S' \in [0,1]^{N \times N}$

PGD拓扑攻击


$l(v_i) = \max\{z[y_i] - \max_{c \neq y_i} z[c], -\kappa\}$
最小化
求解
$S \in \{0,1\}^{N \times N}$
放松约束
$S' \in [0,1]^{N \times N}$
最小化
Projected Gradient Descent （PGD）
学习到
$S' \in [0,1]^{N \times N}$
采样
$S \in \{0,1\}^{N \times N}$
Gradient Descent
$S' \in R^{N \times N}$
Project
$S' \in [0,1]^{N \times N}$

基于积分梯度的攻击



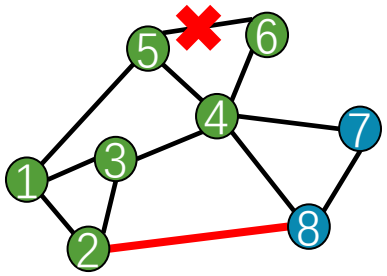
白盒攻击

攻击者可以获得被攻击的模型的完整信息

攻击目标

- 攻击节点分类任务
- 对于某个标签为y_i的节点v_i，攻击者希望GNN能把它分类为其它标签

基于积分梯度的攻击



A, F

A', F'

A

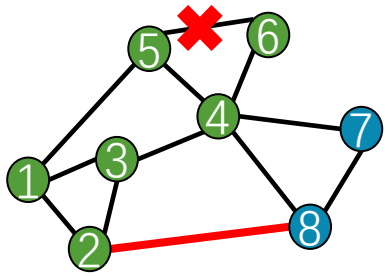
- 加边
- 减边

F

特征被假设为二分值

- $0 \rightarrow 1$
- $1 \rightarrow 0$

基于积分梯度的攻击



A

A'

连续的

GNN

参数固定

预测的节点v_i分类概率

$l(v_i)$
交叉熵损失函数

最大化

梯度上升

$\frac{\partial l(v_i)}{\partial A[i,j]}$

“稍微”增加A[i,j]对损失函数的影响

利用梯度作为指示函数

0

1

$$S(i,j) = \frac{\partial l(v_i)}{\partial A[i,j]} + \frac{\partial l(v_i)}{\partial A[i,j]} + \frac{\partial l(v_i)}{\partial A[i,j]}$$

减边的情况下分数定义为上式的相反数

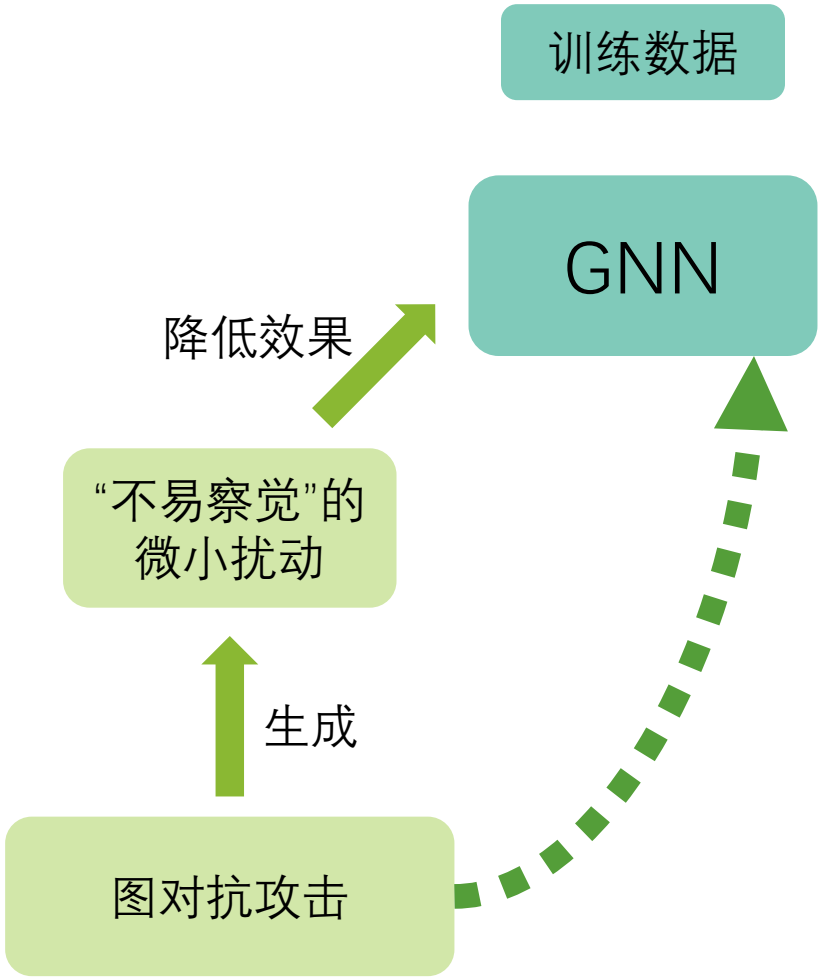
贪心算法

选择 $S(i,j)$最大的节点对进行改变

Adversarial Examples on Graph Data: Deep Insights into Attack and Defense

- 图对抗攻击
  - 白盒攻击
  - 灰盒攻击
  - 黑盒攻击

Nettack



灰盒攻击

攻击者可以获得被攻击的模型的训练数据

攻击目标

- 攻击节点分类任务
- 对于某个标签为y_i的节点v_i，攻击者希望GNN能把它分类为其它标签

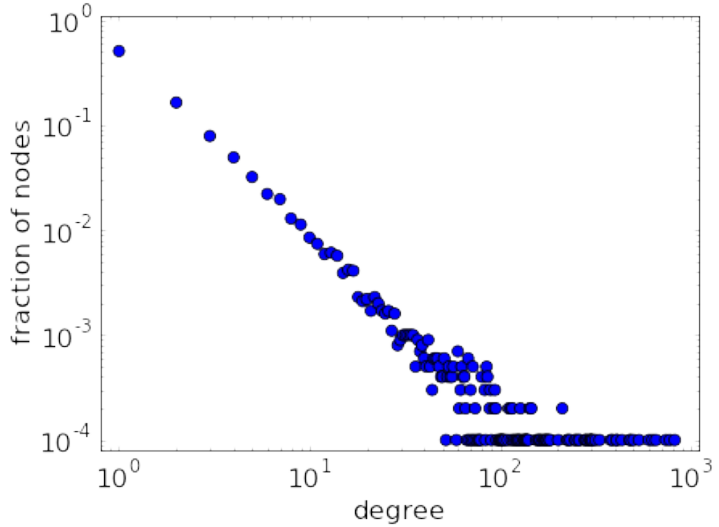
攻击的限制

$$\|\boldsymbol{A}' - \boldsymbol{A}\|_0 + \|\boldsymbol{F}' - \boldsymbol{F}\|_0 \leqslant \Delta$$

- 度的分布
- 特征共现

攻击的限制

度的分布

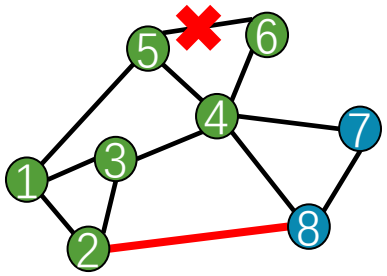


特征共现

机器学习文献

训练	爸爸
梯度下降	家庭
深度学习	牛奶

Nettack



A, F

A', F'

A

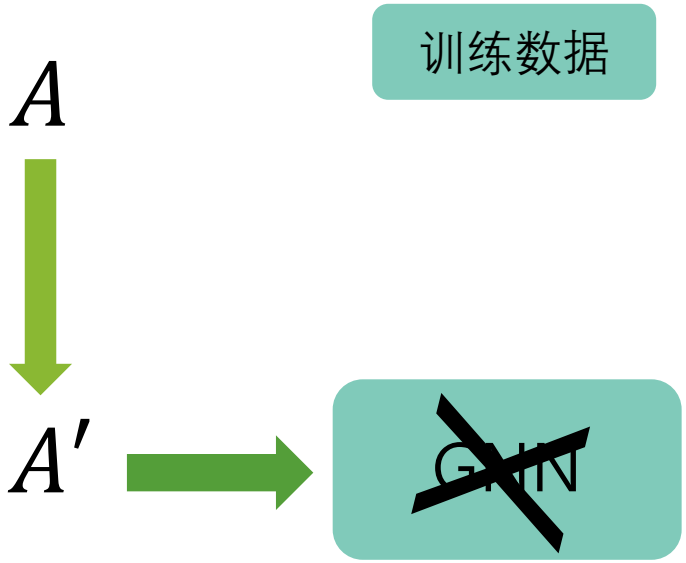
- ❑ 加边
- ❑ 减边

F

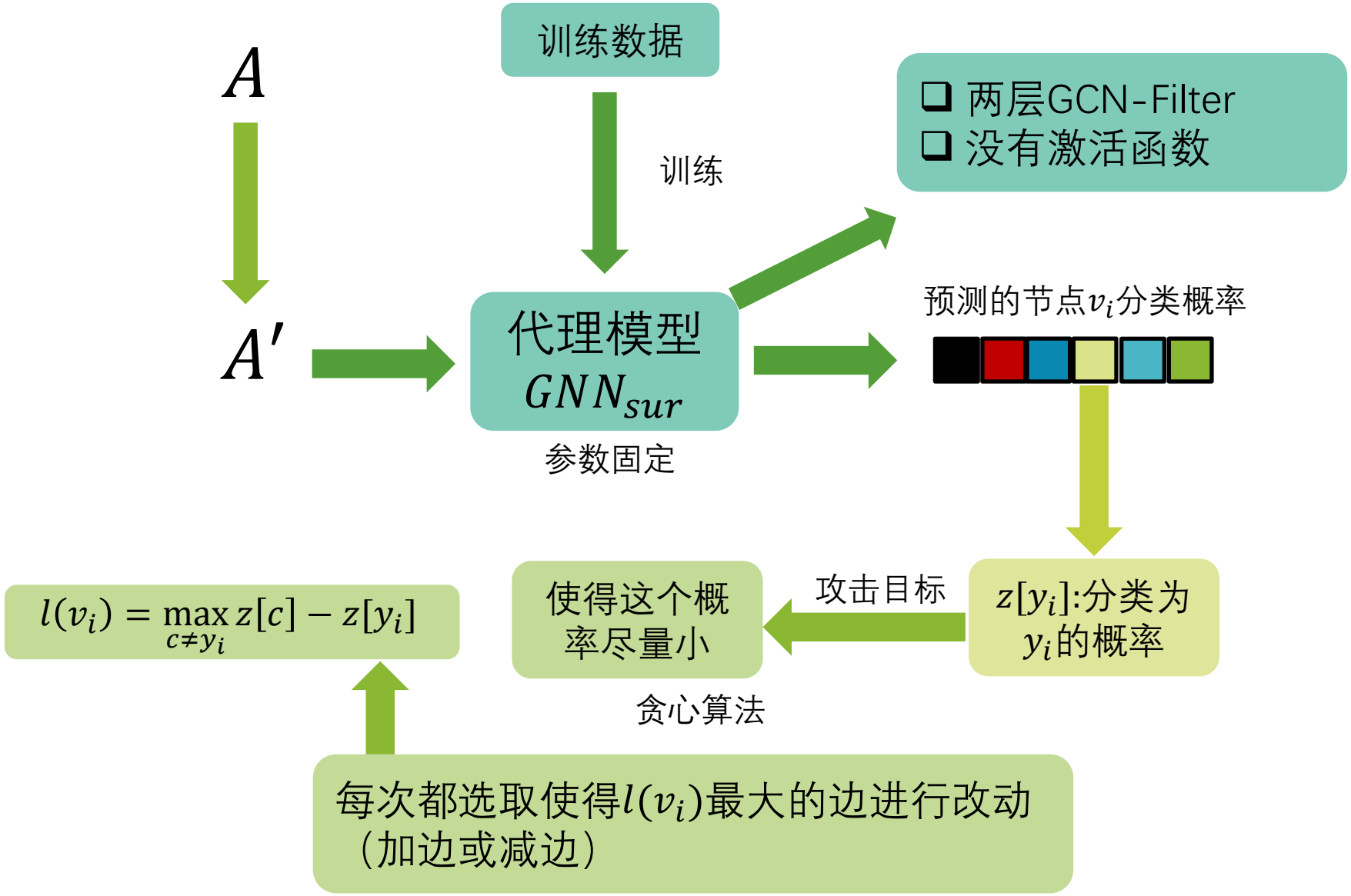
特征被假设为二分值

- ❑ $0 \rightarrow 1$
- ❑ $1 \rightarrow 0$

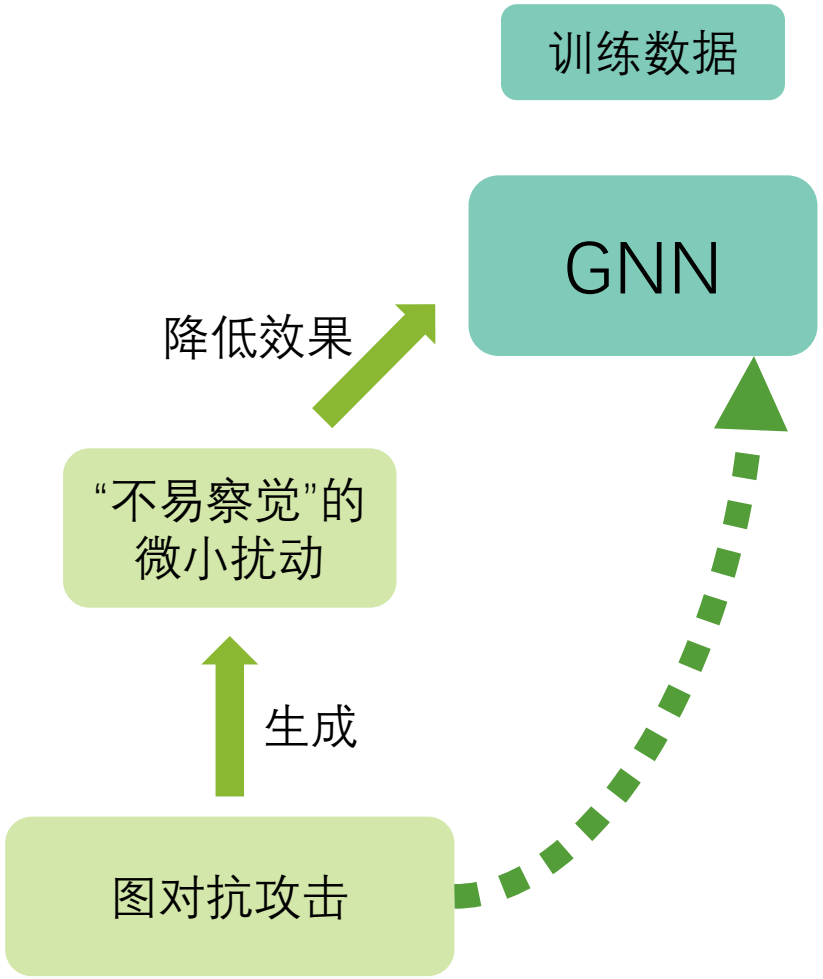
Nettack



Nettack


A
A'
训练数据
训练
代理模型
GNN_{sur}
参数固定
两层GCN-Filter
没有激活函数
预测的节点v_i分类概率
$z[y_i]$:分类为y_i的概率
攻击目标
使得这个概率尽量小
贪心算法
$l(v_i) = \max_{c \neq y_i} z[c] - z[y_i]$
每次都选取使得$l(v_i)$最大的边进行改动
（加边或减边）

基于元梯度的攻击（Meta Attack）



灰盒攻击

攻击者可以获得被攻击的模型的训练数据

攻击目标

- 攻击节点分类任务
- 攻击者希望的目标是降低GNN在测试集上的整体效果

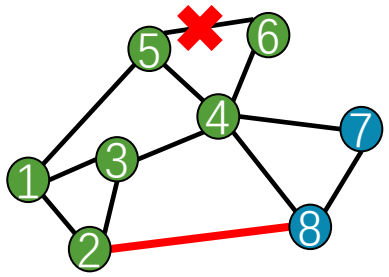
攻击的限制

$$\|\boldsymbol{A}' - \boldsymbol{A}\|_0 + \|\boldsymbol{F}' - \boldsymbol{F}\|_0 \leqslant \Delta$$

- 度的分布
- 特征共现

Adversarial Attacks on Graph Neural Networks via Meta Learning

基于元梯度的攻击（Meta Attack）



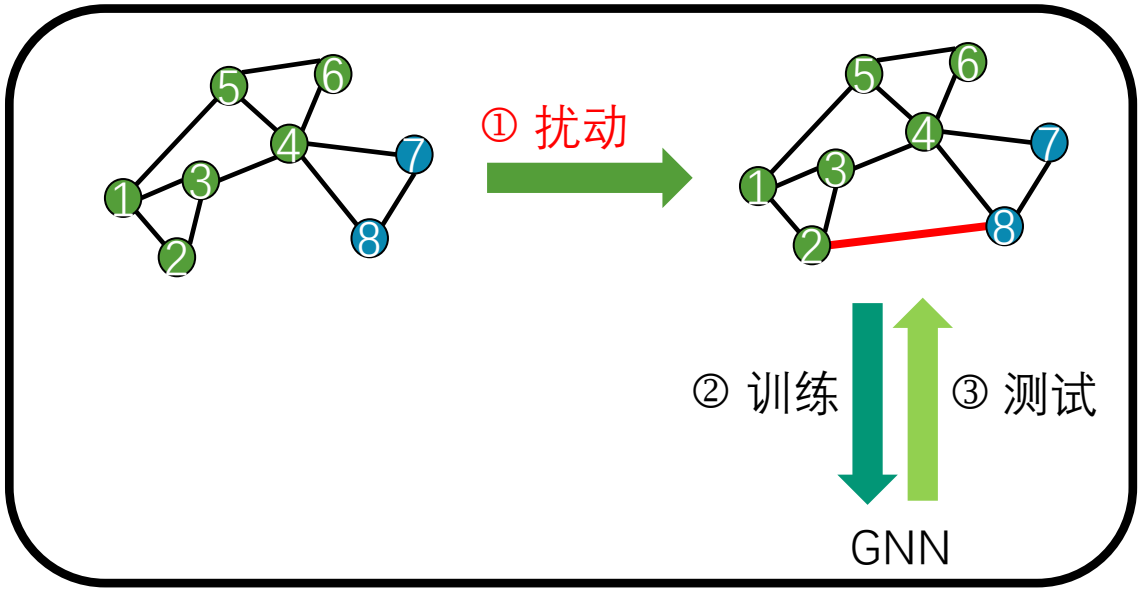
A

A'

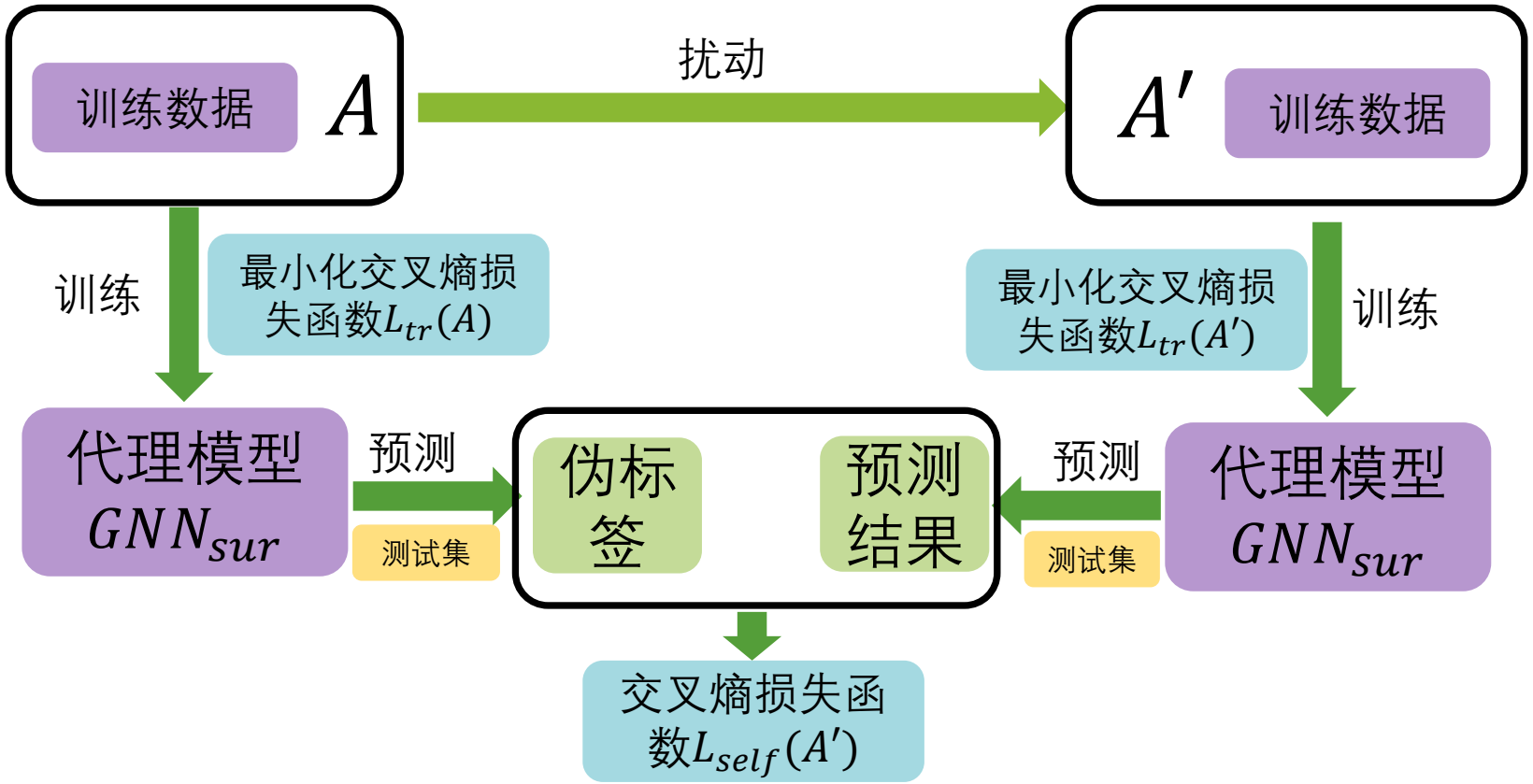
- 加边
- 减边

投毒攻击

- 进行图扰动
- 在被扰动的图上训练模型
- 在被扰动的图上测试模型



基于元梯度的攻击（Meta Attack）



$$\text{最小化 } L_{atk}(A') = -L_{self}(A') \ -L_{tr}(A')$$

$$\min_{\mathbf{A}'} \quad \mathcal{L}_{\text{atk}}\ (GNN_{sur}(\mathbf{A}';\Theta^*)) \qquad \text{s.t.} \qquad \Theta^* = \arg\min_{\Theta} \quad \mathcal{L}_{\text{tr}}\ (GNN_{sur}(\mathbf{A}';\Theta))$$

基于元梯度的攻击（Meta Attack）

$$\min_{\mathbf{A}'} \quad \mathcal{L}_{\text{atk}}\ (GNN_{sur}(\mathbf{A}';\Theta^*)) \qquad \text{s.t.} \qquad \Theta^* = \arg\min_{\Theta} \quad \mathcal{L}_{\text{tr}}\ (GNN_{sur}(\mathbf{A}';\Theta))$$

元梯度

$$\nabla^{meta}_{\mathbf{A}'} := \nabla_{\mathbf{A}'}\mathcal{L}_{\text{atk}}\ (GNN_{\text{sur}}\ (\mathbf{A}';\Theta^*)) \quad \text{s.t.} \quad \Theta^* = \arg\min_{\Theta} \quad \mathcal{L}_{\text{tr}}\ (GNN_{\text{sur}}\ (\mathbf{A}';\Theta))$$

元梯度下降

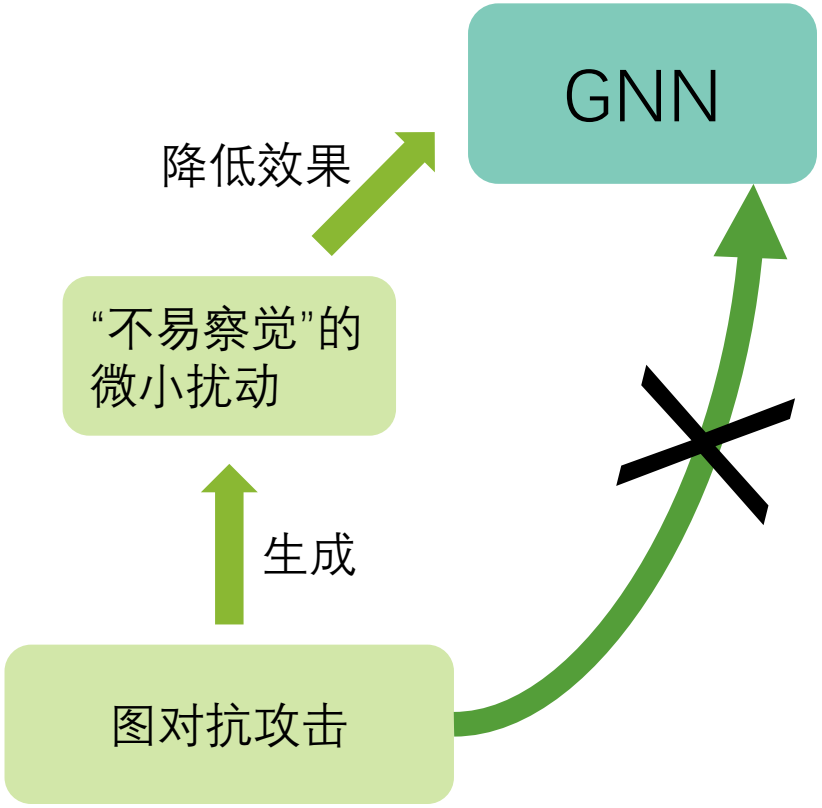
- 元梯度下降求解A'
- 得到的A'是稠密，连续的

用元梯度做指示

- $\nabla^{meta}_{\mathbf{A}'[i,j]}$ 衡量了“稍微”增加$A'[i,j]$对损失函数的影响
- 加边 $S(i,j) = \nabla^{meta}_{\mathbf{A}'[i,j]}$
- 减边 $S(i,j) = -\nabla^{meta}_{\mathbf{A}'[i,j]}$
- 贪心算法：选择S(i,j)最大的节点对进行改变 （加边/减边）

- 图对抗攻击
  - 白盒攻击
  - 灰盒攻击
  - 黑盒攻击

RL-S2V



黑盒攻击

- 攻击者不能可以获得被攻击的模型的任何信息
- 攻击者可以得知模型的预测结果

攻击目标

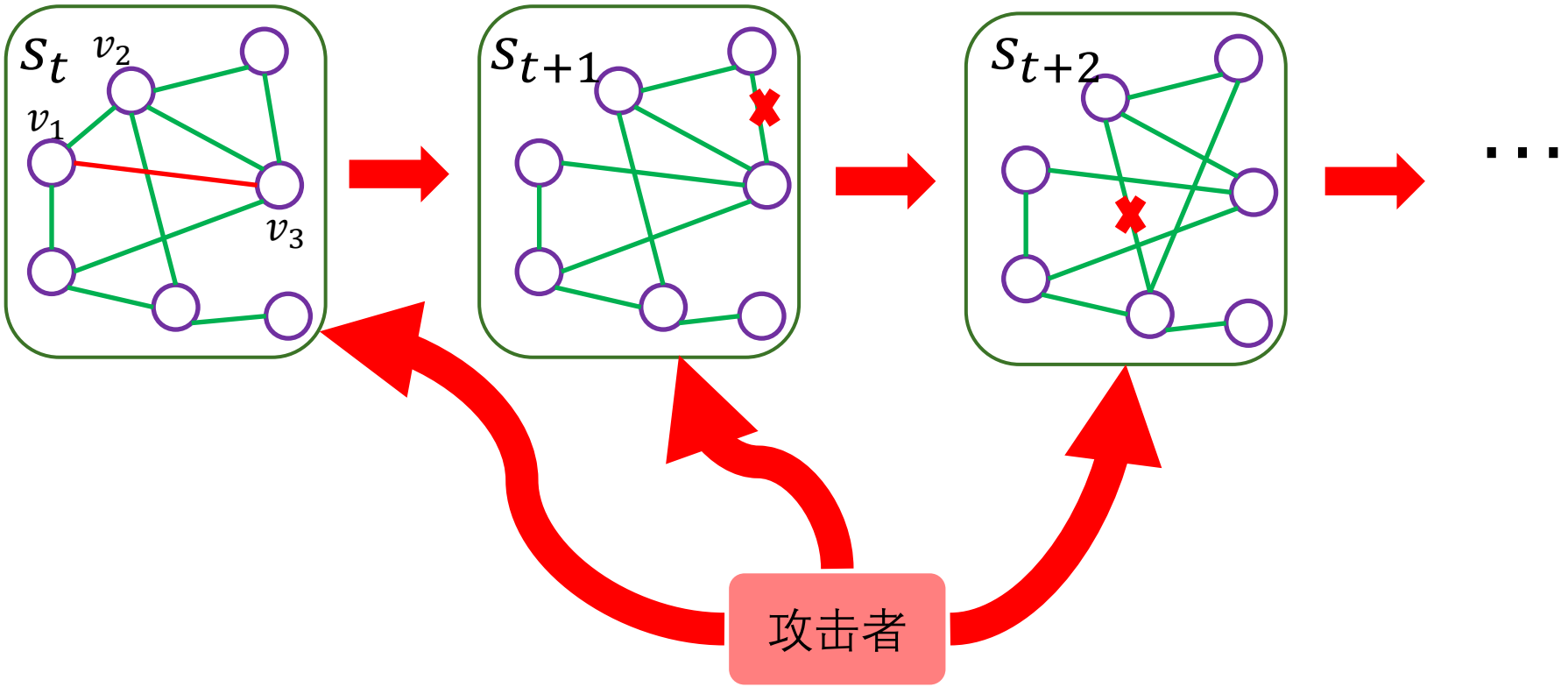
- 攻击节点分类任务
- 对于某个标签为 y_i 的节点 v_i ，攻击者希望GNN能把它分类为其它标签

攻击的限制

$$\left|(\mathcal{E}-\mathcal{E}')\cup(\mathcal{E}'-\mathcal{E})\right|\leqslant\Delta$$

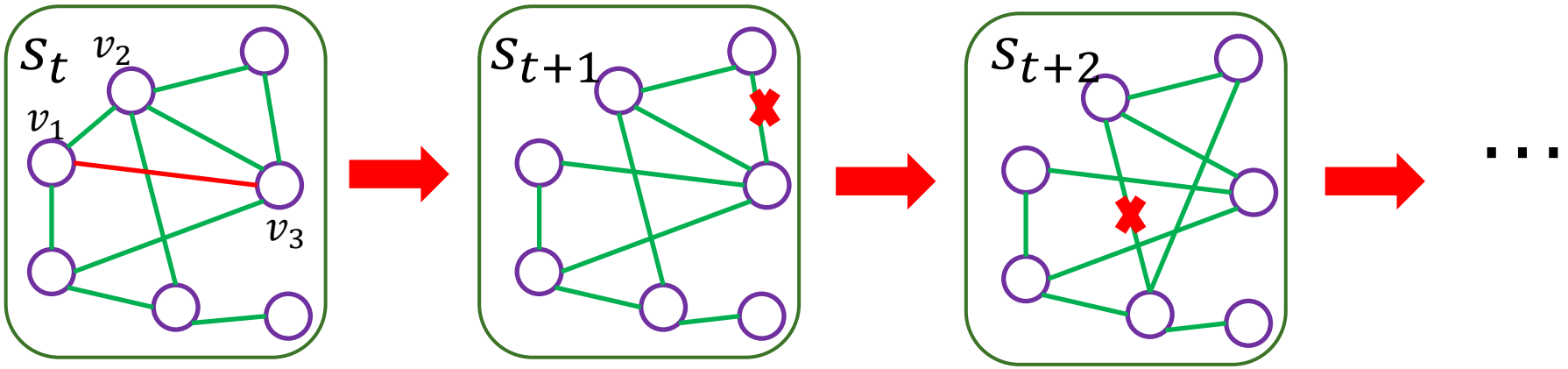
添加的边只能连接在原图中距离少于d的节点

RL-S2V



马尔可夫决策过程：当前的决策只基于当前的状态而与之前的决策无关

RL-S2V



状态空间

- 所有中间图
- 初始状态是$S_1 = G$

行为空间

- 加边或减边
- 第t个行为: a_t

奖励

- 攻击成功: 1
- 攻击失败: -1
- 中间步骤: 0

攻击成功

目标模型基于被攻击的图的预测结果与原图不同

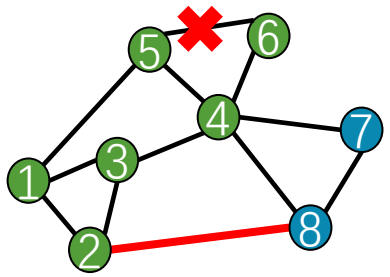
终止条件

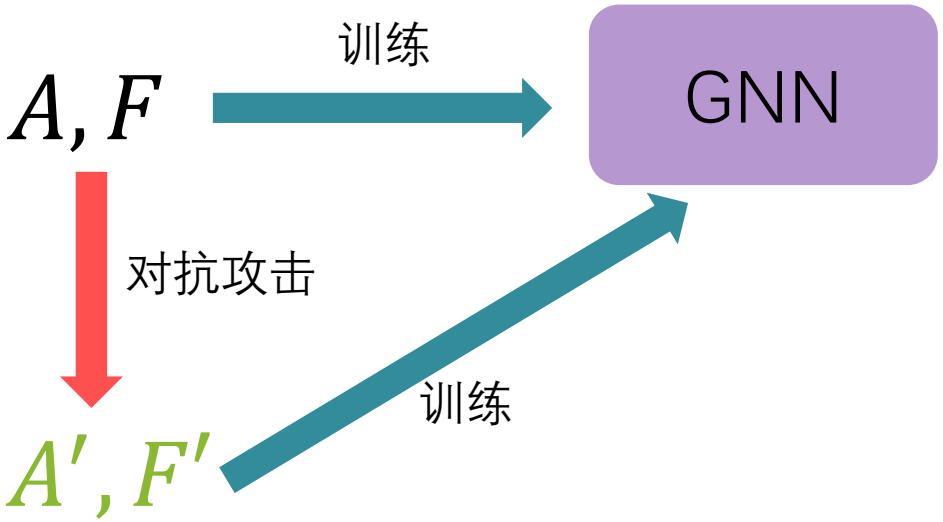
攻击者修改的边的数量达到了预算上限

- 图神经网络鲁棒性简介
- 图对抗攻击
- 图对抗防御

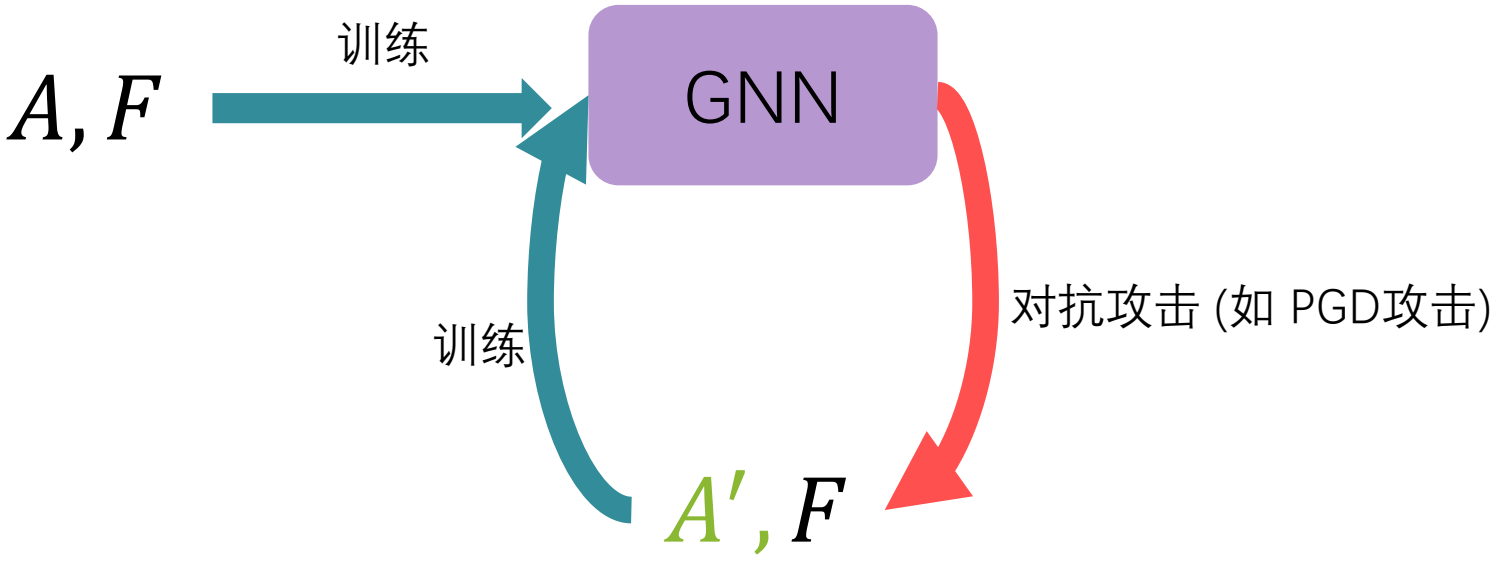
- 图对抗防御
  - 对抗训练
  - 图净化
  - 图结构学习
  - 图注意力机制

对抗训练

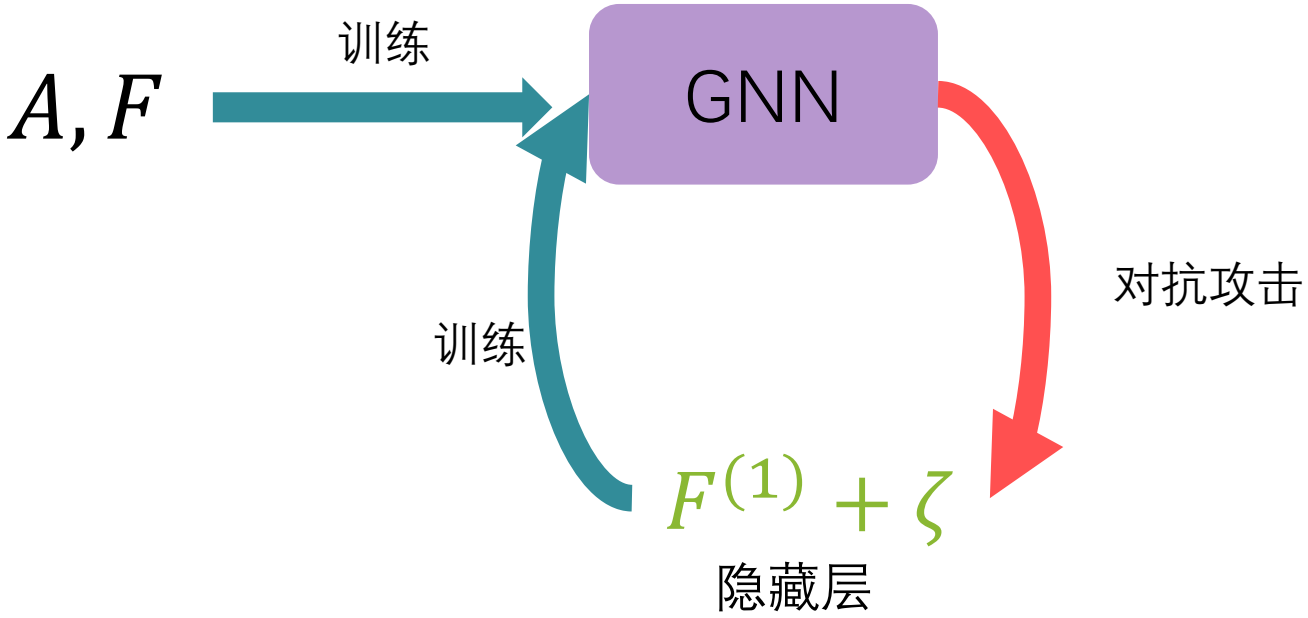




针对图结构的图对抗训练



针对图结构和节点特征的图对抗训练

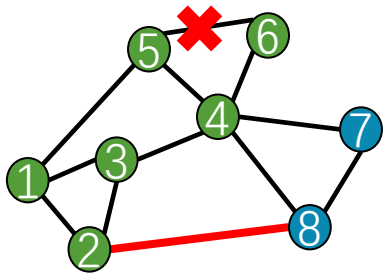


$$\min_{\Theta} \max_{\zeta \in D} \mathcal{L}_{\text{train}} \left(\boldsymbol{A}, \mathbf{F}^{(1)} + \boldsymbol{\zeta}; \boldsymbol{\Theta} \right),$$

$$D = \{\zeta; \|\zeta_i,\|_2 \leqslant \Delta\}$$

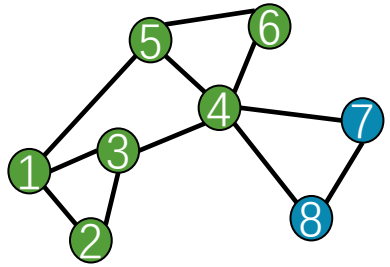
- 图对抗防御
  - 对抗训练
  - 图净化
  - 图结构学习
  - 图注意力机制

图净化的主要思想



预处理

净化



A'

被攻击的图

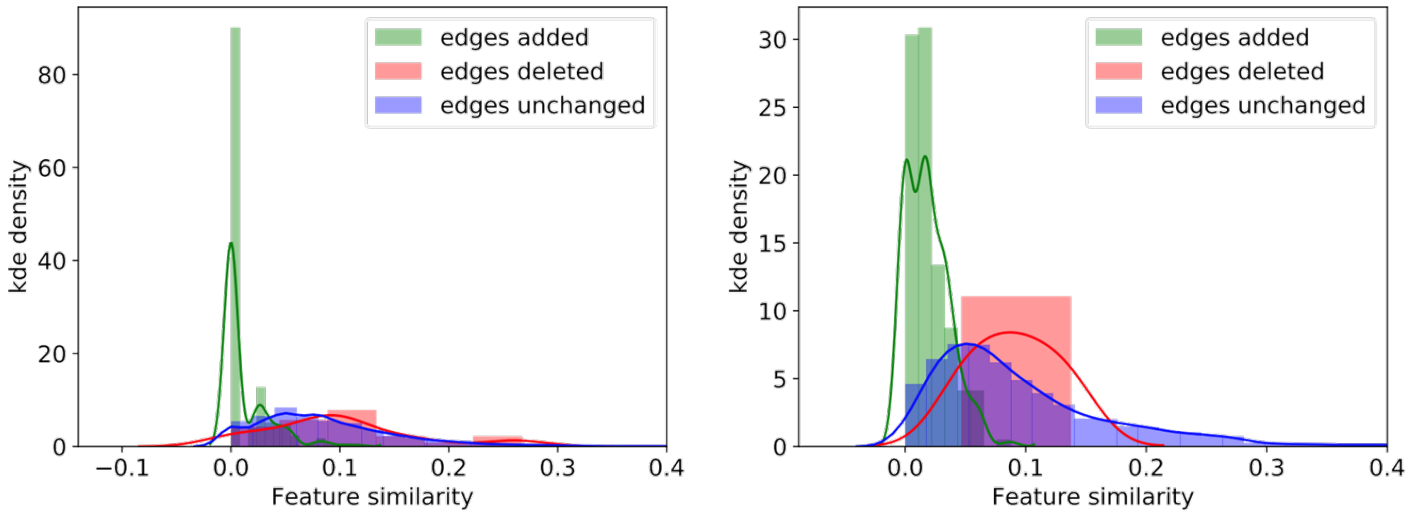
A

“干净”的图

训练

GNN

图净化：去掉“错误”的边



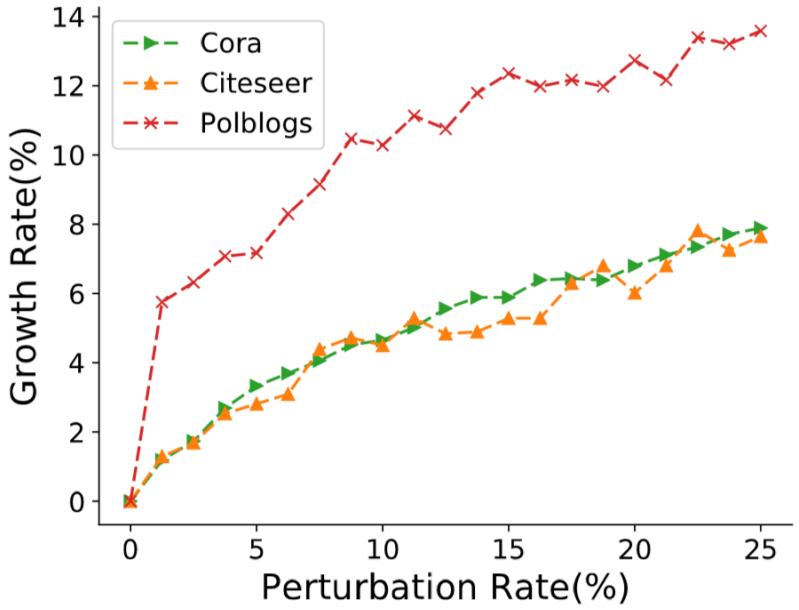
(a) Cora (b) Citeseer

攻击者倾向于添加连接不相似的节点的边

按照一定的阈值去掉连接不相似节点的边

Adversarial Examples on Graph Data: Deep Insights into Attack and Defense

图净化：低秩近似



(b) Rank Growth

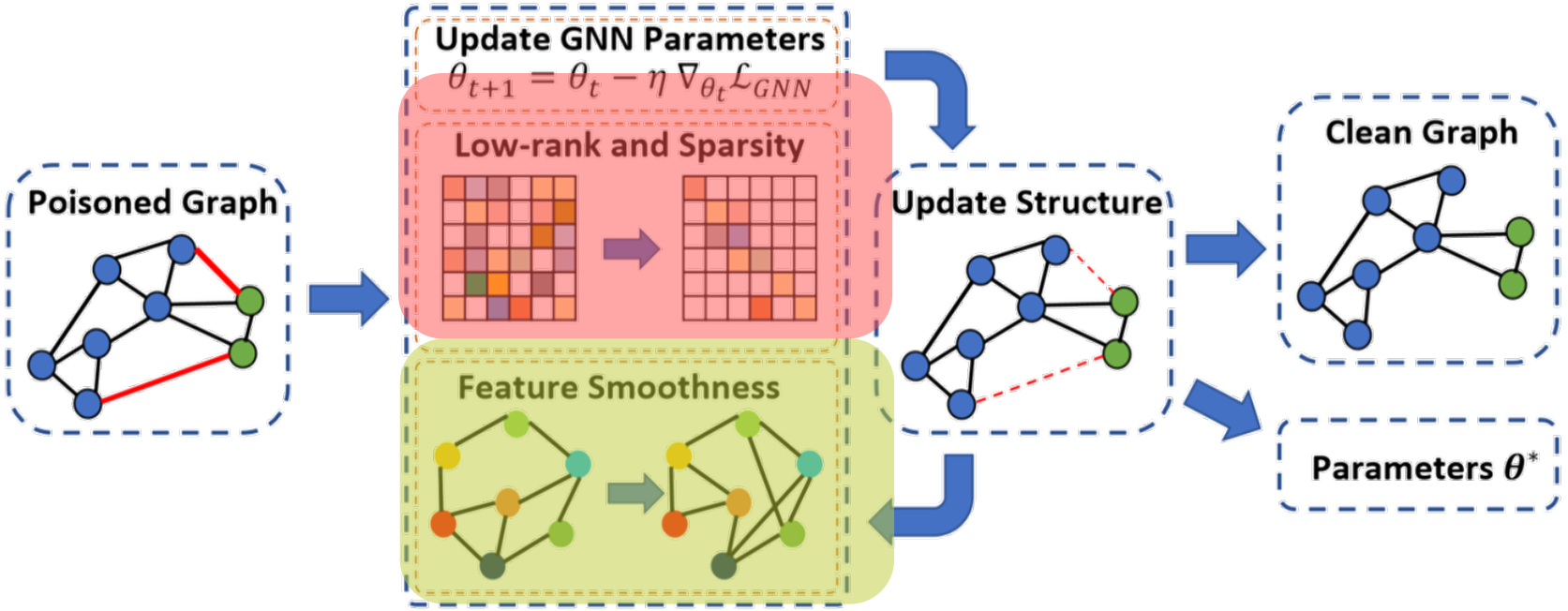
攻击者会增加邻接矩阵的秩

利用SVM获得邻接矩阵的低秩近似

All You Need Is Low (Rank): Defending Against Adversarial Attacks on Graphs

- 图对抗防御
  - 对抗训练
  - 图净化
  - 图结构学习
  - 图注意力机制

图结构学习：Pro-GNN



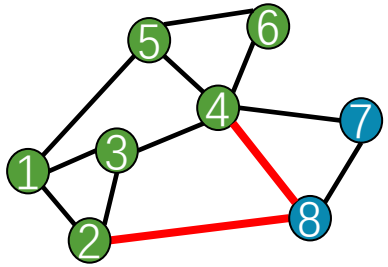
$$\min_{\Theta_S} \mathcal{L}_{\text{train}}\left(\boldsymbol{S}, \boldsymbol{F}; \boldsymbol{\Theta}\right) + \|\boldsymbol{A} - \boldsymbol{S}\|_F^2 + \beta_1 \|\boldsymbol{S}\|_1 + \beta_2 \|\boldsymbol{S}\|_* + \beta_3 \cdot \operatorname{tr}\left(\boldsymbol{F}^T \boldsymbol{L} \boldsymbol{F}\right)$$

同时学习“干净”图结构以及最优的GNN模型

Graph Structure Learning for Robust Graph Neural Networks

- 图对抗防御
  - 对抗训练
  - 图净化
  - 图结构学习
  - 图注意力机制

图注意力机制



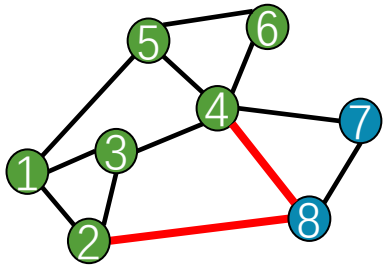
对每条边学习注意力机制

- 希望对被攻击的边或节点学习较小的注意力
- 减少被攻击的边或节点带来的不良影响

Robust Graph Neural Networks (RGCN)

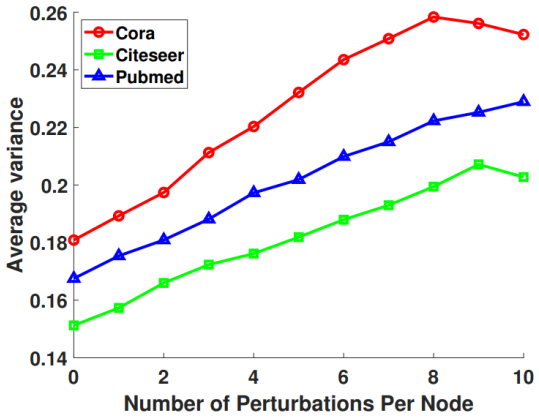
利用高斯分布来建模节点表示

$$\boldsymbol{h}_i^{(l)} \sim N(\boldsymbol{\mu}_i^{(l)}, diag\left(\boldsymbol{\sigma}_i^{(l)}\right))$$



$$\mu_i^{(l+1)} = \sum_{j \in N(i)} \frac{1}{\sqrt{\widetilde{\boldsymbol{D}}_{ii}\widetilde{\boldsymbol{D}}_{jj}}} (\boldsymbol{h}_j^{(l)} \odot \boldsymbol{\alpha}_j^{(l)}) \boldsymbol{W}_{\boldsymbol{\mu}}^{(l)}$$

$$\boldsymbol{\alpha}_j^{(l)} = \exp(-\gamma \boldsymbol{\sigma}_j^{(l)})$$



被攻击过的节点有较高的方差

Robust Graph Convolutional Networks Against Adversarial Attacks

- 图神经网络鲁棒性简介
- 图对抗攻击
- 图对抗防御

深蓝学院
shenlanxueyuan.com

感谢聆听！

Thanks for Listening