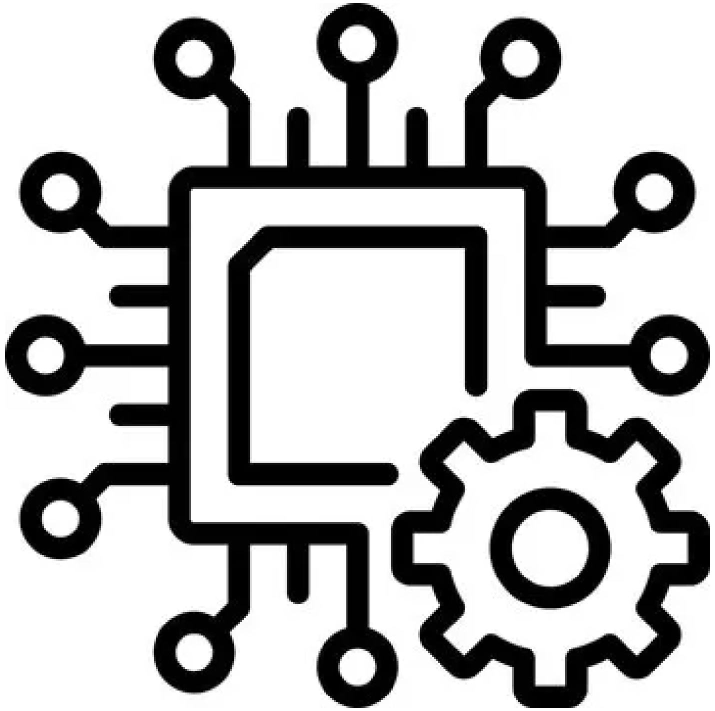
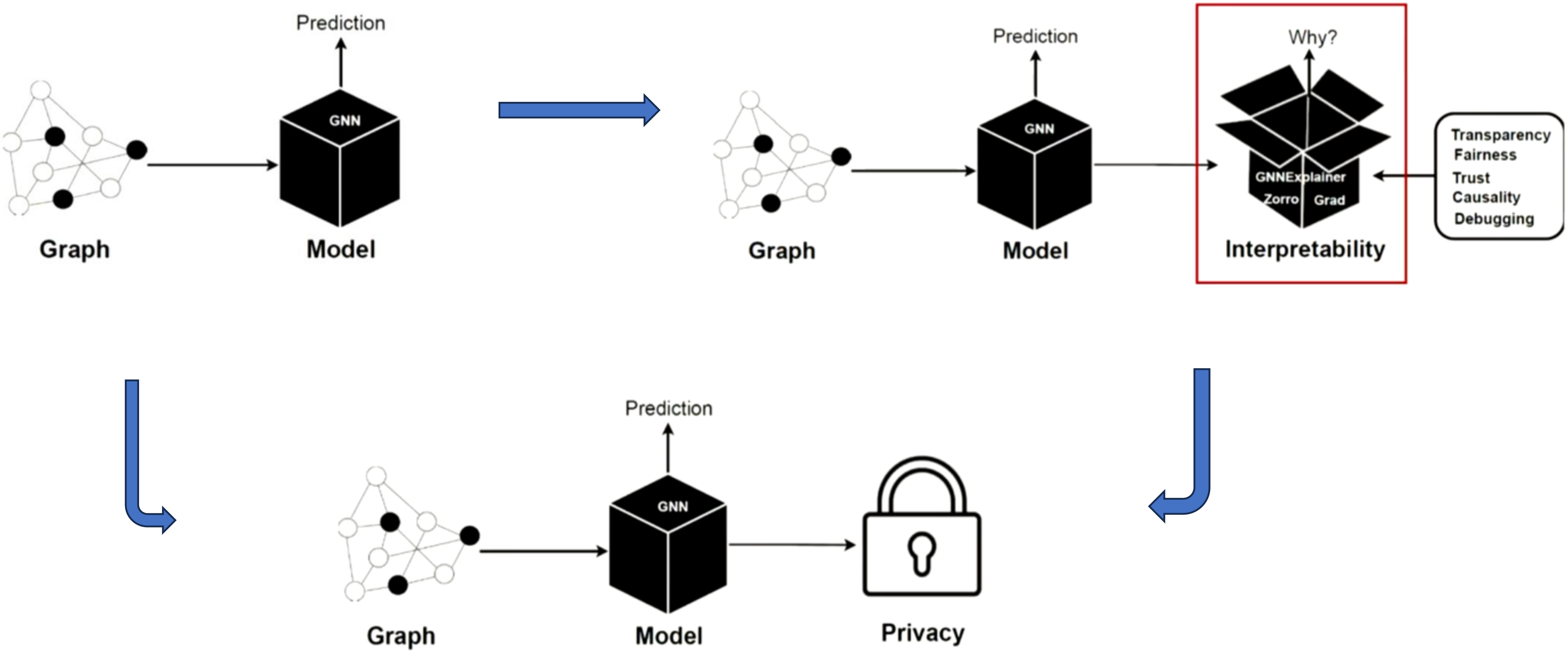
Private Graph Extraction via Feature Explanations



Rishi Raj Sahoo
SMLab Talk
Jan 22, 2025

INTRODUCTION



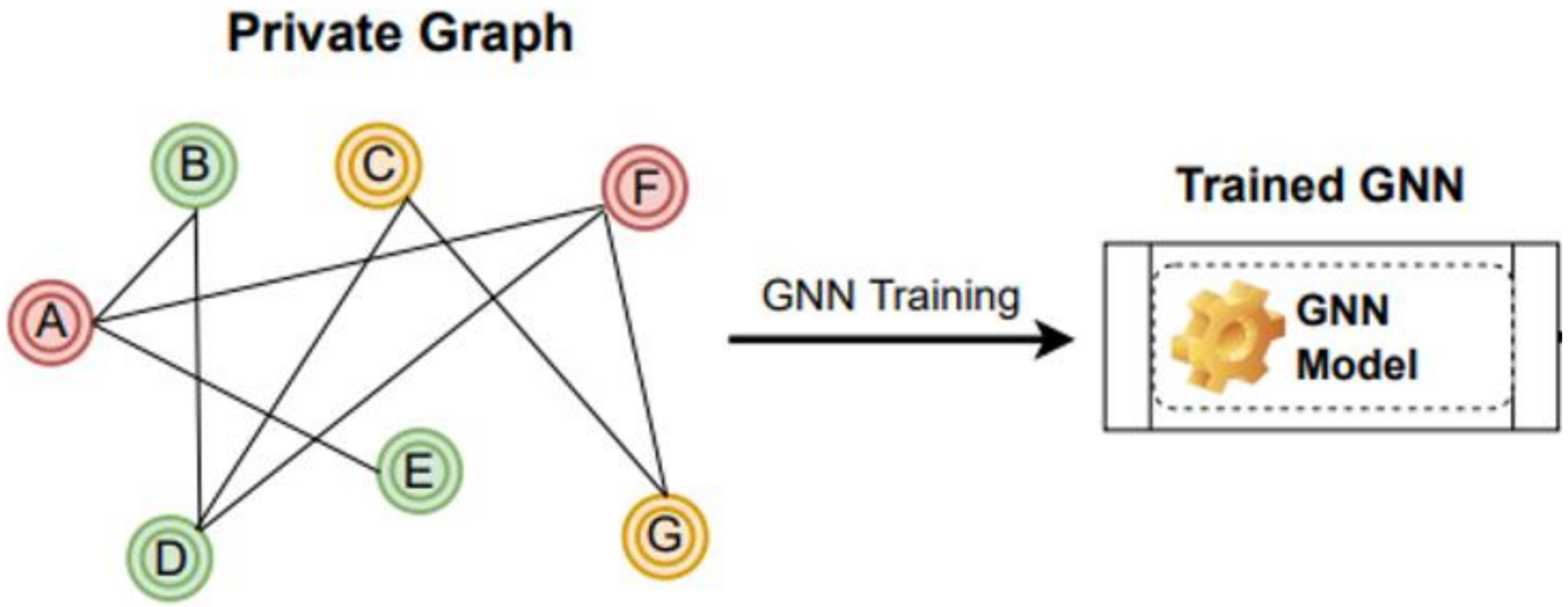
Even black-box model can leak information [1]

PRIVACY vs INTERPRETATBILITY

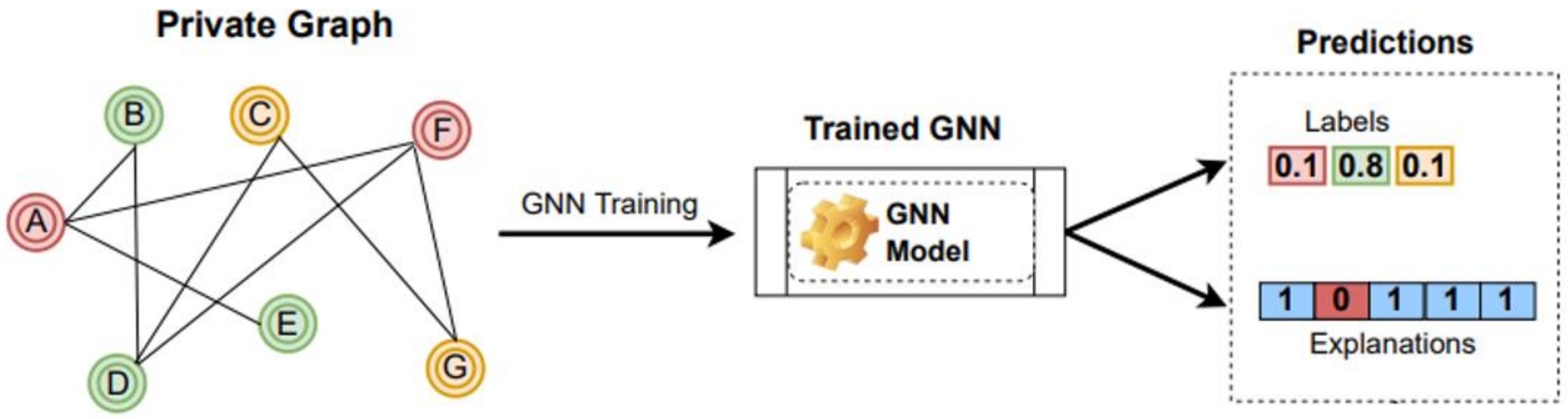
Privacy:
Which tries to preserve everything

Interpretability:
Which release everything(The **why** question)

MOTIVATION

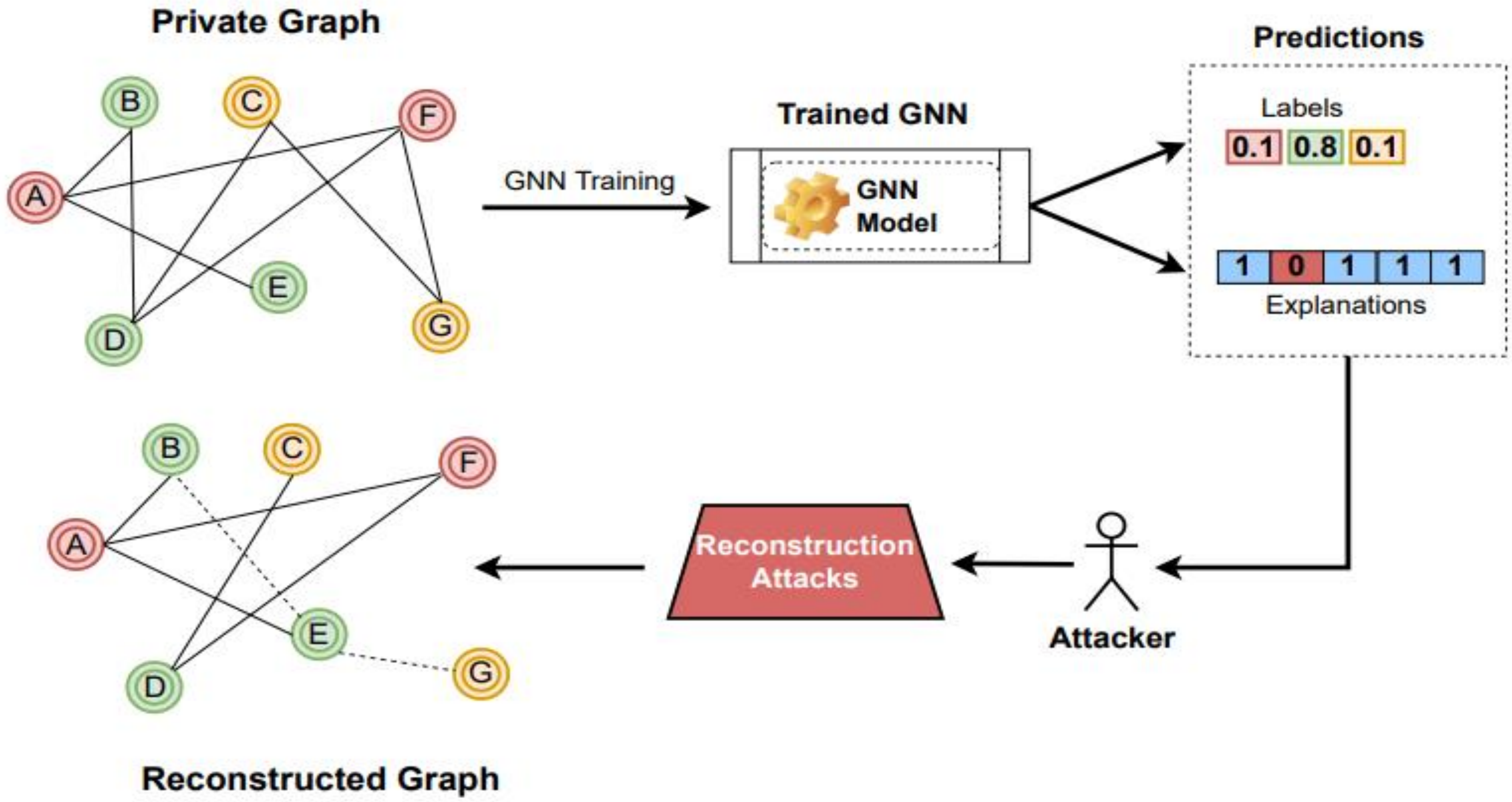


MOTIVATION



Private Graph
B
C
F
A
E
D
G
GNN Training
Trained GNN
GNN
Model
Predictions
Labels
0.1
0.8
0.1
1
0
1
1
1
Explanations

MOTIVATION



Goal: **Reconstruct** the **original graph**, given **explanation** and some **auxiliary information**

THREAT MODEL

Available:

- **Explanations**
- Trained GNN **Model**
- Node **Features**(Optional)
- **Labels** (Optional)

Private:

- **Graphs**/Link

EXPLANATION METHODS

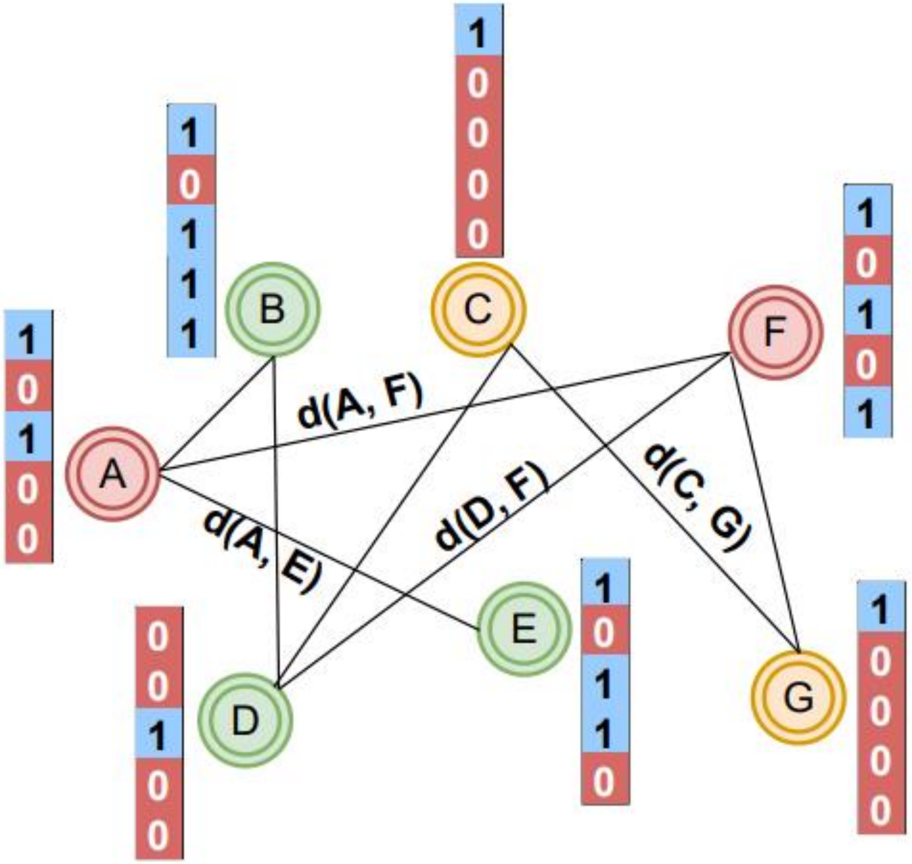
- Feature explanation methods are used.
- Why not node/edge explanations?

Gradient	Perturbation	Surrogate
• Grad • GradInput	• GNNExplainer • Zorro • Zorro-Soft	• GraphLime

ATTACK METHODOLOGIES

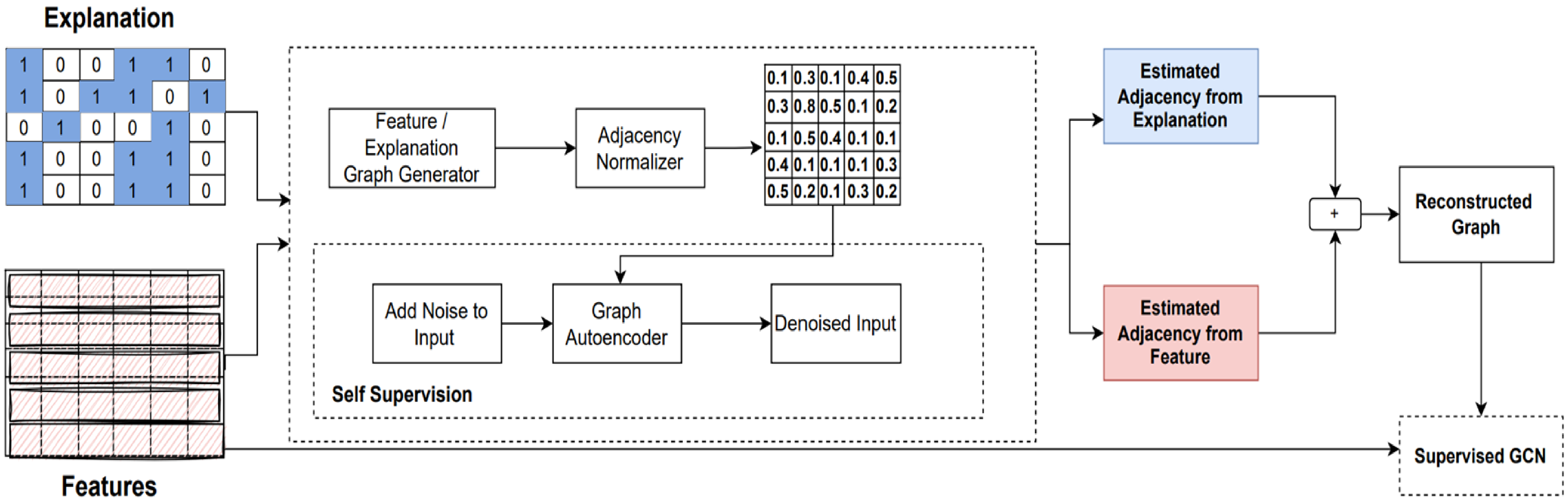
1. Explanation-only Attack (ExplainSim)

- Unsupervised attack
- Access to Explanation only
- Attacker assigns edges between the nodes if the distance between the feature vector is small
- Cosine similarity



ATTACK METHODOLOGIES

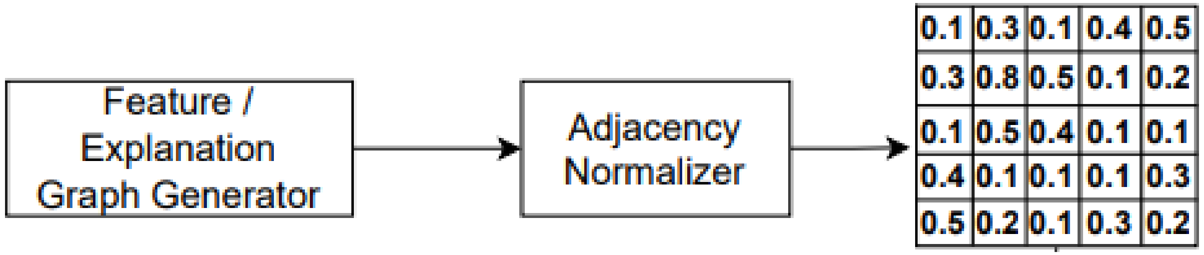
2. Explanation Augmentation Attacks



ATTACK METHODOLOGIES

Generator:

- **Input** = Node features and explanations
- **Output** = Adjacency matrix
- Each element is treated as a separable learnable parameter(**fully parameterised**)



Normalizer:

- **Symmetrize** and make **non-negative**
- **A** is the transformation to obtain symmetric matrix

$$A = D^{-\frac{1}{2}}\left(\frac{\mathrm{P}_{[0,1]}(\tilde{A}) + \mathrm{P}_{[0,1]}(\tilde{A})^T}{2}\right)D^{-\frac{1}{2}},$$

$$\tilde{A} = \mathrm{G}_{FP}(X; \theta_\mathrm{G}) = \theta_\mathrm{G}$$

Where,

$\tilde{A}$ = Adjacency Matrix

$\theta_\mathrm{G} \in \mathbb{R}^{n \times n}$ = Generator

$\mathrm{G}_{FP}(\cdot;\cdot)$ = Generator Function

Where P is a non-negative function defined by:

$$\mathrm{P}_{[0,1]}[x] = \begin{cases} 0 & x < 0, \\ 1 & x > 1, \\ x & \mathit{otherwise}. \end{cases}$$

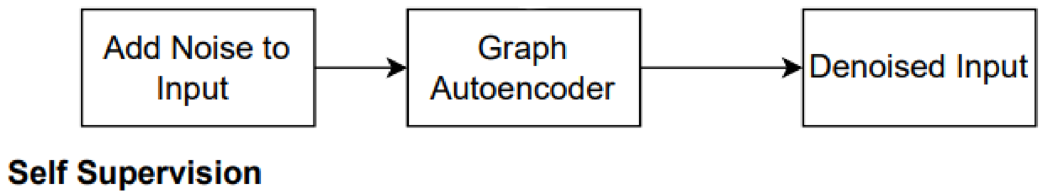
ATTACK METHODOLOGIES

Self-supervision:

- **Denoising autoencoder**
- **Input**: **Noisy features/explanation**

  +

  Graph sampled from the **generator** as input
- **Goal:** To **reconstruct** the true node features and explanations

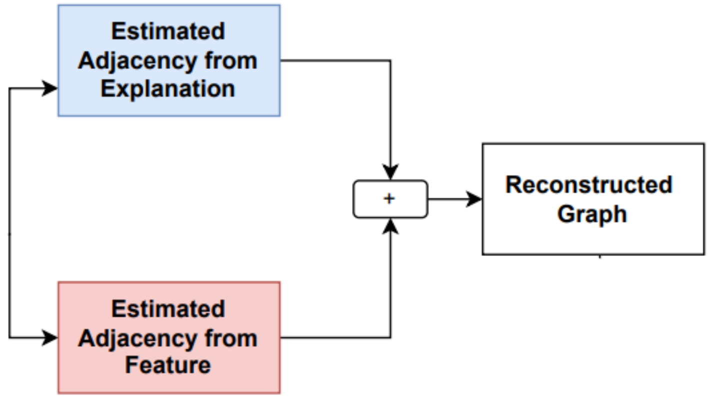


Combining adjacency:

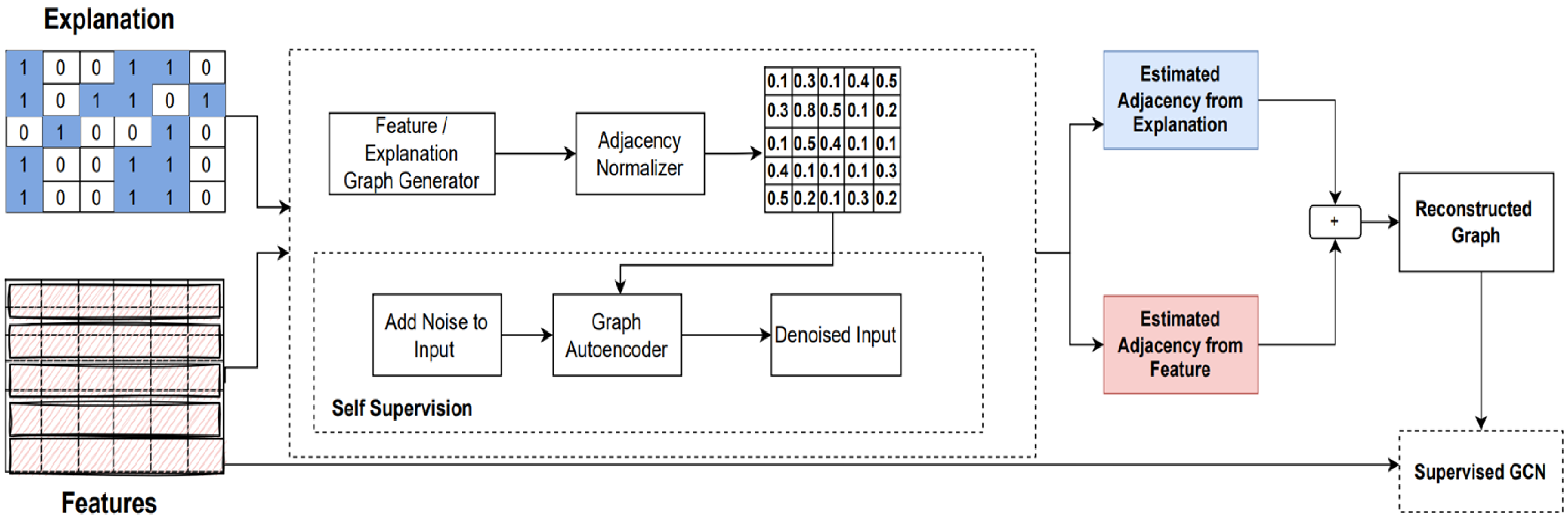
- Add the feature adjacency and explanation adjacency
- **Mult-task approach**: Predicting class label, reconstructing noisy feature and explanations = **reconstructed adjacency**
- Objective: **Minimize Loss**

$$\mathcal{L} = \mathcal{L}_{DAE} + \mathcal{L}_{DAE_{\mathcal{E}_X}} + \mathcal{L}_{C}.$$

Graph Stealing with Explanation and Features(GSEF)

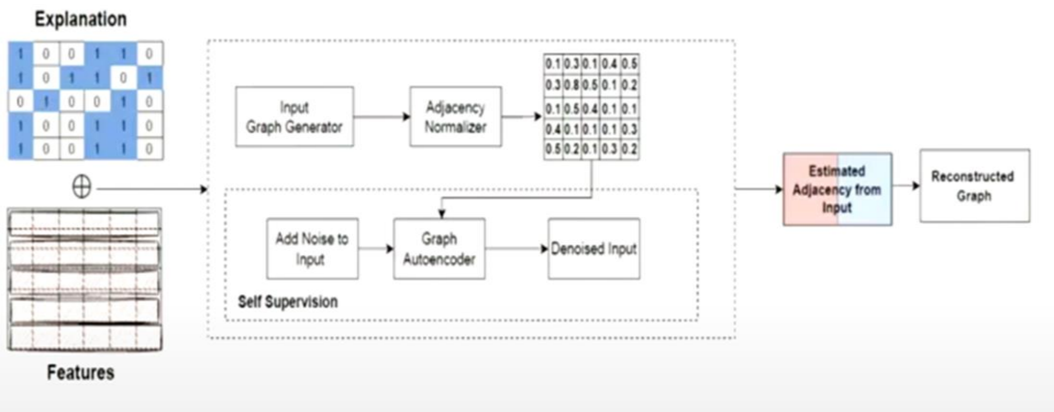


ATTACK METHODOLOGIES

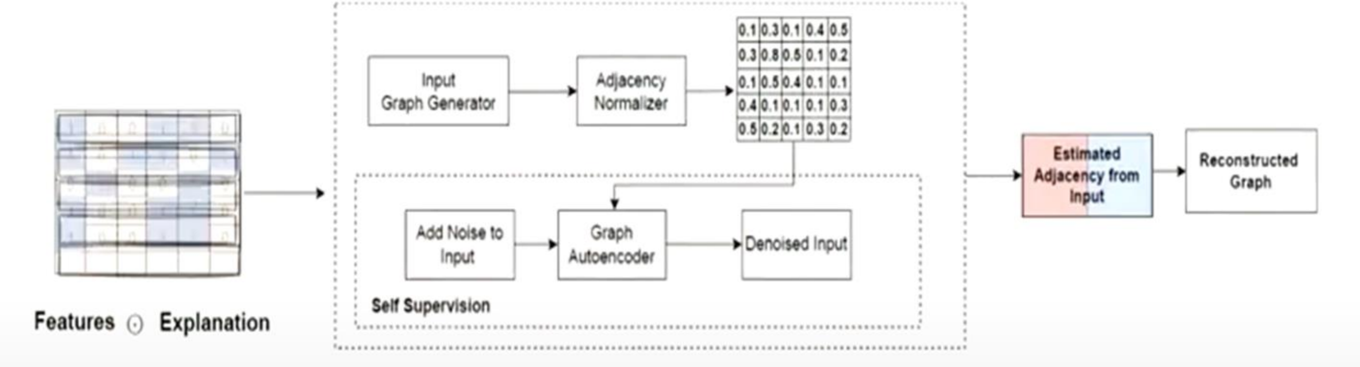


GSEF

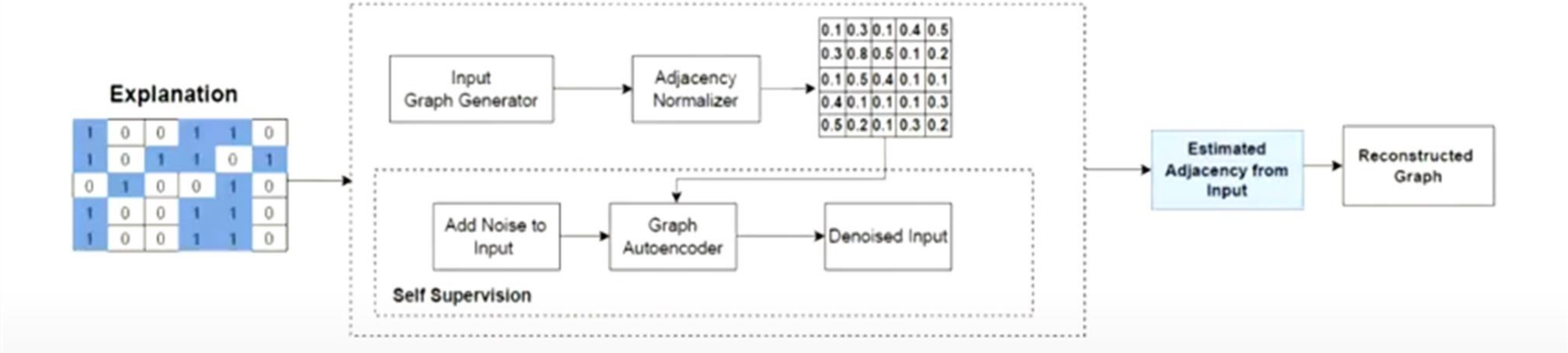
VARIANTS OF GSEF



GSEF-CONCAT



GSEF-MULT



GSE

SUMMARY OF ATTACKS

Attack	X	Y	$\mathcal{E}_X$
ExplainSim	✗	✗	✓
GSEF	✓	✓	✓
GSEF-concat	✓	✓	✓
GSEF-mult	✓	✓	✓
GSE	✗	✓	✓

Table 1: Attack taxonomy based on attacker's knowledge of node features (X), labels (Y) and feature explanations ($\mathcal{E}_X$).

COMPARED BASELINES

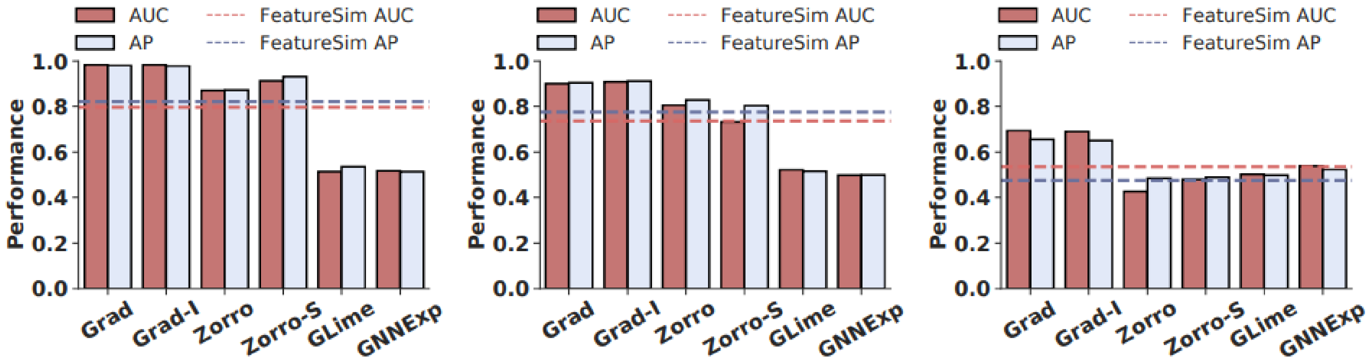
FeatureSim: Assigns links if the distance in feature space is small

GraphMI: Whitebox attack. The goal is to reconstruct the adjacency matrix given the features and labels

Link stealing attack(LSA): Creates a surrogate model and assigns a link if the posterior between the original label and surrogate model are close

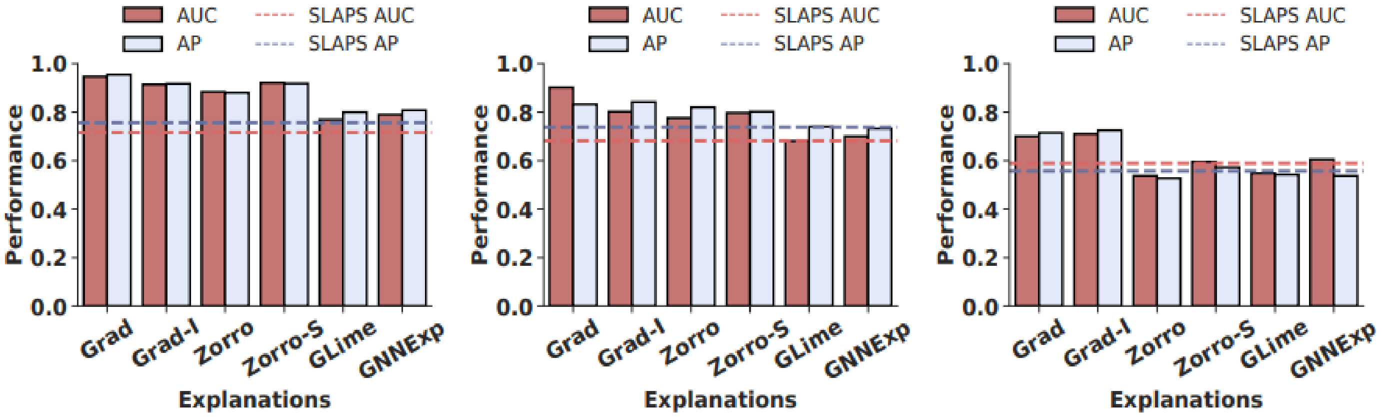
SLAPS: Graph structure learning approach that constructs the appropriate graph given the features and labels

PERFORMANCE



(a) Cora (b) CoraML (c) Bitcoin

ExplainSim vs FeatureSim



(a) Cora (b) CoraML (c) Bitcoin

GSEF vs SLAPS

MEASURING EXPLANATION QUALITY

Fidelity = Measure of the explanation's ability to approximate the model's behaviour (**faithfulness**)

Higher is better

$$\mathcal{F}(\mathcal{E}_X) = \mathbb{E}_{Y_{\mathcal{E}_X}|Z\sim\mathcal{N}}\left[\mathbb{1}_{f(X)=f(Y_{\mathcal{E}_X})}\right]$$

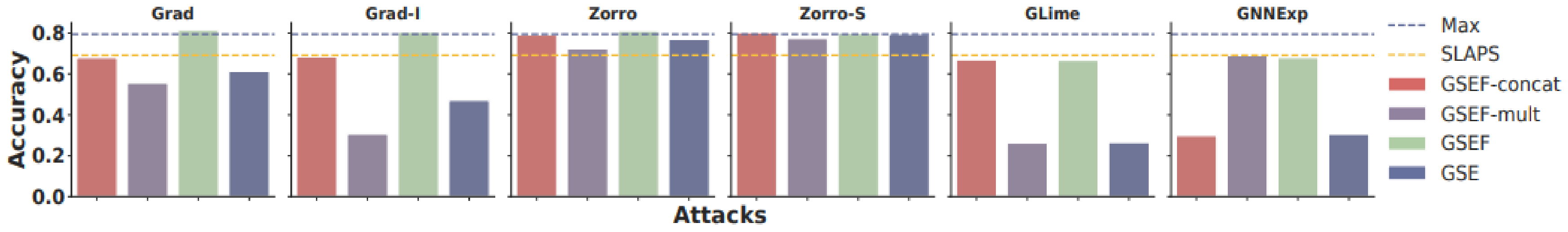
Sparsity = Meaningful explanation should be sparse(contains only subset of the features that is most predictive of the model's decision)

Lower the entropy, sparse the explanation

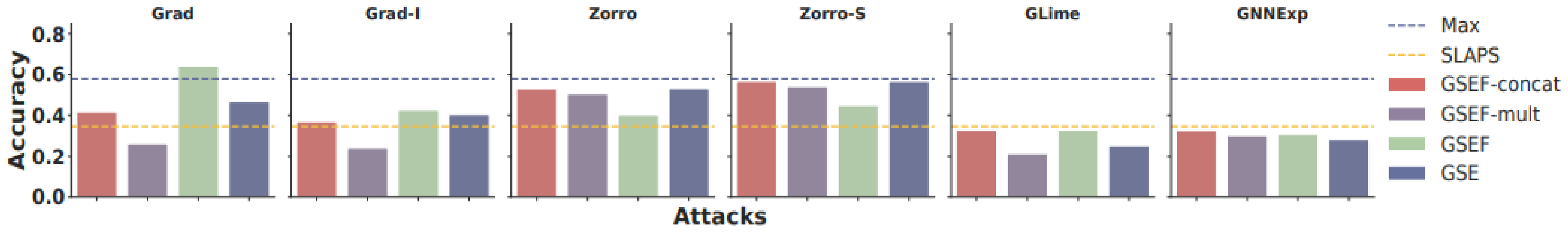
$$H(p) = -\sum_{f\in M} p(f)\log p(f).$$

Exp	Cora		CoraML		Bitcoin	
	Fidelity	Sparsity	Fidelity	Sparsity	Fidelity	Sparsity
Grad	0.23	3.99	0.22	5.24	0.83	0.64
Grad-I	0.19	3.99	0.20	5.30	0.82	0.64
Zorro	0.89	1.83	0.96	3.33	0.99	0.37
Zorro-S	0.98	2.49	0.84	2.75	0.95	0.96
GLime	0.19	0.88	0.20	0.98	0.82	0.13
GNNExp	0.74	7.27	0.55	5.70	0.90	2.05

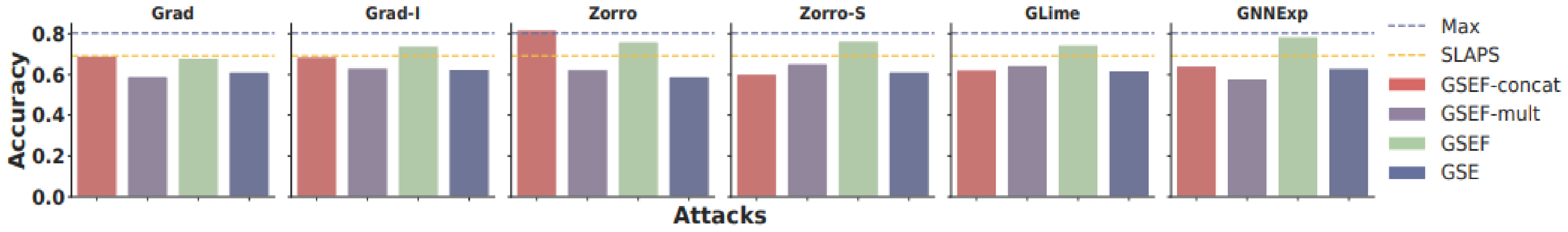
ACCURACY OF RECONSTRUCTED GRAPH



(a) Cora



(b) CoraML



(c) Bitcoin

SUMMARY

Exp	Attack	Cora AUC	Cora AP	CoraML AUC	CoraML AP	Bitcoin AUC	Bitcoin AP
Baseline	FeatureSim	0.799	0.827	0.706	0.753	0.535	0.478
	LsA [20]	0.795	0.810	0.725	0.760	0.532	0.500
	GraphMI [48]	0.856	0.830	0.808	0.814	0.585	0.518
	Slaps [13]	0.736	0.776	0.649	0.702	0.597	0.577
Grad	GSEF-concat	0.734	0.773	0.640	0.705	0.527	0.515
	GSEF-mult	0.678	0.737	0.666	0.730	0.264	0.383
	GSEF	0.948	0.953	**0.902**	0.833	**0.700**	**0.715**
	GSE	0.924	0.939	0.699	0.768	0.229	0.365
	ExplainSim	**0.984**	**0.978**	0.890	**0.891**	0.681	0.644
Grad-I	GSEF-concat	0.734	0.775	0.674	0.734	0.525	0.527
	GSEF-mult	0.691	0.742	0.717	0.756	0.252	0.380
	GSEF	0.949	0.950	0.887	0.832	**0.709**	**0.723**
	GSE	0.903	0.923	0.717	0.781	0.256	0.380
	ExplainSim	**0.984**	**0.979**	**0.903**	**0.899**	0.681	0.644
Zorro	GSEF-concat	0.823	0.860	0.735	0.786	0.575	0.529
	GSEF-mult	0.723	0.756	0.681	0.697	0.399	0.449
	GSEF	**0.884**	**0.880**	0.776	0.820	0.537	0.527
	GSE	0.779	0.810	0.722	0.777	**0.596**	**0.561**
	ExplainSim	0.871	0.873	**0.806**	**0.829**	0.427	0.485
Zorro-S	GSEF-concat	0.907	0.922	0.747	0.791	**0.601**	**0.590**
	GSEF-mult	0.794	0.815	0.712	0.740	0.490	0.491
	GSEF	**0.918**	0.923	**0.776**	**0.819**	0.598	0.565
	GSE	0.893	0.915	0.742	0.784	0.571	0.564
	ExplainSim	0.908	**0.934**	0.732	0.787	0.484	0.496
GLime	GSEF-concat	0.643	0.710	0.610	0.652	0.473	0.493
	GSEF-mult	0.516	0.522	0.517	0.528	0.264	0.371
	GSEF	**0.730**	**0.773**	**0.681**	**0.740**	**0.542**	**0.525**
	GSE	0.558	0.571	0.540	0.555	0.236	0.361
	ExplainSim	0.505	0.524	0.520	0.523	0.504	0.512
GNNExp	GSEF-concat	0.614	0.650	0.653	0.705	0.467	0.489
	GSEF-mult	0.724	0.760	0.637	0.692	0.390	0.454
	GSEF	**0.762**	**0.796**	**0.700**	**0.695**	**0.590**	**0.563**
	GSE	0.517	0.552	0.490	0.508	0.386	0.451
	ExplainSim	0.537	0.541	0.484	0.508	0.551	0.543

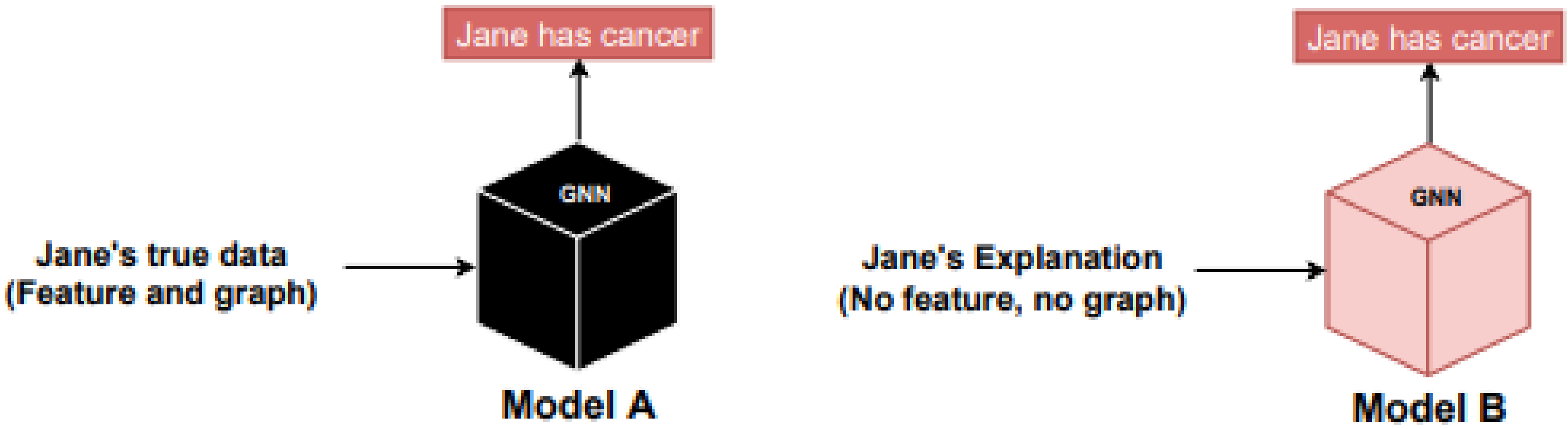
Note:

- **ExplainSim** and **GSEF** attacks for all explanation methods other than **GLIME** and **GNNExp,** outperform all baseline methods.
- Among **baseline** approaches, **GraphMI** performs best followed by **FeatureSim.**
- The information leakage for **BITCOIN** is limited by **small feature size**.
- For **GLIME** and **GNNExp**, we observe that the explanation contains little information about the graph structure. The reason behind this is further revealed in the **fidelity-sparsity** analysis of the obtained explanations.

References

1. Private Graph extraction via feature extraction (Link)
2. Code
3. YouTube video (Link)

ATTACKER’S ADVANTAGE



DEFENSE

$$Pr(\mathcal{E}'_{x_i} = 1) = \begin{cases} \frac{e^\epsilon}{e^\epsilon+1}, & \text{if } \mathcal{E}_{x_i} = 1, \\ \frac{1}{e^\epsilon+1}, & \text{if } \mathcal{E}_{x_i} = 0, \end{cases}$$

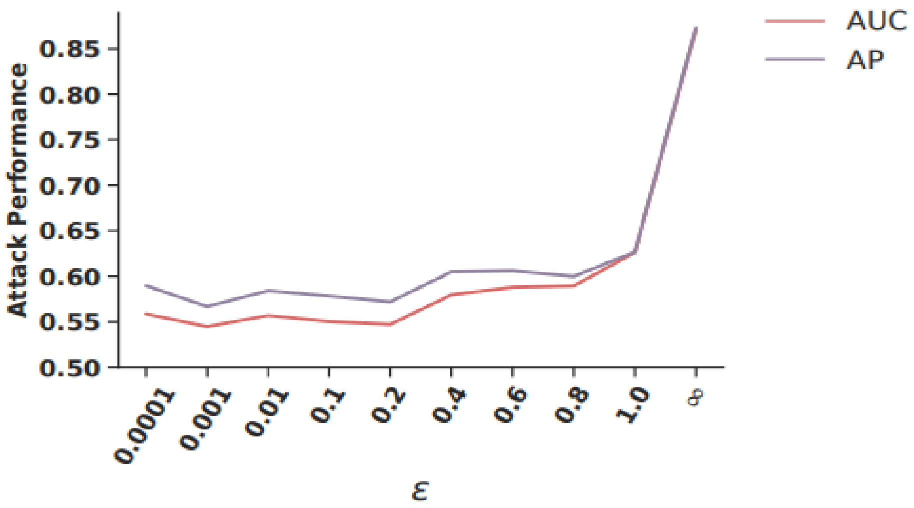


Figure 9: Privacy budget and corresponding attack performance of ExplainSim for Zorro explanation on the Cora dataset. ∞ implies that no perturbation is performed.

ϵ	Fidelity	Sparsity	Intersection
0.0001	0.84	5.91	74.68
0.001	0.84	5.91	74.70
0.01	0.84	5.89	75.03
0.1	0.84	5.80	75.10
0.2	0.83	5.71	75.60
0.4	0.82	5.49	76.45
0.6	0.81	5.25	77.16
0.8	0.81	5.00	78.66
1	0.81	4.73	80.10
∞	0.89	1.83	100

SUMMARY