Interpretable and Generalizable Graph Learning via Stochastic Attention Mechanism

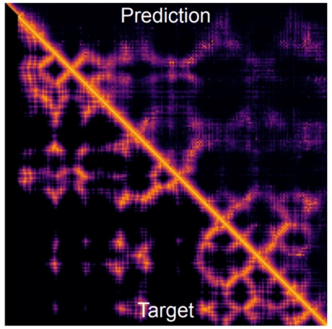
Siqi Miao, Miaoyuan Liu, Pan Li

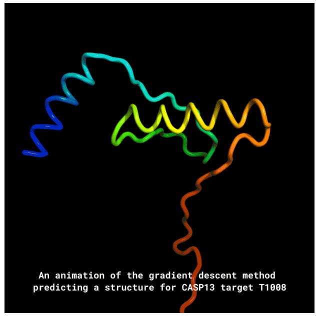
Purdue University

Deep Learning on Graphs in Science

- Protein folding

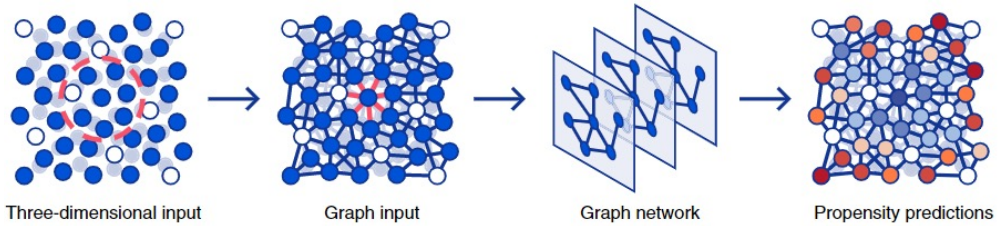
[Senior et al., Nature 2019]
[Jumper et al., Nature 2021]



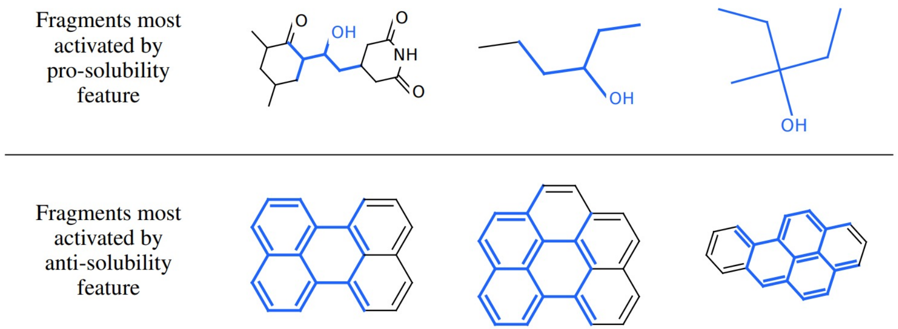


- Simulation of glass dynamics

[Baspt et al, Nature Physics 2021]

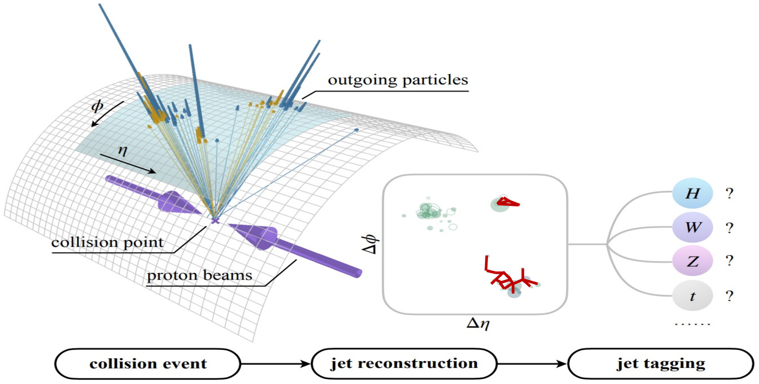


- Molecular Property Prediction



[Duvenaud et al., NeurIPS 2015]

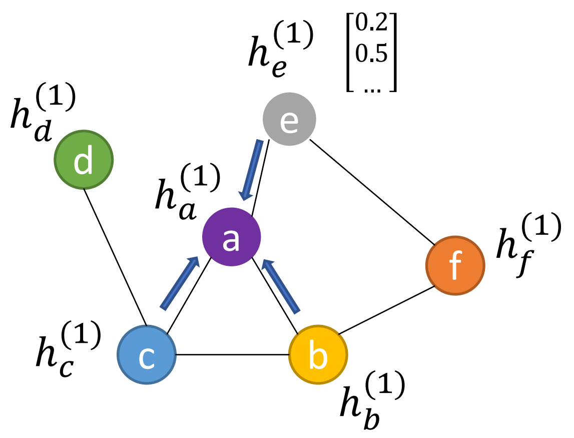
- Jet Tagging in HEP

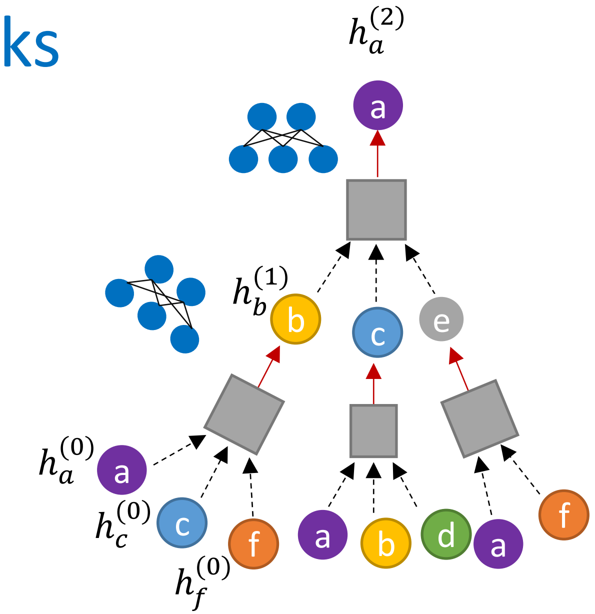


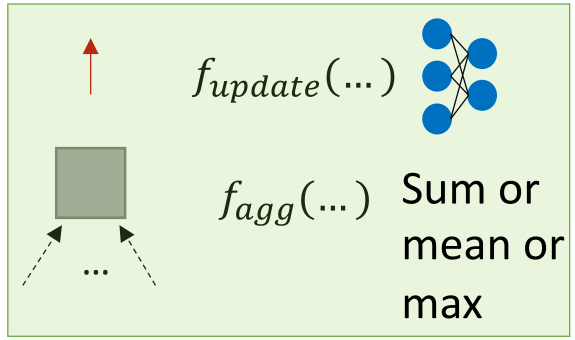
Refined based on [Qu, Li, Qian, 2022]

Can We Trust the GNN models?

- Graph Neural Networks







Graph neural network: one layer

  - ➢ Lack of the model transparency
  - ➢ Unable to tell the effective data patterns
  - ➢ Sensitive to the data distribution shifts

- Many scientific applications need to collect data insights beyond just to achieve high prediction performance.

Recent Efforts on Interpretable GNNs

- Previous works on interpreting GNNs
  - GNNExplainer [Ying et al., 2019]
  - PGExplainer [Luo et al., 2020]
  - PGM-Explainer [Vu et al., 2020]
  - GraphLIME [Huang et al., 2020]
  - SubgraphX [Yuan et al., 2021]
  - GraphMask [Schlichtkrull et al., 2021]
  - ……

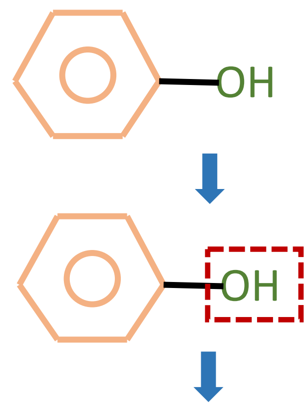
to check what data patterns GNNs capture

- Almost all of them adopt post-hoc approaches…

Step 1. Given a trained GNN predictor f_θ

Step 2. Fix f_θ and train an explainer g_ϕ

Feature selection



G

$G_S = g_\phi(G)$

Soluble or not? $f_\theta(G_S) = f_\theta \circ g_\phi(G)$

Issues of Post-hoc Methods

Our claim: Post-hoc methods can hardly provide <u>trustworthy interpretation</u> for GNN models.

- Post-hoc methods are essentially good at checking sensitivity
- They suffer from
  1. Data distribution shifts
  2. Spuriously correlated patterns

G_S Selected Features

Y The Prediction Task

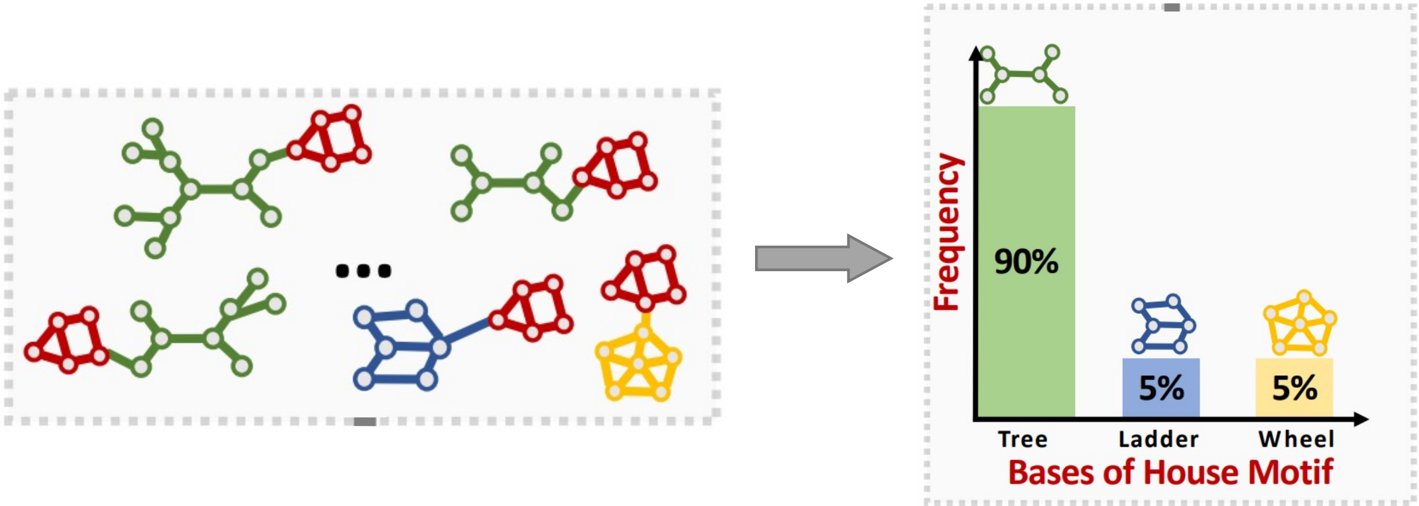
f_θ may not tell if G_S is effective for predicting Y

$\rightarrow$ f_θ is never trained on any subgraph G_S!

Issues of Post-hoc Methods

Our claim: Post-hoc methods can hardly provide trustworthy interpretation for GNN models.

- Post-hoc methods are essentially good at checking sensitivity
- They suffer from
  1. Data distribution shifts
  2. Spuriously correlated patterns



Examples of spurious correlations, DIR [Wu et al., 2022]

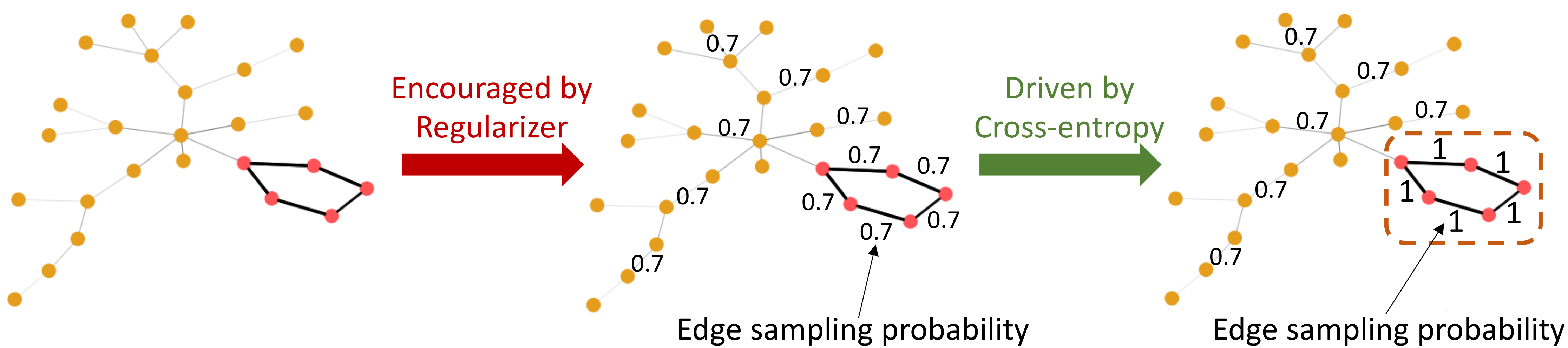
Inherently Interpretable Models

- Our Goal: An inherently interpretable model
- Jointly train both the predictor f_θ and the extractor g_ϕ
  - Input:
    - The original graphs
  - Output:
    - Predictions for the application task
    - Effective data patterns
- Use attention but not vanilla attention!

Graph Stochastic Attention (GSAT)

Graph Stochastic Attention (GSAT)

- Rationale: Inject stochasticity when learning attention
  - A regularizer is used to encourage high randomness
    - High dropping prob.
  - Driven by the classification loss, critical edges should learn to be with low randomness
    - Low dropping prob.
  - The part of G_S with less randomness is indicative to the prediction task Y



Graph Stochastic Attention (GSAT)

- Rationale: Inject stochasticity when learning attention
  - A regularizer is used to encourage high randomness
    - High dropping prob.
  - Driven by the classification loss, critical edges should learn to be with low randomness
    - Low dropping prob.
  - The part of G_S with less randomness is indicative to the prediction task Y
- How to control randomness?
  - Information regularizer to control randomness!
    - i.e., the Information Bottleneck (IB) principle

$$\rightarrow \min_{\theta,\phi} \; -I(f_\theta(G_S), Y) + \underbrace{\beta I(G_S; G)}_{\text{Information regularization}}, \text{ s.t. } \underbrace{G_S \sim g_\phi(G)}_{KL(attention|Q)}$$

Graph Information bottleneck [Wu et al. 2020, Yang et al. 2021]

Graph Stochastic Attention (GSAT)

- Architecture

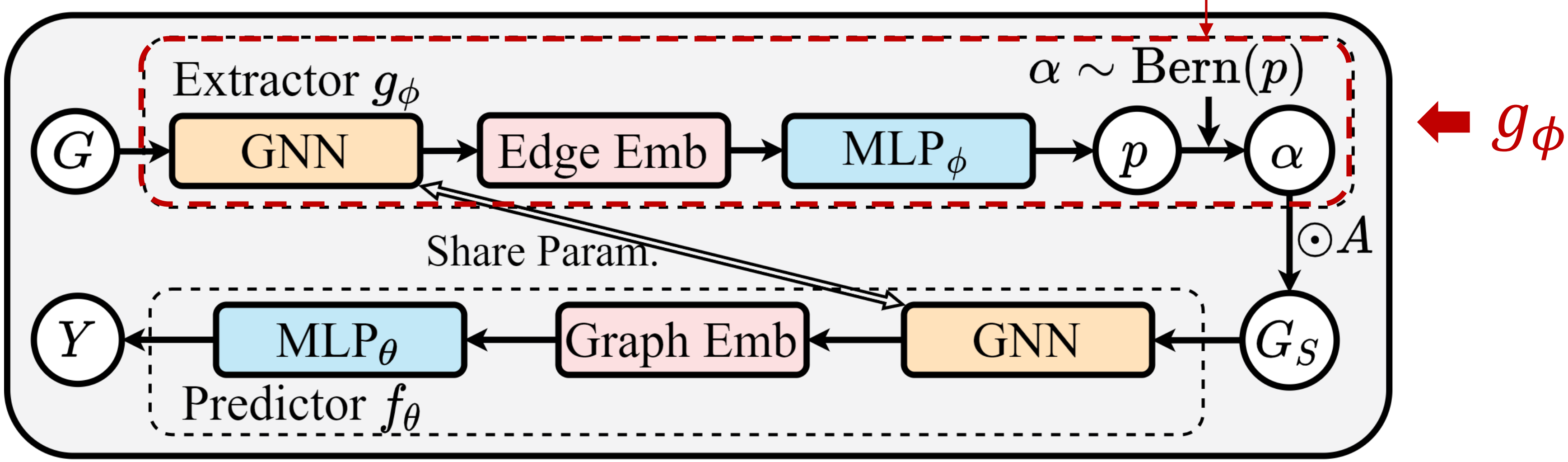
1. Inject stochasticity when learning attention

→ Generate a random graph $G_S \sim g_\phi(G)$

2. The predictor $f_\theta(G_S)$ makes predictions based on G_S

→ To $\min_{\theta,\phi}\ -I(f_\theta(G_S), Y) + \beta I(G_S; G)$

Subgraph Sampling



Graph Stochastic Attention (GSAT)

- Architecture

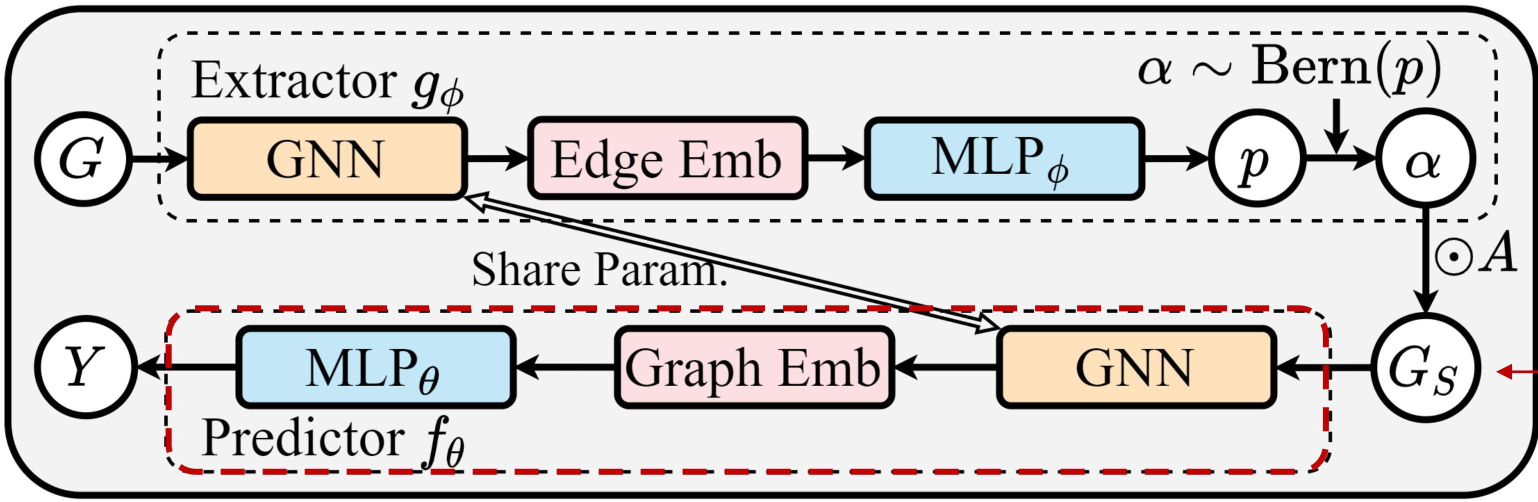
1. Inject stochasticity when learning attention

$\rightarrow$ Generate a random graph $G_S \sim g_\phi(G)$

2. The predictor $f_\theta(G_S)$ makes predictions based on G_S

$\rightarrow$ To $\min_{\theta,\phi} \; -I(f_\theta(G_S), Y) + \beta I(G_S; G)$

f_θ



Sampled Subgraph

Guaranteed Spurious Correlation Removal

- Our IB Objective Provides
  - Guaranteed spurious correlation removal
  - Guaranteed interpretability

Theorem 4.1. Suppose each G contains a subgraph G_S^* such that Y is determined by G_S^* in the sense that $Y = f(G_S^*) + \epsilon$ for some deterministic invertible function f with randomness ϵ that is independent from G. Then, for any $\beta \in [0, 1]$, $G_S = G_S^*$ maximizes the GIB $I(G_S; Y) - \beta I(G_S; G)$, where $G_S \in \mathbb{G}_{\text{sub}}(G)$.

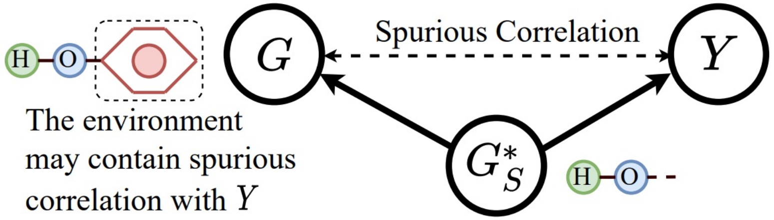


Figure 6. G_S^* determines Y. However, the environment features in $G \backslash G_S^*$ may contain spurious (backdoor) correlation with Y.

Experiments

- Experiments on Interpretability

Table 1. Interpretation Performance (AUC). The <u>underlined</u> results highlight the best baselines. The **bold** font and **bold**† font highlight when GSAT outperform the means of the best baselines based on the mean of GSAT and the mean-2*std of GSAT, respectively.

	BA-2MOTIFS	MUTAG	MNIST-75SP	SPURIOUS-MOTIF $b=0.5$	SPURIOUS-MOTIF $b=0.7$	SPURIOUS-MOTIF $b=0.9$
GNNEXPLAINER	67.35 ± 3.29	61.98 ± 5.45	59.01 ± 2.04	62.62 ± 1.35	62.25 ± 3.61	58.86 ± 1.93
PGEXPLAINER	84.59 ± 9.09	60.91 ± 17.10	69.34 ± 4.32	69.54 ± 5.64	72.33 ± 9.18	$\underline{72.34} \pm 2.91$
GRAPHMASK	$\underline{92.54} \pm 8.07$	62.23 ± 9.01	$\underline{73.10} \pm 6.41$	72.06 ± 5.58	73.06 ± 4.91	66.68 ± 6.96
IB-SUBGRAPH	86.06 ± 28.37	$\underline{91.04} \pm 6.59$	51.20 ± 5.12	57.29 ± 14.35	62.89 ± 15.59	47.29 ± 13.39
DIR	82.78 ± 10.97	64.44 ± 28.81	32.35 ± 9.39	$\underline{78.15} \pm 1.32$	$\underline{77.68} \pm 1.22$	49.08 ± 3.66
GIN+GSAT	$\mathbf{98.74}^{\dagger} \pm 0.55$	$\mathbf{99.60}^{\dagger} \pm 0.51$	$\mathbf{83.36}^{\dagger} \pm 1.02$	$\mathbf{78.45} \pm 3.12$	74.07 ± 5.28	71.97 ± 4.41
GIN+GSAT*	$\mathbf{97.43}^{\dagger} \pm 1.77$	$\mathbf{97.75}^{\dagger} \pm 0.92$	$\mathbf{83.70}^{\dagger} \pm 1.46$	$\mathbf{85.55}^{\dagger} \pm 2.57$	$\mathbf{85.56}^{\dagger} \pm 1.93$	$\mathbf{83.59}^{\dagger} \pm 2.56$
PNA+GSAT	$\mathbf{93.77} \pm 3.90$	$\mathbf{99.07}^{\dagger} \pm 0.50$	$\mathbf{84.68}^{\dagger} \pm 1.06$	$\mathbf{83.34}^{\dagger} \pm 2.17$	$\mathbf{86.94}^{\dagger} \pm 4.05$	$\mathbf{88.66}^{\dagger} \pm 2.44$
PNA+GSAT*	89.04 ± 4.92	$\mathbf{96.22}^{\dagger} \pm 2.08$	$\mathbf{88.54}^{\dagger} \pm 0.72$	$\mathbf{90.55}^{\dagger} \pm 1.48$	$\mathbf{89.79}^{\dagger} \pm 1.91$	$\mathbf{89.54}^{\dagger} \pm 1.78$

*: Apply GSAT to a pretrained GNN and do further co-training.

Improve up to 20%, and 12% on average in interpretation performance

Experiments

- Experiments on Generalizability

Table 2. Prediction Performance (Acc.). The **bold** font highlights the inherently interpretable methods that significantly outperform the corresponding backbone model, GIN or PNA, when the mean-1*std of a method > the mean of its corresponding backbone model.

	MolHiv (AUC)	Graph-SST2	MNIST-75sp	Spurious-Motif $b = 0.5$	Spurious-Motif $b = 0.7$	Spurious-Motif $b = 0.9$
GIN	76.69 ± 1.25	82.73 ± 0.77	95.74 ± 0.36	39.87 ± 1.30	39.04 ± 1.62	38.57 ± 2.31
IB-subgraph	76.43 ± 2.65	82.99 ± 0.67	93.10 ± 1.32	$\mathbf{54.36} \pm 7.09$	$\mathbf{48.51} \pm 5.76$	$\mathbf{46.19} \pm 5.63$
DIR	76.34 ± 1.01	82.32 ± 0.85	88.51 ± 2.57	$\mathbf{45.49} \pm 3.81$	41.13 ± 2.62	37.61 ± 2.02
GIN+GSAT	76.47 ± 1.53	82.95 ± 0.58	$\mathbf{96.24} \pm 0.17$	$\mathbf{52.74} \pm 4.08$	$\mathbf{49.12} \pm 3.29$	$\mathbf{44.22} \pm 5.57$
GIN+GSAT*	76.16 ± 1.39	82.57 ± 0.71	$\mathbf{96.21} \pm 0.14$	$\mathbf{46.62} \pm 2.95$	41.26 ± 3.01	39.74 ± 2.20
PNA (no scalars)	78.91 ± 1.04	79.87 ± 1.02	87.20 ± 5.61	68.15 ± 2.39	66.35 ± 3.34	61.40 ± 3.56
PNA+GSAT	$\mathbf{80.24} \pm 0.73$	$\mathbf{80.92} \pm 0.66$	$\mathbf{93.96} \pm 0.92$	68.74 ± 2.24	64.38 ± 3.20	57.01 ± 2.95
PNA+GSAT*	$\mathbf{80.67} \pm 0.95$	$\mathbf{82.81} \pm 0.56$	$\mathbf{92.38} \pm 1.44$	$\mathbf{69.72} \pm 1.93$	$\mathbf{67.31} \pm 1.86$	61.49 ± 3.46

	molbace	molbbbp	molclintox	moltox21	molsider
PNA	73.52 ± 3.02	67.21 ± 1.34	86.72 ± 2.33	75.08 ± 0.64	56.51 ± 1.90
GSAT	$\mathbf{77.41} \pm 2.42$	$\mathbf{69.17} \pm 1.12$	$\mathbf{87.80} \pm 2.36$	74.96 ± 0.66	$\mathbf{57.58} \pm 1.23$
GSAT*	73.61 ± 1.59	66.30 ± 0.79	$\mathbf{89.26} \pm 1.66$	$\mathbf{75.71} \pm 0.48$	$\mathbf{59.19} \pm 1.03$

Improve 3% on average in prediction accuracy

Experiments

- Comparisons on Spurious Correlation Removal

Table 4. Direct comparison (Acc.) with invariant learning methods on the ability to remove spurious correlations, by applying the backbone model used in (Wu et al., 2022).

SPURIOUS-MOTIF	$b = 0.5$	$b = 0.7$	$b = 0.9$
ERM	39.69 ± 1.73	38.93 ± 1.74	33.61 ± 1.02
V-REx	39.43 ± 2.69	39.08 ± 1.56	34.81 ± 2.04
IRM	41.30 ± 1.28	40.16 ± 1.74	35.12 ± 2.71
DIR	45.50 ± 2.15	43.36 ± 1.64	39.87 ± 0.56
GSAT	$\mathbf{53.27}^{\dagger} \pm 5.12$	$\mathbf{56.50}^{\dagger} \pm 3.96$	$\mathbf{53.11}^{\dagger} \pm 4.64$
GSAT*	43.27 ± 4.58	42.51 ± 5.32	$\mathbf{45.76}^{\dagger} \pm 5.32$

Improve 12% on average in spurious correlation removal

Conclusion

- We propose a novel attention mechanism GSAT
  - ✓ Better interpretation performance
  - ✓ Better generalization capability
  - ✓ Better spurious correlation removal
- Code is available at: https://github.com/Graph-COM/GSAT
  - ✓ Feel free to try it out in Colab: Open in Colab