

# Interpretable and Generalizable Graph Learning via Stochastic Attention Mechanism

Siqi Miao, Miaoyuan Liu, Pan Li

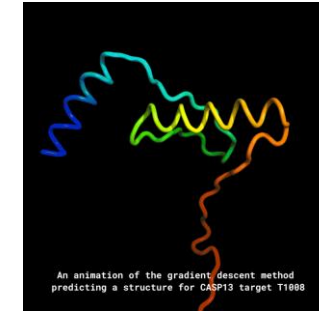
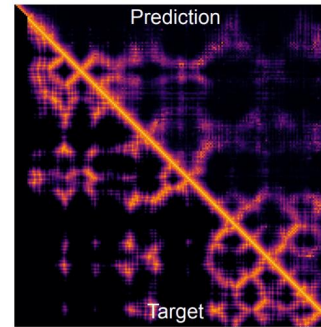
Purdue University



# Deep Learning on Graphs in Science

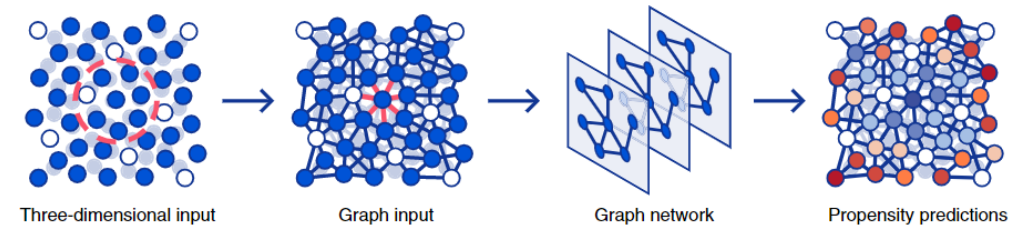
- Protein folding

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



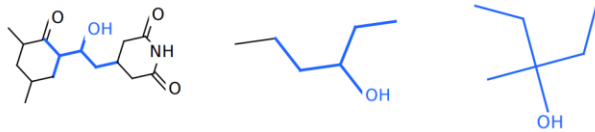
- Simulation of glass dynamics

[Baspt et al, Nature Physics 2021]

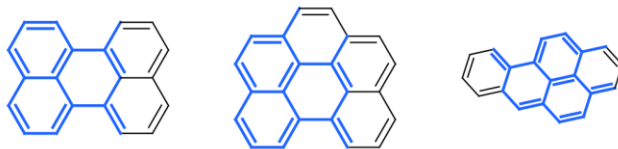


- Molecular Property Prediction

Fragments most activated by pro-solubility feature

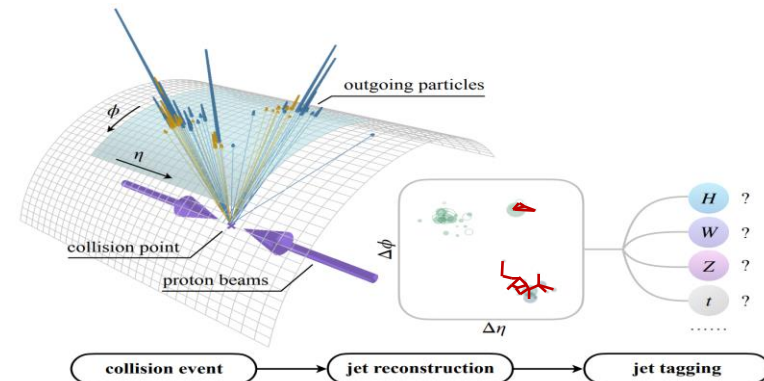


Fragments most activated by anti-solubility feature



[Duvenaud et al., NeurIPS 2015]

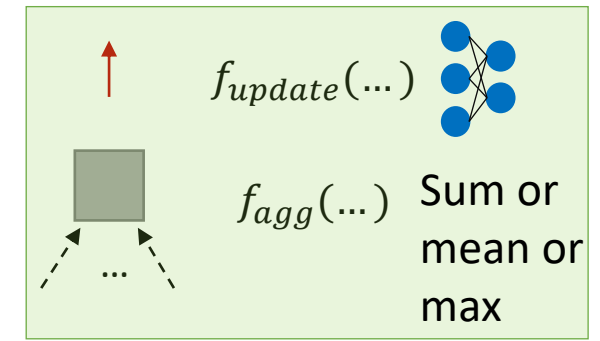
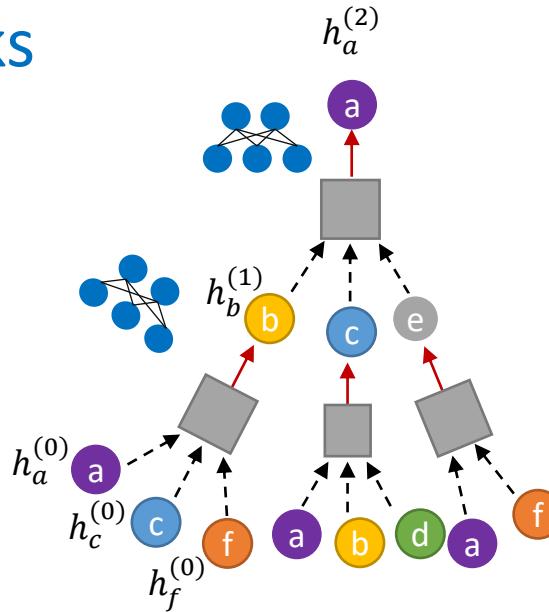
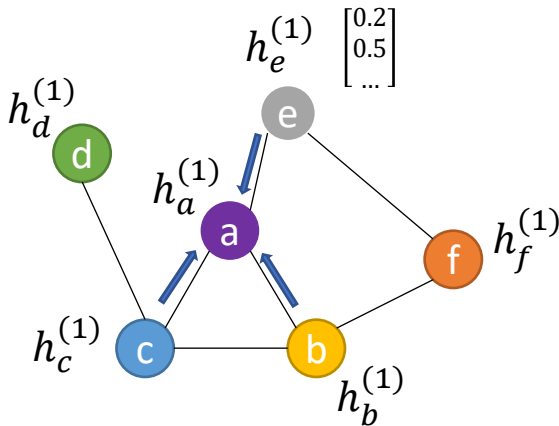
- Jet Tagging in HEP



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

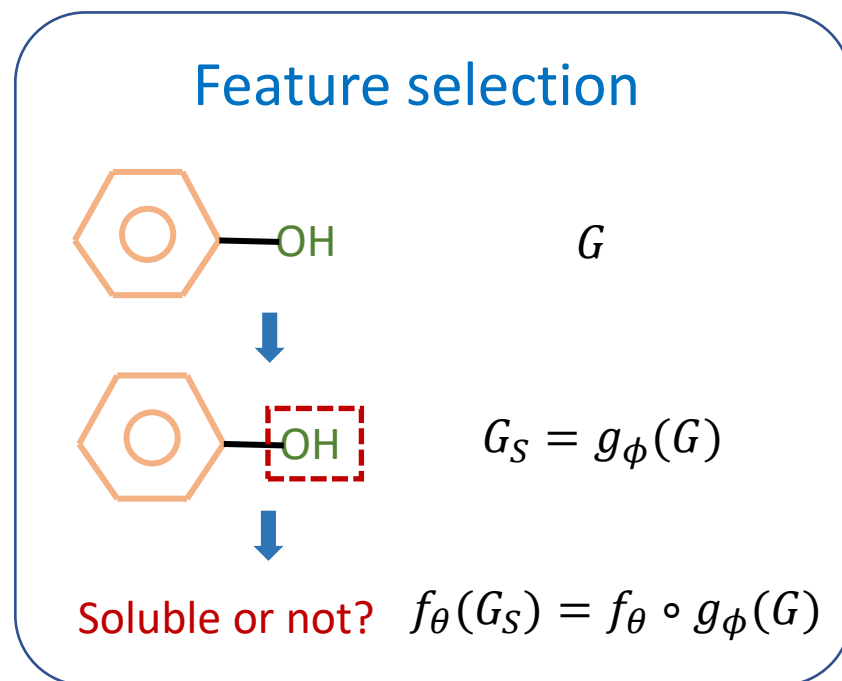
- Previous works on interpreting GNNs

- 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]
- .....

- Almost all previous works adopt post-hoc approaches...

Step 1. Given a trained GNN predictor  $f_\theta$

Step 2. Fix  $f_\theta$  and train an explainer  $g_\phi$



To check what data patterns GNNs capture

# Recent Efforts on Interpretable GNNs

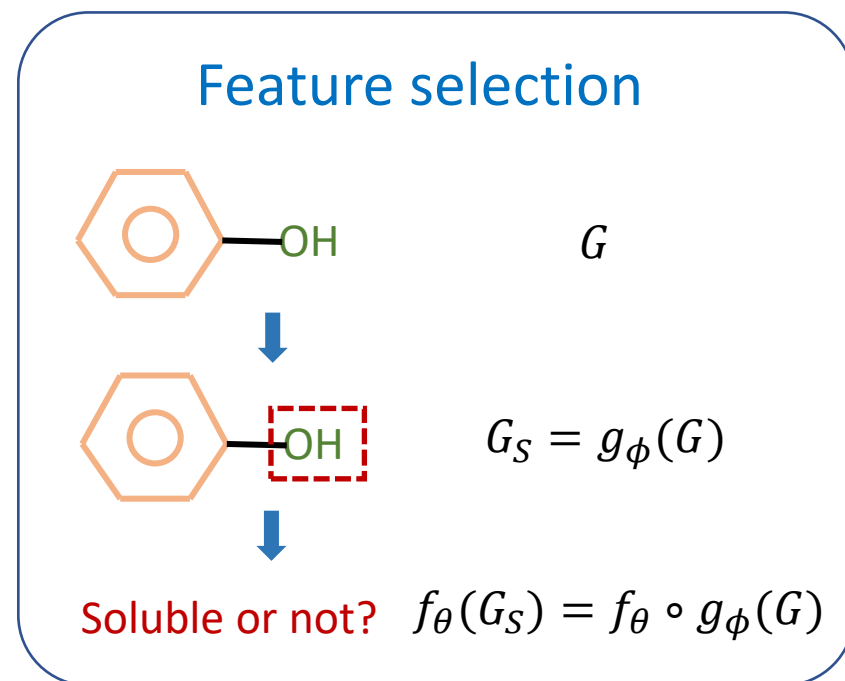
- Previous works on interpreting GNNs

- 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]
- .....

- Almost all previous works adopt post-hoc approaches...

Step 1. Given a trained GNN predictor  $f_\theta$

Step 2. Fix  $f_\theta$  and train an explainer  $g_\phi$



To check what data patterns GNNs capture

# Issues of Post-hoc Methods

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

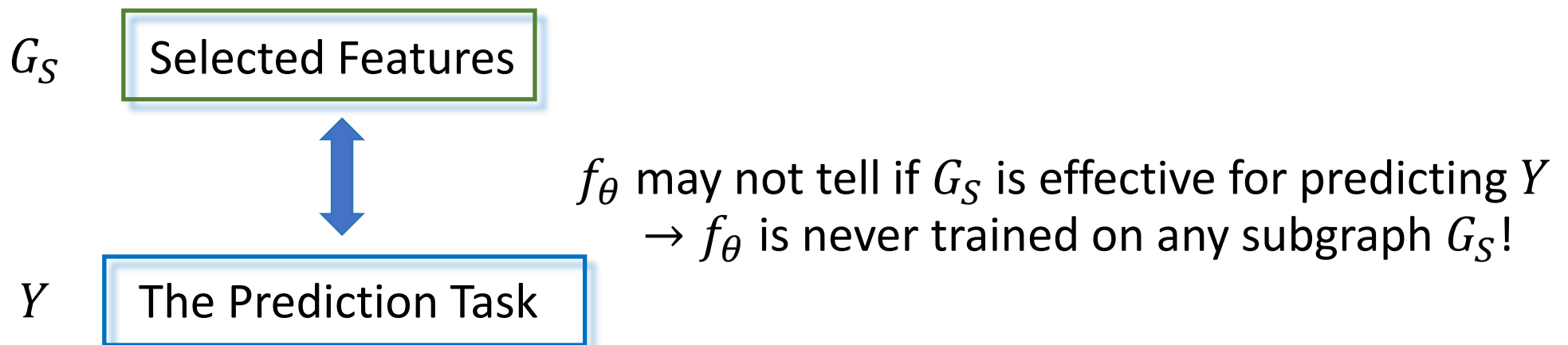
- Post-hoc methods are essentially good at checking **sensitivity**
- They suffer from
  1. Data distribution shifts
  2. Spuriously correlated patterns



# Issues of Post-hoc Methods

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

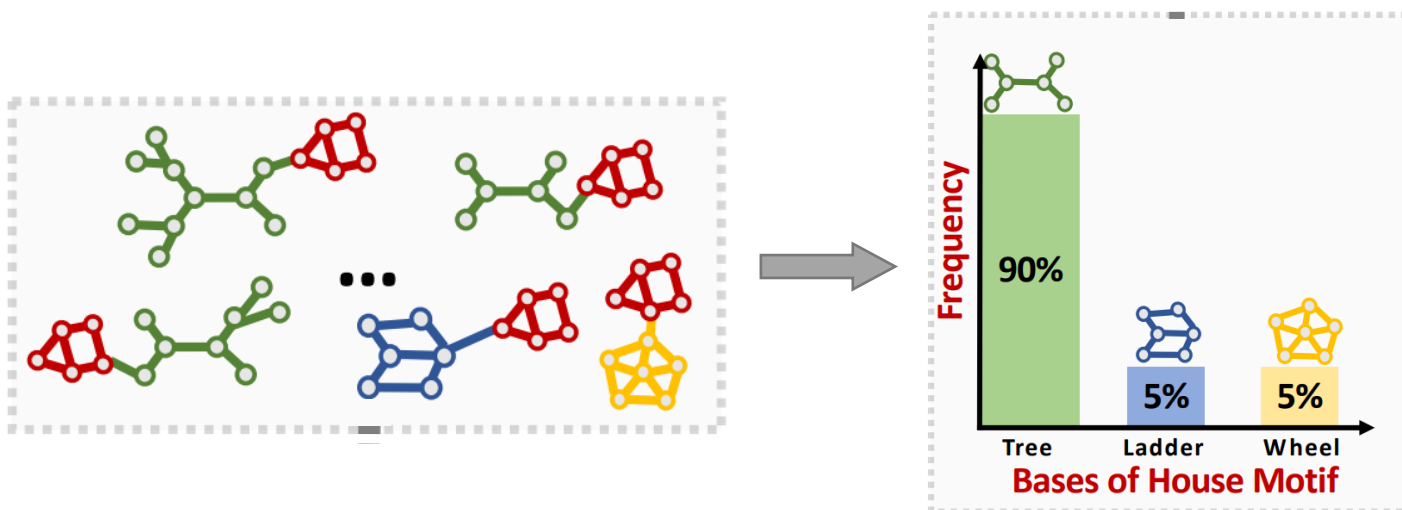
- Post-hoc methods are essentially good at checking **sensitivity**
- They suffer from
  1. Data distribution shifts
  2. Spuriously correlated patterns



# 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]



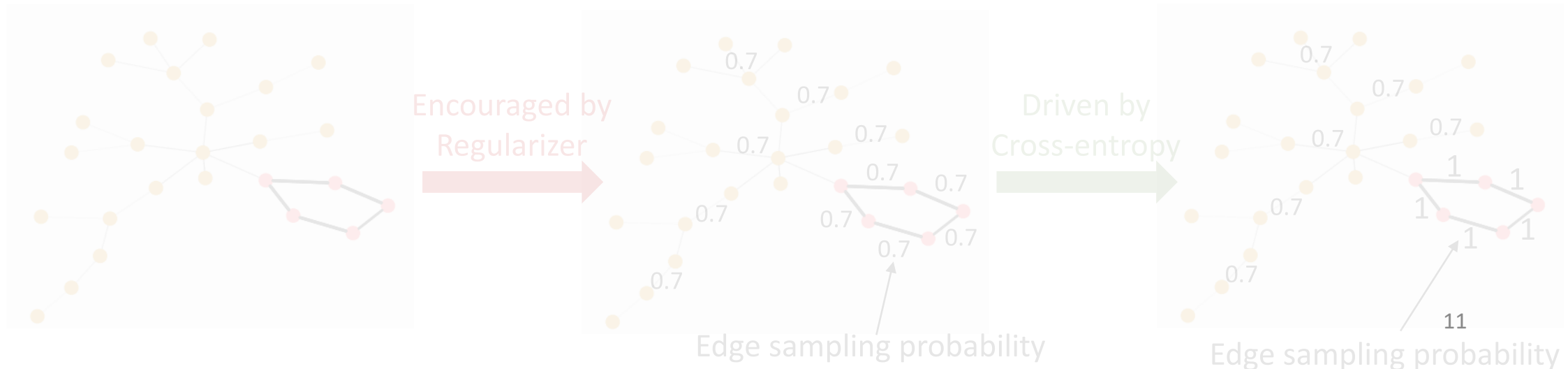
# Inherently Interpretable Models

- Our Goal: An inherently interpretable model
- Jointly train both the predictor  $f_\theta$  and the extractor  $g_\phi$ 
  - Input:
    - The original graphs
  - Output:
    - Predictions for the application task
    - Effective data patterns
- Use **attention** but not vanilla attention!

## Graph Stochastic Attention (GSAT)

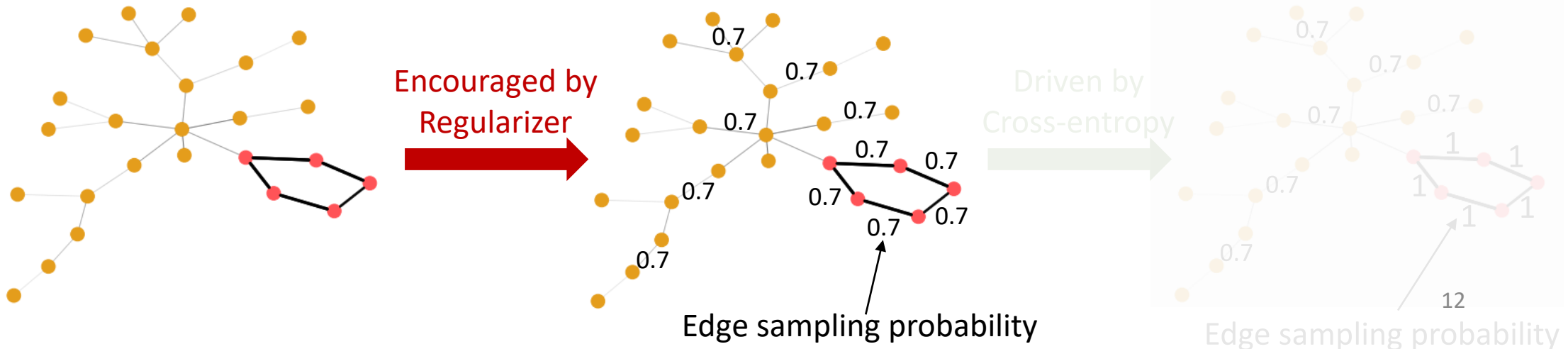
# Graph Stochastic Attention (GSAT)

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



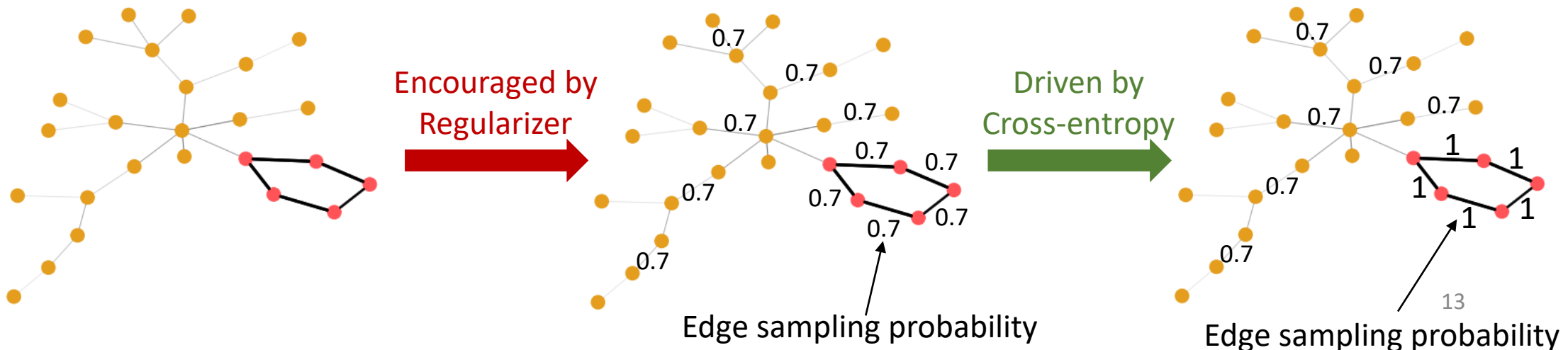
# Graph Stochastic Attention (GSAT)

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



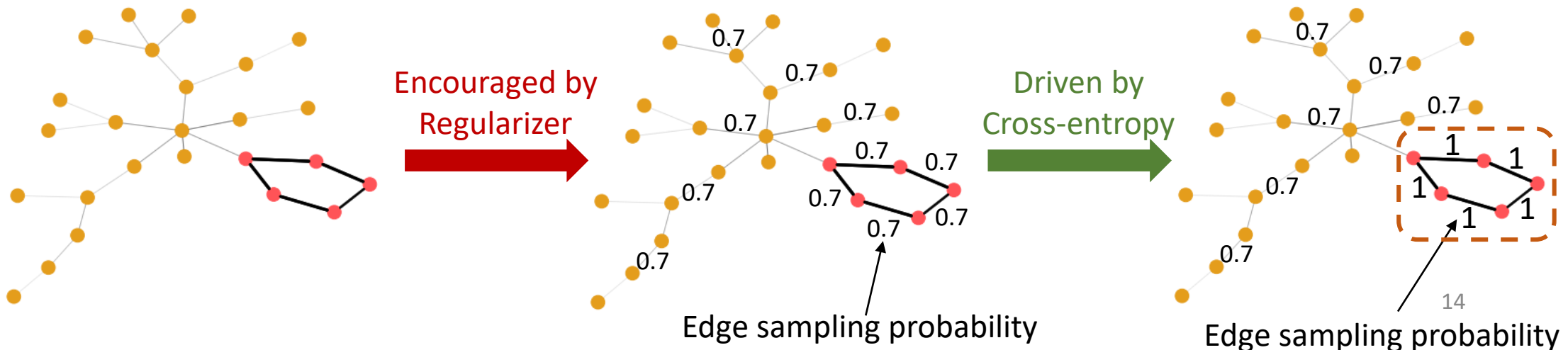
# Graph Stochastic Attention (GSAT)

- Rationale: Inject stochasticity when learning attention
  - A **regularizer** is used to encourage high randomness
    - Low sampling prob.
  - Driven by the **classification loss**, critical edges should learn to be with low randomness
    - High sampling 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
    - Low sampling prob.
  - Driven by the **classification loss**, critical edges should learn to be with low randomness
    - High sampling 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
  - Low sampling prob.
- Driven by the classification loss, critical edges should learn to be with low randomness
  - High sampling 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

$$\rightarrow \min_{\theta, \phi} -I(f_{\theta}(G_S), Y) + \beta I(G_S; G), \text{ s.t. } G_S \sim g_{\phi}(G)$$

Information regularization       $KL(\text{attention}|Q)$

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

# Graph Stochastic Attention (GSAT)

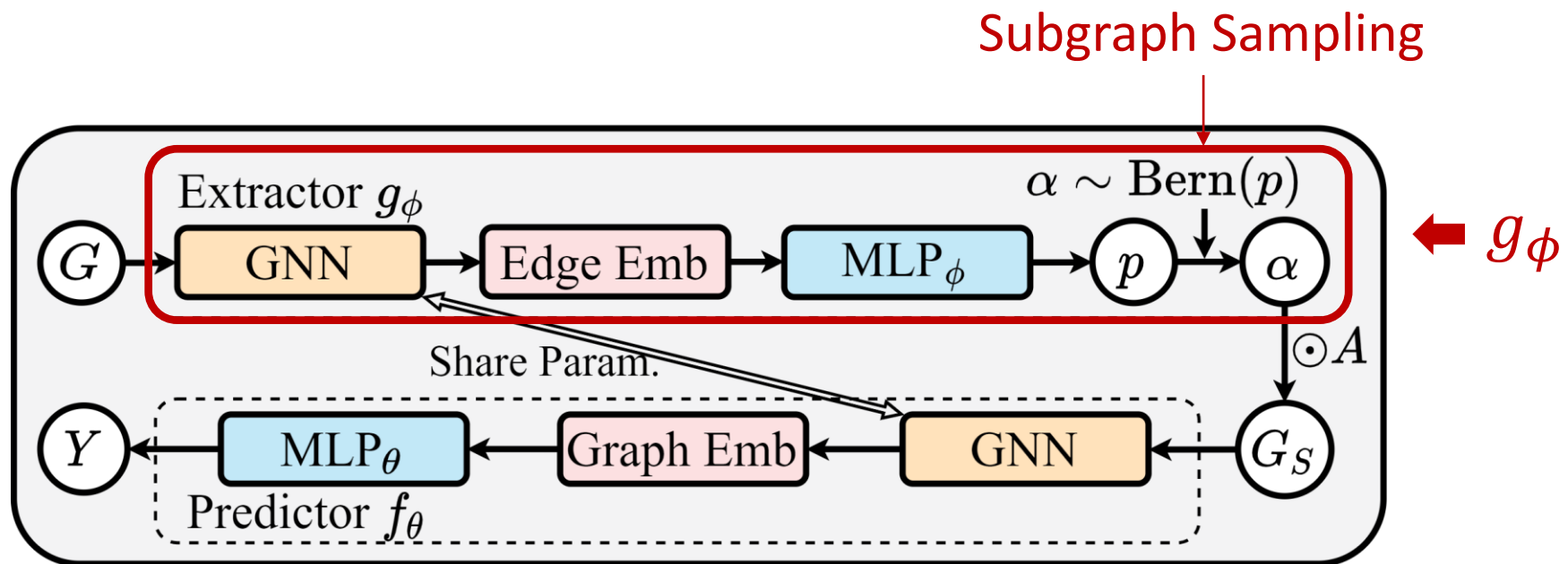
- Architecture

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



# Graph Stochastic Attention (GSAT)

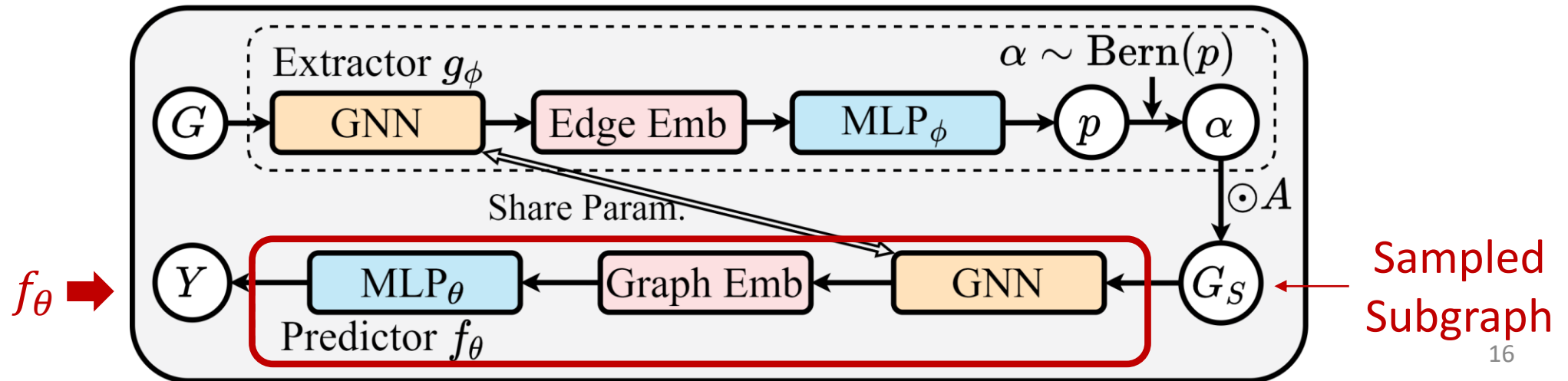
- Architecture

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





# 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  $\epsilon$  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  $G_S \in \mathbb{G}_{\text{sub}}(G)$ .

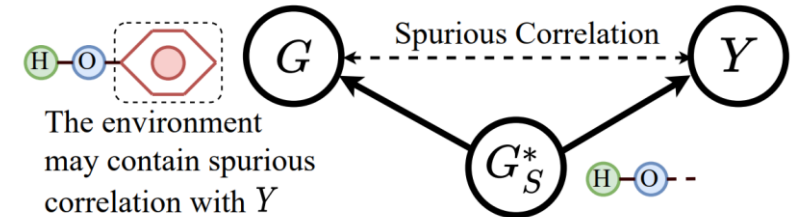


Figure 6.  $G_S^*$  determines  $Y$ . However, the environment features in  $G \setminus G_S^*$  may contain spurious (backdoor) correlation with  $Y$ .

# Experiments

## • Experiments on Interpretability

Table 1. Interpretation Performance (AUC). The underlined results highlight the best baselines. The **bold** font and **bold**<sup>†</sup> 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	$b = 0.5$	SPURIOUS-MOTIF $b = 0.7$	$b = 0.9$
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	<u>72.34</u> ± 2.91
GRAPHMASK	<u>92.54</u> ± 8.07	62.23 ± 9.01	<u>73.10</u> ± 6.41	72.06 ± 5.58	73.06 ± 4.91	66.68 ± 6.96
IB-SUBGRAPH	86.06 ± 28.37	<u>91.04</u> ± 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	<u>78.15</u> ± 1.32	<u>77.68</u> ± 1.22	49.08 ± 3.66
GIN+GSAT	<b>98.74</b> <sup>†</sup> ± 0.55	<b>99.60</b> <sup>†</sup> ± 0.51	<b>83.36</b> <sup>†</sup> ± 1.02	<b>78.45</b> ± 3.12	74.07 ± 5.28	71.97 ± 4.41
GIN+GSAT*	<b>97.43</b> <sup>†</sup> ± 1.77	<b>97.75</b> <sup>†</sup> ± 0.92	<b>83.70</b> <sup>†</sup> ± 1.46	<b>85.55</b> <sup>†</sup> ± 2.57	<b>85.56</b> <sup>†</sup> ± 1.93	<b>83.59</b> <sup>†</sup> ± 2.56
PNA+GSAT	<b>93.77</b> ± 3.90	<b>99.07</b> <sup>†</sup> ± 0.50	<b>84.68</b> <sup>†</sup> ± 1.06	<b>83.34</b> <sup>†</sup> ± 2.17	<b>86.94</b> <sup>†</sup> ± 4.05	<b>88.66</b> <sup>†</sup> ± 2.44
PNA+GSAT*	89.04 ± 4.92	<b>96.22</b> <sup>†</sup> ± 2.08	<b>88.54</b> <sup>†</sup> ± 0.72	<b>90.55</b> <sup>†</sup> ± 1.48	<b>89.79</b> <sup>†</sup> ± 1.91	<b>89.54</b> <sup>†</sup> ± 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	$b = 0.5$	SPURIOUS-MOTIF $b = 0.7$	$b = 0.9$
GIN	76.69 $\pm$ 1.25	82.73 $\pm$ 0.77	95.74 $\pm$ 0.36	39.87 $\pm$ 1.30	39.04 $\pm$ 1.62	38.57 $\pm$ 2.31
IB-SUBGRAPH	76.43 $\pm$ 2.65	82.99 $\pm$ 0.67	93.10 $\pm$ 1.32	<b>54.36</b> $\pm$ 7.09	<b>48.51</b> $\pm$ 5.76	<b>46.19</b> $\pm$ 5.63
DIR	76.34 $\pm$ 1.01	82.32 $\pm$ 0.85	88.51 $\pm$ 2.57	<b>45.49</b> $\pm$ 3.81	41.13 $\pm$ 2.62	37.61 $\pm$ 2.02
GIN+GSAT	76.47 $\pm$ 1.53	82.95 $\pm$ 0.58	<b>96.24</b> $\pm$ 0.17	<b>52.74</b> $\pm$ 4.08	<b>49.12</b> $\pm$ 3.29	<b>44.22</b> $\pm$ 5.57
GIN+GSAT*	76.16 $\pm$ 1.39	82.57 $\pm$ 0.71	<b>96.21</b> $\pm$ 0.14	<b>46.62</b> $\pm$ 2.95	41.26 $\pm$ 3.01	39.74 $\pm$ 2.20
PNA (NO SCALARS)	78.91 $\pm$ 1.04	79.87 $\pm$ 1.02	87.20 $\pm$ 5.61	68.15 $\pm$ 2.39	66.35 $\pm$ 3.34	61.40 $\pm$ 3.56
PNA+GSAT	<b>80.24</b> $\pm$ 0.73	<b>80.92</b> $\pm$ 0.66	<b>93.96</b> $\pm$ 0.92	68.74 $\pm$ 2.24	64.38 $\pm$ 3.20	57.01 $\pm$ 2.95
PNA+GSAT*	<b>80.67</b> $\pm$ 0.95	<b>82.81</b> $\pm$ 0.56	<b>92.38</b> $\pm$ 1.44	<b>69.72</b> $\pm$ 1.93	<b>67.31</b> $\pm$ 1.86	61.49 $\pm$ 3.46

	MOLBACE	MOLBBBP	MOLCLINTOX	MOLTOX21	MOLSIDER
PNA	73.52 $\pm$ 3.02	67.21 $\pm$ 1.34	86.72 $\pm$ 2.33	75.08 $\pm$ 0.64	56.51 $\pm$ 1.90
GSAT	<b>77.41</b> $\pm$ 2.42	<b>69.17</b> $\pm$ 1.12	<b>87.80</b> $\pm$ 2.36	74.96 $\pm$ 0.66	<b>57.58</b> $\pm$ 1.23
GSAT*	73.61 $\pm$ 1.59	66.30 $\pm$ 0.79	<b>89.26</b> $\pm$ 1.66	<b>75.71</b> $\pm$ 0.48	<b>59.19</b> $\pm$ 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	$39.69 \pm 1.73$	$38.93 \pm 1.74$	$33.61 \pm 1.02$
V-REx	$39.43 \pm 2.69$	$39.08 \pm 1.56$	$34.81 \pm 2.04$
IRM	$41.30 \pm 1.28$	$40.16 \pm 1.74$	$35.12 \pm 2.71$
DIR	$45.50 \pm 2.15$	$43.36 \pm 1.64$	$39.87 \pm 0.56$
GSAT	<b><math>53.27^\dagger \pm 5.12</math></b>	<b><math>56.50^\dagger \pm 3.96</math></b>	<b><math>53.11^\dagger \pm 4.64</math></b>
GSAT*	$43.27 \pm 4.58$	$42.51 \pm 5.32$	<b><math>45.76^\dagger \pm 5.32</math></b>

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](#)