Interpretable and Generalizable Graph Learning via Stochastic Attention Mechanism

Siqi Miao [1] **Miaoyuan Liu** [2] **Pan Li** [1]

Abstract

Interpretable graph learning is in need as many scientific applications depend on learning models to collect insights from graph-structured data. Previous works mostly focused on using post-hoc approaches to interpret pre-trained models (graph neural networks in particular). They argue against inherently interpretable models because the good interpretability of these models is often at the cost of their prediction accuracy. However, those post-hoc methods often fail to provide stable interpretation and may extract features that are spuriously correlated with the task. In this work, we address these issues by proposing *Graph Stochastic Attention* (GSAT). Derived from the information bottleneck principle, GSAT injects stochasticity to the attention weights to block the information from task-irrelevant graph components while learning stochasticity-reduced attention to select task-relevant subgraphs for interpretation. The selected subgraphs provably do not contain patterns that are spuriously correlated with the task under some assumptions. Extensive experiments on eight datasets show that GSAT outperforms the state-of-the-art methods by up to 20%↑ in interpretation AUC and 5%↑ in prediction accuracy. Our code is available at `https://github.com/Graph-COM/GSAT`.

1. Introduction

Graph learning models are widely used in science, such as physics (Bapst et al., 2020) and biochemistry (Jumper et al., 2021). In many such disciplines, building more accurate predictive models is typically not the only goal. It is often more crucial for scientists to discover the patterns from the data that induce certain predictions (Cranmer et al., 2020). For example, identifying the functional groups in a molecule that yield its certain properties may provide insights to guide further experiments (Wencel-Delord & Glorius, 2013).

Recently, graph neural networks (GNNs) have become almost the de fato graph learning models due to their great expressive power (Kipf & Welling, 2017; Xu et al., 2019). However, their expressivity is often built upon a highly nonlinear entanglement of irregular graph features. So, it is often quite challenging to figure out the patterns in the data that GNNs use to make predictions.

Many works have been recently proposed to extract critical data patterns for the prediction by interpreting GNNs in post-hoc ways (Ying et al., 2019; Yuan et al., 2020a; Vu & Thai, 2020; Luo et al., 2020; Schlichtkrull et al., 2021; Yuan et al., 2021; Lin et al., 2021; Henderson et al., 2021). They work on a pre-trained model and propose different types of combinatorial search methods to detect the subgraphs of the input data that affect the model predictions the most.

In contrast to the above post-hoc methods, inherently interpretable models have been rarely investigated for graph learning tasks. There are two main concerns regarding such models. First, the prediction accuracy and inherent interpretability of a model often forms a trade-off (Du et al., 2019). Practitioners may not allow sacrificing prediction accuracy for better interpretability. Second, the attention mechanism, a widely-used technique to provide inherent interpretability, often cannot provide faithful interpretation (Lipton, 2018). The rationale of the attention mechanism is to learn weights for different features during the model training, and the rank of the learned weights can be interpreted as the importance of certain features (Bahdanau et al., 2015; Xu et al., 2015). However, recent extensive evaluations in NLP tasks (Serrano & Smith, 2019; Jain & Wallace, 2019; Mohankumar et al., 2020) have shown that the attention may not weigh the features that dominate the model output more than other features. In particular, for graph learning tasks, the widely-used graph attention models (Veličković et al., 2018; Li et al., 2016) seem unable to provide any reliable interpretation of the data (Ying et al., 2019; Yu et al., 2021).

Along another line of research, invariant learning (Pearl

[1]Department of Computer Science, Purdue University, West Lafayette, USA [2]Department of Physics and Astronomy, Purdue University, West Lafayette, USA. Correspondence to: Siqi Miao <miao61@purdue.edu>, Pan Li <panli@purdue.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

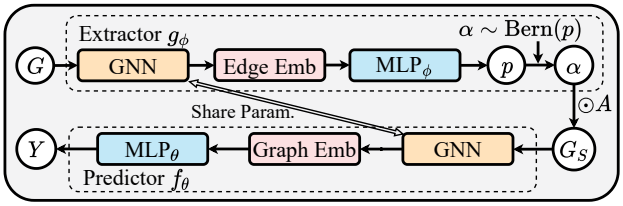


Figure 1. The architecture of GSAT. g_ϕ encodes the input graph G and learns stochastic attention α (from Bernoulli distributions) that randomly drop the edges and obtain a perturbed graph G_S. f_θ encodes G_S to make predictions. GSAT does not constrain the size of G_S but injects stochasticity to constrain information. The subgraph of G_S with learnt reduced-stochasticity (edges with $p_e \to 1$) provides interpretation. GSAT is a unified model by adopting just one GNN for both g_ϕ and f_θ. GSAT can be either trained from scratch or start from a pre-trained GNN predictor f_θ.

et al., 2016; Arjovsky et al., 2019; Chang et al., 2020; Krueger et al., 2021) has been proposed to provide inherent interpretability and better generalizability. They argue that the models naïvely trained over biased data may risk capturing spurious correlations between the input environment features and the labels, and thus suffer from severe generalization issues. So, they propose to train models that align with the causal relations between the signal features and the labels. However, such training approaches to match causal relations typically have high computational complexity.

In this work, we are to address the above concerns by proposing *Graph Stochastic Attention* (GSAT), a novel attention mechanism to build inherently interpretable and well generalizable GNNs. The rationale of GSAT roots in the notion of information bottleneck (IB) (Tishby et al., 2000; Tishby & Zaslavsky, 2015). We formulate the attention as an IB by injecting stochasticity into the attention to constrain the information flow from the input graph to the prediction (Shannon, 1948). Such stochasticity over the label-irrelevant graph components will be kept during the training while that over the label-relevant ones can automatically get reduced. This difference eventually provides model interpretation. By penalizing the amount of information from the input data, GSAT is also expected to be more generalizable.

Our study achieves the following observations and contributions. First, the IB principle frees GSAT from any potentially biased assumptions adopted in previous methods such as the size or the connectivity constraints on the detected graph patterns. Even when those assumptions are satisfied, GSAT still works the best without using such assumptions, while when those assumptions are not satisfied, GSAT achieves significantly better interpretation. See the sampled interpretation result visualizations in Fig. 2 and Fig. 3. Second, from the perspective of IB, all post-hoc interpretation methods are suboptimal. They essentially optimize a model without any information control and then perform a single-step projection to an information-controlled

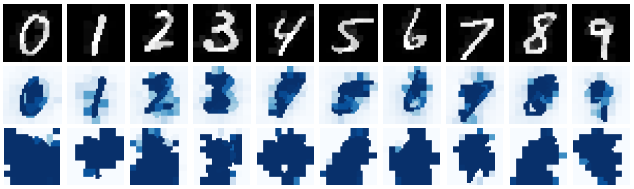


Figure 2. Visualizing attention (normalized to $[0, 1]$) of GSAT (second row) v.s. masks of GraphMask (Schlichtkrull et al., 2021) (third row) on MNIST-75sp. The first row shows the ground-truth. Different digit samples contain interpretable subgraphs of different sizes, while GSAT is not sensitive to such varied sizes.

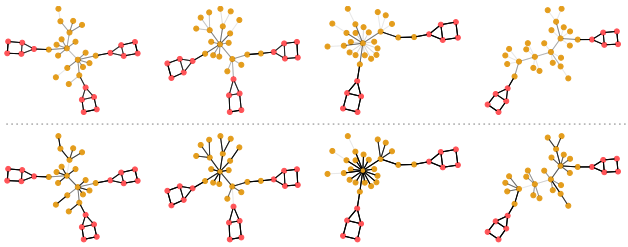


Figure 3. Visualizing attention (normalized to $[0, 1]$) of GSAT (first row) and masks of GraphMask (Schlichtkrull et al., 2021) (second row) on a motif example, where graphs with three house motifs and graphs with two house motifs represent two classes. Samples may contain disconnected interpretable subgraphs, while GSAT detects them accurately. More details can be found in Appendix C.4.

space, which makes the final interpretation performance sensitive to the pre-trained models. Third, by reducing the information from the input graph, GSAT can provably remove spurious correlations in the training data under certain assumptions and achieve better generalization. Fourth, if a pre-trained model is provided, GSAT may further improve both of its interpretation and prediction accuracy.

We evaluate GSAT in terms of both interpretability and label-prediction performance. Experiments over 8 datasets show that GSAT outperforms the state-of-the-art (SOTA) methods by up to $20\%\uparrow$ in interpretation AUC and $5\%\uparrow$ in prediction accuracy. Notably, GSAT achieves the SOTA performance on *molhiv* on OGB (Hu et al., 2020) among the models that do not use manually-designed expert features.

2. Preliminaries

As preliminaries, we define a few notations and concepts.

Graph. An attributed graph can be denoted as $G = (A, X)$ where A is the adjacency matrix and X includes node attributes. Let V and E denote the node set and the edge set, respectively. We focus on graph-level tasks: A training set of graphs with their labels $(G^{(i)}, Y^{(i)})$, $i = 1, ..., n$ are given, where each sample $(G^{(i)}, Y^{(i)})$ is assumed to be IID sampled from some unknown distribution $\mathbb{P}_{\mathcal{Y}\times\mathcal{G}} = \mathbb{P}_{\mathcal{Y}|\mathcal{G}}\mathbb{P}_{\mathcal{G}}$.

Label-relevant Subgraph. A label-relevant subgraph refers to the subgraph G_S of the input graph G that mostly indi-

cates the label Y. For example, to determine the solubility of a molecule, the hydroxy group -OH is a positive-label-relevent subgraph, as if it exists, the molecule is often soluble to the water. Finding label-relevant subgraphs is a common goal of interpretable graph learning.

Attention Mechanism. Attention mechanism has been widely used in interpretable neural networks for NLP and CV tasks (Bahdanau et al., 2015; Xu et al., 2015; Vaswani et al., 2017). However, GNNs with attention (Veličković et al., 2018) often generate low-fidelity attention weights. As it learns multiple weights for every edge, it is far from trivial to combine those weights with the irregular graph structure to perform graph label-relevant feature selection.

There are two types of attention models: One normalizes the attention weights to sum to one (Bahdanau et al., 2015), while the other learns weights between $[0, 1]$ without normalization (Xu et al., 2015). As the counterparts in GNN models, GAT adopts the normalized one (Veličković et al., 2018) while GGNN adopts the unnormalized one (Li et al., 2016). Our method belongs to the second category.

Graph Neural Network. GNNs are neural network models that encode graph-structured data into node representations or graph representations. They initialize each node feature representation with its attributes $h_v^{(0)} = X_v$ and then gradually update it by aggregating representations from its neighbors, i.e., $h_v^{(l+1)} \leftarrow q(h_v^{(l)}, \{h_u^{(l)} | u : (u, v) \in E\})$ where $q(\cdot)$ denotes a function implemented by NNs (Gilmer et al., 2017). Graph representations are often obtained via an aggregation (sum/mean) of node representations.

Learning to Explain (L2X). L2X (Chen et al., 2018) studies the feature selection problem in the regular feature space and proposed a mutual information (MI) maximization rule to select a fixed number of features. Specifically, let $I(a; b) \triangleq \sum_{a,b} \mathbb{P}(a, b) \log \frac{\mathbb{P}(a,b)}{\mathbb{P}(a)\mathbb{P}(b)}$ denote the MI between two random variables a and b. Large MI indicates certain high correlation between two random variables. Hence, with input features $X \in \mathbb{R}^F$, L2X is to search a k-sized set of indices $S \subseteq \{1, 2, ..., F\}$, where $k = |S| < F$, such that the features in the subspace indexed by S (denoted by X_S) maximizes the mutual information with the labels Y, i.e.,

$$\max_{S \subseteq \{1,2,...,F\}} I(X_S; Y), \quad \text{s.t. } |S| \leq k. \tag{1}$$

Our model is inspired by L2X. However, as graph features and their interpretable counterparts are in an irregular space without a fixed dimension, directly applying L2X may achieve subpar performance in graph learning tasks. We propose to use information constraint instead in Sec. 3.1.

Later, we will also use the *entropy* defined as $H(a) \triangleq -\sum_a \mathbb{P}(a) \log \mathbb{P}(a)$ and the *KL-divergence* defined as $\mathrm{KL}(\mathbb{P}(a)||\mathbb{Q}(a)) \triangleq \sum_a \mathbb{P}(a) \log \frac{\mathbb{P}(a)}{\mathbb{Q}(a)}$ (Cover, 1999).

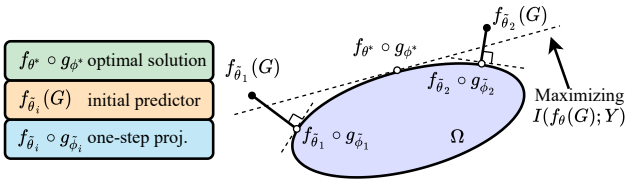


Figure 4. Post-hoc methods just perform one-step projection to the information-constrained space, which is always suboptimal and the interpretation performance is sensitive to the pre-trained model.

3. Graph Learning Interpretation via GIB

In this section, we will first propose the GIB-based objective for interpretable graph learning and point out the issues of post-hoc GNN interpretation methods.

3.1. GIB-based Objective for Interpretation

Finding label-relevant subgraphs in graph learning tasks has unique challenges. As for the irregularity of graph structures, graph learning models often have to deal with the input graphs of various sizes. The critical subgraph patterns may be also of different sizes and be highly irregular. Consider the example of molecular solubility again, although the functional groups for positive solubility such as -OH, -NH_2 are of similar sizes, those for negative solubility range from small groups (e.g., -Cl) to extremely large ones (e.g. -$C_{10}H_9$). And, a molecule may contain multiple functional groups scattered in the graph that determine its properties. Given these observations, it is not proper to just mimic the cardinality constraint used for a regular dimension space (Eq. (1)) and select subgraphs of certain sizes potentially with a connectivity constraint as done in (Ying et al., 2019). Inspired by the graph information bottleneck (GIB) principle (Wu et al., 2020; Yu et al., 2021), we propose to use information constraint instead to select label-relevant subgraphs, i.e., solving

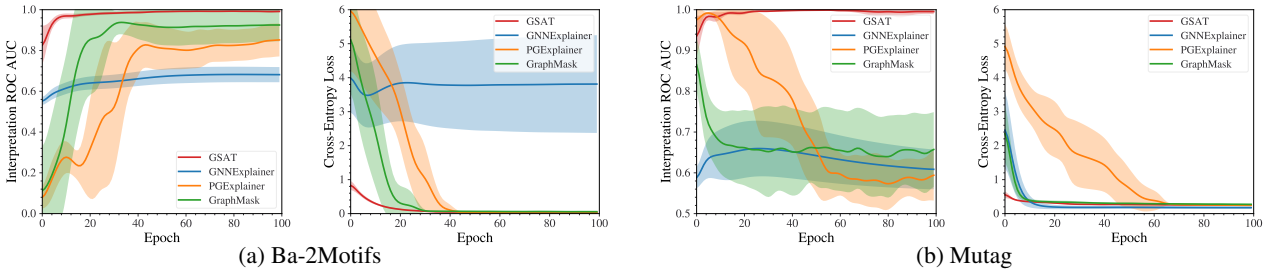
$$\max_{G_S} I(G_S; Y), \text{s.t. } I(G_S; G) \leq \gamma, G_S \in \mathbb{G}_{sub}(G) \tag{2}$$

where $\mathbb{G}_{sub}(G)$ denotes the set of the subgraphs of G. Note that GIB does not impose any potentially biased constraints such as the size or the connectivity of the selected subgraphs. Instead, GIB uses the information constraint $I(G_S; G) \leq \gamma$ to select G_S that inherits only the most indicative information from G to predict the label Y by maximizing $I(G_S; Y)$. As thus, G_S provides model interpretation.

Yu et al. (2021) also considered using GIB to select subgraphs. However, we adopt a fundamentally different mechanism that we will provide a detailed comparison in Sec. 4.4.

3.2. Issues of Post-hoc GNN Interpretation Methods

Almost all previous GNN interpretation methods are post-hoc, such as GNNExplainer (Ying et al., 2019), PGEx-



(a) Ba-2Motifs (b) Mutag

Figure 5. Issues of post-hoc interpretation methods. All methods are trained with 10 random seeds; post-hoc methods are also provided with models pre-trained with different seeds. Interpretation performance and the training losses of Eq. 2 for GSAT and Eq. 4 for others are shown. We guarantee that all the pre-trained models are well-trained in their pre-training stage (Acc. ∼100% Ba-2Motif, ∼90% Mutag).

plainer (Luo et al., 2020) and GraphMask (Schlichtkrull et al., 2021). Given a pre-trained predictor $f_\theta(\cdot): \mathcal{G} \to \mathcal{Y}$, they try to find out the subgraph G_S that impacts the model predictions the most, while keeping the pre-trained model unchanged. This procedure essentially first maximizes the MI between $f_\theta(G)$ and Y and obtains a model parameter

$$\tilde{\theta} \triangleq \arg\max_{\theta} I(f_\theta(G); Y), \tag{3}$$

and then optimizes a subgraph extractor g_ϕ via

$$\tilde{\phi} \triangleq \arg\max_{\phi} I(f_{\tilde{\theta}}(G_S); Y), \text{ s.t. } G_S = g_\phi(G) \in \Omega. \tag{4}$$

where Ω implies a subset of the subgraphs $\mathbb{G}_{sub}(G)$ that satisfy some constraints, e.g., the cardinality constraint adopted by GNNExplainer and PGExplainer. Let us temporarily ignore the difference between different constraints and just focus on the optimization objective. The post-hoc objective Eq. (4) and GIB (Eq. (2)) share some similar spirits. However, the post-hoc methods may not give or even approximate the optimal solution to Eq. (2) because $f_\theta \circ g_\phi$ is not jointly trained. From the optimization perspective, post-hoc methods just perform *one-single step projection* (see Fig. 4) from the model $f_{\tilde{\theta}}$ in an unconstrained space to $f_{\tilde{\theta}} \circ g_{\tilde{\phi}}$ in the information-constrained space Ω where the projection rule follows that the induced MI decrease $I(f_{\tilde{\theta}}(G); Y) - I(f_{\tilde{\theta}}(g_{\tilde{\phi}}(G)); Y)$ gets minimized.

In practice, such a suboptimal behavior will yield two undesired consequences. First, $f_{\tilde{\theta}}$ may not fully extract the information from $G_S = g_\phi(G)$ to predict Y during the optimization of Eq. (4) because $f_{\tilde{\theta}}$ is originally trained to make $I(f_{\tilde{\theta}}(G); Y)$ approximate $I(G, Y)$ while $(G_S, Y) = (g_\phi(G), Y)$ follows a distribution different from (G, Y). Therefore, $I(f_{\tilde{\theta}}(G_S); Y)$ may not well approximate $I(G_S; Y)$, and thus may mislead the optimization of g_ϕ and disable g_ϕ to select G_S that indeed indicates Y. GNNExplainer suffers from this issue over Ba-2Motif as shown in Fig. 5: The training loss, $-I(f_{\tilde{\theta}}(G_S); Y)$ keeps high and the interpretation performance is subpar. It is possible to further decrease the training loss via a more aggressive optimization of g_ϕ. However, the models may risk overfitting the data, which yields the second issue.

An aggressive optimization of g_ϕ may give a large *empirical* MI $\hat{I}\left(f_{\tilde{\theta}}(g_\phi(G)); Y\right)$ (or a small training loss equivalently) by selecting features that help to distinguish labels for training but are essentially irrelevant to the labels or spuriously correlated with the labels in the population level. Previous works have shown that label-irrelevant features are known to be discriminative enough to even identify each graph in the training dataset let alone the labels (Suresh et al., 2021). Empirically, we indeed observe such overfitting problems of all post-hoc methods over Mutag as shown in Fig. 5, especially PGExplainer and GraphMask. In the first 5 to 10 epochs, these two models succeed in selecting good explanations while having a large training loss. Further training successfully decreases the loss (after 10 epochs) but degenerates the interpretation performance substantially. This might also be the reason why in the original literatures of these post-hoc methods, training over only a small number of epochs is suggested. However, in practical tasks, it is hard to have the ground truth interpretation labels to verify the results and decide a trusty stopping criterion.

Another observation of Fig. 5 also matches our expectation: From the optimization perspective, post-hoc methods suffer from an initialization issue. Their interpretability can be highly sensitive to the pre-trained model $f_{\tilde{\theta}}$, as empirically demonstrated by the large variances in Fig. 5. Only if the pre-trained $f_{\tilde{\theta}}$ approximates the optimal f_{θ^*}, the performance can be roughly guaranteed. So, a joint training of $f_\theta \circ g_\phi$ according to the GIB principle Eq. (2) is typically needed.

4. Stochastic Attention Mechanism for GIB

In this section, we will first give a tractable variational bound of the GIB objective (Eq. (2)), and then introduce our model GSAT with the stochastic attention mechanism. We will further discuss how the stochastic attention mechanism improves both model interpretation and generalization.

4.1. A Tractable Objective for GIB

GSAT is to learn an extractor g_ϕ with parameter ϕ to extract $G_S \in \mathbb{G}_{\text{sub}}(G)$. g_ϕ blocks the label-irrelevant informa-

tion in the data G via injected stochasticity while allowing the label-relevant information kept in G_S to make predictions. In GSAT, $g_\phi(G)$ essentially gives a distribution over $\mathbb{G}_{\text{sub}}(G)$. We also denote this distribution as $\mathbb{P}_\phi(G_S|G)$. Later, $g_\phi(G)$ and $\mathbb{P}_\phi(G_S|G)$ are used interchangeably.

Putting the constraint into the objective (Eq.(2)), we obtain the optimization of g_ϕ via GIB, i.e., for some $\beta > 0$,

$$\min_{\phi} -I(G_S;Y) + \beta I(G_S;G), \text{ s.t. } G_S \sim g_\phi(G). \quad (5)$$

Next, we follow Alemi et al. (2016); Poole et al. (2019); Wu et al. (2020) to derive a tractable variational upper bound of the two terms in Eq. (5). Detailed derivation is given in Appendix B. For the term $I(G_S;Y)$, we introduce a parameterized variational approximation $\mathbb{P}_\theta(Y|G_S)$ for $\mathbb{P}(Y|G_S)$. We obtain a lower bound:

$$I(G_S;Y) \geq \mathbb{E}_{G_S,Y}[\log \mathbb{P}_\theta(Y|G_S)] + H(Y). \quad (6)$$

Note that $\mathbb{P}_\theta(Y|G_S)$ essentially works as the predictor $f_\theta : \mathcal{G} \to \mathcal{Y}$ with parameter θ in our model. For the term $I(G_S;G)$, we introduce a variational approximation $\mathbb{Q}(G_S)$ for the marginal distribution $\mathbb{P}(G_S) = \sum_G \mathbb{P}_\phi(G_S|G)\mathbb{P}_\mathcal{G}(G)$. And, we obtain an upper bound:

$$I(G_s;G) \leq \mathbb{E}_G[\text{KL}(\mathbb{P}_\phi(G_S|G)||\mathbb{Q}(G_S))] \quad (7)$$

Plugging in the above two inequalities, we obtain a variational upper bound of Eq. (5) as the objective of GSAT:

$$\min_{\theta,\phi} -\mathbb{E}[\log \mathbb{P}_\theta(Y|G_S)] + \beta \mathbb{E}[\text{KL}(\mathbb{P}_\phi(G_S|G)||\mathbb{Q}(G_S))], \quad \text{s.t.} \quad G_S \sim \mathbb{P}_\phi(G_S|G). \quad (8)$$

Next, we specify $\mathbb{P}_\theta$ (aka f_θ), $\mathbb{P}_\phi$ (aka g_ϕ) and $\mathbb{Q}$ in GSAT.

4.2. GSAT and Stochastic Attention Mechanism

For clarity, we introduced the predictor f_θ and the extractor g_ϕ separately. Actually, GSAT is a unified model as f_θ, g_ϕ share the same GNN encoder except their last layers.

Stochastic Attention via $\mathbb{P}_\phi$. The extractor g_ϕ first encodes the input graph G via the GNN into a set of node representations $\{h_v|v \in V\}$. For each edge $(u,v) \in E$, g_ϕ contains an MLP layer plus sigmoid that maps the concatenation (h_u, h_v) into $p_{uv} \in [0,1]$. Then, for each forward pass of the training, we sample stochastic attention from Bernoulli distributions $\alpha_{uv} \sim \text{Bern}(p_{uv})$. To make sure the gradient w.r.t. p_{uv} is computable, we apply the gumbel-softmax reparameterization trick (Jang et al., 2017). The extracted graph G_S will have an attention-selected subgraph as $A_S = \alpha \odot A$. Here α is the matrix with entries α_{uv} for $(u,v) \in E$ or zeros for the non-edge entries. A is the adjacency matrix of G and $\odot$ is entry-wise product. The distribution of G_S given G through the above procedure characterizes $\mathbb{P}_\phi(G_S|G)$, so $\mathbb{P}_\phi(G_S|G) = \prod_{u,v\in E} \mathbb{P}(\alpha_{uv}|p_{uv})$, where p_{uv} is a function of G. This essentially makes the attention α_{uv} to be conditionally independent across different edges given the input graph G.

似乎就是随机选边

Prediction via $\mathbb{P}_\theta$. The predictor f_θ adopts the same GNN to encode the extracted graph G_S to a graph representation, and finally passes such representation through an MLP layer plus softmax to model the distribution of Y. This procedure gives the variational distribution $\mathbb{P}_\theta(Y|G_S)$.

Marginal Distribution Control via $\mathbb{Q}$. The bound Eq.(7) is always true for any $\mathbb{Q}(G_S)$. We define $\mathbb{Q}(G_S)$ as follows. For every graph $G \sim \mathbb{P}_\mathcal{G}$ and every two directed node pair (u,v) in G, we sample $\alpha'_{uv} \sim \text{Bern}(r)$ where $r \in [0,1]$ is a hyperparameter. We remove all edges in G and add all edges (u,v) if $\alpha'_{uv} = 1$. Suppose the obtained graph is G_S. This procedure defines the distribution $\mathbb{Q}(G_S) = \sum_G \mathbb{P}(\alpha'|G)\mathbb{P}_\mathcal{G}(G)$. As α' is independent from the graph G given its size n, $\mathbb{Q}(G_S) = \sum_n \mathbb{P}(\alpha'|n)\mathbb{P}_\mathcal{G}(G = n) = \mathbb{P}(n)\prod_{u,v=1}^n \mathbb{P}(\alpha'_{uv})$. The probability of an n-sized graph $\mathbb{P}(n)$ is a constant and thus will not affect the model. Note that our choice of $\mathbb{Q}(G_S)$ shares the similar spirit of using standard Gaussian as the latent distribution with variational auto-encoders (Kingma & Welling, 2014).

Using the above $\mathbb{P}_\theta$, the first term in Eq.(8) reduces to a standard cross entropy loss. Using $\mathbb{P}_\phi$ and $\mathbb{Q}$, the KL-divergence term becomes, for every $G \sim \mathbb{P}_\mathcal{G}$, n as the size of G,

$$\text{KL}(\mathbb{P}_\phi(G_S|G)||\mathbb{Q}(G_S)) = \sum_{(u,v)\in E} p_{uv}\log\frac{p_{uv}}{r} + (1-p_{uv})\log\frac{1-p_{uv}}{1-r} + c(n,r). \quad (9)$$

where $c(n,r)$ is a constant without any trainable parameters.

4.3. The Interpretation Mechanism of GSAT

The interpretability of GSAT essentially comes from the information control: GSAT decreases the information from the input graphs by injecting stochasticity via attention into G_S. In the training, the regularization term Eq.(9) would try to assign large stochasticity for all edges, yet driven by the classification loss $\min -I(G_S;Y)$ (equivalent to cross-entropy loss), GSAT can learn to reduce such stochasticity of the attention on the task-relevant subgraphs. So, it is not the entire G_S but the part of G_S with the stochasticity-reduced attention, aka $p_{uv} \to 1$, that provide model interpretation. Therefore, when GSAT provides interpretation, in practice, one can rank all edges according to p_{uv} and use those top ranked ones (given a certain budget if needed) as the detected subgraph for interpretation. The contribution of injecting stochasticity to the performance is so significant as shown in experiments (Table 5), so is the contribution of our regularization term (Eq. (9)) when we compare it with the sparsity-driven ℓ_1-norm (Fig. 7).

GSAT is substantially different from previous methods, as we do not use any sparsity constraints such as ℓ_1-norm (Ying et al., 2019; Luo et al., 2020), ℓ_0-norm (Schlichtkrull et al., 2021) or ℓ_2-regression to $\{0, 1\}$ (Yu et al., 2021) to select size-constrained (or connectivity-constrained) subgraphs. We actually observe that setting r away from 0 in the marginal regularization (Eq. (9)), i.e., pushing G_S away from being sparse often provides more robust interpretation. This matches our intuition that GIB by definition does not make any assumptions on the selected subgraphs but just constrains the information from the original graphs. Our experiments show that GSAT outperform baselines significantly without leveraging those assumptions in the optimization even if the label-relevant subgraphs satisfy these assumptions. If the label-relevant subgraphs are indeed disconnected or vary in sizes, the improvement of GSAT is expected to be even more.

4.4. Further Comparison on Interpretation Mechanism

PGExplainer and GraphMask also have stochasticity in their models (Luo et al., 2020; Schlichtkrull et al., 2021). However, their main goal is to enable a gradient-based search over a discrete subgraph-selection space rather than control the information as GSAT does. Hence, they did not in principle derive the information regularization as ours (Eq. (9)) but adopt sparsity constraints to extract a small subgraph G_S directly used for interpretation.

IB-subgraph (Yu et al., 2021) considers using GIB as the objective but does not inject any stochasticity to generate G_S, so its selected subgraph G_S is a deterministic function of G. Specifically, IB-subgraph samples batches of graphs G to estimate $I(G_S; G)$ and optimize a deterministic function $G_S = g_\phi(G)$ to minimize such MI estimation. In this case $I(G_S; G)(= H(G_S) - H(G_S|G))$ reduces to the entropy $H(G_S)$, which tends to give a small-sized G_S, because the space of small graphs is small and has a lower upper bound of the entropy. By contrast, $G_S \sim g_\phi(G)$ is random in GSAT, and GSAT implements GIB mainly by increasing $H(G_S|G)$ via injecting stochasticity.

4.5. Guaranteed Spurious Correlation Removal

GSAT can remove spurious correlations in the training data and has guaranteed interpretability. We may prove that if there exists a correspondence between a subgraph pattern G_S^* and the label Y, the pattern G_S^* is the optimal solution of the GIB objective (Eq. (2)).

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

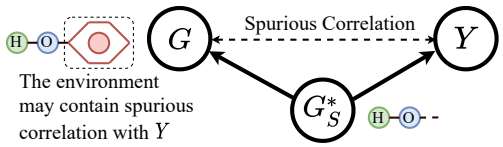


Figure 6. G_S^* determines Y. However, the environment features in $G \backslash G_S^*$ may contain spurious (backdoor) correlation with Y.

Proof. Consider the following derivation:

$$
\begin{aligned}
& I(G_S; Y) - \beta I(G_S; G) \\
=& I(Y; G, G_S) - I(G; Y|G_S) - \beta I(G_S; G) \\
=& I(Y; G, G_S) - (1-\beta) I(G; Y|G_S) - \beta I(G; G_S, Y) \\
=& I(Y; G) - (1-\beta) I(G; Y|G_S) - \beta I(G; G_S, Y) \\
=& (1-\beta) I(Y; G) - (1-\beta) I(G; Y|G_S) - \beta I(G; G_S|Y),
\end{aligned}
$$

where the third equality is because $G_S \in \mathbb{G}_{sub}(G)$, then (G_S, G) holds no more information than G.

If $\beta \in [0, 1]$, G_S that maximizes $I(G_S, Y) - \beta I(G_S; G)$ can also minimize $(1-\beta) I(G; Y|G_S) + \beta I(G; G_S|Y)$. As $I(G; Y|G_S) \geq 0$, $I(G; G_S|Y) \geq 0$, the lower bound of $(1-\beta) I(G; Y|G_S) + \beta I(G; G_S|Y)$ is 0.

G_S^* is the subgraph that makes $(1-\beta) I(G; Y|G_S^*) + \beta I(G; G_S^*|Y) = 0$. This is because (a) $Y = f(G_S^*) + \epsilon$ where ϵ is independent of G so $I(G; Y|G_S^*) = 0$ and (b) $G_S^* = f^{-1}(Y - \epsilon)$ where ϵ is independent of G so $I(G; G_S^*|Y) = 0$. Therefore, $G_S = G_S^*$ maximizes GIB $I(G_S; Y) - \beta I(G_S; G)$, where $G_S \in \mathbb{G}_{\text{sub}}(G)$. □

Although G_S^* determines Y, in the training dataset the data G and Y may have some spurious correlation caused by the environment (Pearl et al., 2016; Arjovsky et al., 2019; Chang et al., 2020; Krueger et al., 2021). That is, $G \backslash G_S^*$ may have some correlation with the label, but this correlation is spurious and is not the true reason that determines its label (illustrated in Fig. 6). A model trained over G to predict Y via just MI maximization may capture such spurious correlation. If such correlation is changed during the test phase, the model suffers from performance decay.

However, Theorem 4.1 indicates that GSAT by optimizing the GIB objective has the capability to address the above issue by only extracting G_S^*, which removes the spurious correlation and also provides guaranteed interpretability.

4.6. Fine-tuning and Interpreting a Pre-trained Model

GSAT can also fine-tune and interpret a pre-trained GNN. Given a GNN $f_{\tilde{\theta}}$ pre-trained by $\max_\theta I(f_\theta(G); Y)$, GSAT can fine-tune it via $\max_{\theta,\phi} I(f_\theta(G_S); Y) - \beta I(G_S; G)$, $G_S \sim g_\phi(G)$ by initializing the GNN used in g_ϕ and f_θ as the one in the pre-trained model $f_{\tilde{\theta}}$.

We observe that this framework almost never hurts the original prediction performance (and sometimes even boosts it).

Moreover, this framework often achieves better interpretation results compared with training the GNN from scratch.

5. Other Related Works

Besides the models (Ying et al., 2019; Luo et al., 2020; Schlichtkrull et al., 2021; Yu et al., 2021) that we have compared with in detail in Sec. 3.2 and Sec. 4.4, we review some other interpretation methods here.

Most previous works on GNN interpretation are post-hoc (Ribeiro et al., 2016). Some works strongly rely on the connectivity assumption and only search over the space of connected subgraphs for interpretation. They adopt either reinforcement learning (Yuan et al., 2020a) or Monte Carlo tree search (Yuan et al., 2021). Other methods including PGM-Explainer (Vu & Thai, 2020) leveraging graphical models, Gem (Lin et al., 2021) checking Granger causality and Graphlime (Huang et al., 2020) using HSIC Lasso are only applied to node-level task interpretation. Some works check the gradients w.r.t. the input features to find important features (Pope et al., 2019; Baldassarre & Azizpour, 2019).

Much fewer works have considered intrinsic interpretation. Recently, Wu et al. (2022) has proposed DIR to make the model avoid overfitting spurious correlations and only capture invariant rationales to provide interpretability. However, DIR needs to iteratively break graphs into subgraphs and assemble subgraphs into graphs during the model training, which is far more complicated than GSAT.

6. Experiments

We evaluate our method for both interpretability and prediction performance. We will compare our method with both state-of-the-art (SOTA) post-hoc interpretation methods and inherently interpretable models. We will also compare with several invariant learning methods to demonstrate the ability of GSAT to remove spurious correlations. We briefly introduce datasets, baselines and experiment settings here, and more details can be found in Appendix C.

6.1. Datasets

Mutag (Debnath et al., 1991) is a molecular property prediction dataset. Following (Luo et al., 2020), -NO_2 and -NH_2 in mutagen graphs are labeled as ground-truth explanations.

BA-2Motifs (Luo et al., 2020) is a synthetic dataset with binary graph labels. House motifs and cycle motifs give class labels and thus are regarded as ground-truth explanations for the two classes respectively.

Spurious-Motif (Wu et al., 2022) is a synthetic dataset with three graph classes. Each class contains a particular motif that can be regarded as the ground-truth explanation. Some spurious correlation between the rest graph components (other than the motifs) and the labels also exists in the training data. The degree of such correlation is controlled by b, and we include datasets with $b = 0.5, 0.7$ and 0.9.

MNIST-75sp (Knyazev et al., 2019) is an image classification dataset, where each image in MNIST is converted to a superpixel graph. Nodes with nonzero pixel values provide ground-truth explanations. Note that the subgraphs that provide explanations are of different sizes in this dataset.

Graph-SST2 (Socher et al., 2013; Yuan et al., 2020b) is a sentiment analysis dataset, where each text sequence in SST2 is converted to a graph. Following the splits in (Wu et al., 2022), this dataset contains degree shifts and no ground-truth explanation labels. So, we only evaluate prediction performance and provide interpretation visualizations.

OGBG-Molhiv (Wu et al., 2018; Hu et al., 2020) is a molecular property prediction datasets. We also evaluate GSAT on molbace, molbbbp, molclintox, moltox21 and molsider datasets from OGBG. As there are no ground truth explanation labels for these datasets, we only evaluate the prediction performance of GSAT.

6.2. Baselines and Setup

Interpretability Baselines. We compare interpretability with post-hoc methods GNNExplainer (Ying et al., 2019), PGExplainer (Luo et al., 2020), GraphMask (Schlichtkrull et al., 2021), and inherently interpretable models DIR (Wu et al., 2022) and IB-subgraph (Yu et al., 2021).

Prediction Baselines. We compare prediction performance with the backbone models GIN (Xu et al., 2019) and PNA (Corso et al., 2020), and inherently interpretable models DIR (Wu et al., 2022) and IB-subgraph (Yu et al., 2021).

Invariant Learning Baselines. We compare the ability to remove spurious correlations with invariant learning methods IRM (Arjovsky et al., 2019), V-REx (Krueger et al., 2021) and DIR (Wu et al., 2022). Baseline results yielded by empirical risk minimization (ERM) are also included.

Metrics. For interpretation evaluation, we report explanation ROC AUC following (Ying et al., 2019; Luo et al., 2020). For prediction performance, we report classification ROC AUC for all OGBG datasets and report accuracy for all other datasets. All the results are averaged over 10 times tests with different random seeds. For the post-hoc methods, we do not cherry pick a pre-trained model. Instead, in each test, we interpret a model pre-trained independently that achieves the best validation performance.

Setup. Since we focus on graph classification tasks, GIN (Xu et al., 2019) is used as the backbone model for both baselines and GSAT. We also apply PNA (Corso et al., 2020) to further test the wide applicability of GSAT, for which we adopt the no-scalars version since the scalars used in PNA

Table 1. Interpretation Performance (AUC). The underlined 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	$\mathbf{96.22}^\dagger \pm 2.08$	$\mathbf{88.54}^\dagger \pm 0.72$	$\mathbf{90.55}^\dagger \pm 1.48$	$\mathbf{89.79}^\dagger \pm 1.91$	$\mathbf{89.54}^\dagger \pm 1.78$

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

are essentially a type of attention, which may conflict with our method. GIN+GSAT denotes using GIN as the base GNN encoder of GSAT, and PNA+GSAT means replacing the GNN encoder with PNA. In addition, we apply GSAT to fine-tune and interpret pre-trained models as described in Sec. 4.6, which is highlighted as GSAT*. In all the experiments, we use $r = 0.7$ in Eq. (9) by default or otherwise specified. Our studies have shown that GSAT is generally robust when $r \in [0.5, 0.9]$ (see Fig. 7 later).

6.3. Result Comparison and Analysis

Interpretability Results. As shown in Table 1, our methods significantly outperform the baselines by 9%↑ on average and up to 20%↑. If we just compare among inherently interpretable models, the boost is even more significant. Moreover, GSAT also provides much stabler interpretation than the baselines as for the much smaller variance. GSAT* via fine-tuning a pre-trained model can often further boost the interpretation performance. Also, when the more expressive model PNA is used as the backbone, we find the posthoc methods are likely to suffer from the overfitting issue as explained in Sec. 3.2. However, GSAT does not suffer from that and can yield even better interpretation results. Over Ba-2Motifs and Mutag, GNNExplainer and PGExplainer work worse than what reported in (Luo et al., 2020) as we do not cherry pick the pre-trained model. However, GSAT still significantly outperforms their reported performance in the Appendix C.4. We also provide visualizations of the subgraphs discovered by GSAT in Appendix D.

Prediction Results. As explained in Sec. 4.5, being trained via the GIB principle, GSAT is more generalizable and thus may achieve even better prediction performance. As shown in Table 2, GIN+GSAT significantly outperforms the backbone GIN over the Spurious-Motif datasets, where spurious correlation exists in the training data. For other datasets, GIN+GSAT can achieve comparable results, which matches our claim that GSAT provides interpretation without hurting the prediction. IB-subgraph, trained via the GIB principle, also achieves good prediction performance though its interpretability is poor (Table 1). When PNA is used, GSAT improves it by about $1-5\%$ on the datasets in the first three columns. Notably, GSAT* achieves the SOTA performance on *molhiv* among all models that do not incorporate expert knowledge according to the leaderboard. Unexpectedly, PNA achieves very good performance on Spurious-Motif and GSAT* just slightly improves it. Our results on the other 5 molecular datasets from OGBG are showed in Table 3, where GSAT and GSAT* mostly outperform PNA.

Invariant Learning Results. We note that DIR achieves a bit lower prediction performance in Table 2 than what reported in (Wu et al., 2022) even after we extensively tune its

Table 3. Generalization ROC AUC on other OGBG-Mol datasets. The **bold** font highlights when GSAT outperforms PNA.

	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	$\mathbf{89.26} \pm 1.66$	$\mathbf{75.71} \pm 0.48$	$\mathbf{59.19} \pm 1.03$

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	$\mathbf{45.76}^{\dagger} \pm 5.32$

parameters, which is probably due to the different backbone models used. Hence, we also compare with DIR by using their backbone model. And we include several invariant learning baselines reported in DIR to further demonstrate the ability of GSAT to remove spurious correlations. Results are shown in Table 4. GSAT significantly outperforms all invariant learning methods on spurious correlation removal, even without utilizing causality analysis, which further validates our claims in Sec. 4.5. A comparison of interpretability of these models is shown in Table 7 in the appendix.

Ablation Studies. We conduct ablation studies from three aspects: First, the importance of stochasticity in GSAT, where we replace the Bernoulli sampling procedure with setting attention $\alpha_{uv} = p_{uv}$ without stochasticity; Second, the importance of the information regularization term (Eq. (9)), where we set its coefficient $\beta = 0$ in Eq. (8); Third, the superiority of the information regularization term over the sparsity-driven term ℓ_1-norm.

As shown in Table 5, the performance drops significantly when there is either no stochasticity or $\beta = 0$. Specifically, GSAT-NoStoch means applying deterministic attention $\in [0, 1]$, which causes the most performance drop. GSAT-NoStoch-$\beta = 0$ corresponds to using deterministic attention without the regularization term in Eq. (9), which causes the second most performance drop. GSAT-$\beta = 0$ denotes applying stochastic attention with no regularization, which performs better than baselines but worse than original GSAT and suffers from large variance. Overall, no stochasticity yields the biggest drop, which well matches our theory. This also implies that directly using the deterministic attention mechanisms such as GAT (Veličković et al., 2018) or GGNN (Li et al., 2016) may not yield good interpretability.

Fig. 7 shows that our information regularization term can achieve consistently better performance than the sparsity-driven ℓ_1-norm regularization even when the grid search is used to tune hyperparameters. We also observe that when r is close to 0, the results often get decreased or have higher variance. The best performance is often achieved when $r \in [0.5, 0.9]$, which matches our theory. More results on other datasets can be found in Fig. 8 in the appendix.

Table 5. Ablation study on β and stochasticity in GSAT (GIN as the backbone model) on Spurious-Motif. We report both interpretation ROC AUC (top) and prediction accuracy (bottom).

SPURIOUS-MOTIF	$b = 0.5$	$b = 0.7$	$b = 0.9$
GSAT	79.81 ± 3.98	74.07 ± 5.28	71.97 ± 4.41
GSAT-$\beta = 0$	66.00 ± 11.04	65.92 ± 3.28	66.31 ± 6.82
GSAT-NOSTOCH	59.64 ± 5.33	55.78 ± 2.84	55.27 ± 7.49
GSAT-NOSTOCH-$\beta = 0$	63.37 ± 12.33	60.61 ± 10.08	66.19 ± 7.76
GIN	39.87 ± 1.30	39.04 ± 1.62	38.57 ± 2.31
GSAT	51.86 ± 5.51	49.12 ± 3.29	44.22 ± 5.57
GSAT-$\beta = 0$	45.97 ± 8.37	49.67 ± 7.01	49.84 ± 5.45
GSAT-NOSTOCH	40.34 ± 2.77	41.90 ± 3.70	37.98 ± 2.64
GSAT-NOSTOCH-$\beta = 0$	43.41 ± 8.05	45.88 ± 9.54	42.25 ± 9.77

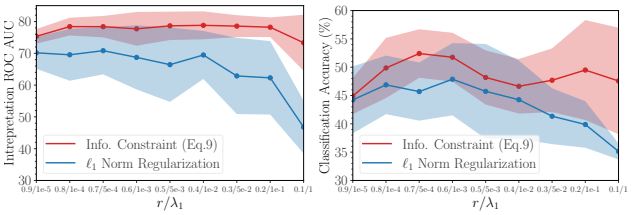


Figure 7. Comparison between (a) using the information constraint in Eq. (9) and (b) replacing it with ℓ_1-norm. Results are shown for Spurious-Motif $b = 0.5$, where r is tuned from 0.9 to 0.1 and the coefficient of the ℓ_1-norm λ_1 is tuned from $1e$-5 to 1.

7. Conclusion

Graph Stochastic Attention (GSAT) is a novel attention mechanism to build interpretable graph learning models. GSAT injects stochasticity to block label-irrelevant information and leverages the reduction of stochasticity to select label-relevant subgraphs. Such rationale is grounded by the information bottleneck principle. GSAT has many transformative characteristics. For example, it removes the sparsity, continuity or other potentially biased assumptions in graph learning interpretation without performance decay. It can also remove spurious correlation to better the model generalization. As a by-product, we also reveal a potentially severe issue behind post-hoc interpretation methods from the optimization perspective of information bottleneck.

ACKNOWLEDGMENTS

We greatly thank the actionable suggestions given by reviewers. S. Miao and M. Liu are supported by the National Science Foundation (NSF) award HDR-2117997. P. Li is supported by the JPMorgan Faculty Award.

References

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2016.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, 2015.

Baldassarre, F. and Azizpour, H. Explainability techniques for graph convolutional networks. In *International Conference on Machine Learning Workshops, 2019 Workshop on Learning and Reasoning with Graph-Structured Representations*, 2019.

Bapst, V., Keck, T., Grabska-Barwińska, A., Donner, C., Cubuk, E. D., Schoenholz, S. S., Obika, A., Nelson, A. W., Back, T., Hassabis, D., et al. Unveiling the predictive power of static structure in glassy systems. *Nature Physics*, 16(4):448–454, 2020.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. Curriculum learning. In *International Conference on Machine Learning*, pp. 41–48, 2009.

Chang, S., Zhang, Y., Yu, M., and Jaakkola, T. Invariant rationalization. In *International Conference on Machine Learning*, pp. 1448–1458. PMLR, 2020.

Chen, J., Song, L., Wainwright, M., and Jordan, M. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning*, pp. 883–892. PMLR, 2018.

Corso, G., Cavalleri, L., Beaini, D., Liò, P., and Veličković, P. Principal neighbourhood aggregation for graph nets. In *Advances in Neural Information Processing Systems*, pp. 13260–13271, 2020.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Cranmer, M., Sanchez Gonzalez, A., Battaglia, P., Xu, R., Cranmer, K., Spergel, D., and Ho, S. Discovering symbolic models from deep learning with inductive biases. In *Advances in Neural Information Processing Systems*, pp. 17429–17442, 2020.

Debnath, A. K., Lopez de Compadre, R. L., Debnath, G., Shusterman, A. J., and Hansch, C. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34 (2):786–797, 1991.

Du, M., Liu, N., and Hu, X. Techniques for interpretable machine learning. *Communications of the ACM*, 63(1): 68–77, 2019.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pp. 1263–1272. PMLR, 2017.

Henderson, R., Clevert, D.-A., and Montanari, F. Improving molecular graph neural network explainability with orthonormalization and induced sparsity. In *International Conference on Machine Learning*, pp. 4203–4213. PMLR, 2021.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems*, pp. 22118–22133, 2020.

Huang, Q., Yamada, M., Tian, Y., Singh, D., Yin, D., and Chang, Y. Graphlime: Local interpretable model explanations for graph neural networks. *arXiv preprint arXiv:2001.06216*, 2020.

Jain, S. and Wallace, B. C. Attention is not explanation. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3543–3556, 2019.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Knyazev, B., Taylor, G. W., and Amer, M. Understanding attention and generalization in graph neural networks. In *Advances in Neural Information Processing Systems*, pp. 4204–4214, 2019.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Priol, R. L., and Courville, A. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

Li, Y., Zemel, R., Brockschmidt, M., and Tarlow, D. Gated graph sequence neural networks. In *International Conference on Learning Representations*, 2016.

Lin, W., Lan, H., and Li, B. Generative causal explanations for graph neural networks. In *International Conference on Machine Learning*, pp. 6666–6679. PMLR, 2021.

Lipton, Z. C. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., and Zhang, X. Parameterized explainer for graph neural network. In *Advances in Neural Information Processing Systems*, pp. 19620–19631, 2020.

Mohankumar, A. K., Nema, P., Narasimhan, S., Khapra, M. M., Srinivasan, B. V., and Ravindran, B. Towards transparent and explainable attention models. In *Association for Computational Linguistics*, pp. 4206–4216, 2020.

Pearl, J., Glymour, M., and Jewell, N. P. *Causal Inference in Statistics: A Primer*. John Wiley & Sons, 2016.

Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, pp. 5171–5180. PMLR, 2019.

Pope, P. E., Kolouri, S., Rostami, M., Martin, C. E., and Hoffmann, H. Explainability methods for graph convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10772–10781, 2019.

Ribeiro, M. T., Singh, S., and Guestrin, C. Model-agnostic interpretability of machine learning. In *International Conference on Machine Learning Workshops, 2016 Workshop on Human Interpretability in Machine Learning*, 2016.

Schlichtkrull, M. S., Cao, N. D., and Titov, I. Interpreting graph neural networks for {nlp} with differentiable edge masking. In *International Conference on Learning Representations*, 2021.

Serrano, S. and Smith, N. A. Is attention interpretable? In *Association for Computational Linguistics*, pp. 2931–2951, 2019.

Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Suresh, S., Li, P., Hao, C., and Neville, J. Adversarial graph augmentation to improve graph contrastive learning. In *Advances in Neural Information Processing Systems*, pp. 15920–15933, 2021.

Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. In *IEEE Information Theory Workshop*, pp. 1–5. IEEE, 2015.

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Vu, M. and Thai, M. T. Pgm-explainer: Probabilistic graphical model explanations for graph neural networks. In *Advances in Neural Information Processing Systems*, pp. 12225–12235, 2020.

Wencel-Delord, J. and Glorius, F. C–h bond activation enables the rapid construction and late-stage diversification of functional molecules. *Nature chemistry*, 5(5):369–375, 2013.

Wu, T., Ren, H., Li, P., and Leskovec, J. Graph information bottleneck. In *Advances in Neural Information Processing Systems*, pp. 20437–20448, 2020.

Wu, Y., Wang, X., Zhang, A., He, X., and Chua, T.-S. Discovering invariant rationales for graph neural networks. In *International Conference on Learning Representations*, 2022.

Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., and Pande, V. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pp. 2048–2057. PMLR, 2015.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

Ying, Z., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. Gnnexplainer: Generating explanations for graph neural networks. In *Advances in Neural Information Processing Systems*, pp. 9240–9251, 2019.

Yu, J., Xu, T., Rong, Y., Bian, Y., Huang, J., and He, R. Graph information bottleneck for subgraph recognition. In *International Conference on Learning Representations*, 2021.

Yuan, H., Tang, J., Hu, X., and Ji, S. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 430–438, 2020a.

Yuan, H., Yu, H., Gui, S., and Ji, S. Explainability in graph neural networks: A taxonomic survey. *arXiv preprint arXiv:2012.15445*, 2020b.

Yuan, H., Yu, H., Wang, J., Li, K., and Ji, S. On explainability of graph neural networks via subgraph explorations. In *International Conference on Machine Learning*, pp. 12241–12252. PMLR, 2021.

A. Supplementary Notations for Information Theory and Graph Neural Networks

Entropy. Given a discrete random variable a, its entropy is defined as $H(a) \triangleq -\sum_a \mathbb{P}(a)\log\mathbb{P}(a)$. If a is a continuous random variable, its differential entropy is defined as $H(a) \triangleq -\int_a \mathbb{P}(a)\log\mathbb{P}(a)da$.

KL-Divergence. Given two distributions $\mathbb{P}(x)$ and $\mathbb{Q}(x)$, KL-Divergence is used to measure the difference between $\mathbb{P}$ and $\mathbb{Q}$, and it is defined as $\mathrm{KL}(\mathbb{P}(x)||\mathbb{Q}(x)) \triangleq \sum_x \mathbb{P}(x)\log\frac{\mathbb{P}(x)}{\mathbb{Q}(x)}$.

Mutual Information. Given two random variables a and b, the mutual information (MI) $I(a;b)$ is a measure of the mutual dependence between them. MI quantifies the amount of information regarding one random variable if another random variable is known. Formally, $I(a;b) \triangleq \sum_{a,b}\mathbb{P}(a,b)\log\frac{\mathbb{P}(a,b)}{\mathbb{P}(a)\mathbb{P}(b)}$, where $\mathbb{P}(a,b)$ is the joint distribution and $\mathbb{P}(a)$, $\mathbb{P}(b)$ are the marginal distributions. By definition, $I(a,b) = \mathrm{KL}(\mathbb{P}(a,b)||\mathbb{P}(a)\mathbb{P}(b)) = \sum_{a,b}\mathbb{P}(a,b)\log\mathbb{P}(a|b) - \sum_b \mathbb{P}(b)\log\mathbb{P}(b) = -H(a|b) + H(b)$.

Graph Neural Networks (GNNs). Given an L-layer GNN, let $h_v^{(l)}$ denote the node representation for node v in the i^{th} layer and $\mathcal{N}(v)$ denote a set of nodes adjacent to node v. Let $h_v^{(0)}$ be the node feature X_v. Most GNNs follow a message passing scheme, where there are two main steps in each layer: (1) neighbourhood aggregation, $m_v^{(l)} = \mathrm{AGG}(\{h_u^{(l-1)}|u \in \mathcal{N}(v)\})$; (2) node representation update, $h_v^{(l)} = \mathrm{UPDATE}(m_v^{(l)}, h_v^{(l-1)})$. For graph classification tasks, after obtaining $h_v^{(L)}$ for each node, the graph representation is given by $h_G = \mathrm{POOL}(\{h_v^{(L)}|v \in V\})$ and h_G will be used to make predictions. The above AGG, UPDATE, POOL are three functions. AGG and POOL are typically implemented via SUM, MEAN and MAX while UPDATE is a fully connected (typically shallow) neural network. In some cases, edge representations may be in need, and they are often given by $h_{u,v}^{(l)} = \mathrm{CONCAT}(h_u^{(l)}, h_v^{(l)})$.

B. Variational Bounds for the GIB Objective — Eq. (6) and Eq. (7)

From Eq. (5), the IB objective is:

$$\min_{\phi} -I(G_S;Y) + \beta I(G_S;G), \text{ s.t. } G_S \sim g_\phi(G). \tag{10}$$

To optimize it, we introduce two variational bounds on the two terms, respectively.

For the first term $I(G_S;Y)$, by definition:

$$I(G_S;Y) = \mathbb{E}_{G_S,Y}\left[\log\frac{\mathbb{P}(Y|G_S)}{\mathbb{P}(Y)}\right]. \tag{11}$$

Since $\mathbb{P}(Y|G_S)$ is intractable, we introduce a variational approximation $\mathbb{P}_\theta(Y|G_S)$ for it. Then, we obtain a lower bound for Eq. (6):

$$\begin{aligned} I(G_S;Y) &= \mathbb{E}_{G_S,Y}\left[\log\frac{\mathbb{P}_\theta(Y|G_S)}{\mathbb{P}(Y)}\right] + \mathbb{E}_{G_S}\left[\mathrm{KL}(\mathbb{P}(Y|G_S)||\mathbb{P}_\theta(Y|G_S))\right] \\ &\geq \mathbb{E}_{G_S,Y}\left[\log\frac{\mathbb{P}_\theta(Y|G_S)}{\mathbb{P}(Y)}\right] \\ &= \mathbb{E}_{G_S,Y}\left[\log\mathbb{P}_\theta(Y|G_S)\right] + H(Y). \end{aligned} \tag{12}$$

For the second term $I(G;G_S)$, by definition:

$$I(G;G_S) = \mathbb{E}_{G_S,G}\left[\log\frac{\mathbb{P}(G_S|G)}{\mathbb{P}(G_S)}\right]. \tag{13}$$

Since $\mathbb{P}(G_S)$ is intractable, we introduce a variational approximation $\mathbb{Q}(G_S)$ for the marginal distribution $\mathbb{P}(G_S) = \sum_G \mathbb{P}_\phi(G_S|G)\mathbb{P}_\mathcal{G}(G)$. Then, we obtain an upper bound for Eq. (7):

$$\begin{aligned} I(G;G_S) &= \mathbb{E}_{G_S,G}\left[\log\frac{\mathbb{P}_\phi(G_S|G)}{\mathbb{Q}(G_S)}\right] - \mathrm{KL}\left(\mathbb{P}(G_S)||\mathbb{Q}(G_S)\right) \\ &\leq \mathbb{E}_G\left[\mathrm{KL}\left(\mathbb{P}_\phi(G_S|G)||\mathbb{Q}(G_S)\right)\right]. \end{aligned} \tag{14}$$

Table 6. Direct comparison with the interpretation ROC AUC of GNNExplainer and PGExplainer reported in (Luo et al., 2020), which are given a selected pre-trained model.

	BA-2MOTIFS	MUTAG
GNNEXPLAINER	74.2	72.7
PGEXPLAINER	92.6	87.3
GSAT	$\mathbf{98.74}^{\dagger} \pm 0.55$	$\mathbf{99.60}^{\dagger} \pm 0.51$
GSAT*	$\mathbf{97.43}^{\dagger} \pm 0.02$	$\mathbf{97.75}^{\dagger} \pm 0.92$

Table 7. Direct comparison with the interpretation precision@5 of DIR reported in (Wu et al., 2022) based on the backbone model in (Wu et al., 2022).

	SPURIOUS-MOTIF		
	$b = 0.5$	$b = 0.7$	$b = 0.9$
GNNEXPLAINER	0.203 ± 0.019	0.167 ± 0.039	0.066 ± 0.007
DIR	0.255 ± 0.016	0.247 ± 0.012	0.192 ± 0.044
GSAT	$\mathbf{0.519}^{\dagger} \pm 0.022$	$\mathbf{0.503}^{\dagger} \pm 0.034$	$\mathbf{0.416}^{\dagger} \pm 0.081$
GSAT*	$\mathbf{0.532}^{\dagger} \pm 0.019$	$\mathbf{0.512}^{\dagger} \pm 0.011$	$\mathbf{0.520}^{\dagger} \pm 0.022$

C. Supplementary Experiments

C.1. Details of the Datasets

Mutag (Debnath et al., 1991) is a molecular property prediction dataset, where nodes are atoms and edges are chemical bonds. Each graph is associated with a binary label based on its mutagenic effect. Following (Luo et al., 2020), -NO_2 and -NH_2 in mutagen graphs are labeled as ground-truth explanations.

BA-2Motifs (Luo et al., 2020) is a synthetic dataset, where the base graph is generated by Barabási-Albert (BA) model. Each base graph is attached with a house-like motif or a five-node cycle motif. House motifs and cycle motifs give class labels and thus are regarded as ground-truth explanations for the two classes respectively.

Spurious-Motif (Wu et al., 2022) is a synthetic dataset with three graph classes. Following the notations in (Wu et al., 2022), each graph consists of a base graph (tree/ladder/wheel denoted by $\bar{G}_S = 0, 1, 2$ respectively, with some abuse of notations) and a motif (cycle/house/crane denoted by $G_S = 0, 1, 2$, respectively, with some abuse of notations). The label is determined only by G_S, while there also exists spurious correlation between the label and $\bar{G}_S$. Specifically, to construct a graph in the training set, G_S will be sampled uniformly, while $\bar{G}_S$ will be sampled with probability $\mathbb{P}(\bar{G}_S)$, where $\mathbb{P}(\bar{G}_S) = b$ if $\bar{G}_S = G_S$; otherwise $\mathbb{P}(\bar{G}_S) = (1 - b)/2$. So, b is a parameter used to control the degree of such spurious correlation. When $b = 1/3$, there is no spurious correlation. We include datasets with $b = 0.5$, $b = 0.7$ and $b = 0.9$. Note that for testing data, the motifs and bases are randomly attached to each other, which can test if the model overfits the spurious correlation.

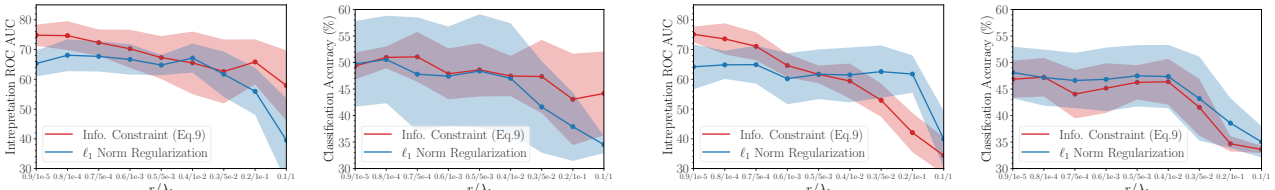
MNIST-75sp (Knyazev et al., 2019) is a image classification dataset, where each image in MNIST is converted to a superpixel graph. Each node in the graph represents a superpixel and edges are formed based on spatial distance between superpixel centers. Node features are the coordinates of their centers of masses. Nodes with nonzero pixel values provide ground-truth explanations. Note that the subgraphs that provide explanations are of different sizes in this dataset.

Graph-SST2 (Socher et al., 2013; Yuan et al., 2020b) is a sentiment analysis dataset, where each text sequence in SST2 is converted to a graph. Each node in the graph represents a word and edges are formed based on relationships between different words. We follow the dataset splits in (Wu et al., 2022) to create degree shifts in the training set, which can better test generalizability of models. Specifically, graphs with higher average node degree will be used to train and validate models, while graphs with fewer nodes will be used to test models. And this dataset contains no ground-truth explanation labels, so we only evaluate prediction performance here and provide interpretation visualizations in Appendix D.

OGBG-Molhiv (Wu et al., 2018; Hu et al., 2020) is a molecular property prediction datasets, where nodes are atoms and edges are chemical bonds. A binary label is assigned to each graph according to whether a molecule inhibits HIV virus replication or not. We also evaluate GSAT on molbace, molbbbp, molclintox, moltox21 and molsider datasets from OGBG. As there are no ground truth explanation labels for these datasets, we only evaluate the prediction performance of GSAT.

Table 8. Ablation study on β and stochasticity in GSAT (PNA as the backbone model) on Spurious-Motif. We report both interpretation ROC AUC (top) and prediction accuracy (bottom).

SPURIOUS-MOTIF	$b = 0.5$	$b = 0.7$	$b = 0.9$
PNA+GSAT	83.34 ± 2.17	86.94 ± 4.05	88.66 ± 2.44
PNA+GSAT-$\beta = 0$	82.01 ± 6.43	78.88 ± 6.74	80.53 ± 5.03
PNA+GSAT-NOSTOCH	79.72 ± 3.86	76.36 ± 2.57	80.21 ± 3.76
PNA+GSAT-NOSTOCH-$\beta = 0$	78.69 ± 10.77	78.97 ± 13.95	79.91 ± 13.11
PNA	68.15 ± 2.39	66.35 ± 3.34	61.40 ± 3.56
PNA+GSAT	68.74 ± 2.24	64.38 ± 3.20	57.01 ± 2.95
PNA+GSAT-$\beta = 0$	59.68 ± 7.28	58.03 ± 11.84	53.94 ± 8.11
PNA+GSAT-NOSTOCH.	51.92 ± 11.17	41.22 ± 7.72	39.56 ± 2.74
PNA+GSAT-NOSTOCH.-$\beta = 0$	56.54 ± 6.88	48.93 ± 10.33	45.82 ± 9.60



(a) Spurious-Motif, $b = 0.7$ (b) Spurious-Motif, $b = 0.9$

Figure 8. Ablation study on (a) using the info. constraint in Eq. (9) and (b) replacing it with ℓ_1-norm, where r is tuned from 0.9 to 0.1 and the coefficient of the ℓ_1-norm λ_1 is tuned from $1e$-5 to 1.

C.2. Details on Hyperparameter Tuning

C.2.1. BACKBONE MODELS

Backbone Architecture. We use a two-layer GIN (Xu et al., 2019) with 64 hidden dimensions and 0.3 dropout ratio. We use the setting from (Corso et al., 2020) for PNA, which has 4 layers with 80 hidden dimensions, 0.3 dropout ratio, and no scalars are used. For OGBG-Mol datasets, we directly follow (Corso et al., 2020) using (mean, min, max, std) aggregators for PNA; yet we find PNA has convergence issues on other datasets when sum aggregator is not used. Hence, PNA uses (mean, min, max, std, sum) aggregators for all other datasets.

Dataset Splits. For Ba-2Motifs, we split it randomly into three sets (80%/10%/10%). For Mutag, we split it randomly into 80%/20% to train and validate models, and following (Luo et al., 2020) we use mutagen molecules with -NO_2 or -NH_2 as test data (because only these samples have explanation labels). For MNIST-75sp, we use the default splits given by (Knyazev et al., 2019); due to its large size in the graph setting, we also reduce the number of training samples following (Wu et al., 2022) to speed up training. For Graph-SST2, Spurious-Motifs and OGBG-Mol, we use the default splits given by (Yuan et al., 2020b) and (Wu et al., 2022). Following (Corso et al., 2020), edge features are not used for all OGBG-Mol datasets.

Epoch. We tune the number of epochs to make sure the convergence of all models. When GIN is used as the backbone model, MNIST-75sp and OGBG-Molhiv are trained for 200 epochs, and all other datasets are trained for 100 epochs. When PNA is used, Mutag and Ba-2Motifs are trained for 50 epochs and all other datasets are trained for 200 epochs. We report the performance of the epoch that achieves the best validation prediction performance and use the models that achieve such best validation performance as the pre-trained models. When multiple epochs achieve the same best performance, we report the one with the lowest validation prediction loss.

Batch Size. All datasets use a batch size of 128; except for MNIST-75sp we use a batch size of 256 to speed up training due to its large size in the graph setting.

Learning Rate. GIN uses 0.003 learning rate for Spurious-Motifs and 0.001 for all other datasets. PNA uses 0.01 learning rate with scheduler following (Corso et al., 2020), 0.003 learning rate for Graph-SST2 and Spurious-Motifs, and 0.001 learning rate for all other datasets.

C.2.2. GSAT

Basic Setting. If not specified, GSAT uses the same settings mentioned for the backbone models. All Spurious-Motif datasets share the same hyperparameters, which are tuned based on $b = 0.5$.

Learning Rate. When PNA is used, GSAT uses 0.001 learning rate for all OGBG-Mol datasets; otherwise it uses the same learning rate as mentioned above.

r in Equation (9). Ba-2Motif and Mutag use $r = 0.5$, and all other datasets use $r = 0.7$. We find $r = 0.7$ can generally provide great performance for all datasets. Inspired by curriculum learning (Bengio et al., 2009), r will initially set to 0.9 and gradually decay to the tuned value. We adopt a step decay, where r will decay 0.1 for every 10 epochs.

β in Equation (8). β is not tuned and is set to $\frac{1}{|E|}$ for all datasets.

Temperature. Temperature used in the Gumbel-softmax trick (Jang et al., 2017) is not tuned, and we use 1 for all datasets.

C.2.3. Baseline Interpretable Methods/Models

Basic Setting. If not specified, baselines use the same settings mentioned for the backbone models. All Spurious-Motif datasets share the same hyperparameters, which are tuned based on $b = 0.5$.

GNNExplainer. We tune the learning rate from $(1, 0.1, 0.01, 0.001)$ and the coefficient of the ℓ_1-norm from $(0.1, 0.01, 0.001)$, based on validation interpretation ROC AUC. The coefficient of the entropy regularization term is set to the recommended value 1. Again, in a real-world setting, post-hoc methods have no clear metric to tune hyper-parameters.

PGExplainer. We use the tuned recommended settings from (Luo et al., 2020), including the temperature, the coefficient of ℓ_1-norm regularization and the coefficient of entropy regularization.

GraphMask. We use the recommended settings from (Schlichtkrull et al., 2021), including the temperature, gamma, zeta and the coefficient of ℓ_0-norm regularization.

DIR. Causal ratio is tuned for Ba-2Motif and Mutag. Since the other datasets we use are the same, we use the recommended settings from (Wu et al., 2022). However, even though datasets are the same, we find the same α specified in their source code do not work well in our setting. Hence, we tune α from $(10, 1, 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001)$.

IB-subgraph. Due to the extreme inefficiency of IB-subgraph, we are only able to tune its mi-weight around the recommended value from $(2, 0.2, 0.02)$. And we use the default inner loop iterations and con-weight as specified in their source code. IB-subgraph needs $\sim$40 hours to train 100 epochs for 1 seed on Spurious-Motif and $\sim$150 hours for OGBG-Molhiv on a Quadro RTX 6000. By contrast, GSAT only needs $\sim$15 minutes to train 100 epochs on OGBG-Molhiv.

Random Seed. All methods are trained with 10 different random seeds; except for IB-subgraph we train it for 5 different random seeds due to its inefficiency. For post-hoc methods, the pre-trained models are also trained with 10 different random seeds instead of a fixed pre-trained model in (Luo et al., 2020). For inherently interpretable models, GSAT, IB-subgraph and DIR, we average the best epoch's performance according to their validation prediction performance. For post-hoc baselines, we average their last epoch's performance. For IB-subgraph, we stop training when there is no improvement over 20 epochs to make the training possible on large datasets.

C.3. Node/Edge Attention

We also explore node-level attention, and we find it is especially useful for molecular datasets and datasets with large graph sizes. Hence, we use node-level attention for on Mutag, MNIST-75sp and OGBG-Mol datasets, and for all other datasets we use edge attention. Specifically, when node attention is used, the MLP layers in $\mathbb{P}_\phi$ will take as input the node embeddings and output p_v for each $v \in V$. Then, the stochastic node attention is sampled for each node $\alpha_v \sim \text{Bern}(p_v)$. After that, α_{uv} is obtained by $\alpha_{uv} = \alpha_u \alpha_v$.

C.4. Further Supplementary Experiments

Fig. 3 shows an experiment with disconnected critical subgraphs, where the dataset is generated in a similar way used to generate Ba-2Motifs. Specifically, each base graph is generated using the BA model and will be attached with two house motifs or three house motifs randomly. The number of house motifs represents the graph class. Both GSAT and GraphMask

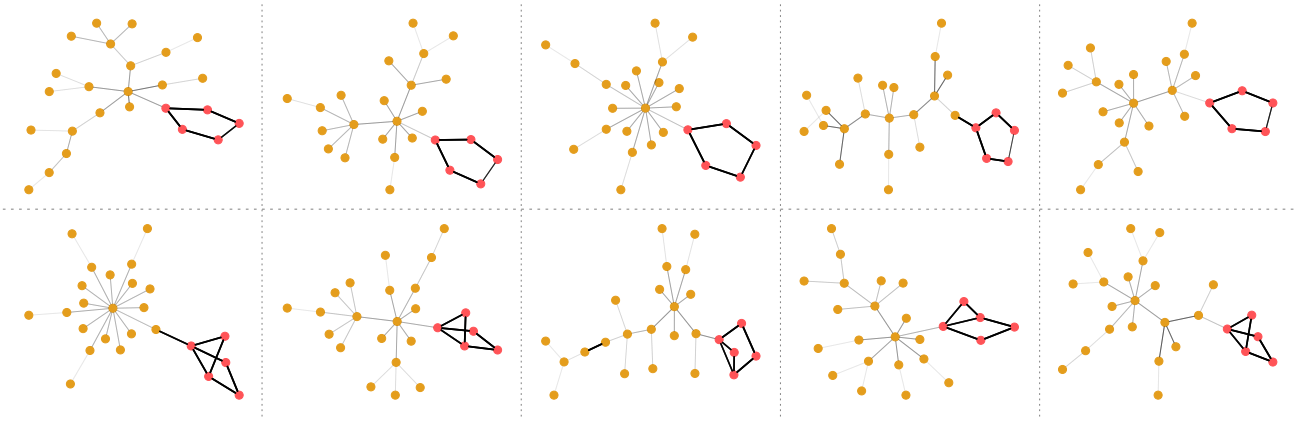


Figure 9. Visualizing label-relevant subgraphs discovered by GSAT for Ba-2Motifs. Nodes colored pink are ground-truth explanations, and each row represents a graph class.

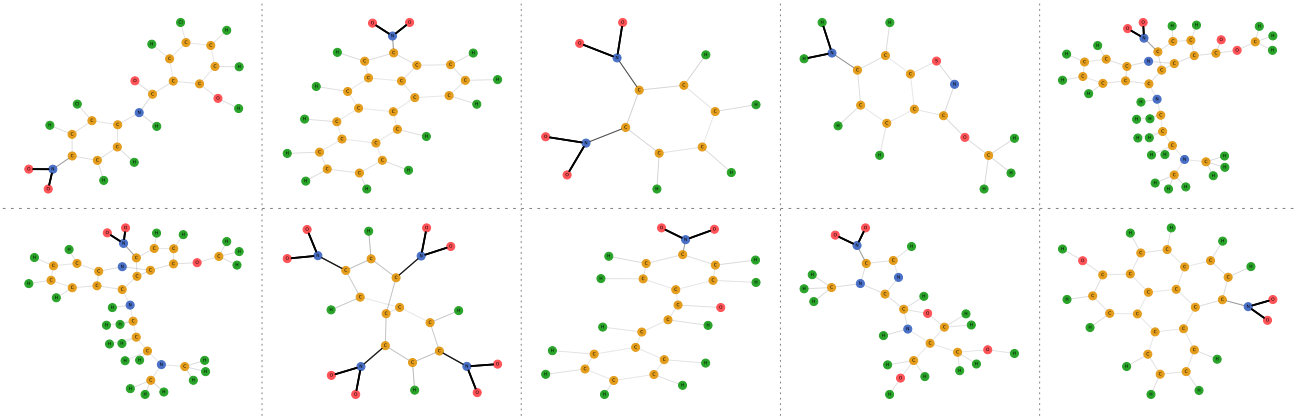


Figure 10. Visualizing label-relevant subgraphs discovered by GSAT for Mutag. -NO_2 and -NH_2 are ground-truth explanations. We only present mutagen graphs as only these graphs are with ground-truth explanation labels.

are trained with the same settings used on Ba-2Motifs.

Table 6 shows a direct comparison with PGExplainer and GNNExplainer between the interpretation ROC AUC reported in (Luo et al., 2020) and the performance of GSAT. And GSAT still outperforms their methods significantly.

Table 4 and Table 7 show direct comparisons with DIR, where we apply GSAT with the backbone model used in DIR. And GSAT still greatly outperforms their method.

Table 8 shows the ablation study on β and stochasticity in GSAT, where PNA is the backbone model. Figure 8 shows the ablation study of the information constraint introduced in Eq. (9) on Spurious-Motif $b = 0.7$ and $b = 0.9$. We observe the same trends from these ablation studies as discussed in Sec. 6.3.

D. Interpretation Visualization

We provide visualizations of the label-relevant subgraphs discovered by GSAT on eight datasets, as shown from Fig. 9 to Fig. 16. The transparency of the edges shown in the figures represents the normalized attention weights learned by GSAT. The normalized attention weights are to rescale the learnt weights $\{p_{uv}|(u,v) \in E\}$ to $[0, 1]$: For each graph, denote $p_{\min} = \min\{p_{uv}|(u,v) \in E\}$ and $p_{\max} = \max\{p_{uv}|(u,v) \in E\}$. We rescale the weights according to

$$\hat{p}_{uv} = \frac{p_{uv} - p_{\min}}{p_{\max} - p_{\min}} \tag{15}$$

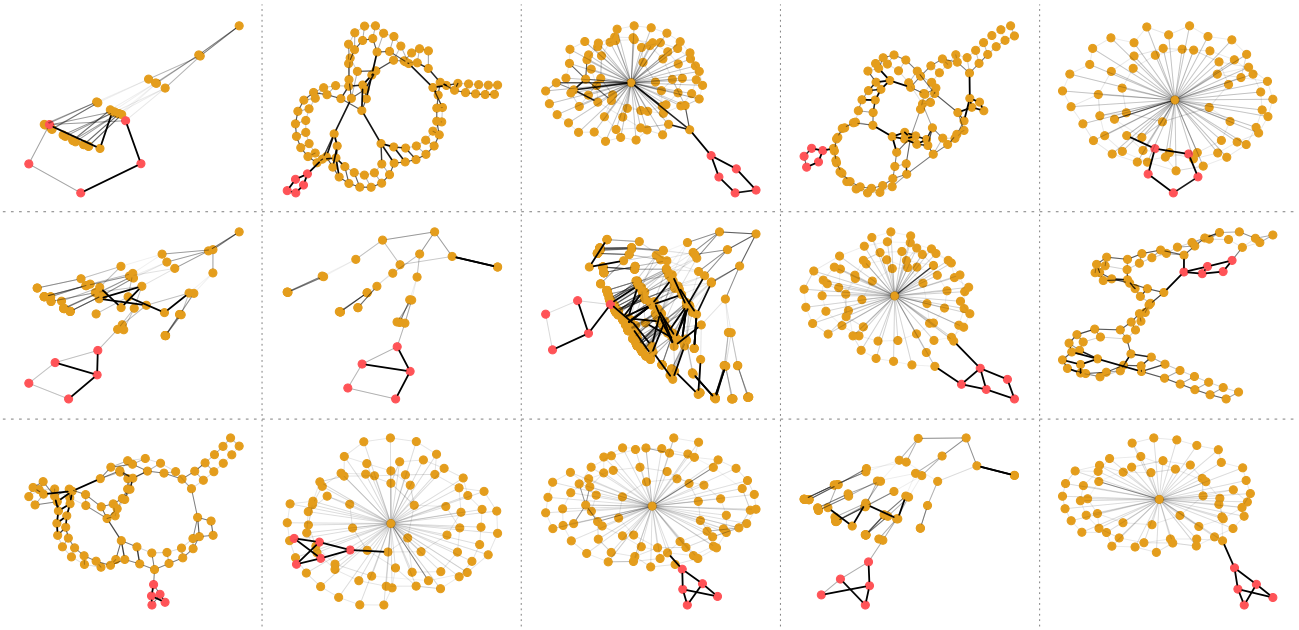


Figure 11. Visualizing label-relevant subgraphs discovered by GSAT for Spurious-Motif $b = 0.5$. Nodes colored pink are ground-truth explanations, and each row represents a graph class.

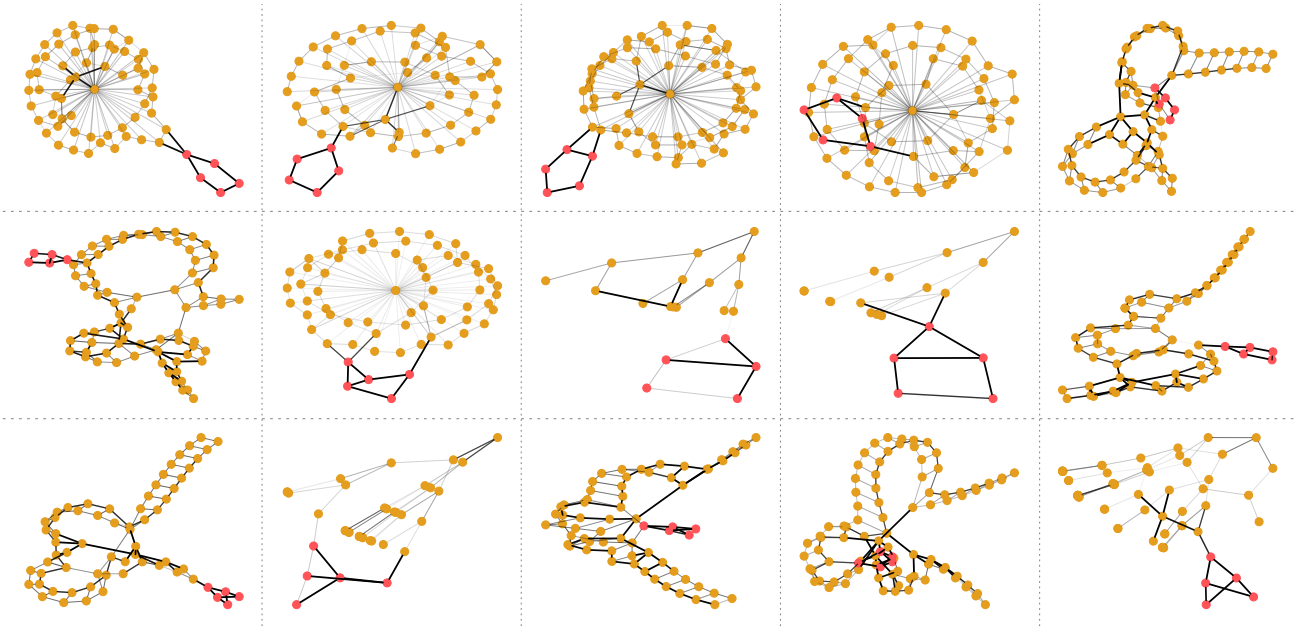


Figure 12. Visualizing label-relevant subgraphs discovered by GSAT for Spurious-Motif $b = 0.7$. Nodes colored pink are ground-truth explanations, and each row represents a graph class.

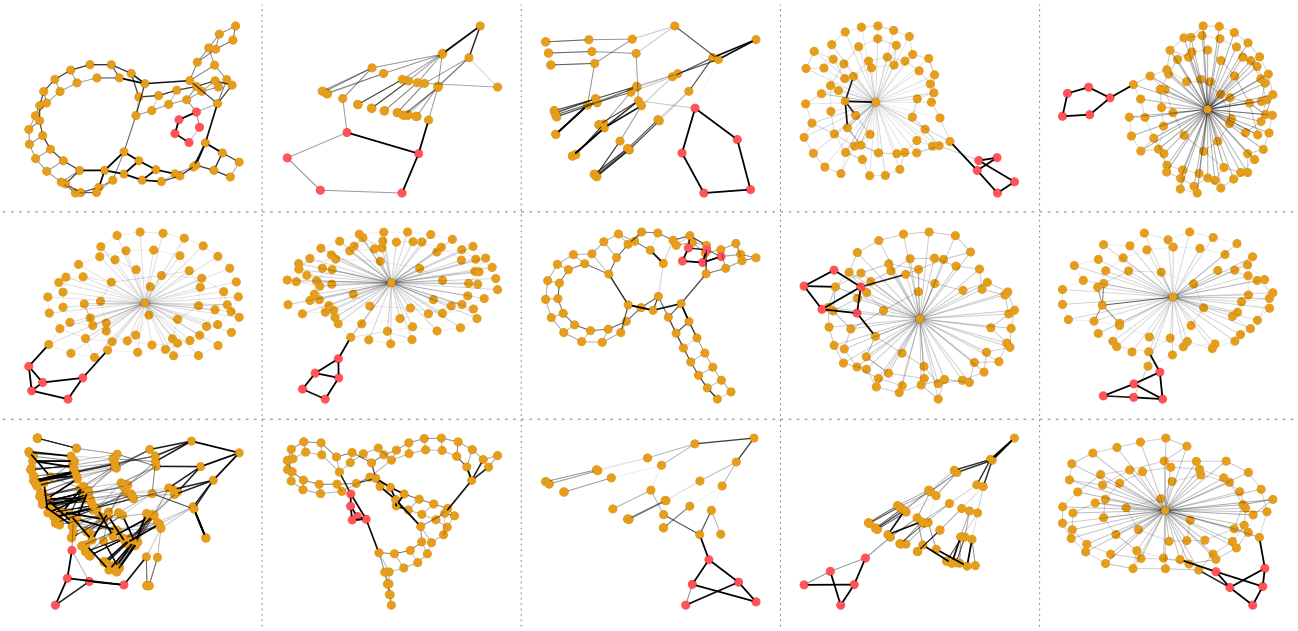


Figure 13. Visualizing label-relevant subgraphs discovered by GSAT for Spurious-Motif $b = 0.9$. Nodes colored pink are ground-truth explanations, and each row represents a graph class.

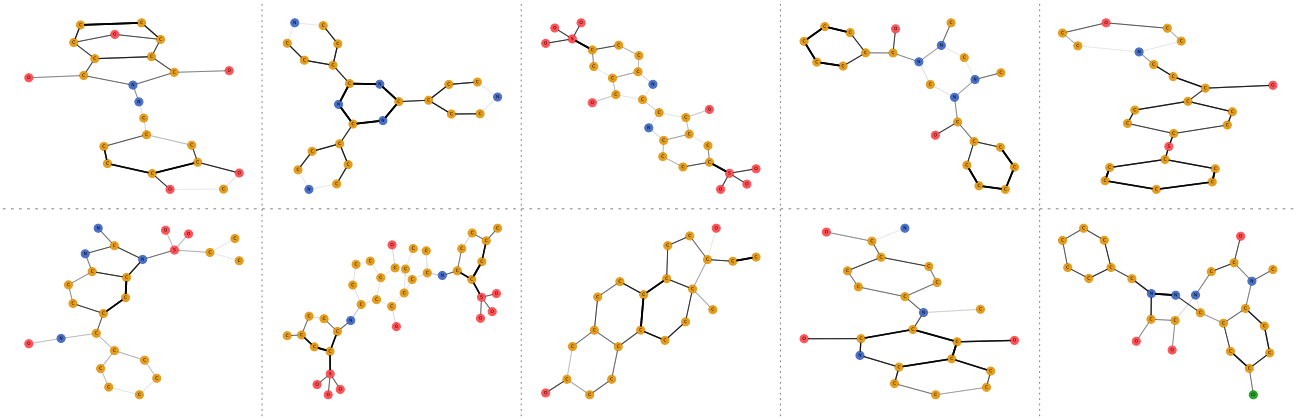


Figure 14. Visualizing label-relevant subgraphs discovered by GSAT for OGBG-Molhiv. Each row represents a graph class.

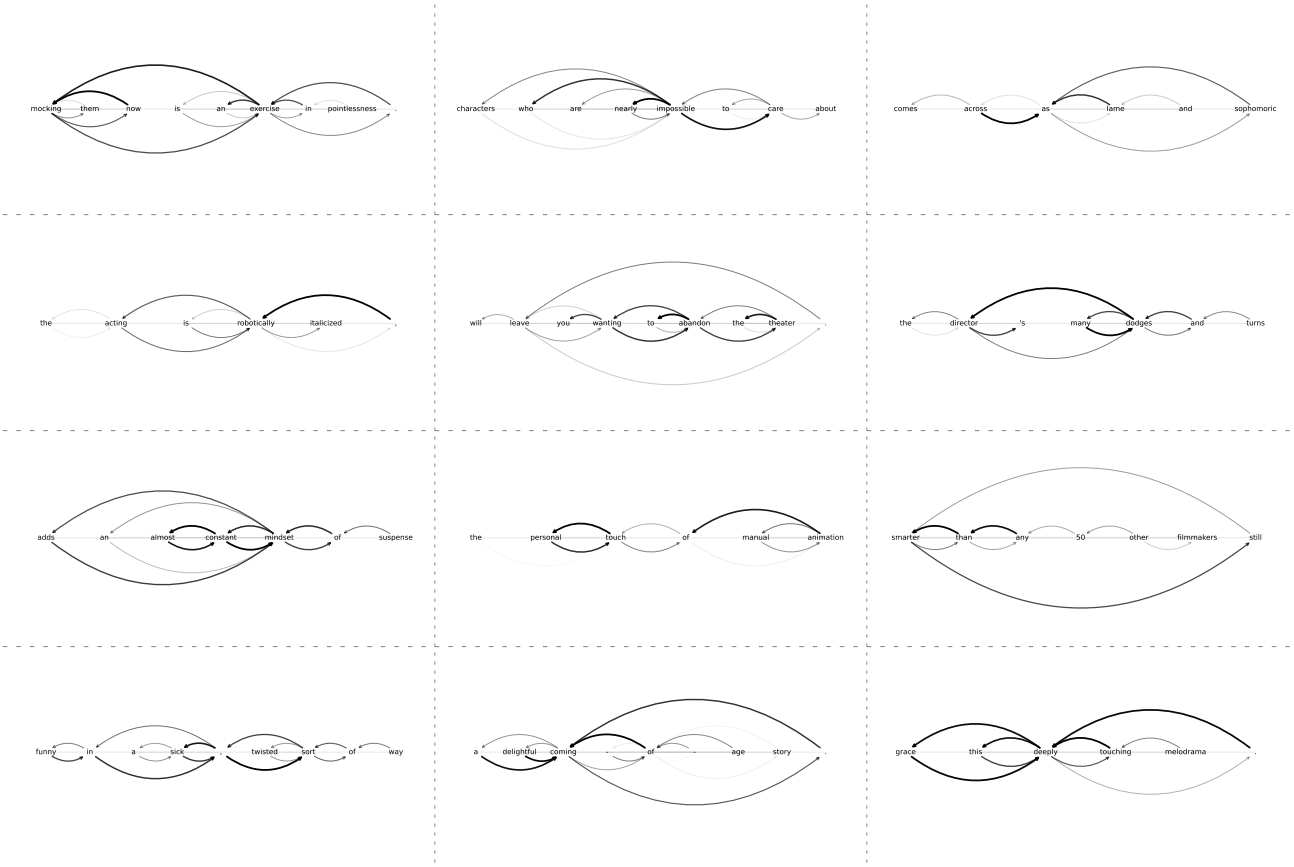


Figure 15. Visualizing label-relevant subgraphs discovered by GSAT for Graph-SST2. The top two rows show sentences with negative sentiment, and the bottom two rows show sentences with positive sentiment.

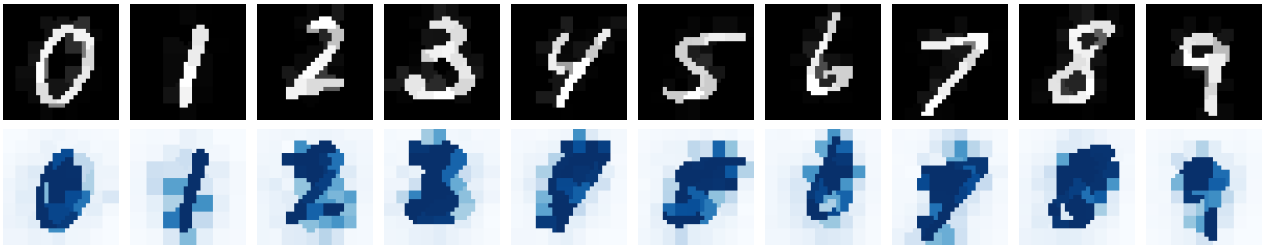


Figure 16. Visualizing label-relevant subgraphs discovered by GSAT for MNIST-75sp. The first row shows the raw images and the second row shows the normalized attention weights learned by GSAT.