

[NeurIPS 2022 Spotlight] Learning Substructure Invariance for Out-of-Distribution Molecular Representations

图分类。

没有环境标签。

总体目标

$$\max_{\omega, \Phi} I(\mathbf{z}; \mathbf{y}), \text{ s.t. } \min_{\omega, \Phi} I(\mathbf{y}; \mathbf{e} | \mathbf{z})$$

$$\text{即 } \max_{q_\theta(\mathbf{y}|\mathbf{z}), q_\theta(\mathbf{z}|\mathbf{G})} I(\mathbf{z}; \mathbf{y}), \text{ s.t. } \min_{q_\theta(\mathbf{y}|\mathbf{z}), q_\theta(\mathbf{z}|\mathbf{G})} I(\mathbf{y}; \mathbf{e} | \mathbf{z}) \quad (1),$$

max项表示希望z有足够的预测能力，min项表示希望z与环境无关。其中 $q_\theta(z|G)$ 和 $q_\theta(y|z)$ 分别表示encoder Φ 和 classifier w 的输出。

$q_\theta(z|G)$ 和 $q_\theta(y|z)$ 二者的具体实现：

1. encoder Φ ：包括一个molecular encoder和substructure encoder，它们二者的分别得到整个分子的表示和子结构的表示，之后在通过一个attentive pooling，拿着molecular encoder的输出当query去找substructure encoder的输出
2. classifier w ：一个MLP.

直接优化上述目标是不可处理的。文章对此的解决是先推了一个ELBO用来优化得到一个环境划分器来获得环境标签e，然后证明了(1)可以用如下的目标来具体优化。

$$\mathcal{L}_{inv}(\theta; \mathcal{G}, \tau) = \underbrace{\frac{1}{|\mathcal{G}|} \sum_{(G, y) \in \mathcal{G}} |\log q_\theta(y|G) - \mathbb{E}_{p(\mathbf{e}|\mathbf{G})} [\log p_\tau(y|G, e)]|}_{\textcircled{1}} + \beta \mathbb{E}_{\mathbf{e}} \left[\underbrace{\frac{1}{|\mathcal{G}^e|} \sum_{(G, y) \in \mathcal{G}^e} [-\log q_\theta(y|G)]}_{\textcircled{2}} \right],$$

①是invariance项 ②是sufficiency项

文章证明了：

1. min ① 等于 $\min_{q_\theta(\mathbf{y}|\mathbf{z}), q_\theta(\mathbf{z}|\mathbf{G})} I(\mathbf{y}; \mathbf{e} | \mathbf{z})$ ，也就是让z在所有测试环境表现一样，即 $p(y|z, e) = p(y|z)$
2. min ② 等于 $\max_{q_\theta(\mathbf{y}|\mathbf{z}), q_\theta(\mathbf{z}|\mathbf{G})} I(\mathbf{z}; \mathbf{y})$ ，也就是让z有足够的预测能力

环境划分器

通过最大化如下的ELBO（等于是在最小化环境分类器和真实后验分布的KL散度

$$D_{KL}(q_\kappa(e | G, y) \| p(e | G)) :$$

$$\mathcal{L}_{elbo}(\tau, \kappa; \mathcal{G}) = \frac{1}{|\mathcal{G}|} \sum_{(G, y) \in \mathcal{G}} [\mathbb{E}_{q_\kappa} [\log p_\tau(y | G, e)] - D_{KL}(q_\kappa(e | G, y) \| p(e | G))]$$

来训练环境划分器 $q_\kappa(e|G, y)$ 和 conditioned-GNN $p_\tau(y|G, e)$

具体实现：

环境分类器 $q_\kappa(e|G, y)$ ：Graph Isomorphism Network (GIN)，接受一个图和标签，产生环境的概率分布

先验分布 $p_\tau(e|G)$: 均匀分布或高斯分布

条件GNN $p_\tau(y|G, e)$: 一个GIN

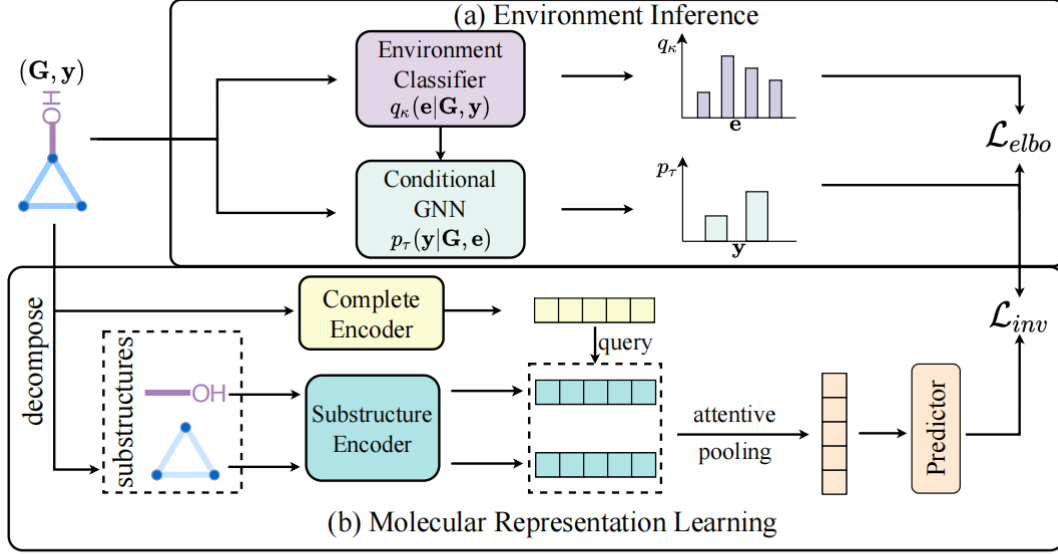


Figure 2: Overview of our model. The whole training procedure is divided into two stages: 1) Optimize the environment-inference model. Given an input molecule (G, y) , we first infer the latent environment variable e . This stage is trained under the guidance of \mathcal{L}_{elbo} . 2) Optimize the molecule encoder and the final predictor guided by \mathcal{L}_{inv} .

训练过程

1) **optimizing the environment-inference model:** $\kappa^*, \tau^* \leftarrow \arg \max_{\kappa, \tau} \mathcal{L}_{elbo}(\tau, \kappa; \mathcal{G}^{train})$.

2) **optimizing the molecule encoder and the predictor:** $\theta^* \leftarrow \arg \min_{\theta} \mathcal{L}_{inv}(\theta; \mathcal{G}^{train}, \tau)$.

Algorithm 1: The training procedure.

Input: Dataset $\mathcal{G}^{train} = \{(G_i, y_i)\}_{i=1}^{N^{train}}$; Number of training epochs for environment inference module E_1 ; Number of training epochs for the molecule encoder and the predictor E_2 ; Batch size B .

Output: Trained parameters θ .

- 1 Initialize parameters θ, τ and κ ;
 - 2 **for** $i \leftarrow 1$ **to** E_1 **do**
 - 3 Sample data batches $\mathcal{B} = \{G_1, G_2, \dots, G_k\}$ from \mathcal{G}^{train} with batch size B ;
 - 4 **for** $j \leftarrow 1$ **to** k **do**
 - 5 Compute batch loss $\mathcal{L}_{elbo}(\tau, \kappa; \mathcal{G}_j)$ according to Eq. 6;
 - 6 Backpropagate $-\mathcal{L}_{elbo}$ and optimize parameters τ, κ ;
 - 7 Freeze the parameters κ, τ ;
 - 8 **for** $i \leftarrow 1$ **to** E_2 **do**
 - 9 Sample data batches $\mathcal{B} = \{G_1, G_2, \dots, G_k\}$ from \mathcal{G}^{train} with batch size B ;
 - 10 **for** $j \leftarrow 1$ **to** k **do**
 - 11 Determine the environment of each sample (G, y) in \mathcal{G}_k by $\arg \max_e q_\kappa(e|G, y)$;
 - 12 Compute batch loss $\mathcal{L}_{inv}(\theta; \mathcal{G}_k, \tau)$ according to Eq. 7;
 - 13 Backpropagate \mathcal{L}_{inv} and optimize parameters θ ;
 - 14 Output the parameters θ ;
-