Global Human-guided Counterfactual Explanations for Molecular Properties via Reinforcement Learning

Danqing Wang*
Carnegie Mellon University
Language Technologies Institute
Pittsburgh, USA
danqingw@andrew.cmu.edu

Antonis Antoniades*
University of California, Santa Barbara
Santa Barbara, USA
antonis@ucsb.edu

Kha-Dinh Luong
University of California, Santa Barbara
Santa Barbara, USA

Edwin Zhang
Harvard University
Cambridge, USA
Founding
Austin, USA

Mert Kosan†
University of California, Santa Barbara
Santa Barbara, USA

Jiachen Li
University of California, Santa Barbara
Santa Barbara, USA

Ambuj Singh
University of California, Santa Barbara
Santa Barbara, USA

William Yang Wang
University of California, Santa Barbara
Santa Barbara, USA

Lei Li
Carnegie Mellon University
Language Technologies Institute
Pittsburgh, USA

ABSTRACT

Counterfactual explanations of Graph Neural Networks (GNNs) offer a powerful way to understand data that can naturally be represented by a graph structure. Furthermore, in many domains, it is highly desirable to derive data-driven global explanations or rules that can better explain the high-level properties of the models and data in question. However, evaluating global counterfactual explanations is hard in real-world datasets due to a lack of human-annotated ground truth, which limits their use in areas like molecular sciences. Additionally, the increasing scale of these datasets provides a challenge for random search-based methods. In this paper, we develop a novel global explanation model RLHEX for molecular property prediction. It aligns the counterfactual explanations with human-defined principles, making the explanations more interpretable and easy for experts to evaluate. RLHEX includes a VAE-based graph generator to generate global explanations and an adapter to adjust the latent representation space to human-defined principles. Optimized by Proximal Policy Optimization (PPO), the global explanations produced by RLHEX cover 4.12% more input graphs and reduce the distance between the counterfactual explanation set and the input set by 0.47% on average across three molecular datasets. RLHEX provides a flexible framework to incorporate different human-designed principles into the counterfactual explanation generation process, aligning these explanations with domain expertise. The code and data are released at https://github.com/dqwang122/RLHEX.

*Both authors contributed equally to this research.
†Work done prior to joining Visa Inc.

KDD '24, August 25–29, 2024, Barcelona, Spain.

ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3672045

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; • **Applied computing** → **Molecular sequence analysis**.

KEYWORDS

Graph Neural Network; Counterfactual Explanation; Reinforcement Learning

ACM Reference Format:
Danqing Wang, Antonis Antoniades, Kha-Dinh Luong, Edwin Zhang, Mert Kosan, Jiachen Li, Ambuj Singh, William Yang Wang, and Lei Li. 2024. Global Human-guided Counterfactual Explanations for Molecular Properties via Reinforcement Learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3637528.3672045

1 INTRODUCTION

Graph Neural Networks (GNNs) have shown promise in fields such as cheminformatics and molecular sciences [10, 45]. A crucial application of GNNs in these fields is molecule property prediction [26, 33, 34], where the task is to predict a molecule's properties based on its structural or functional groups. This is essential for various scientific research aspects, including drug discovery [11], environmental monitoring [27], and materials engineering [6].

However, the intricate complexity of Graph Neural Networks (GNNs) poses challenges in fully leveraging the rich information embedded within the structural properties and feature representations of nodes and edges [46, 53, 54]. Moreover, it is challenging to interpret and understand the underlying rationale behind GNNs' prediction due to non-transparency. This complexity underlines a growing need to understand GNN predictions, particularly with counterfactual (CF) explanations, which present the conditions that need to change to alter the model's decisions [1, 24]. They can highlight the influential sub-graph affecting an individual graph's

prediction (local explanations) [24, 25, 41], or a collection of factors affecting the whole dataset that outlines the model's learned principles (global explanations) [19, 52].

Local explanations are known to be vulnerable to a small noise in the input graph [4] and often fail to generalize to new graphs [25]. Moreover, without ground-truth labels for local explanations in real-world datasets, it is challenging to evaluate the performance on large datasets. For example, Szymanski et al. [40] predicts 2.2 million molecules as crystals, making it feasible to manually check the explanation for each prediction without the ground-truth labels. This underscores a significant need for more comprehensive and global explanations.

Global explainers offer a small set of explanations for the whole input set's behavior. They are more resistant to changes and easier for humans to assess. However, finding global explanations that adequately explain the GNN decision is challenging. Kosan et al. [19] shows it is an NP-hard problem to find the best CF explanation set that can cover as many input graphs as possible regarding the size limit of the set. Additionally, previous CF explainers often use individual nodes, edges, or features as explanations, which are hard for domain experts to verify. For example, Wu et al. [47] states that previous explanations for predicting molecule properties do not match chemists' intuition as these features are not chemically meaningful. Chemists rely on bioisosteres and functional groups to identify the properties of a molecule. Resolving this issue could significantly improve the work of chemists, allowing them to make well-informed decisions based on interpreted results from molecule property prediction.

In response to these challenges, we introduce a global CF explanation model to aid domain experts in understanding the high-level behavior of GNN predictors, which is called **R**einforcement **L**earning via **H**uman-guided **EX**planations (RLHEX). RLHEX first identifies several principles for the ideal global CF explanation to align with the preference of domain experts. For example, the global CF explanations should be chemically valid and cover as many input molecules as possible. RLHEX then employs a Variational Autoencoder (VAE)-based molecule generation model to generate global CF explanations and introduces an adapter to align it with the human-designed principles. Specifically, the adapter learns the policy for aligning the initial molecule latent space with the desired CF explanations' space by leveraging Proximal Policy Optimization (PPO). By sampling from the aligned latent space, the CF explanations generated by RLHEX outperform other strong baselines on three molecular datasets. In summary, our contributions are:

- We propose a novel counterfactual explanation model for GNNs to generate global CF explanations that align with human-designed principles. These concise explanations offer domain experts a better insight into GNNs' prediction rationale.
- By optimizing the latent space via PPO, RLHEX can sample diverse explanations satisfying desirable and diverse principles.
- Experimental results show that RLHEX can achieve the best performance on three real-world molecule datasets: AIDS [32], Mutagenicity [15] and Dipole [30] in terms of several evaluation metrics.

2 RELATED WORK

Counterfactual Explanations of GNNs. Counterfactual reasoning presents a necessary condition that would, if not met, alter the prediction [20, 31]. Studies aimed at providing counterfactual explanations for GNNs can be divided into two categories: *local* and *global* explainers. Local explainers select sub-structures from a given graph that contribute to its GNN's prediction [24, 28, 41], whereas global explainers produce new graphs to illustrate the model behavior across a set of graphs [19, 52]. There are two typical ways to generate counterfactual explanations. One is to perturb nodes and edges of the input graph to get a different prediction. For example, Lucic et al. [24] and Tan et al. [41] learn a mask matrix to select the sub-graph or feature from the input graph, while Kosan et al. [19] and [50] explore different graph edits. The other one is to model it as a generative task. For example, Yuan et al. [52], Numeroso and Bacciu [28] and Ma et al. [25] generates the key pattern for counterfactual explanations directly. However, the explanations created by previous methods are difficult to evaluate without the ground-truth labels. In this paper, we align a graph generative model with human-designed principles to make the explanations more human-friendly.

Graph-based Molecule Generation. Graph neural networks are widely used in 2D molecule tasks [10, 45]. A molecule can be graphically represented with atoms as vertices and chemical bonds as edges. The atom-based generation methods take the atom as the basic generation units [22, 51], while the fragment-based methods build their vocabulary based on the chemical substructure [14, 18, 23]. The fragment-based generation is more likely to produce meaningful molecules with chemically desirable properties, which are reflected in their substructure [9, 47]. Additionally, it can make the edit-based sampling more effective and efficient [48]. In this paper, we use a fragment-based generative model to ensure that the generated global explanations are valid molecules, making them understandable for domain experts.

Align Explainability with Humans. Aligning models to make them helpful and friendly to humans has gained increasing attention recently, especially in large language models [2, 29]. The alignment can be formulated in the reinforcement learning framework to optimize the model policy towards the reward functions based on human values [3] or principles [38, 39]. [21] explores the selective explanations based on what aligns with the recipient's preferences. Xu et al. [49] investigates explainable metrics and aligns the explanation of the mistakes with humans by the failure mode summarized from human feedback. However, few studies have investigated how to align the global interpretation of GNNs with the preferences of domain experts.

3 PRELIMINARIES

3.1 Molecular Property Prediction

A molecule can be intuitively represented as a graph $G = (V, E)$, where $V \in \mathbb{R}^{|V| \times d_v}$ is a set of nodes corresponding to atoms and $E \in \mathbb{R}^{|V| \times |V| \times d_e}$ is a set of edges corresponding to chemical bonds. $|V|$ is the number of atoms. d_v and d_e are the feature dimensions of the atom and the bonds respectively. The functional groups of the

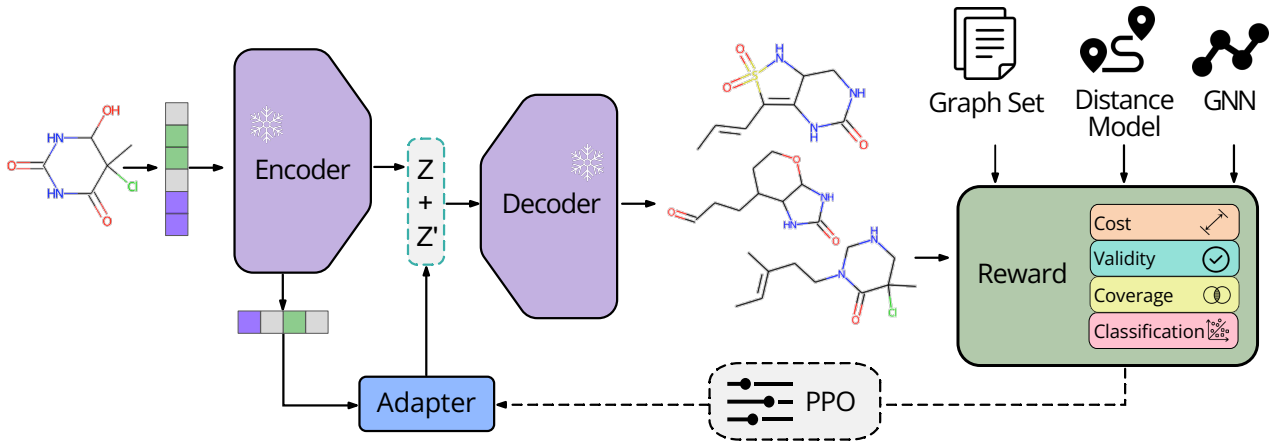

Figure 1: RLHEX has three main parts - the VAE-based generation model, the adapter, and the reward module. The reward module contains several reward functions based on principles designed by humans, which make the generated explanations easier for domain experts to interpret. The adapter modifies the latent representation z by adding the delta z', which is optimized using PPO to align with the principles designed by humans. The RLHEX model uses the molecule to be explained as the input and creates the CF explanations from the modified latent representation $z + z'$.

molecule are often denoted by a subgraph $S = (V_s, E_s)$ of the graph G, which satisfies $V_s \in V$ and $E_s \in E$.

The molecular property prediction can be modeled as a binary graph classification task. It aims to predict whether the input molecule has a certain chemical property, such as whether the molecule is active against AIDS. Given a set of n molecular graphs $\mathbb{G} = \{G_0, G_2, \cdots, G_n\}$ and the ground-truth labels $\{y_0, y_1, \cdots, y_n\}$ where $y_i \in \{0, 1\}$, the GNN classifier $f_\phi(\cdot)$ is trained to predict the estimated label $\hat{y}_i = f_\phi(G_i)$ for each input graph G_i.

Typically, GNN learns the representation of each node $h_v \in \mathbb{R}^d$ by aggregating the information of its neighbors $N(v)$ [10]. d is the dimension of the hidden state. By identifying different aggregation functions $\mathrm{M}_\phi(\cdot)$ and node update functions $\mathrm{U}_\phi(\cdot)$, the update mechanism in each layer l can be denoted as:

$$m_v^l = \sum_{w \in N(v)} \mathrm{M}_\phi(h_v^{l-1}, h_w^{l-1}, e_{vw}) \tag{1}$$

$$h_v^l = \mathrm{U}_\phi(h_v^{l-1}, m_v^l). \tag{2}$$

Here e_{vw} is the feature vector of the edge between node v and w. Finally, a pooling layer is added on top of the last hidden layer L to get the representation of the whole graph:

$$h_G = \mathrm{Pooling}(\{h_v^L | v \in V\}), \tag{3}$$

and a classification head is used to predict binary label $\hat{y}$ for the given chemical property.

3.2 GNN Counterfactual Explanation

The explanation of the GNN classifier is to analyze and interpret how it makes predictions. A local counterfactual explanation (CF explanation) of the GNN classifier $f_\phi(\cdot)$ on the input molecular graph G is defined as an instance C that $f_\phi(C) \neq f_\phi(G)$. Here, C can be the sub-graph of the original G or another graph similar to G. The optimal CF explanation C^* is one that minimizes the distance between G and the CF explanation [24]. Ideally, the optimal CF explanation should be very close to the input graph and have a different prediction. It reveals the minimal perturbation the classifier needs to change its decision.

For the global CF explanation, it aims to provide a set of K instances $\mathbb{C} = \{C_1, C_2, \cdots, C_K\}$ that can explain the global behavior of the classifier. Different from the local CF explanations which is specifically designed for each input molecule, the limited size of the global CF explanation set makes it easier for domain experts to check the classifier behavior on large molecule datasets. For example, given an undesirable molecule property $y = 0$ and a set of molecules $\{G_i | f_\phi(G_i) = 0, G_i \in \mathbb{G}\}$, the global CF explanation set is a set of K instances $\{C_k | f_\phi(C_k) \neq 0, k \in \{0, 1, \cdots, K\}\}$. Kosan et al. [19] proposes that the global CF set should have a high *coverage*, a low *cost* and a small *size*.

3.3 VAE-based Graph Generation

Given the input graph $G = (V, E)$, the variational auto-encoders first embed the graph into continuous latent representation $\boldsymbol{z} \in \mathbb{R}^{d_z}$ by the encoder $p_\varphi(\boldsymbol{z}|G)$. d_z is the dimension of the latent representation. The graph decoder then outputs the graph from the sampled point in the latent space $q_\psi(\hat{G}|\boldsymbol{z})$ [37]. The model is trained by minimizing the training objective:

$$L = -E_{q_\psi(\boldsymbol{z}|G)}[\log p_\varphi(G|\boldsymbol{z})] + \mathrm{KL}(q_\psi(\boldsymbol{z}|G)||p(\boldsymbol{z})). \tag{4}$$

Here, $p(\boldsymbol{z})$ is the prior distribution $\mathcal{N}(0, 1)$. It maximizes the likelihood of the input graph G and regularizes the latent space with the KL divergence.

Encoder The graph neural network is often used as the encoder to map the input graph into the latent representation $\boldsymbol{z}$. The hidden representation of the graph h_G is obtained by Eqn 3. It is then mapped to the μ_G and log variance σ_G of variational posterior approximation $q_\psi(\boldsymbol{z}|G)$ for the reparameterization [17]. The latent representation is sampled from $\mathcal{N}(\mu_G, \sigma_G|G)$.

Decoder Based on how the graph is generated, there are two typical types of graph decoder. One is to pre-define the number of nodes $|V|$ and then predict the node feature matrix $\hat{V}$ and edge matrix $\hat{E}$. The other is to autoregressively generate nodes $P(v_i|v_{<i}, \boldsymbol{z}), i \in \{1, \cdots, |V|\}$ and then predict the edge between the nodes $P(e_{v,w}|\boldsymbol{z})$.

4 METHODOLOGY

In this section, we propose a novel global explanation framework RLHEX to align global CF explanations with human-designed principles. We first discuss several desirable properties for the optimal global CF explanation of molecules. Then we introduce the backbone framework to generate CF candidates and use Proximal Policy Optimization (PPO) to align the candidates with these human principles.

4.1 Principle of Global Molecule Explanation

Inspired by previous studies, we investigate three principles to guide the generation to make the global explanation more understandable and interpretable to domain experts such as chemists.

(1) *The generated explanations should be counterfactual to the input molecules* [31].
(2) *The generated explanations should be valid molecule.* [47].
(3) *The explanation set should be small enough for an expert to manually evaluate while covering as many input molecules as possible* [19].

The first one is the basic principle for CF explanation, requiring the counterfactual explanation C to have a different prediction with the input molecule G: $f_\phi(C) \neq f_\phi(G)$. The second one ensures that the generated explanations are chemically meaningful structures to chemists. This interpretability is more compatible with the domain knowledge and easy for the chemists to understand. For example, the generated graphs should not violate the implicit valence and ring information. The last one is derived from the definition of the optimal local CF explanation. The CF explanation set should be similar to the input molecule so that it can reveal the necessary features the GNN predictor relies on to change their prediction. The size of the explanation should also be small to ensure it is durable for domain experts to check. Here we follow Kosan et al. [19] to introduce three metrics to formally define the requirement: **cov**, **cost**, and **size**. The size is denoted as $|\mathbb{C}|$.

Coverage is a measure of the proportion of input graphs $G \in \mathbb{G}$ can be covered by explanations in $\mathbb{C}$ under a given distance threshold δ:

$$\mathbf{cov}(\mathbb{C}) = \frac{|\{G \in \mathbb{G} | \min_{C \in \mathbb{C}} d(G, C) \leq \delta\}|}{|\mathbb{G}|}, \tag{5}$$

where $d(G, C)$ is the function to calculate the similarity between two graphs, and $|\mathbb{G}|$ indicates the size of the set $\mathbb{G}$. The cost is the mean distance between the input graph set $\mathbb{G}$ and the explanation set $\mathbb{C}$:

$$\mathbf{cost}(\mathbb{C}) = \frac{1}{|\mathbb{G}|} \sum_{i=1}^{|\mathbb{G}|} \min_{C \in \mathbb{C}} d(G, C). \tag{6}$$

4.2 Adapter-enhanced Molecule Generator

To align these human-designed principles with CF generation, we propose Reinforcement Learning via Human-guided EXplanations RLHEX, a flexible CF generation framework for molecules. It formulates the search for an optimal global counterfactual explanation as a graph set generation task. Given the input graph set $\mathbb{G}$ and the GNN classifier $f_\phi(\cdot)$, RLHEX generates a set of graphs $\mathbb{C}$ as its CF explanations.

As shown in Figure 1, RLHEX includes three modules: a VAE-based generation model, an adapter, and a reward module.

- **VAE-based Generation Model**. It aims to sample diverse valid molecules as CF candidates.
- **Adapter** module. It is a parameterized policy π_θ to steer the generator into producing explanations that meet the human-designed principle.
- **Reward Module**. It provides the reward signal based on the human-designed principles to guide the generation. Both heuristic-based and parameterized criteria can be flexibly integrated to meet specific experimental needs or to customize explanations as required.

4.2.1 VAE-based Generation Model. We backbone our model with a fragment-based molecule generation model Principal Subgraph VAE (PSVAE) [18] M_ψ. It first mines principal subgraphs from the molecule datasets and then generates new molecules based on the subgraphs. Essentially, principal subgraphs are frequent and large fragments. The sub-graph-based generation is more interpretable and chemically meaningful, resulting in valid molecules.

PSVAE decomposes one molecule into a set of unordered non-overlapped principal sub-graphs $[F_0, \cdots, F_n]$. n is the number of subgraphs. To generate a new molecule, PSVAE autoregressively predicts chemical sub-graphs $P(F_i|F_{<i}, \boldsymbol{z})$. It then non-autoregressively predicts the inter-subgraph edges e_{vw} via GNN, where v and w are nodes from different sub-graphs. Therefore, the likelihood of the generated molecule can be formulated as:

$$\sum_{i=0}^{n} \log P(F_i|F_{<i}, \boldsymbol{z}) + \sum_{v \in F_i, w \in F_j, i \neq j} \log P(e_{vw}|\boldsymbol{z}). \tag{7}$$

By replacing the first term of Eqn 4 with Eqn 7, we obtain the training objective of PSVAE. We use the pre-trained checkpoint of PSVAE and freeze the parameters in the encoder and decoder.

4.2.2 Latent Distribution Adaptor. To steer the molecule generation model to create desired CF explanations, we add a lightweight adaptor as our parameterized policy π_θ to adjust the latent distribution. The adaptor can either be initialized as a copy of the initial PSVAE, denoted as M_ψ, or as a randomly initialized model, in our case a lightweight transformer encoder [42]. It takes the graph hidden state h_G as the input and maps it to a shift on the mean of the latent distribution $\mu'_G = \pi_\theta(h_G)$. The new latent representation $\boldsymbol{z'}$ is sample from the distribution $\mathcal{N}(\mu_G + \mu'_G, \sigma_G)$. The decoder takes z' as the input to generate $q_\psi(G|\boldsymbol{z'})$. The complete generative process, starting from the input molecule G_t and ending at the

explanation molecule C_t can be formalized as follows:

$$\mu_{G_t}, \sigma_{G_t} = M_\psi^{(E)}(G_t) \tag{8}$$

$$\mu'_G = \mu_G + \pi_\theta(h_{G_t}) \tag{9}$$

$$z' \sim \mathcal{N}(\mu_{G_t} + \mu'_{G_t}, \sigma_{G_t}) \tag{10}$$

$$C_t = M_\psi^{(D)}(z'), \tag{11}$$

where and $M_\psi^{(E)}, M_\psi^{(D)}$ stand for the respective encoder and decoder of the PSVAE. This framework allows for the sequential generation of molecules, with each step informed by the previous state, thus enabling a guided exploration of the molecular space that is coherent with the desired properties encoded by the policy π_θ.

4.2.3 Principle Modeling. We design our reward module based on the principles in Section 4.1. For (1), we take the prediction probability of the opposite class as the reward. For example, if the input molecules are predicted as negative $p\left(f_\phi(G) = 0\right)$, we take the probability of $p\left(f_\phi(C) = 1\right)$ as the reward. We use $p_\phi(C)$ for short. For validity in (2), we use the Rdkit library [1] to build an indicator function $\mathbb{I}(\cdot)$ as the reward signal [2]. If the molecule is valid, $\mathbb{I}(C) = 1$, otherwise, $\mathbb{I}(C) = 0$.

The objective of (3) can be written as:

$$\max_{\mathbb{C}} \mathbf{cov}(\mathbb{C}) \quad s.t. |\mathbb{C}| = k. \tag{12}$$

Here, the size of $\mathbb{C}$ is limited by k, and the cost is constrained based on the threshold δ in coverage. In practice, we first maximize the local reward for each input molecule G and get a set of CF explanation $\mathbb{C}$. Then we greedily select top-k explanations from the candidate set as $\mathbb{C}_k$. The local reward $s(C)$ for each candidate C is defined as:

$$\mathbf{score}(C) = \mathbb{I}(C) * [\alpha p_\phi(C) + \beta \mathbf{cov}(C)] \tag{13}$$

Here α, β are the coefficients of the prediction probability and the local coverage. If we use $R(\mathbb{C})$ to indicate the sum of local reward on the set $\mathbb{C}$, the optimal global score of the CF explanation set with size k can be written as:

$$R^*(\mathbb{C}_k) = \max_{\mathbb{C}_k} \sum_{C \in \mathbb{C}_k} \mathbf{score}(C), \quad s.t. |\mathbb{C}_k| = k. \tag{14}$$

4.3 Tailor Latent Distribution via PPO

We formulate the alignment to the human-designed principles as a Markov decision process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, p)$, with state space $\mathcal{S}$, action space $\mathcal{A}$, reward function $\mathcal{R}$, and transition probability matrix p. The state $s \in \mathcal{S}$ is the CF candidates, which are all molecules. The action is the modification of the latent distribution $\mu'_G \in \mathcal{A} \subseteq \mathbb{R}^{d_z}$. d_z is the dimension of the latent space. $p(\cdot|s,a) : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ indicates the probability of CF candidates based on the new latent distribution. The reward $r \in \mathcal{R}$ is the local reward function defined in Eqn 13.

At each time step t, RLHEX employs the VAE encoder to get the embedding of the input molecule G, which is the state s_t. It then uses $\pi_\theta(s_t, a_t)$ to sample the mean shift of the latent distribution $\mu'(G)$. The reward $r(s_t, a_t)$ is based on the scores from the human-designed principle (Eqn 13). Our goal is to learn a policy $a \sim \pi_\theta(s)$ that can find the optimal latent distribution for the CF explanations.

We leverage Proximal Policy Optimization (PPO) [36] to generate molecules that satisfy different principles. It has been extensively used to steer models into producing human-desired outputs in language models, using RLHF (Reinforcement Learning from Human Feedback) [7, 29, 55], and we take inspiration from these methods to build a more flexible system within which many different forms of human-guided principles can be used to optimize our explanation model.

PPO operates by optimizing an objective that balances exploration and exploitation of the current policy π_{old}, to gain more reward and explore new policies π_θ. This balance is achieved through clipped probability ratios, ensuring that updates do not deviate too far from the current policy. The standard PPO objective is the following:

$$L(\theta) = \mathop{\mathbb{E}}_{\substack{s_t \sim p(\cdot|s_{t-1}, a_{t-1}) \\ a_t \sim \pi_{old}(\cdot|s_{t-1})}} \left[\min \left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\text{old}}(a_t|s_t)} \hat{A}(s_t, a_t), \right. \right. \tag{15}$$

$$\left. \left. \text{clip}\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\text{old}}(a_t|s_t)}, 1-\epsilon, 1+\epsilon \right) \hat{A}(s_t, a_t) \right) \right],$$

where ϵ is a hyperparameter that defines the clipping range, and $\mathbb{E}_{s_t, a_t}$ represents the expectation for an on-policy batch sample. We train an additional critic model $V(s)$ to estimate the actual reward of the current state and action $Q(s_t, a_t)$ during the training. Here $Q(s_t, a_t)$ is the expected local rewards of the CF candidates sampled from the new latent distribution: $Q(s_t, a_t) = \mathbb{E}_{C \sim p(\cdot|s_t, a_t)}[\mathbf{score}(C)]$. $\mathbf{score}(C)$ is from Eqn 13. $\hat{A}(s_t, a_t)$ denotes the advantage function, which is defined as $A(s,a) = Q(s,a) - V(s)$.

In our case, the property of a constrained policy update is essential to our goal of keeping newly generated molecules close in distribution to the original molecule distribution to ensure their validity.

Algorithm 1 RLHEX Inference

The input molecule set $\mathbb{G}$, the encoder $M_\psi^{(E)}$ and the decoder $M_\psi^{(D)}$, the policy π_θ, the CF set $\mathbb{C}$
$\mathbb{C} \leftarrow \emptyset$
for $G \in \mathbb{G}$ **do**
  for $t \in 1 : T$ **do**
    Map G to the latent distribution by $M_\psi^{(E)}$ and π_θ via Eqn 8 and 9
    Sample $z' \leftarrow \mathcal{N}(\mu_G + \mu'_G, \sigma_G)$ via Eqn 10
    Get one candidate C by $M_\psi^{(D)}$ via Eqn 11
    $G \leftarrow C$
  $\mathbb{C} \leftarrow \mathbb{C} + \{C\}$
for $i \in 1 : k$ **do**
  $C \leftarrow \text{argmax}_{C \in \mathbb{C}} \Delta r(C; \mathbb{C}_{k-1})$
  $\mathbb{C}_k \leftarrow \mathbb{C}_k + \{C\}$

[1] https://www.rdkit.org/

[2] *rdkit.Chem.detectChemistryProblems* is used to check molecule validity. It inspects molecular properties such as atom valence. Experts can easily add more constraints such as QED [5] by replacing this function.

4.4 Greedy Selection

For each molecule G in $\mathbb{G}$, we apply RLHEX to optimize the local reward $\textbf{score}(C)$ in Eqn 13 and get the CF candidate set $\mathbb{C}$. To get an optimal CF candidate set $\mathbb{C}_k$ with size k, we greedily choose the top-k candidates from $\mathbb{C}$. Start from an empty set $\mathbb{C}_0$, we add the candidate with the maximum gain, which is defined as:

$$\textbf{Gain}(C; \mathbb{C}) = R(\mathbb{C} + C) - R(\mathbb{C}). \tag{16}$$

The detailed phase is described in Alg. 1.

5 EXPERIMENT

5.1 Datasets

We focus on molecule property prediction and conduct our experiments on three real-world molecule datasets: AIDS [32], Mutagenicity [15] and Dipole [30]. AIDS and Mutagenicity have been used in previous works [19, 43], however, qualitative evaluation of the counterfactuals generated for these tasks is challenging without domain expertise. For that reason, we seek to formulate a task in which the chemical characteristics can be quickly evaluated by observing the structure of the generated graphs. AIDS is a binary dataset where the label=0 indicates the molecule is active against AIDS. The activity is the desirable attribution for molecules, so we flip the label to make the negative class correspond to the undesirable property. The mutagenicity dataset classifies molecules by whether they are mutagenetic and labels the mutagenetic with label 0. Dipole is a binary classification dataset we curated from a subset of molecules reported by Pereira and Aires-de Sousa [30], in which the dipole moment of each molecule is recorded. In particular, we extract a subset of the most polar molecules to form the positive class and a subset of the least polar molecules to form the negative class. Polarity is a comparably simpler chemical property to assess from only the molecular structure. Following previous work, we keep atom types that appear at least 50 times in the dataset, resulting in 9 common atoms in AIDS and Dipole and 10 in Mutagenicity. [3] For the graph dataset AIDS and Mutagenicity, we convert the graph representation to SMILES and remove the duplicated instances.

We randomly split the dataset by 0.8:0.1:0.1 for training, validation and testing. We follow Kosan et al. [19] to train separate GNN-based predictors $f_\phi(\cdot)$ on these datasets. We use 3 convolution layers as the aggregation function and add the message to the previous node representation for update. One max pooling layer is used to get the graph representation h_G from the node representation and a full-connected layer is added on top of it for classification. The model is trained with the Adam optimizer [16] and a learning rate of 0.001 for 1000 epochs. Detailed information is listed in Table 1.

5.2 Baselines

Since most CF explanation methods for GNNs focus on local explanations, which are not directly comparable, we present the state-of-the-art global explanation method **GCFExplainer** [19] and elaborate two sampling-based generative baselines. **GCFExplainer** functions employ vertex-reinforced random walks on an edit map of graphs and a greedy summary to deliver high-coverage, low-cost candidate sets for input graphs. The maximum steps of the random walk is set to 6000. Two non-RL generative baselines are tested: **PSVAE** and **PSVAE-SA**. PSVAE iteratively encodes the input molecule to the latent space and samples from the latent space to generate candidates. We terminate the sampling process when its generation has a different prediction from the input molecule, or when it arrives at the maximum iteration. PSVAE-SA applies simulated annealing to optimize the sampling process toward the reward function. For each iteration, it samples from the latent representation of the input molecule and accepts the new generation based on the Metropolis criterion:

$$A_t = \min(1, e^{\frac{\textbf{score}_t - \textbf{score}_{t-1}}{T}}), \tag{17}$$

Here, $\textbf{score}_t$ represents the local reward at step t defined in Eqn 13, while T is temperature, initially set at 0.1 and halved every 10 steps. We follow the standard settings of other hyperparameters in original papers. We check the validity of the generated candidate explanations and only keep valid molecules with opposite predictions for the final candidate set.

Table 1: Dataset Statistic. Here # indicates the number size. We train the GNN classifiers to predict the molecule property. We ignore atoms that appear less than 50 times in the dataset and filter the duplicated molecules by SMILES representation.

	AIDS	Mutagenicity	Dipole
#Graphs	1562	3461	3539
#Nodes per graph	15.73	30.34	9.16
#Edges per graph	16.32	30.80	18.53
#Atom Type	9	10	9
#GNN Accuracy	97.81%	80.00%	89.37%

5.3 Implementation and Evaluation

We initialize the encoder and decoder of RLHEX based on PSVAE [18]. We use the released checkpoint trained on ZINC250K [13] and freeze the parameters. Note that our method does not limit us to a specific architecture, allowing for the use of other models for molecule generation purposes. We set the dimension of the latent representation space as 56 and the hidden size of the adaptor to 400 for both the policy model and the critic model. We use the Adam optimizer for training and set the learning rate to 1e-5. We use linear warmup and decay the learning rate to 1/10 of the maximum. To facilitate good explorations vs. exploitation strategy, we employ Upper Confidence Bound sampling Wang et al. [44], which preferably samples input molecules according to the mean and variance of the scores they yield over episodes.

We follow previous studies to use Tanimoto similarity to calculate the distance between molecules [28, 46]. It is a commonly used similarity function to compare chemical structures based on fingerprints. We set the distance threshold δ as 0.87 based on the dataset distribution.

We evaluate the model performance based on the coverage and cost described in Section 4.2.3. We set the coefficient in Eqn 13 to $\alpha = 1, \beta = 10$ to balance the scale of the prediction probability and

[3] We further remove molecules with atom *Na* because it is not a common atom in ZINC250K pre-trained dataset [13].

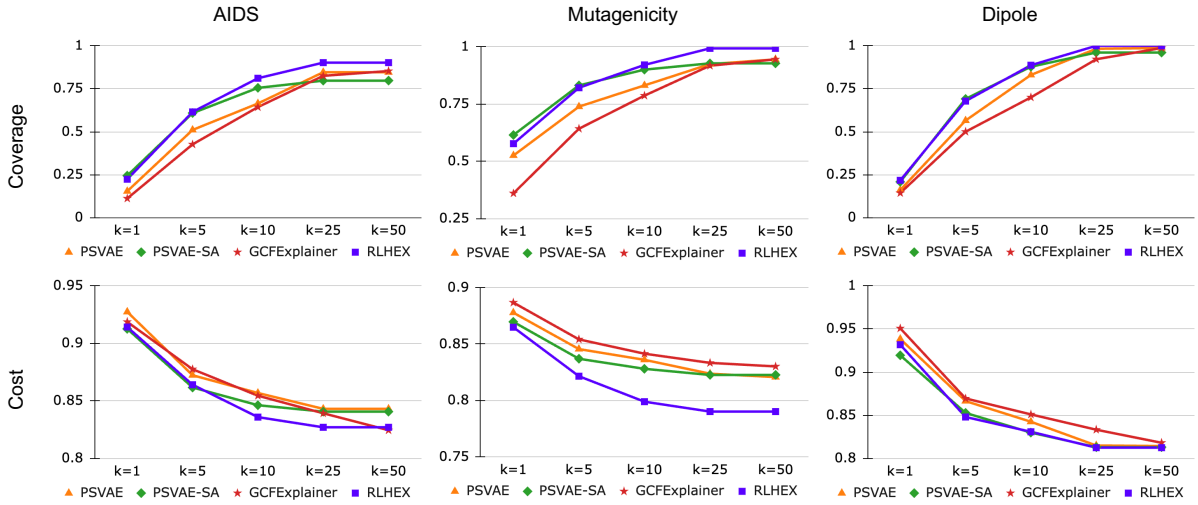

Figure 2: Coverage and Cost with different size k of explanation set. RLHEX generally outperforms the baselines on coverage and cost. We use iteration $i = 20$ for generation.

Table 2: Coverage measures the percentage of the input molecules covered by the top-10 counterfactual explanations. RLHEX achieves the highest coverage on three datasets.

Coverage ↑	AIDS	Mutagenicity	Dipole
GCFExplainer	0.647±0.016	0.806±0.043	0.762±0.020
PSVAE	0.682±0.030	0.850±0.023	0.836±0.018
PSVAE-SA	0.761±0.024	0.894±0.011	0.872±0.010
RLHEX	**0.822**±0.021	**0.909**±0.024	**0.896**±0.017

Table 3: Cost indicates the minimum distance between the input set and the top-10 explanation set. RLHEX creates counterfactual explanations that are more similar to the input graphs but have different GNN predictions.

Cost ↓	AIDS	Mutagenicity	Dipole
GCFExplainer	0.853±0.008	0.917±0.046	0.964±0.068
PSVAE	0.854±0.003	0.836±0.004	0.839±0.005
PSVAE-SA	0.847±0.003	0.816±0.008	**0.832**±0.002
RLHEX	**0.837**±0.001	**0.813**±0.006	0.833±0.004

the individual coverage. PSVAE-based baselines maintain a beam size of 10 and a temperature of 1 for decoding. After getting the counterfactual candidate set, we use the greedy algorithm to select top-k counterfactual explanations as described in Alg. 1. The main results come from sampling $T = 20$ iterations per input molecule. We use $k = 10$ for the main experiment.

5.4 Main Results

Table 2 and 3 show the coverage and cost on the test set of three datasets with $k = 10$ after 20 iterations. We ran the experiments 5 times with different random seeds and calculated the average and standard deviations. Our method, RLHEX, performs better than the best baseline model, PSVAE-SA, with a gain of 8% increase in coverage and a cost reduction of 1.23% on AIDS. It also performs best on Mutagenicity in terms of coverage and cost. Dipole has the highest coverage with a similar cost. It means that the explanations from RLHEX are similar to the input molecules with different GNN predictions. This helps people understand the GNN predictor's behaviors on the whole input dataset with limited molecules.

Figure 2 shows how the candidate set size k impacts performance. RLHEX has the highest coverage with different k. When k increases, the performance gap grows. All methods reach their maximum coverage on the input graph set after $k = 25$. The candidate set covers almost the entire input graph set. Two generative baselines, PSVAE and PSVAE-SA, do better than GCFExplainer when k is small. But GCFExplainer reduces the performance gap as the number of candidates grows.

In general, RLHEX outperforms other baselines in cost. On AIDS and Dipole, GCFExplainer can match the input set at a lower cost than PSVAE and PSVAE-SA with $k = 50$. This is similar to RLHEX. However, on Mutagenicity, our method RLHEX has a clear edge over the other baselines for all k. PSVAE-SA has a small cost at $k = 1$ on three datasets while it lags when k grows larger. This means that the simulated annealing starting from individual input is good at finding the local optima explanation but not at the global optima. However, RLHEX can find a better global explanation set closer to the input set for different k.

5.5 Analysis

Ablation Studies In Table 4, we further investigate the performance of our model with several ablation studies. These studies delve deeper into the fundamental components of the RLHEX model and their contributions to the model's overall effectiveness. To better understand the role of the PSVAE in RLHEX, we consider a variant of RLHEX, namely RLHEX w/o PSVAE. This variant starts

Table 4: Ablation Study on RLHEX. Cov. inidated Coverage. All experiments are based on $i = 20$, $k = 10$ and beam size 10.

	AIDS		Mutagenicity		Dipole	
	Cov. ↑	Cost ↓	Cov. ↑	Cost ↓	Cov. ↑	Cost ↓
RLHEX	0.811	0.836	0.914	0.813	0.887	0.831
w/o PSVAE	0.168	0.923	0.900	0.799	0.713	0.846
w/o Adapter	0.615	0.857	0.887	0.791	0.817	0.833

with randomly initialized parameters and is trained from scratch. Noticeably, the absence of the pre-trained PSVAE results in a significant reduction in coverage on the AIDS and Dipole datasets. Nonetheless, it is still able to generate valid explanations because of the subgraph-based generation method. We also study the influence of the trained adapter by examining a version of RLHEX (referred to as RLHEX w/o trained Adaptor) that includes a randomly initialized adapter without any further training. The adapter in this context is directly used for inference with a random shift on the latent space. The results highlight that the removal of the adapter leads to decreases in coverage and an increase in cost. This is primarily due to the alignment mismatch with the human-designed principles.

RLHEX achieves the highest coverage after one iteration Figure 3 illustrates how the performance is impacted by the number of iterations during the inference process. For the PSVAE-based model, an iteration is defined as one pass over the set of input graphs, $\mathbb{G}$. For GCFExplainer, the maximum step of the random walk is defined as $i * |\mathbb{G}|$, where i denotes the iteration number. As shown in Figure 3, RLHEX reaches optimal performance after one iteration and maintains a stable performance after 20 iterations. However, GCFExplainer shows the best performance at the first iteration. As the iteration number increases, other models start to outperform GCFExplainer. This performance decrease can be attributed to GCFExplainer's process of exploring every possible perturbation on nodes and edges for the current molecule at each step, which includes investigating up to 100,000 neighbors. In comparison, PSVAE-based methods explore 10 candidates (beam size = 10) for each input graph. Consequently, GCFExplainer's large search space at each step makes it more effective when the iteration number is small. However, as the number of iterations increases, its performance deteriorates due to its inefficient exploration strategy.

Generated CF explanations are more interpretable Figure 4 displays two cases produced by RLHEX, specifically focusing on the AIDS and Dipole datasets. In the AIDS dataset, the GNN predictor classifies the input graphs as negative, or label=0, indicating that these molecules are inactive against AIDS. Conversely, in the Dipole dataset, the input molecules are deemed non-polar, while the CF explanation shows them as polar. As seen in Figure 4, the CF explanation closely resembles these input molecules, as they share several common chemical sub-graphs. This similarity allows RLHEX to encapsulate the behavior of the GNN predictor across various input molecules by grouping multiple negative molecules together. It further uncovers the necessary conditions that would prompt the GNN predictor to alter its prediction. Another highlight of our work is that by comparing the varying sub-graphs between the CF explanation and the covered input molecules, domain experts can more easily identify which functional group the GNN predictor uses to make its decisions. This feature makes RLHEX a valuable tool for enhancing understanding and facilitating decision-making in chemists' work.

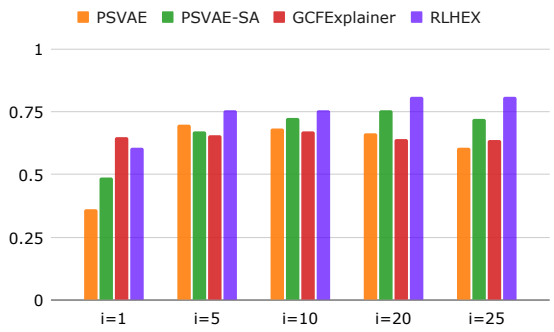

Figure 3: Coverage on AIDS dataset with different iterations i. We limit the explanation set with $k = 10$ to calculate the coverage.

5.6 Expert Assessment on CF Explanation

We also engaged molecular chemists to evaluate our CF explanations. Although the evaluations lacked empirical laboratory testing, the expert feedback was in general alignment with the explanations about specific classes of molecules. Through this procedure, the chemists could assess the efficacy of the GNN classifier through its alignment with known chemical knowledge. Based on the case shown in Figure 4, chemists made the following observations:

AIDS The CF candidate is predicted to be active against AIDS. Although the input negative molecules also have the fused aromatic rings and hydroxyl (-OH) groups that could potentially form important interactions with viral targets, the fused 3-ring system of the CF candidate increases its possibility of being against AIDS. The fused 3-ring system resembles known HIV integrase inhibitor pharmacophores [8].

Dipole The CF candidate and the covered input molecules have O-H and N-H bonds, which are polar due to the electronegativity difference between oxygen and hydrogen[35]. However, the CF candidate has a bent geometry, which allows the bond dipoles to add up and create a net molecular dipole [12].

6 CONCLUSION AND DISCUSSION

In this paper, we introduced RLHEX, a global counterfactual explanation method that strives to aid domain experts to better understand GNN predictions. Our proposed model aligns with the domain experts' criteria and uses Proximal Policy Optimization (PPO) to generate chemically valid explanations that can cover the highest number of input molecules. Importantly, our method takes into account the interpretability requirement of domain experts, an aspect often overlooked in CF explanations, making it suitable for understanding molecular property predictions. Experimental results show that RLHEX outperforms other strong baselines on three real-world molecular datasets.

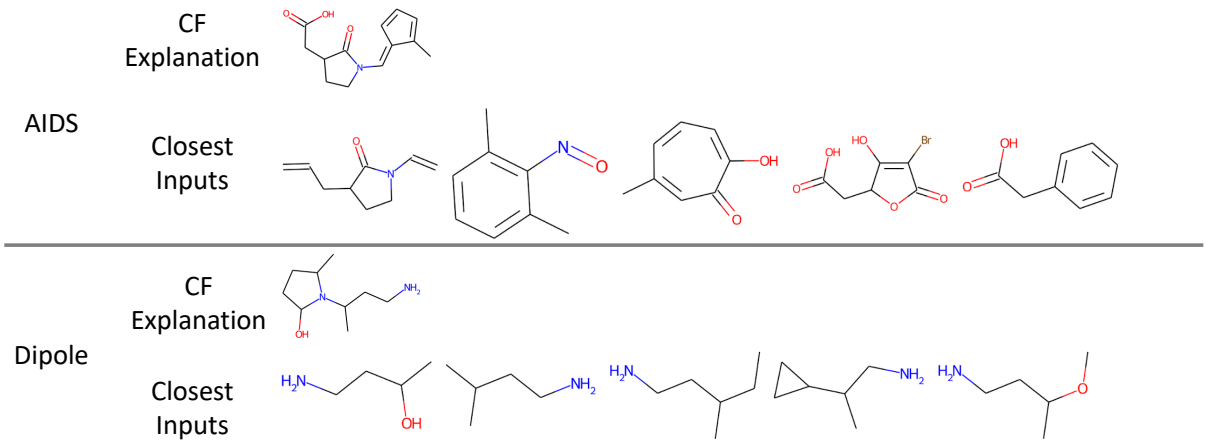

Figure 4: The counterfactual (CF) explanation generated for the closest input molecules from the AIDS and Dipole datasets. For each CF explanation, we compute the distance between it and the input molecules, selecting the top 5 input molecules for display. The generated CF explanation for AIDS exhibits a coverage of 0.231 over the input molecule set, while the CF explanation for the Dipole dataset shows a coverage of 0.209.

Further work can focus on the scalability of our method. We plan to investigate RLHEX's performance on larger datasets and more complex molecular structures. Although the primary application of our method is in cheminformatics, we anticipate that the underlying principle of RLHEX could be broadly applied across various fields requiring interpretability in the use of GNNs. Furthermore, more human-designed principles can be flexibly integrated into RLHEX to align different preferences of domain experts.

To conclude, RLHEX represents a significant step towards creating more interpretable and understandable GNN models. By generating global counterfactual explanations through human-aligned principles, RLHEX offers a promising avenue for domain experts to better utilize and comprehend the findings of GNN predictions, particularly in the evolving generative landscape. Through our work, we aspire to bridge the gap between the scale and complexity of sequence-based prediction models and the intuitiveness required by experts to make new scientific discoveries.

ACKNOWLEDGEMENT

We gratefully acknowledge the support of the National Science Foundation (# 2229876) for funding this research.

REFERENCES

[1] Carlo Abrate and Francesco Bonchi. 2021. Counterfactual graphs for explainable classification of brain networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2495–2504.
[2] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862* (2022).
[3] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* (2022).
[4] Mohit Bajaj, Lingyang Chu, Zi Yu Xue, Jian Pei, Lanjun Wang, Peter Cho-Ho Lam, and Yong Zhang. 2021. Robust counterfactual explanations on graph neural networks. *Advances in Neural Information Processing Systems* 34 (2021), 5644–5655.
[5] G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. 2012. Quantifying the chemical beauty of drugs. *Nature chemistry* 4, 2 (2012), 90–98.
[6] Keith T Butler, Daniel W Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. 2018. Machine learning for molecular and materials science. *Nature* 559, 7715 (2018), 547–555.
[7] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. Deep reinforcement learning from human preferences. arXiv:1706.03741 [stat.ML]
[8] Kalyan Das, Paul J Lewi, Stephen H Hughes, and Eddy Arnold. 2005. Crystallography and the design of anti-AIDS drugs: conformational flexibility and positional adaptability are important in the design of non-nucleoside HIV-1 reverse transcriptase inhibitors. *Prog Biophys Mol Biol* 88, 2 (Jun 2005), 209–231. https://doi.org/10.1016/j.pbiomolbio.2004.07.001
[9] Zijie Geng, Shufang Xie, Yingce Xia, Lijun Wu, Tao Qin, Jie Wang, Yongdong Zhang, Feng Wu, and Tie-Yan Liu. 2023. De novo molecular generation via connection-aware motif mining. *arXiv preprint arXiv:2302.01129* (2023).
[10] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.
[11] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. 2018. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science* 4, 2 (2018), 268–276.
[12] David J. Griffiths and Darrell F. Schroeter. 2018. *Introduction to Quantum Mechanics* (3 ed.). Cambridge University Press.
[13] John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. 2012. ZINC: a free tool to discover chemistry for biology. *Journal of*

chemical information and modeling 52, 7 (2012), 1757–1768.
[14] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2018. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*. PMLR, 2323–2332.
[15] Jeroen Kazius, Ross McGuire, and Roberta Bursi. 2005. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of medicinal chemistry* 48, 1 (2005), 312–320.
[16] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980
[17] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
[18] Xiangzhe Kong, Wenbing Huang, Zhixing Tan, and Yang Liu. 2022. Molecule generation by principal subgraph mining and assembling. *Advances in Neural Information Processing Systems* 35 (2022), 2550–2563.
[19] Mert Kosan, Zexi Huang, Sourav Medya, Sayan Ranu, and Ambuj Singh. 2023. Global Counterfactual Explainer for Graph Neural Networks. In *WSDM*.
[20] Mert Kosan, Samidha Verma, Burouj Armgaan, Khushbu Pahwa, Ambuj Singh, Sourav Medya, and Sayan Ranu. 2024. GNNX-BENCH: Unravelling the Utility of Perturbation-based GNN Explainers through In-depth Benchmarking. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=VJvbOSXRUq
[21] Vivian Lai, Yiming Zhang, Chacha Chen, Q Vera Liao, and Chenhao Tan. 2023. Selective explanations: Leveraging human input to align explainable ai. *arXiv preprint arXiv:2301.09656* (2023).
[22] Yujia Li, Oriol Vinyals, Chris Dyer, Razvan Pascanu, and Peter Battaglia. 2018. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324* (2018).
[23] Tairan Liu, Misagh Naderi, Chris Alvin, Supratik Mukhopadhyay, and Michal Brylinski. 2017. Break down in order to build up: decomposing small molecules for fragment-based drug design with e molfrag. *Journal of chemical information and modeling* 57, 4 (2017), 627–631.
[24] Ana Lucic, Maartje A Ter Hoeve, Gabriele Tolomei, Maarten De Rijke, and Fabrizio Silvestri. 2022. Cf-gnnexplainer: Counterfactual explanations for graph neural networks. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 4499–4511.
[25] Jing Ma, Ruocheng Guo, Saumitra Mishra, Aidong Zhang, and Jundong Li. 2022. Clear: Generative counterfactual explanations on graphs. *Advances in Neural Information Processing Systems* 35 (2022), 25895–25907.
[26] Oscar Méndez-Lucio, Christos Nicolaou, and Berton Earnshaw. 2022. MolE: a molecular foundation model for drug discovery. arXiv:2211.02657 [q-bio.QM]
[27] Bryan N Nguyen, Elaine W Shen, Janina Seemann, Adrienne MS Correa, James L O'Donnell, Andrew H Altieri, Nancy Knowlton, Keith A Crandall, Scott P Egan, W Owen McMillan, et al. 2020. Environmental DNA survey captures patterns of fish and invertebrate diversity across a tropical seascape. *Scientific Reports* 10, 1 (2020), 6729.
[28] Danilo Numeroso and Davide Bacciu. 2021. Meg: Generating molecular counterfactual explanations for deep graph networks. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.
[29] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. arXiv:2203.02155 [cs.CL]
[30] Florbela Pereira and João Aires-de Sousa. 2018. Machine learning for the prediction of molecular dipole moments obtained by density functional theory. *Journal of cheminformatics* 10 (2018), 1–11.
[31] Mario Alfonso Prado-Romero, Bardh Prenkaj, Giovanni Stilo, and Fosca Giannotti. [n. d.]. A survey on graph counterfactual explanations: definitions, methods, evaluation, and research challenges. *Comput. Surveys* ([n. d.]).
[32] Kaspar Riesen, Horst Bunke, et al. 2008. IAM Graph Database Repository for Graph Based Pattern Recognition and Machine Learning.. In *SSPR/SPR*, Vol. 5342. 287–297.
[33] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. 2021. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences* 118, 15 (2021), e2016239118. https://doi.org/10.1073/pnas.2016239118 arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.2016239118
[34] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. 2022. Large-Scale Chemical Language Representations Capture Molecular Structure and Properties. arXiv:2106.09553 [cs.LG]
[35] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. 2012. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Phys. Rev. Lett.* 108 (Jan 2012), 058301. Issue 5. https://doi.org/10.1103/PhysRevLett.108.058301
[36] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).
[37] Martin Simonovsky and Nikos Komodakis. 2018. Graphvae: Towards generation of small graphs using variational autoencoders. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part I 27*. Springer, 412–422.
[38] Zhiqing Sun, Yikang Shen, Hongxin Zhang, Qinhong Zhou, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. 2023. SALMON: Self-Alignment with Principle-Following Reward Models. arXiv:2310.05910 [cs.LG]
[39] Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. 2023. Principle-Driven Self-Alignment of Language Models from Scratch with Minimal Human Supervision. arXiv:2305.03047 [cs.LG]
[40] Nathan J. Szymanski, Bernardus Rendy, Yuxing Fei, Rishi E. Kumar, Tanjin He, David Milsted, Matthew J. McDermott, Max Gallant, Ekin Dogus Cubuk, Amil Merchant, Haegyeom Kim, Anubhav Jain, Christopher J. Bartel, Kristin Persson, Yan Zeng, and Gerbrand Ceder. 2023. An autonomous laboratory for the accelerated synthesis of novel materials. *Nature* 624, 7990 (2023), 86–91. https://doi.org/10.1038/s41586-023-06734-w
[41] Juntao Tan, Shijie Geng, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Yunqi Li, and Yongfeng Zhang. 2022. Learning and evaluating graph neural network explanations based on counterfactual and factual reasoning. In *Proceedings of the ACM Web Conference 2022*. 1018–1027.
[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
[43] Xiang Wang, Yingxin Wu, An Zhang, Fuli Feng, Xiangnan He, and Tat-Seng Chua. 2022. Reinforced causal explainer for graph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 2 (2022), 2297–2309.
[44] Zhi Wang, Chicheng Zhang, and Kamalika Chaudhuri. 2022. Thompson Sampling for Robust Transfer in Multi-Task Bandits. arXiv:2206.08556 [cs.LG]
[45] Oliver Wieder, Stefan Kohlbacher, Mélaine Kuenemann, Arthur Garon, Pierre Ducrot, Thomas Seidel, and Thierry Langer. 2020. A compact review of molecular property prediction with graph neural networks. *Drug Discovery Today: Technologies* 37 (2020), 1–12.
[46] Felix Wong, Erica J Zheng, Jacqueline A Valeri, Nina M Donghia, Melis N Anahtar, Satotaka Omori, Alicia Li, Andres Cubillos-Ruiz, Aarti Krishnan, Wengong Jin, et al. 2023. Discovery of a structural class of antibiotics with explainable deep learning. *Nature* (2023), 1–9.
[47] Zhenxing Wu, Jike Wang, Hongyan Du, Dejun Jiang, Yu Kang, Dan Li, Peichen Pan, Yafeng Deng, Dongsheng Cao, Chang-Yu Hsieh, et al. 2023. Chemistry-intuitive explanation of graph neural networks for molecular property prediction with substructure masking. *Nature Communications* 14, 1 (2023), 2585.
[48] Yutong Xie, Chence Shi, Hao Zhou, Yuwei Yang, Weinan Zhang, Yong Yu, and Lei Li. 2021. MARS: Markov Molecular Sampling for Multi-objective Drug Discovery. In *International Conference on Learning Representations*.
[49] Wenda Xu, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Yang Wang, and Lei Li. 2023. Instructscore: Towards explainable text generation evaluation with automatic feedback. *arXiv preprint arXiv:2305.14282* (2023).
[50] Qiang Yang, Changsheng Ma, Qiannan Zhang, Xin Gao, Chuxu Zhang, and Xiangliang Zhang. 2023. Counterfactual Learning on Heterogeneous Graphs with Greedy Perturbation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 2988–2998. https://doi.org/10.1145/3580305.3599289
[51] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. 2018. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems* 31 (2018).
[52] Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. 2020. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 430–438.
[53] Ziwei Zhang, Peng Cui, and Wenwu Zhu. 2020. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering* 34, 1 (2020), 249–270.
[54] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI open* 1 (2020), 57present–81.
[55] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. 2019. Fine-Tuning Language Models from Human Preferences. *CoRR* abs/1909.08593 (2019). arXiv:1909.08593 http://arxiv.org/abs/1909.08593