

# GDiffRetro: Retrosynthesis Prediction with Dual Graph Enhanced Molecular Representation and Diffusion Generation

Shengyin Sun<sup>1\*†</sup>, Wenhao Yu<sup>2\*†</sup>, Yuxiang Ren<sup>3‡</sup>, Weitao Du<sup>4</sup>, Liwei Liu<sup>3</sup>, Xuecang Zhang<sup>3</sup>, Ying Hu<sup>5</sup>, Chen Ma<sup>1</sup>

<sup>1</sup>Department of Computer Science, City University of Hong Kong

<sup>2</sup>Department of Computer Science and Engineering, The Chinese University of Hong Kong

<sup>3</sup>Advance Computing and Storage Lab, Huawei Technologies

<sup>4</sup>Academy of Mathematics and Systems Science, Chinese Academy of Sciences

<sup>5</sup>Department of Electronic Engineering, Tsinghua University

shengysun4-c@my.cityu.edu.hk, whyu24@cse.cuhk.edu.hk, {renyuxiang1, liuliwei5, zhangxuecang}@huawei.com, duweitao@amss.ac.cn, yinghu\_yh@mail.tsinghua.edu.cn, chenma@cityu.edu.hk

## Abstract

Retrosynthesis prediction focuses on identifying reactants capable of synthesizing a target product. Typically, the retrosynthesis prediction involves two phases: Reaction Center Identification and Reactant Generation. However, we argue that most existing methods suffer from two limitations in the two phases: (i) Existing models do not adequately capture the “face” information in molecular graphs for the reaction center identification. (ii) Current approaches for the reactant generation predominantly use sequence generation in a 2D space, which lacks versatility in generating reasonable distributions for completed reactive groups and overlooks molecules’ inherent 3D properties. To overcome the above limitations, we propose GDiffRetro. For the reaction center identification, GDiffRetro uniquely integrates the original graph with its corresponding dual graph to represent molecular structures, which helps guide the model to focus more on the faces in the graph. For the reactant generation, GDiffRetro employs a conditional diffusion model in 3D to further transform the obtained synthon into a complete reactant. Our experimental findings reveal that GDiffRetro outperforms state-of-the-art semi-template models across various evaluative metrics.

## 1 Introduction

The retrosynthesis task aims to find a set of reactants capable of synthesizing a given product, which is a one-to-many problem. Even for experienced chemists, addressing such a one-to-many problem is still extremely challenging. Recently, benefitting from the rapid advancement of deep learning (DL), researchers have resorted to DL models to design efficient methods for retrosynthesis tasks. Existing DL-based methods (Somnath et al. 2021) can be divided into template-based, template-free, and semi-template methods. Template-based methods rely on predefined templates (extracted from a large-scale chemical database). For example,

the GLN (Dai et al. 2019) treats chemical knowledge of reaction templates as logical rules, and then models the joint probability between rules and reactants. As template-based methods are constrained by predefined templates, template-free approaches are proposed. For example, Chemformer (Irwin et al. 2022) formulates the retrosynthesis prediction as a translation task, where the product SMILES (strings describing molecular compositions) and the set of reactant SMILES serve as the “source language strings” and “target language strings”, respectively. Template-free methods typically generate reactant SMILES by sequentially outputting individual symbols, which makes their predictions limited in diversity.

To alleviate issues present in both template-based and template-free methods, the semi-template framework has recently been adopted, which does not utilize reaction templates and has good interpretability. It predicts the final reactants through the intermediates (synthons) in two steps: first identifying the reaction center to form synthons, then completing the synthons into reactants. For instance, G2Gs (Shi et al. 2020) first employs the Relational Graph Convolutional Network (RGCN) (Schlichtkrull et al. 2017) for reaction center identification, and then generates products through the variational graph translation.

Although existing semi-template methods have achieved success in some scenarios, we argue that there are still avenues to improve. First, classical methods (Shi et al. 2020; Dai et al. 2019) solely focus on the nodes within the molecular graph, neglecting the features associated with faces (edges divide the entire plane into a set of regions, called faces) in the graph. The features of faces play a crucial role in the reaction center identification. For example, in a benzene ring, all carbon atoms reside on one face, and the bonds connecting these carbons exhibit high stability, making them less likely to serve as reaction centers. Secondly, existing methods generate reactants based on 2D graphs, ignoring the 3D structural information of molecules to some extent.

To address the above shortcomings, we propose the retrosynthesis prediction with dual graph-enhanced molecular representation and diffusion generation (**GDiffRetro**). We first introduce dual graphs to the reaction center identifica-

\*These authors contributed equally.

†Work done as interns at Huawei ACS Lab.

‡Corresponding author.

tion. The dual graph is a way to describe a graph from the perspective of its faces. In the dual graph, each node corresponds to a face in the original graph. Our primary motivation for introducing the dual graph is to integrate the face information into the node representations, enabling the model to focus on faces within molecular graphs, such as distinguishing whether different nodes are on the same faces. Considering the powerful capability of generative models in capturing realistic patterns (Zhang, Rao, and Agrawala 2023; Sun et al. 2023), we then employ the 3D diffusion model to generate final reactants. Specifically, we generate reactants conditioned on the synthons obtained from the reaction center identification stage, and then conduct the 3D diffusion process to preserve the reactants’ inherent structural properties. Our contributions are summarized below:

- To better extract information from molecular graphs, we introduce dual graphs to guide the model in focusing on the faces in the molecular structure, which enables the model to precisely identify reaction centers.
- To better transform intermediates obtained from the stage of reaction center identification into reactants, we make the first attempt at resorting to the conditional diffusion model in the semi-template retrosynthesis prediction.

## 2 Related Work

**Retrosynthesis Prediction.** Some significant template-based works include GLN (Dai et al. 2019), LocalRetro (Chen and Jung 2021), and Dual-TB (Sun et al. 2020). To overcome the constraints of external knowledge in template-based methods, template-free methods have been developed. Key examples in the template-free category include Chemformer (Irwin et al. 2022), RetroBridge (Igashov et al. 2024), and Dual-TF (Sun et al. 2020). Considering that template-free methods lack interpretability, semi-template methods have emerged. Notable works in the semi-template category include MEGAN (Sacha et al. 2021), G2Gs (Shi et al. 2020), and RetroDiff (Wang et al. 2024).

**Molecular Generation.** The molecular generation is closely related to generative models (Zhang et al. 2024a; Guo et al. 2024; Zhang et al. 2024b). For example, You et al. modeled the molecular generation as a sequential decision process on graphs (You et al. 2018). Recent works introduced diffusion models to molecular data. GeoDiff (Xu et al. 2022) and ConfGF (Shi et al. 2021) condition the model on the adjacency matrix of the molecular, enabling them to optimize torsion angles between atoms. The equivariant diffusion model (Hoogeboom et al. 2022) generates 3D molecules from scratch, conditioned on predefined scalar properties. Another noteworthy work is DiffLinker (Igashov et al. 2022), a 3D diffusion model for designing linkers. Other models include LatentDiff (Cong et al. 2024) for protein design and AbX (Tian, Ren, and Zhang 2024) for antibody design.

## 3 Methodology

### 3.1 Graph-based Reaction Center Identification

A molecule  $\mathcal{M}$  containing  $n$  atoms and  $q$  types of chemical bonds can be written as  $\mathcal{M} = \{\mathbf{A}, \mathbf{X}\}$ , where  $\mathbf{X} \in \mathbb{R}^{n \times d}$  is

a  $d$ -dimensional node feature matrix and  $\mathbf{A} \in \mathbb{R}^{n \times n \times q}$  is an adjacency matrix ( $A_{i,j,k} = 1$  if there exists a bond of type  $k$  between atom  $i$  and atom  $j$ ). Based on the above formulation, a chemical reaction can be described as a pair of sets  $(\mathcal{G}^r, \mathcal{G}^p)$ , where  $\mathcal{G}^r = \{\mathcal{M}_i^r\}_{i=1}^l$  is a set containing  $l$  reactants and  $\mathcal{G}^p = \{\mathcal{M}_j^p\}_{j=1}^m$  is a set containing  $m$  products. Following previous work, we focus only on standard single-output chemical reactions, i.e.,  $|\mathcal{G}^p| = 1$ . For a single-output reaction  $(\{\mathcal{M}_i^r\}_{i=1}^l, \mathcal{M}^p)$ , the goal of retrosynthesis is to predict the set of reactants  $\{\mathcal{M}_i^r\}_{i=1}^l$  corresponding to the given product  $\mathcal{M}^p$ . The overview of our solution for the retrosynthesis task is shown in Fig. 1. We first conduct reaction center identification to partition the products into synthons (subgraphs of the product molecule, often not valid molecules). Then, we utilize a diffusion model to generate reactants based on the previously obtained synthons. Notations and chemical terms are summarized in Section A and Section B of the supplementary material, respectively. The supplementary material is in arXiv version of the paper.

Given embeddings of two atoms in the product, the prediction model for reaction center identification is required to output a score, i.e., the probability of a reaction center existing between these two atoms. The higher the probability, the more it indicates that the product needs to break the bond between these two atoms to generate synthons.

Considering the heterogeneity of the molecular graph, we use the RGCN to encode atoms in the given product  $\mathcal{M}^p = \{\mathbf{A}^p, \mathbf{X}^p\}$ . The node  $i$ ’s representation is updated as follows (starting with node  $i$ ’s initial feature  $\mathbf{h}_i^0 = \mathbf{X}^p[i, :]$ ):

$$\{\mathbf{h}_i^l\}_{l=1}^L = \sigma \left( \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \mathbf{W}_r^{l-1} \mathbf{h}_j^{l-1} + \mathbf{W}_0^{l-1} \mathbf{h}_i^{l-1} \right), \quad (1)$$

where  $\mathcal{R}$  is the set of all edge types (chemical bonds),  $\mathcal{N}_i^r$  is the set of neighbors of node  $i$  under relation  $r$  (can be obtained through  $\mathbf{A}^p[:, :, r]$ ),  $\sigma(\cdot)$  is an activation function,  $\mathbf{W}_r$  is the learnable weight matrix corresponding to the edge type  $r$ , and  $\mathbf{W}_0$  is the learnable weight matrix for the self-loop edge. An RGCN with  $L$  layers can only aggregate information from nodes within  $L$  hops, while the reactivity of a reaction center may also be related to more distant nodes. Therefore, we also compute the graph-level embedding (by applying the Readout( $\cdot$ ) function proposed in (Velickovic et al. 2019)) to introduce the influence of remote atoms, i.e.,

$$\mathbf{h}_{\mathcal{M}^p} = \text{Readout}(\mathbf{H}^L), \quad (2)$$

where  $\mathbf{H}^L$  is a node embedding matrix constructed from  $\mathbf{h}_i^L$ .

### 3.2 Dual Graph Enhanced Representation

Encoding strategies in Section 3.1 is designed from the perspective of the nodes, while neglecting the faces in the molecular graph. Faces in the molecular graph are equally important. For example, carbon atoms in a benzene ring are all in one face, and the bonds between these carbons are very stable, making them unlikely to become reaction centers. To make up this shortcoming, we introduce the dual graph  $\mathcal{D}^p = \{\mathbf{A}_d^p, \mathbf{X}_d^p\}$  of  $\mathcal{M}^p$ , which is constructed as follows:

- **Topological structure construction.** Given an original planar graph  $\mathcal{M}^p$ , the dual graph  $\mathcal{D}^p$  is a graph that has a

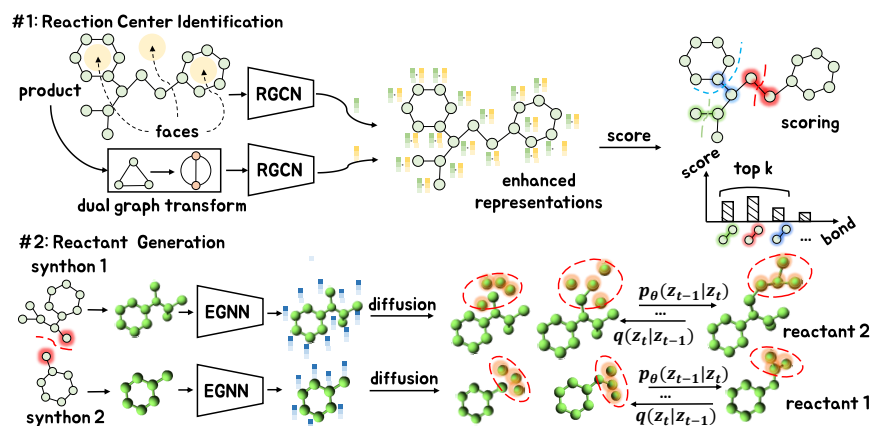


Figure 1: The framework of GDiffRetro. In the stage #1, we utilize the dual graph to enhance the representations. In the stage #2, we employ the 3D diffusion model (conditioned on the obtained synthon) to convert synthons into reactants.

node for each face of  $\mathcal{M}^P$ . Additionally,  $\mathcal{D}^P$  has an edge connecting two nodes if the corresponding faces in  $\mathcal{M}^P$  are separated by an edge in the  $\mathcal{M}^P$ . The type of an edge in  $\mathcal{D}^P$  corresponds to the type of the edge it crosses in  $\mathcal{M}^P$ . As shown in Fig. 2, five nodes in the original graph divide the entire space into three parts. Consequently, its corresponding dual graph contains three nodes, with each node representing a face. Furthermore, the original graph contains two types of edges (denoted as blue and green, respectively). Similarly, its dual graph also comprises two types of edges. Specifically, edges crossing the blue edges in the original graph belong to one type (marked as red), whereas edges crossing the green edges in the original graph belong to another type (marked as orange).

- **Node feature construction.** The feature of a node in the dual graph depends on the surface where the node is located. Formally, the feature of node  $i$  in  $\mathcal{D}^P$  is:

$$\mathbf{X}_d^P[i, :] = \frac{1}{|\mathcal{S}_i|} \sum_{j \in \mathcal{S}_i} \mathbf{X}^P[j, :], \quad (3)$$

where  $\mathcal{S}_i$  is the set of nodes in  $\mathcal{M}^P$  on the face where node  $i$  in  $\mathcal{D}^P$  is located.

After obtaining  $\mathcal{D}^P$ , we encode the nodes in the dual graph in the same way as before (through an L-layer RGCN), i.e.,

$$\mathbf{D}^L = \text{RGCN}(\mathcal{D}^P), \quad (4)$$

where each row in  $\mathbf{D}^L$  is the final embedding of a node in  $\mathcal{D}^P$ . Combining  $\mathcal{M}^P$  and  $\mathcal{D}^P$ , the final embedding of node  $i$  in the original molecular graph can be expressed as:

$$\mathbf{m}_i = \mathbf{H}^L[i, :] \parallel \sum_{j \in \mathcal{F}_i} \mathbf{D}^L[j, :] \parallel \mathbf{h}_{\mathcal{M}^P}, \quad (5)$$

where  $\mathcal{F}_i$  is the set of nodes in  $\mathcal{D}^P$  on the face where node  $i$  in  $\mathcal{M}^P$  is located. In order to estimate the reactivity probability between a pair of nodes  $i$  and  $j$ , we formulate the edge embedding as follows:

$$\mathbf{e}_{ij} = \mathbf{m}_i \parallel \mathbf{m}_j \parallel \mathbf{A}^P[i, j, :]. \quad (6)$$

Then, the reactivity score can be calculated as  $s_{ij} = \text{Sigmoid}(\phi(\mathbf{e}_{ij}))$ , where  $\phi(\cdot)$  is a network for converting edge embeddings to scalar scores. For training, the proposed module is optimized by maximizing the following loss:

$$\mathcal{L}^{(1)} = -\mathbb{E}_{\mathcal{P}_r} \left[ \sum_i \sum_{j \neq i} \lambda Y_{ij} \log(s_{ij}) + (1 - Y_{ij}) \log(1 - s_{ij}) \right], \quad (7)$$

where  $\mathcal{P}_r$  is the set of all chemical reactions in the training data,  $Y_{ij}$  is the true label indicating whether a reaction center exists between atoms  $i$  and  $j$ , and  $\lambda$  is a hyper-parameter for alleviating class imbalance issues.

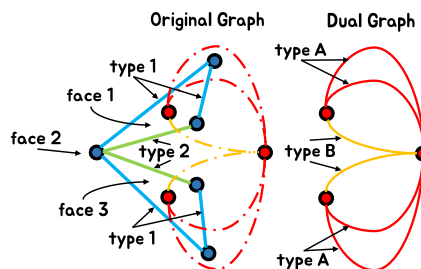


Figure 2: An example of dual graph construction. Each node in the dual graph corresponds to a face in the original graph, and the type of each edge in the dual graph depends on the type of the edge it crosses in the original graph (Type 1  $\Leftrightarrow$  Type A, Type 2  $\Leftrightarrow$  Type B). More details about dual graphs and faces are in Section E (supplementary material).

In Fig. 2, each node in the dual graph represents a face in the original graph. When we input the dual graph into the RGCN, the outputted node representations of the dual graph are actually the face representations of the original graph. The obtained face representations are then integrated into the node representations of the original graph through Eq. 5. In this way, the face information within the original graph can be effectively used. When two nodes in the original graph exist within the same face, their corresponding face repre-

sentations included in the Eq. 5 will be the same (i.e., they have the same face information), which aligns with intuition.

### 3.3 Conditional Diffusion Reactant Generation

In Fig. 1, the conditional diffusion generation involves two processes: *i*) A forward process corrupts the structure and features of a synthon by adding Gaussian noise. *ii*) A reverse process learns the denoise process and outputs a reactant.

• **Forward process.** An atom  $\mathbf{s}$  can be represented by a 3D coordinates  $\mathbf{u}^{(x)} \in \mathbb{R}^3$  and  $d$ -dimensional features  $\mathbf{u}^{(h)} \in \mathbb{R}^d$ , i.e.,  $\mathbf{s} = [\mathbf{u}^{(x)}, \mathbf{u}^{(h)}]$ . Setting  $\mathbf{z}_0 = \mathbf{s}$  as the initial state and parameterize a fixed noise process as:  $q(\mathbf{z}_t | \mathbf{z}_0) = \mathcal{N}(\mathbf{z}_t; \alpha_t \mathbf{z}_0, \sigma_t^2 \mathbf{I})$ ,  $t = 1, \dots, T$ , where  $\mathbf{z}_t$  is a latent noised representation,  $\mathcal{N}(\cdot)$  denotes a Gaussian distribution,  $\mathbf{I}$  is an identity matrix,  $\alpha_t$  controls the proportion of the original input to be retained, and  $\sigma_t^2$  controls the intensity of added Gaussian noise. Inspired by (Song et al. 2021), we adopt a variance-preserving noise adding process, i.e.,  $\alpha_t = \sqrt{1 - \sigma_t^2}$ . In addition, a numerically stable polynomial noise schedule (Hoogeboom et al. 2022) is applied, i.e.,  $\alpha_t = (1 - 2s) \left[ 1 - \left( \frac{t-1}{T-1} \right)^2 \right]$ , where  $s$  is set to  $10^{-5}$  to ensure numerical stability.

• **Reverse process.** The reverse process takes  $\mathbf{x}_T$  as the starting point and attempts to learn a network with  $\theta$  as a trainable parameter for denoising, as follows:

$$p_\theta(\mathbf{z}_{t-1} | \mathbf{z}_t) = \mathcal{N}(\mathbf{z}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{z}_t, t), \mathbf{R}_\theta(\mathbf{z}_t, t)), \quad (8)$$

where  $\boldsymbol{\mu}_\theta(\mathbf{z}_t, t)$  and  $\mathbf{R}_\theta(\mathbf{z}_t, t)$  are obtained from a network parameterized by  $\theta$ .

• **Optimization process.** To maximize the likelihood of observed data, we optimize the variational lower bound, i.e.,

$$\begin{aligned} -\log p_\theta(\mathbf{z}_0) &= -\log \int p_\theta(\mathbf{z}_{0:T}) d\mathbf{z}_{1:T} \\ &\leq \sum_{t>1}^T \underbrace{\mathbb{E}_{q(\mathbf{z}_t | \mathbf{z}_0)} [D_{\text{KL}}(q(\mathbf{z}_{t-1} | \mathbf{z}_t, \mathbf{z}_0) \| p_\theta(\mathbf{z}_{t-1} | \mathbf{z}_t))]}_{\text{(diffusion loss } \mathcal{L}_t)} \\ &\quad + \underbrace{D_{\text{KL}}(q(\mathbf{z}_T | \mathbf{z}_0) \| p(\mathbf{z}_T))}_{\text{(prior loss } \mathcal{L}_p)} - \underbrace{\mathbb{E}_{q(\mathbf{z}_1 | \mathbf{z}_0)} [\log p_\theta(\mathbf{z}_0 | \mathbf{z}_1)]}_{\text{(reconstruction loss } \mathcal{L}_1)}, \end{aligned} \quad (9)$$

where  $p_\theta(\mathbf{z}_{0:T}) = p_\theta(\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_T)$ ,  $d\mathbf{z}_{1:T} = d\mathbf{z}_1 d\mathbf{z}_2 \dots d\mathbf{z}_T$ , and  $D_{\text{KL}}(\cdot)$  is the KL divergence. The  $\mathcal{L}_t$  encourages approximating the  $q(\mathbf{z}_{t-1} | \mathbf{z}_t, \mathbf{z}_0)$  through a distribution  $p_\theta(\mathbf{z}_{t-1} | \mathbf{z}_t)$  associated with a network. The closed form of  $q(\mathbf{z}_{t-1} | \mathbf{z}_t, \mathbf{z}_0)$  can be expressed as:

$$q(\mathbf{z}_{t-1} | \mathbf{z}_t, \mathbf{z}_0) = \mathcal{N}(\mathbf{z}_{t-1}; \boldsymbol{\mu}_q(\mathbf{z}_t, \mathbf{z}_0, t), \sigma_q^2(t) \mathbf{I}), \quad (10)$$

where

$$\boldsymbol{\mu}_q(\mathbf{z}_t, \mathbf{z}_0, t) = \frac{\alpha_t \sigma_{t-1}^2}{\alpha_{t-1} \sigma_t^2} \mathbf{z}_t + \frac{\alpha_{t-1}^2 \sigma_t^2 - \alpha_t^2 \sigma_{t-1}^2}{\alpha_{t-1} \sigma_t^2} \mathbf{z}_0, \quad (11)$$

and  $\sigma_q^2(t) = \sigma_{t-1}^2 - \frac{\alpha_t^2 \sigma_{t-1}^4}{\alpha_{t-1}^2 \sigma_t^2}$ . Substituting Eq. 8 and Eq. 10 into the diffusion loss at the time step  $t$  yields:

$$\begin{aligned} \mathcal{L}_t &= \mathbb{E}_{q(\mathbf{z}_t | \mathbf{z}_0)} [D_{\text{KL}}(q(\mathbf{z}_{t-1} | \mathbf{z}_t, \mathbf{z}_0) \| p_\theta(\mathbf{z}_{t-1} | \mathbf{z}_t))] \\ &= \mathbb{E}_{q(\mathbf{z}_t | \mathbf{z}_0)} \left[ \frac{1}{2\sigma_q^2(t)} \left[ \left\| \boldsymbol{\mu}_\theta(\mathbf{z}_t, t) - \boldsymbol{\mu}_q(\mathbf{z}_t, \mathbf{z}_0, t) \right\|_2^2 \right] \right], \end{aligned} \quad (12)$$

where  $\boldsymbol{\mu}_\theta(\mathbf{z}_t, t)$  can be expressed according to the Eq. 11:

$$\boldsymbol{\mu}_\theta(\mathbf{z}_t, t) = \frac{\alpha_t \sigma_{t-1}^2}{\alpha_{t-1} \sigma_t^2} \mathbf{z}_t + \frac{\alpha_{t-1}^2 \sigma_t^2 - \alpha_t^2 \sigma_{t-1}^2}{\alpha_{t-1} \sigma_t^2} \hat{\mathbf{z}}_\theta(\mathbf{z}_t, t), \quad (13)$$

with  $\hat{\mathbf{z}}_\theta(\mathbf{z}_t, t)$  as the predicted initial state  $\mathbf{z}_0$  (output by a network). Plugging Eq. 11 and Eq. 13 into Eq. 12 results in

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{z}_t | \mathbf{z}_0)} \left[ \frac{1}{2} \left( \frac{\alpha_{t-1}^2}{\sigma_{t-1}^2} - \frac{\alpha_t^2}{\sigma_t^2} \right) \left\| \hat{\mathbf{z}}_\theta(\mathbf{z}_t, t) - \mathbf{z}_0 \right\|_2^2 \right]. \quad (14)$$

Considering that  $\mathbf{z}_t$  can be reparameterized as  $\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \boldsymbol{\epsilon}$ , where  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ , (Ho, Jain, and Abbeel 2020) suggests that using a network to predict  $\boldsymbol{\epsilon}$  instead of  $\mathbf{z}_0$  will lead to a better result, i.e.,  $\mathcal{L}_t$  can be simplified to:

$$\mathcal{L}_t = \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[ \frac{1}{2} \left( \frac{\alpha_{t-1}^2 \sigma_t^2}{\alpha_t^2 \sigma_{t-1}^2} - 1 \right) \left\| \hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t, t) - \boldsymbol{\epsilon} \right\|_2^2 \right]. \quad (15)$$

According to (Hoogeboom et al. 2022), both  $\mathcal{L}_p$  and  $\mathcal{L}_0$  are close to 0 (due to  $\alpha_T = 0$ ,  $\alpha_1 \approx 1$ , and  $\mathbf{z}_0$  is discrete). Furthermore, Ho *et al.* (Ho, Jain, and Abbeel 2020) found that removing the weight in Eq. 15 is conducive to improving sample quality. Therefore, an unweighted version of the final loss  $\mathcal{L}^{(2)}$  used in the reactant generation phase is:

$$\mathcal{L}^{(2)} = \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t \sim \mathcal{U}(1, T)} \left[ \left\| \hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_t, t) - \boldsymbol{\epsilon} \right\|_2^2 \right]. \quad (16)$$

• **Modeling of the  $\hat{\boldsymbol{\epsilon}}_\theta$ .** During the reactant generation stage, each atom contains both a feature vector and 3D coordinates. To preserve equivariance of  $\hat{\boldsymbol{\epsilon}}_\theta$  to coordinate rotations and translations (details about the equivariance are provided in Proposition 1), we utilize the EGNN (Satorras et al. 2021) to model  $\hat{\boldsymbol{\epsilon}}_\theta$ . Define the feature of atom  $i$  in the denoising time step  $t$  as  $\mathbf{z}_{i,t} = [\mathbf{z}_{i,t}^{(x)}, \mathbf{z}_{i,t}^{(h)}]$  and the final reactant  $\mathcal{R}$  containing  $n$  atoms as  $\mathcal{R} = \{v_i\}_{i=1}^n = \{\mathcal{S}, \mathcal{Q}\}$ , where  $\mathcal{S} = \{v_i\}_{i=1}^m$  is a set of  $m$  atoms in the synthon (obtained from the first stage),  $\mathcal{Q} = \{v_i\}_{i=m+1}^n$  is a set of  $n - m$  atoms need to be generated. The process of determining the number of atoms is detailed in Section G (supplementary material). Following the previous work (Hoogeboom et al. 2022),  $\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_{i,t}, t)$  can be written as:

$$\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_{i,t}, t) = \left[ \mathbf{e}_{i,t}^{(x),L}, \mathbf{e}_{i,t}^{(h),L} \right] - \left[ \mathbf{z}_{i,t}^{(x)}, \mathbf{0} \right], \quad (17)$$

where  $\left[ \mathbf{e}_{i,t}^{(x),L}, \mathbf{e}_{i,t}^{(h),L} \right]$  is the embedding of atom  $i$  output by an  $L$ -layer EGNN (time  $t$ ), and its computation process is:

$$\begin{aligned} \mathbf{m}_{ij} &= \phi_e \left( \mathbf{e}_{i,t}^{(h),l-1}, \mathbf{e}_{j,t}^{(h),l-1}, \left\| \mathbf{d}_{ij}^{l-1} \right\|_2^2 \right), \\ \mathbf{e}_{i,t}^{(h),l} &= \phi_h \left( \mathbf{e}_{i,t}^{(h),l-1}, \sum_{j \neq i} \mathbf{m}_{ij} \right), \\ \mathbf{e}_{i,t}^{(x),l} &= \begin{cases} \mathbf{e}_{i,t}^{(x),l-1} + \sum_{i \neq j} \frac{\mathbf{d}_{ij}^{l-1} \phi_r \left( \mathbf{e}_{i,t}^{(h),l-1}, \mathbf{e}_{j,t}^{(h),l-1} \right)}{1 + \left\| \mathbf{d}_{ij}^{l-1} \right\|_2^2}, & v_i \notin \mathcal{S}, \\ \mathbf{e}_{i,t}^{(x),l-1}, & v_i \in \mathcal{S}, \end{cases} \end{aligned} \quad (18)$$

with  $\mathbf{d}_{ij}^{l-1} = \mathbf{e}_{i,t}^{(x),l-1} - \mathbf{e}_{j,t}^{(x),l-1}$ ,  $\phi_e/\phi_h/\phi_r$  is a multi-layer perceptron. In the reactant generation, a node contains 3D coordinates  $\mathbf{u}^{(x)}$  and features  $\mathbf{u}^{(h)}$ . Processing such features associated with 3D coordinates requires operations that respect the symmetry of the data, i.e.,  $\mathbf{u}^{(h)}$  should be invariant to group transformations and  $\mathbf{u}^{(x)}$  should be related to rotations/translations. Details as shown in Proposition 1:

**Proposition 1** *The update of node features in Eq. 18 satisfies permutation invariance, while the update of coordinates changes with the input coordinate attributes. Formally, denote the update rules in Eq. 18 as an abstract function  $f(\cdot)$ , then we have  $f(\cdot)$  satisfies  $\mathbf{z}^{(x)}, \mathbf{z}^{(h)} = f(\mathbf{u}^{(x)}, \mathbf{u}^{(h)})$  and  $\mathbf{U}\mathbf{z}^{(x)} + \mathbf{t}, \mathbf{z}^{(h)} = f(\mathbf{U}\mathbf{u}^{(x)} + \mathbf{t}, \mathbf{u}^{(h)})$ , where  $\mathbf{U}$  is an orthogonal matrix to rotate  $\mathbf{u}^{(x)}$ , and  $\mathbf{t}$  is a translation vector.*

*Proof.* See Section D.1 (supplementary material).

It can be seen that we keep the coordinates of the atoms in the  $\mathcal{S}$  unchanged (Eq. 18), so that the generation of reactants is conditioned on the synthon obtained in the first stage. Essentially, introducing conditions (defined as  $\tilde{\mathbf{c}}$ ) is equivalent to shifting the mean (Sohl-Dickstein et al. 2015) of the reverse sampling process, as follows:

**Proposition 2** *The benefit of introducing conditions  $\tilde{\mathbf{c}}$  lies in injecting prior knowledge into the reverse sampling process, i.e.,  $p_{\theta}(\mathbf{z}_{t-1}|\mathbf{z}_t, \tilde{\mathbf{c}}) = \mathcal{N}(\mathbf{z}_{t-1}; \boldsymbol{\mu}_{\theta}(\mathbf{z}_t, t) + \mathbf{R}_{\theta}(\mathbf{z}_t, t)\nabla_{\mathbf{z}_t} \log p(\tilde{\mathbf{c}}|\mathbf{z}_t), \mathbf{R}_{\theta}(\mathbf{z}_t, t))$ .*

*Proof.* See Section D.2 (supplementary material).

The Proposition 2 indicates that the reverse sampling process with the mean  $\boldsymbol{\mu}_{\theta}(\mathbf{z}_t, t) + \mathbf{R}_{\theta}(\mathbf{z}_t, t)\nabla_{\mathbf{z}_t} \log p(\tilde{\mathbf{c}}|\mathbf{z}_t)$  and variance  $\mathbf{R}_{\theta}(\mathbf{z}_t, t)$ , where the additional term  $\mathbf{R}_{\theta}(\mathbf{z}_t, t)\nabla_{\mathbf{z}_t} \log p(\tilde{\mathbf{c}}|\mathbf{z}_t)$  guides the model to generate reactants that are more line with the prior (i.e., conditions). The prior  $\mathbf{R}_{\theta}(\mathbf{z}_t, t)\nabla_{\mathbf{z}_t} \log p(\tilde{\mathbf{c}}|\mathbf{z}_t)$  makes the generation of reactants more controllable (preventing deviation from chemical principles), thereby improving the final hit rate such as the top-1 accuracy.

## 4 Experiments

### 4.1 Experiment Setup

We utilize the USPTO-50k dataset (Lowe 2017), detailed in Section C of the supplementary material, to assess the proposed method. The split of the dataset follows previous work (Liu et al. 2017; Shi et al. 2020). Baselines are selected as follows: For *Template-Based* methods, we select MHNreact (Seidl et al. 2022), GLN (Dai et al. 2019), LocalRetro (Chen and Jung 2021), GraphRetro (Somnath et al. 2021), RetroComposer (Yan et al. 2022), and DualTB (Sun et al. 2020). For *Template-Free* methods, we select Transformer (Vaswani et al. 2017), SCROP (Zheng et al. 2020), Retroformer (Wan et al. 2022), GTA (Seo et al. 2021), Graph2SMILES (D-GCN) (Tu and Coley 2022), Transformer (*Aug.*) (Tetko et al. 2020), Dual-TF (Sun et al. 2020), Chemformer (Irwin et al. 2022), and RetroBridge (Igashov et al. 2024). For *Semi-Template* methods, we select MEGAN (Sacha et al. 2021), G2Gs (Shi et al. 2020), RetroXpert (Yan et al. 2020), G<sup>2</sup>Retro (Chen et al. 2023), RetroPrime (Wang et al. 2021), and RetroDiff (Wang et al. 2024). The definition

of different types is given in Section 1. Following previous work (Liu et al. 2017), we employ the top- $k$  exact match accuracy as evaluation metric. More implementation details can be seen in Section F (supplementary material). Code available at <https://github.com/sunshy-1/GDiffRetro>.

### 4.2 Performance Comparison

As shown in Table 1, the top-1 result of GDiffRetro surpasses all template-free/semi-template based baselines, and most of the state-of-the-art template-based baselines. It’s important to note that template-based methods rely heavily on external knowledge compared to template-free/semi-template based methods, making a direct comparison between template-based methods and template-free/semi-template based methods inherently unfair. To ensure fairness, we focus on performance gains within the category. Within the “Semi-Template” category, GDiffRetro achieves a relative improvement of 12.0% (unknown class) and 4.3% (known class, i.e. assuming the reaction class is known) in terms of the top-1 metric compared to the second-best method. This demonstrates that GDiffRetro can provide the most accurate retrosynthesis prediction with just a single attempt (this advantage is explained in the Proposition 2).

In real-world applications, a relatively high single-attempt success rate (i.e., top-1 accuracy) is extremely important. This is because the reactants obtained a single retrosynthetic prediction usually not commercially available. Typically, we need to recursively conduct multiple retrosynthesis predictions to obtain the final synthesis route (akin to a search tree). Obviously, under this tree-like structure, a higher top-1 accuracy can greatly narrow the search space, thereby enhancing efficiency and reducing resource consumption. When examining the top-3 and top-5 metrics, GDiffRetro performs on par with all the template-free and semi-template baselines. Particularly, GDiffRetro almost outperforms all template-free/semi-template baselines in the top-3 and top-5 performances. It is worth noting that certain approaches, such as MEGAN, may exhibit significantly higher performance than GDiffRetro in terms of the top-5 result (known class). We attribute this difference to the limited number of sampling iterations in GDiffRetro, which may hinder its ability to generate an ample set of candidates. Mathematically, the diversity of results generated by a diffusion model is closely related to the sampling PDF’s peakiness. In the conditional diffusion model, the peakiness depends on the mutual information between the condition and target. In our setting, the synthons (conditions) and reactants (targets) are similar, this leads to more concentrated sampling results, which in turn causes a decrease in the top-5 accuracy.

### 4.3 Analysis and Visualization

To assess the proficiency of GDiffRetro in learning reaction templates, we visualize the end-to-end retrosynthesis prediction for two examples from the same reaction class in Fig. 3. For two products belonging to the “protections” class, GDiffRetro accurately predicts the reaction centers. Then, it generates the reactants using two similar sets of synthons, formulating identical completed parts. These examples demonstrate that GDiffRetro’s results are informed by

Baselines		Top- $k$ accuracy (unknown class)			Top- $k$ accuracy(known class)		
		$k=1$	$k=3$	$k=5$	$k=1$	$k=3$	$k=5$
Template-Based	MHNreact (Seidl et al. 2022)	51.8	74.6	81.2	-	-	-
	GLN (Dai et al. 2019)	52.5	69.0	75.6	64.2	79.1	85.2
	LocalRetro (Chen and Jung 2021)	53.4	<b>77.5</b>	<b>85.9</b>	63.9	<b>86.8</b>	<b>92.4</b>
	GraphRetro (Somnath et al. 2021)	53.7	68.3	72.2	63.9	81.5	85.2
	RetroComposer (Yan et al. 2022)	54.5	77.2	83.2	65.9	85.8	89.5
	Dual-TB (Sun et al. 2020)	<b>55.2</b>	74.6	80.5	<b>67.7</b>	84.8	88.9
Template-Free	Transformer (Vaswani et al. 2017)	43.7	59.7	65.1	-	-	-
	SCROP (Zheng et al. 2020)	43.7	60.0	65.2	59.0	74.8	78.1
	Transformer ( <i>Aug.</i> ) (Tetko et al. 2020)	48.3	-	73.4	-	-	-
	RetroBridge (Igashov et al. 2024)	50.8	<b>74.1</b>	<b>80.6</b>	-	-	-
	GTA (Seo et al. 2021)	51.1	67.6	74.8	-	-	-
	Retroformer (Wan et al. 2022)	52.9	68.2	72.5	64.0	<b>82.5</b>	<b>86.7</b>
	Graph2SMILES (Tu and Coley 2022)	52.9	66.5	70.0	-	-	-
	Dual-TF (Sun et al. 2020)	53.6	70.7	74.6	<b>65.7</b>	81.9	84.7
Semi-Template	Chemformer (Irwin et al. 2022)	<b>54.3</b>	-	62.3	-	-	-
	MEGAN (Sacha et al. 2021)	48.1	70.7	78.4	60.7	82.0	<b>87.5</b>
	G2Gs (Shi et al. 2020)	48.9	67.6	72.5	61.0	81.3	86
	RetroXpert (Yan et al. 2020)	50.4	61.1	62.3	62.1	75.8	78.5
	G <sup>2</sup> Retro (Chen et al. 2023)	51.4	72.1	78.2	59.4	79.4	84.2
	RetroPrime (Wang et al. 2021)	51.4	70.8	74.0	64.8	82.7	85
	RetroDiff (Wang et al. 2024)	52.6	71.2	81.0	-	-	-
	GDiffRetro (Ours ♣)	<b>58.9 ♣</b>	<b>79.1 ♣</b>	<b>81.9 ♣</b>	<b>67.6 ♣</b>	<b>84.1 ♣</b>	86.1 ♣

Table 1: Top- $k$  exact match accuracy (%) on reaction dataset USPTO-50k. The best result for each category is bolded.

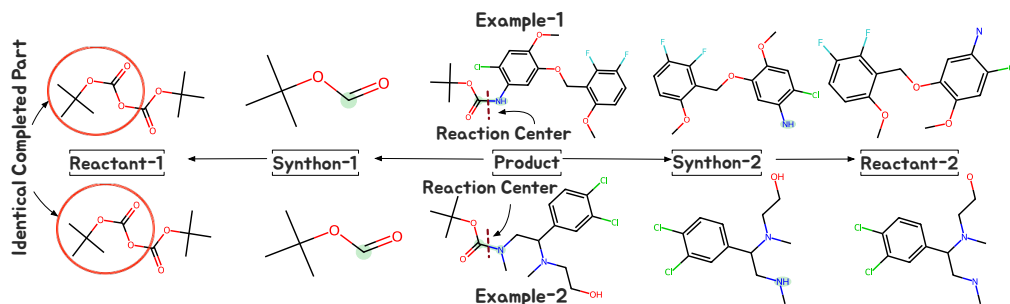


Figure 3: Depiction of the overall *Retrosynthesis Prediction* process for examples in the *protections* reaction class. Reaction centers are highlighted on the products and synthons, while the completed parts are outlined with a circle on the reactants.

Setting ( <i>Dual-G</i> )	Top- $k$ Accuracy %				
	$k=1$	$k=2$	$k=3$	$k=5$	$k=10$
<i>w/o</i>	81.4	93.4	96.5	98.6	99.6
<i>w</i>	86.2	95.1	97.4	98.8	99.6

a. Top- $k$  accuracy of *Reaction Center Prediction*.

b. Top- $k$  accuracy of *Retrosynthesis Prediction*.

Table 2: Results of single *Reaction Center Prediction* and end-to-end *Retrosynthesis Prediction* (unknown class). *w Dual-G* means the *Dual-Graphs* considered. *w/o Dual-G* means the vanilla RGCN without *Dual-Graphs*.

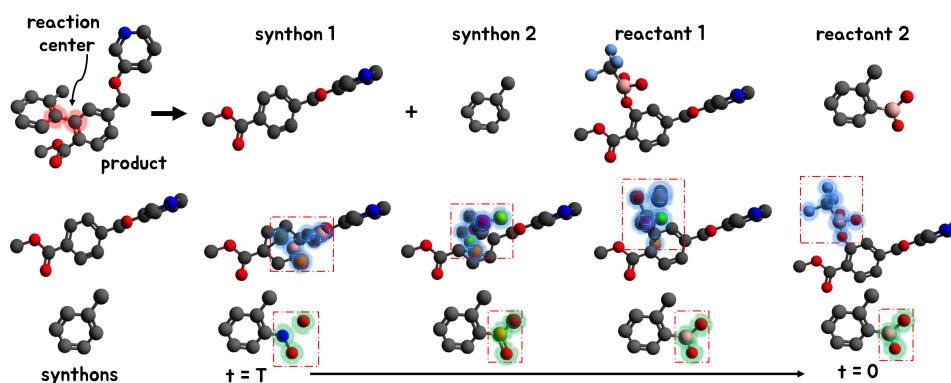


Figure 4: Trajectory of reactant generation. The atoms undergoing changes are highlighted with a rectangle.

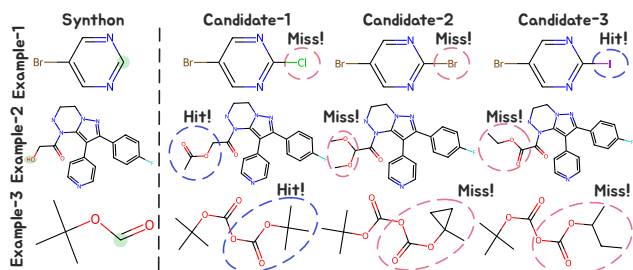


Figure 5: Top-3 generated reactants for distinct synthons. Correctly completed parts are denoted with *Hit!*, while incorrect completions are marked with *Miss!*.

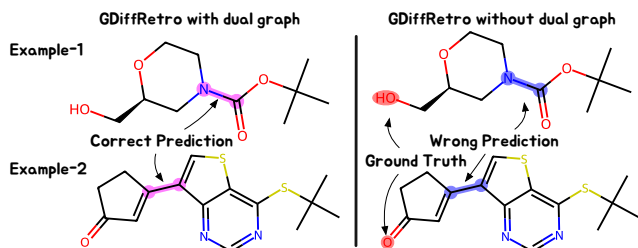


Figure 6: Illustration of reaction center on two products predicted by GDiffRetro, with dual graphs considered (left pair, the prediction and ground truth overlap) and without them (right pair, the prediction and ground truth are inconsistent).

its understanding of the reaction class, indicating its potential in capturing the underlying reaction template. The trajectory of reactant generation is provided in Fig. 4. Atoms to be generated start from a cluster of noise (with random coordinates and categories). As the denoising time step changes, reasonable atom types and coordinates gradually emerge.

To demonstrate the validity and diversity of generated reactant candidates, we provide the top-3 results given by GDiffRetro on 3 different synthons in Fig. 5. It can be seen that GDiffRetro not only always matches in a one-shot trial, but also gives some reasonable results in the following sub-candidates. Among them, some examples generate

all reactants with the same number of completed atoms, but GDiffRetro still provides diverse meaningful candidates.

#### 4.4 Ablation Study

To verify the effectiveness of the dual graphs we introduced to the RGCNs, we compare two sets of ablation experiments (**a.** top- $k$  accuracy of reaction center prediction, **b.** top- $k$  accuracy of end-to-end retrosynthesis prediction), with and without dual graphs introduced in the inference process. The results are shown in Table 2a and Table 2b, respectively.

In Table 2a, it is evident that GDiffRetro achieves a significant improvement in top-1 accuracy when incorporating dual graphs. GDiffRetro with *Dual-G* also consistently surpasses its counterpart without *Dual-G* among all other metrics. A clear correlation between the performance of the end-to-end retrosynthesis prediction and the *Dual-G* configuration can be observed in Table 2b, following a similar trend to Table 2a. GDiffRetro with *Dual-G* outperforms the version without it, showing approximately a 6% increase in top-1 accuracy, and a slight improvement in top-3, top-5, and top-10 accuracy. The relatively modest improvement is attributed to the already high accuracy observed in these metrics.

The predictions for the reaction center of two test molecules, containing 1 and 3 rings, respectively, are depicted in Fig. 6. With enhanced information provided by the dual graphs, GDiffRetro with *Dual-G* can offer more precise predictions. More examples from 10 distinct reaction classes are illustrated in Section H (supplementary material).

## 5 Conclusion

In this paper, we introduce GDiffRetro, a novel framework designed for retrosynthesis prediction. GDiffRetro notably incorporates a dual graph enhanced molecular representation for the reaction center identification, and introduces the 3D conditional diffusion model for reactant generation. Experimental results show that GDiffRetro not only surpasses current state-of-the-art models in top-1 accuracy, including those heavily reliant on templates, but also achieves competitive performance across top-3 and top-5 rankings. Through comprehensive ablation studies and detailed visualization, we confirm that the two key components proposed in GDiffRetro function independently and effectively.

## Acknowledgments

This work was supported by the NSFC Young Scientists Fund (No. 9240127), the Donations for Research Projects (No. 9229129) of the City University of Hong Kong, the Early Career Scheme (No. CityU 21219323), and the General Research Fund (No. CityU 11220324) of the University Grants Committee (UGC).

## References

- Chen, S.; and Jung, Y. 2021. Deep retrosynthetic reaction prediction using local reactivity and global attention. *JACS Au*, 1(10): 1612–1620.
- Chen, Z.; Ayinde, O. R.; Fuchs, J. R.; Sun, H.; and Ning, X. 2023. G2Retro as a two-step graph generative models for retrosynthesis prediction. *Communications Chemistry*.
- Cong, F.; Yan, K.; Wang, L.; Au, W. Y.; McThrow, M. C.; Komikado, T.; Maruhashi, K.; Uchino, K.; Qian, X.; and Ji, S. 2024. A latent diffusion model for protein structure generation. In *Learning on Graphs Conference*.
- Dai, H.; Li, C.; Coley, C.; Dai, B.; and Song, L. 2019. Retrosynthesis prediction with conditional graph logic network. In *Advances in Neural Information Processing Systems*.
- Guo, Z.; Liu, J.; Wang, Y.; Chen, M.; Wang, D.; Xu, D.; and Cheng, J. 2024. Diffusion models in bioinformatics and computational biology. *Nature reviews bioengineering*, (2): 136–154.
- Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *arXiv:2006.11239*.
- Hoogeboom, E.; Satorras, V. G.; Vignac, C.; and Welling, M. 2022. Equivariant diffusion for molecule generation in 3D. In *Proc. Int. Conf. Machine Learning*.
- Igashov, I.; Schneuing, A.; Segler, M.; Bronstein, M.; and Correia, B. 2024. Retrobridge: Modeling retrosynthesis with Markov bridges. In *Proc. Int. Conf. Learning Representations*.
- Igashov, I.; Stärk, H.; Vignac, C.; Satorras, V. G.; Frossard, P.; Welling, M.; Bronstein, M.; and Correia, B. 2022. Equivariant 3D-conditional diffusion models for molecular linker design. In *arXiv:2210.05274*.
- Irwin, R.; Dimitriadis, S.; He, J.; and Bjerrum, E. J. 2022. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1): 015022.
- Liu, B.; Ramsundar, B.; Kawthekar, P.; Shi, J.; Gomes, J.; Luu Nguyen, Q.; Ho, S.; Sloane, J.; Wender, P.; and Pande, V. 2017. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS Central Science*, 3(10): 1103–1113.
- Lowe, D. 2017. Chemical reactions from US patents (1976-Sep2016).
- Sacha, M.; Błaz, M.; Byrski, P.; Dabrowski-Tumanski, P.; Chrominski, M.; Loska, R.; Włodarczyk-Pruszyński, P.; and Jastrzebski, S. 2021. Molecule edit graph attention network: modeling chemical reactions as sequences of graph edits. *Journal of Chemical Information and Modeling*, 61(7): 3273–3284.
- Satorras, V. G.; Hogeboom, E.; Fuchs, F.; Posner, I.; and Welling, M. 2021. E(n) equivariant normalizing flows. In *Advances in Neural Information Processing Systems*.
- Schlichtkrull, M.; Kipf, T. N.; Bloem, P.; van den Berg, R.; Titov, I.; and Welling, M. 2017. Modeling relational data with graph convolutional networks. In *arXiv:1703.06103*.
- Seidl, P.; Renz, P.; Dyubankova, N.; Neves, P.; Verhoeven, J.; Wegner, J. K.; Segler, M.; Hochreiter, S.; and Klambauer, G. 2022. Improving few- and zero-shot reaction template prediction using modern Hopfield networks. *Journal of Chemical Information and Modeling*.
- Seo, S.-W.; Song, Y. Y.; Yang, J. Y.; Bae, S.; Lee, H.; Shin, J.; Hwang, S. J.; and Yang, E. 2021. GTA: Graph truncated attention for retrosynthesis. In *AAAI*.
- Shi, C.; Luo, S.; Xu, M.; and Tang, J. 2021. Learning gradient fields for molecular conformation generation. In *Proc. Int. Conf. Machine Learning*.
- Shi, C.; Xu, M.; Guo, H.; Zhang, M.; and Tang, J. 2020. A graph to graphs framework for retrosynthesis prediction. In *Proc. Int. Conf. Machine Learning*.
- Sohl-Dickstein, J.; Weiss, E. A.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proc. Int. Conf. Machine Learning*.
- Somnath, V. R.; Bunne, C.; Coley, C.; Krause, A.; and Barzilay, R. 2021. Learning graph models for retrosynthesis prediction. In *Advances in Neural Information Processing Systems*.
- Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-based generative modeling through stochastic differential equations. In *Proc. Int. Conf. Learning Representations*.
- Sun, R.; Dai, H.; Li, L.; Kearnes, S.; and Dai, B. 2020. Energy-based view of retrosynthesis. In *arXiv:2007.13437*.
- Sun, S.; Ren, Y.; Ma, C.; and Zhang, X. 2023. Large Language Models as Topological Structure Enhancers for Text-Attributed Graphs. In *arXiv:2311.14324*.
- Tetko, I. V.; Karpov, P.; Van Deursen, R.; and Godin, G. 2020. State-of-the-art augmented NLP transformer models for direct and single-step retrosynthesis. *Nature Communications*, 11(1): 5575.
- Tian, Z.; Ren, M.; and Zhang, H. 2024. Antibody design using a score-based diffusion model guided by evolutionary, physical and geometric constraints. In *Proc. Int. Conf. Machine Learning*.
- Tu, Z.; and Coley, C. W. 2022. Permutation invariant graph-to-sequence model for template-free retrosynthesis and reaction prediction. *Journal of Chemical Information and Modeling*, 62(15): 3503–3513.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*.
- Velickovic, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep graph infomax. In *Proc. Int. Conf. Learning Representations*.



Wan, Y.; Hsieh, C.-Y.; Liao, B.; and Zhang, S. 2022. Retroformer: Pushing the limits of end-to-end retrosynthesis transformer. In *Proc. Int. Conf. Machine Learning*.

Wang, X.; Li, Y.; Qiu, J.; Chen, G.; Liu, H.; Liao, B.; Hsieh, C.-Y.; and Yao, X. 2021. RetroPrime: A diverse, plausible and Transformer-based method for single-step retrosynthesis predictions. *Chemical Engineering Journal*.

Wang, Y.; Song, Y.; Xu, M.; Wang, R.; Zhou, H.; and Ma, W.-Y. 2024. RetroDiff: Retrosynthesis as multi-stage distribution interpolation. <https://openreview.net/forum?id=dUTwqiEked>.

Xu, M.; Yu, L.; Song, Y.; Shi, C.; Ermon, S.; and Tang, J. 2022. Geodiff: A geometric diffusion model for molecular conformation generation. In *arXiv:2203.02923*.

Yan, C.; Ding, Q.; Zhao, P.; Zheng, S.; Yang, J.; Yu, Y.; and Huang, J. 2020. Retroxpert: Decompose retrosynthesis prediction like a chemist. In *Advances in Neural Information Processing Systems*.

Yan, C.; Zhao, P.; Lu, C.; Yu, Y.; and Huang, J. 2022. RetroComposer: Composing templates for template-based retrosynthesis prediction. *Biomolecules*.

You, J.; ; Liu, B.; Ying, R.; Pande, V.; and Leskovec, J. 2018. Graph convolutional policy network for goal-directed molecular graph generation. In *Advances in Neural Information Processing Systems*.

Zhang, K.; Lin, Y.; Wu, G.; Ren, Y.; Zhang, X.; Zhang, X.; and Du, W. 2024a. Sculpting Molecules in 3D: A Flexible Substructure Aware Framework for Text-Oriented Molecular Optimization. In *arXiv:2403.03425*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proc. IEEE Int. Conf. Computer Vision*.

Zhang, T.; Ren, Y.; Hou, C.; Lv, H.; and Zhang, X. 2024b. Molecular graph representation learning integrating large language models with domain-specific small models. In *International Conference on Bioinformatics and Biomedicine*.

Zheng, S.; Zheng, S.; Rao, J.; Zhang, Z.; Xu, J.; and Yang, Y. 2020. Predicting retrosynthetic reactions using self-corrected transformer neural networks. *Journal of chemical information and modeling*, 60(1): 47–55.