Data-Efficient Molecular Generation with Hierarchical Textual Inversion

Seojin Kim [1] **Jaehyun Nam** [1] **Sihyun Yu** [1] **Younghoon Shin** [2] **Jinwoo Shin** [1]

Abstract

Developing an effective molecular generation framework even with a limited number of molecules is often important for its practical deployment, e.g., drug discovery, since acquiring task-related molecular data requires expensive and time-consuming experimental costs. To tackle this issue, we introduce *Hierarchical textual Inversion for Molecular generation* (HI-Mol), a novel data-efficient molecular generation method. HI-Mol is inspired by the importance of hierarchical information, e.g., both coarse- and fine-grained features, in understanding the molecule distribution. We propose to use multi-level embeddings to reflect such hierarchical features based on the adoption of the recent textual inversion technique in the visual domain, which achieves data-efficient image generation. Compared to the conventional textual inversion method in the image domain using a single-level token embedding, our multi-level token embeddings allow the model to effectively learn the underlying low-shot molecule distribution. We then generate molecules based on the interpolation of the multi-level token embeddings. Extensive experiments demonstrate the superiority of HI-Mol with notable data-efficiency. For instance, on QM9, HI-Mol outperforms the prior state-of-the-art method with 50× less training data. We also show the effectiveness of molecules generated by HI-Mol in low-shot molecular property prediction. Code is available at https://github.com/Seojin-Kim/HI-Mol.

1. Introduction

Finding novel molecules has been a fundamental yet crucial problem in chemistry (Xue et al., 2019; Xu et al., 2019b) due to its strong relationship in achieving important applications, such as drug discovery (Segler et al., 2018; Bongini et al., 2021) and material design (Hamdia et al., 2019; Tagade et al., 2019). However, generating molecules poses a challenge due to their highly complicated nature and the vast size of the input space (Drew et al., 2012). To tackle this issue, several works have considered training deep generative models to learn the molecule distribution using large molecular datasets (Jin et al., 2018; Kong et al., 2022; Ahn et al., 2022; Geng et al., 2023). This is inspired by the recent breakthroughs of generative models in other domains, e.g., images and videos (Rombach et al., 2022; Singer et al., 2022; Yu et al., 2023), in learning high-dimensional data distributions. Intriguingly, such deep molecular generation methods have demonstrated reasonable performance (Jin et al., 2020; Ahn et al., 2022; Kong et al., 2022) on the large-scale benchmarks (Ramakrishnan et al., 2014; Polykovskiy et al., 2020) in finding chemically valid and novel molecules, showing great potential to solve the challenge.

Unfortunately, existing molecular generation frameworks tend to fail in limited data regimes (Guo et al., 2022). This restricts the deployment of existing approaches to practical scenarios, because task-related molecular data for the target real-world applications are often insufficient to train such molecular generative models. For example, drug-like molecules for a specific organ are inherently scarce in nature (Schneider & Fechner, 2005; Altae-Tran et al., 2017), and the drug-likeness of each candidate molecule should be verified through years of extensive wet experiments and clinical trials (Drews, 2000; Hughes et al., 2011). This time-consuming and labor-intensive data acquisition process of new task-related molecules limits the number of training data available for a model to learn the desired molecule distribution (Stanley et al., 2021). Thus, it is often crucial to develop an effective *data-efficient molecular generation* framework, yet this direction has been overlooked in the field of deep molecular generation (Guo et al., 2022).

In this paper, we aim to address the aforementioned shortcomings of existing molecular generation frameworks in the low-shot regimes by designing a method to leverage knowledge in a limited number of molecules extensively. To this end, inspired by the chemical prior that molecules can be hierarchically clustered (Alexander et al., 2011), we introduce multi-level embeddings that capture coarse- and fine-grained

[1]Korea Advanced Institute of Science and Technology (KAIST) [2]Korea University. Correspondence to: Seojin Kim <osikjs@kaist.ac.kr>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

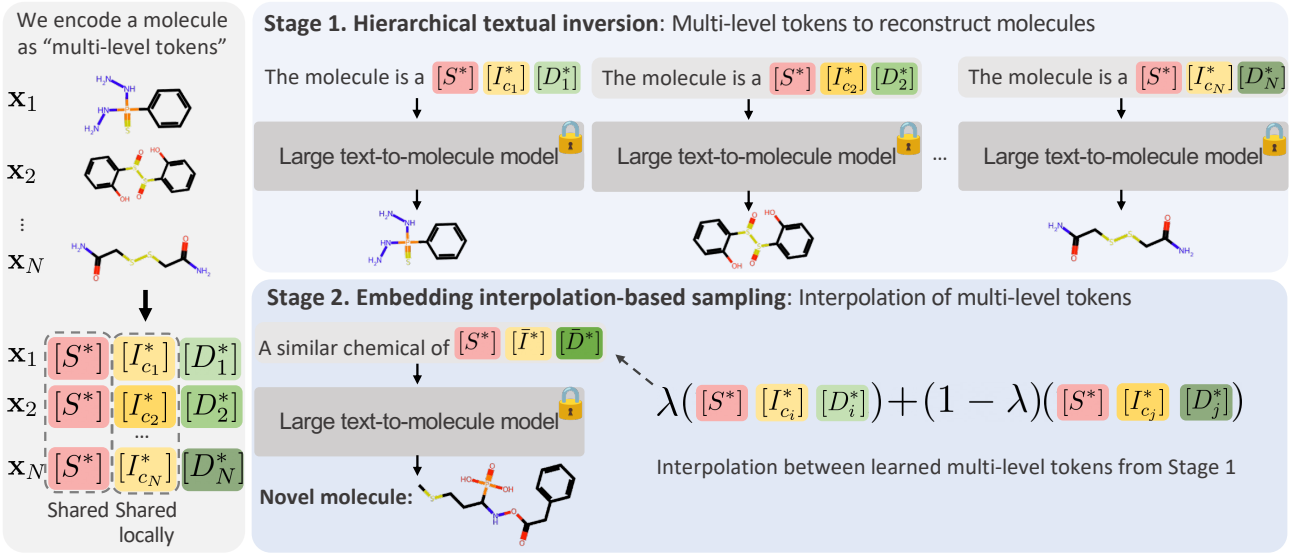


Figure 1. Overview of HI-Mol framework. (1) Hierarchical textual inversion: we encode low-shot molecules into multi-level token embeddings. (2) Embedding interpolation-based sampling: we generate novel molecules using interpolation of low-level token embeddings.

features among low-shot molecules, where a selective assignment of the multi-level tokens for each molecule allows to incorporate hierarchical features of molecules.

We learn this hierarchical embedding in the token space of the recent text-to-molecule model (Edwards et al., 2022). Specifically, we adopt *textual inversion* (Gal et al., 2022)—which learns low-shot image distribution by introducing a new single token within a text-to-image model. A key difference here is that the consideration of "multi-level" tokens (see Figure 1) are essential in learning the low-shot molecule distribution due to the complicatedly structured nature of molecules compared to images (see Table 1).

Contribution. We introduce a novel data-efficient molecular generation method, coined **H**ierarchical textual **I**nversion for **Mol**ecular generation (**HI-Mol**). Specifically, HI-Mol is composed of the following components:

- **Hierarchical textual inversion:** We propose a molecule-specialized textual inversion scheme to capture the hierarchical information of molecules (Alexander et al., 2011). In contrast to textual inversion for the visual domain that optimizes a single shared token on given training data, we design multi-level tokens for the inversion. Thus, the shared token learns the common features among molecules and the low-level tokens learn cluster-specific or molecule-specific features.
- **Embedding interpolation-based sampling:** We propose to use low-level tokens in addition to the shared token for molecular generation. In particular, we consider using the interpolation of low-level token embeddings. The mixing approach is designed to extensively utilize the information of given molecules, and thus effectively alleviates the issue of the limited number of molecules.

We extensively evaluate HI-Mol by designing several data-efficient molecular generation tasks on the datasets in the MoleculeNet benchmark (Wu et al., 2018) and on the QM9 dataset (Ramakrishnan et al., 2014). For instance, in the HIV dataset in MoleculeNet, HI-Mol improves Frechet ChemNet Distance (FCD, lower is better; Preuer et al., 2018) as 20.2 $\rightarrow$ 16.6 from prior arts. On the QM9 dataset, HI-Mol already outperforms the previous state-of-the-arts, e.g., STGG (Ahn et al., 2022) by 0.585 $\rightarrow$ 0.434 in FCD, with 50$\times$ less training data. Finally, we validate the effectiveness of the molecules generated by our HI-Mol framework on the low-shot property prediction tasks in MoleculeNet.

2. Related Work

Molecular generation. Most molecular generation methods fall into three categories. First, graph-based methods (Jo et al., 2022; Hoogeboom et al., 2022; Luo et al., 2022; Zhang et al., 2023; Vignac et al., 2023) formalize molecular generation as a graph generation problem by representing each molecule as an attributed graph. Next, fragment-based methods (Jin et al., 2018; Kong et al., 2022; Geng et al., 2023) define a dictionary of chemically meaningful fragments, e.g., functional groups. A molecule is represented as a tree structure of dictionary elements and the distribution of connected fragments is then modeled. Finally, string-based methods (Gómez-Bombarelli et al., 2016; Flam-Shepherd et al., 2022; Ahn et al., 2022) utilize the Simplified Molecular-Input Line-Entry System (SMILES, Weininger, 1988) to write molecules as strings and learn the distribution of molecules in this string space. Our method takes the string-based approach based on the recent large-scale text-to-molecule models that use the SMILES representation.

Hierarchical generation methods. Recent molecular generation methods introduce the notion of hierarchy in molecular generation (Jin et al., 2020; Zhu et al., 2023). Specifically, they incorporate the hierarchy *within* a single molecule, e.g., atom- and motif-level. However, they do not consider the hierarchy *among* molecules, e.g., dataset-, cluster-, and molecule-level, which is indeed crucial to understand the molecular dataset, i.e., target distribution (Alexander et al., 2011). To overcome this limitation, we carefully design our multi-level embeddings to reflect the hierarchy *among* the target molecules through our hierarchical tokens.

Molecular language model. Following the recent progress in text-conditional generative models, e.g., text-to-text (Raffel et al., 2020; Touvron et al., 2023) and text-to-image (Ramesh et al., 2021; Rombach et al., 2022), there exist several attempts to train text-to-molecule models, i.e., molecular language models (Bagal et al., 2021; Christofidellis et al., 2023; Liu et al., 2023). Specifically, these works exploit popular language model architectures to have pre-trained models for molecules, based on the SMILES (Weininger, 1988) representation that interprets a given molecule as a string. For instance, MolT5 (Edwards et al., 2022) proposes to fine-tune a large text-to-text language model, T5 (Raffel et al., 2020), with SMILES representations of large-scale molecular data and description-SMILES pair data to have a text-to-molecule model. Notably, it results in a highly effective pre-trained model for molecules, demonstrating superior performance across several text-to-molecule generation tasks. Building on its success, we mainly utilize the Large-Caption2Smiles model trained with this MolT5 approach for our goal of data-efficient molecular generation.[1]

Low-shot generation. In the field of generative models, there have been considerable efforts to design a low-shot generation framework for generating new samples from a given small number of data (Wang et al., 2018; Noguchi & Harada, 2019). Intriguingly, recent works on large-scale text-to-image diffusion models have surprisingly resolved this challenge, even enabling "personalization" of the model to a few in-the-wild images through simple optimization schemes that update only a few parameters of a pre-trained model (Gal et al., 2022; Cohen et al., 2022; Wei et al., 2023). In particular, textual inversion (Gal et al., 2022) exhibits that the personalization of large-scale text-to-image diffusion models with a small number of images can be achieved even with a very simple optimization of a single additional text token without updating any pre-trained model parameters.

In contrast to the recent advances of low-shot generation in the image domain, developing a low-shot (or data-efficient) molecular generation framework is relatively under-explored despite its importance in practical applications (Altae-Tran et al., 2017; Guo et al., 2022). Hence, our method tackles this problem by designing a molecule-specific inversion method using the recent large-scale text-to-molecule models. Specifically, we incorporate the concept of "hierarchy" of molecular structures (Alexander et al., 2011) into the textual inversion framework. Due to this unique motivation, our method effectively learns the molecule distribution with the shared concept in low-shot molecules with diverse molecular structures, while the applications of prior works, e.g., Guo et al. (2022), are limited to structurally similar low-shot molecules such as monomers and chain-extenders (see Table 7 for comparison).

3. HI-Mol: Hierarchical Textual Inversion for Molecular Generation

In Section 3.2, we provide descriptions of textual inversion to explain our method. In Section 3.1, we provide an overview of our problem and the main idea. In Section 3.3, we provide a component-wise description of our method.

3.1. Problem Description and Overview

We formulate our problem of *data-efficient molecular generation* as follows. Consider a given molecular data $\mathcal{M} := \{\mathbf{x}_n\}_{n=1}^{N}$, where each molecule $\mathbf{x}_n$ is drawn from an unknown task-related molecule distribution $p(\mathbf{x}|\mathbf{c})$. Here, $\mathbf{c}$ represents the common underlying chemical concept among molecules in the dataset for the target task, e.g., blood-brain barrier permeability. We aim to learn a model distribution $p_{\text{model}}(\mathbf{x})$ that matches $p(\mathbf{x}|\mathbf{c})$, where the number of molecules N is small, e.g., $N = 691$ in the BACE dataset.

To solve this problem, we take the recent approach of textual inversion (Gal et al., 2022) from the text-to-image model literature—a simple yet powerful technique in low-shot image generation that learns a common concept in given images as a single token in the text embedding space. Motivated by its success, we aim to learn the common chemical concept of molecules as text tokens and use them for our goal of data-efficient generation. However, we find that the naïve applications of inversion fail in molecules (see Table 1). Unlike images, molecules with similar semantics often have entirely different structures (see Figure 1), making it difficult to simply learn the common concept as a single text token. Our contribution lies in resolving this challenge by adopting molecule-specific priors, i.e., hierarchy, into the framework to enjoy the power of textual inversion techniques in achieving data-efficient molecular generation.

3.2. Preliminary: Textual Inversion

Recent text-to-image generation methods have proposed textual inversion (Gal et al., 2022), which aims to learn a common concept $\mathbf{c}$, i.e., the distribution $p(\mathbf{x}|\mathbf{c})$, from a

[1] We provide the results utilizing other text-to-molecule models (Christofidellis et al., 2023; Pei et al., 2023) in Appendix E.

Table 1. The ratio of valid generated molecules (Validity) based on naïve adoption of inversion methods in visual domain for data-efficient molecular generation in the HIV dataset (Wu et al., 2018).

Inversion method	Validity (%)
Textual Inversion (Gal et al., 2022)	0.4
DreamBooth (Ruiz et al., 2022)	0.0

small set of images and use it for the concept-embedded (or personalized) generation. To achieve this, they optimize a *single* text embedding of a new token $[S^*]$ shared among images to learn $\mathbf{c}$ using a frozen pre-trained text-to-image model $f_{\texttt{t2i}}$. Specifically, they put $[S^*]$ with a short text description, e.g., "A photo of $[S^*]$", as the text prompt to $f_{\texttt{t2i}}$, and then optimize this token embedding using given low-shot images with the exact same training objective that is used for training $f_{\texttt{t2i}}$. We propose to adapt the textual inversion method into the data-efficient molecular generation framework based on the recently proposed large-scale pre-trained text-to-molecule model (Edwards et al., 2022).

3.3. Detailed Description of HI-Mol

Failure of conventional inversion in molecules. We conduct an experiment to explore the applicability of existing inversion methods (Gal et al., 2022; Ruiz et al., 2022) in the visual domain for our goal of data-efficient molecular generation. These methods use a text prompt with a single shared token $[S^*]$ for the inversion of low-shot images based on the recent text-to-image models (Rombach et al., 2022; Saharia et al., 2022). Similarly, we apply their training objectives to low-shot molecules with a molecular language model (Edwards et al., 2022). In contrast to the success in the low-shot image generation tasks, in Table 1, we show that such naïve applications of inversion methods fail in the molecular domain, i.e., they do not generate enough valid molecules, which motivates us to design a molecule-specialized inversion for data-efficient molecular generation.

Hierarchical textual inversion. We first propose a molecule-specific textual inversion to learn the desired distribution of low-shot molecules. Unlike the original textual inversion (Gal et al., 2022) that assumes a single shared token $[S^*]$ only, we propose to use "hierarchical" tokens $[S^*], \{[I_k^*]\}_{k=1}^K, \{[D_n^*]\}_{n=1}^N$ (with parametrizations $\theta := (\mathbf{s}, \{\mathbf{i}_k\}_{k=1}^K, \{\mathbf{d}_n\}_{n=1}^N)$) by introducing additional intermediate tokens $\{[I_k^*]\}_{k=1}^K$ and detail tokens $\{[D_n^*]\}_{n=1}^N$ (with $K < N$) to extensively incorporate the hierarchical features in training molecules. For instance, intermediate and detail tokens enable to learn different level of features, i.e., cluster-wise and molecule-wise, respectively.

To learn these hierarchical tokens, we consider a frozen text-to-molecule model f, e.g., Large-Caption2Smiles (Edwards et al., 2022), to apply our proposed hierarchical textual inversion objective. Specifically, we optimize θ by minimizing the following objective on the given molecular dataset $\mathcal{M}$:

$$\mathcal{L}(\theta; \mathbf{x}_n) := \min_{k \in [1,K]} \mathcal{L}_{\texttt{CE}}\Big(f(\text{"The molecule is a } [S^*][I_k^*][D_n^*]\text{"}),\ \mathbf{x}_n\Big), \quad (1)$$

where $\mathcal{L}_{\texttt{CE}}$ denotes cross-entropy loss and $\mathbf{x}_n$ is represented as its corresponding SMILES (Weininger, 1988) string.[2] Thus, after training, each molecule $\mathbf{x}_n$ is interpreted as text tokens $[S^*][I_{c_n}^*][D_n^*]$, where we assign the intermediate token index $c_n \in [1, K]$ (for given $\mathbf{x}_n$ and the corresponding $[D_n^*]$) during optimization to minimize the training objective $\mathcal{L}$ (see Eq. (1)). We note that the selection of $[I_{c_n}^*]$ is achieved in an unsupervised manner so that it does not require specific information about each molecule. Intriguingly, we find that $[I_{c_n}^*]$ can learn some of the informative cluster-wise features through this simple selection scheme although we have not injected any prior chemical knowledge of the given molecular data (see Figure 2 for an example).

Our "multi-level" token design is particularly important for the successful inversion with molecules because molecules have a different nature from images that are typically used in the existing textual inversion method. Image inputs in the conventional textual inversion are visually similar, e.g., pictures of the same dog with various poses, whereas molecules often have entirely different structures even if they share the common chemical concept, e.g., activeness on the blood-brain membrane permeability (Wu et al., 2018). This difference makes it difficult to learn the common concept as a simple single token; we mitigate it by adopting hierarchy in the inversion scheme by incorporating the principle of the chemistry literature highlighting that molecular data can be clustered hierarchically (Alexander et al., 2011).

Embedding interpolation-based sampling. We propose a sampling strategy from the learned distribution via our hierarchical textual inversion framework. We find that the naïve application of the sampling schemes used in existing textual inversion for images, e.g., putting a text prompt including the shared token $[S^*]$ such as "A similar chemical of $[S^*]$" into the molecular language model f, does not show reasonable performance in molecular generation (see Table 1).

To alleviate this issue, we propose to utilize the learned hierarchy information of molecules obtained in our textual inversion, i.e., intermediate tokens $\{[I_k^*]\}_{k=1}^K$ and detail tokens $\{[D_n^*]\}_{n=1}^N$, to sample from our target distribution. We consider the interpolation of each of $[I_{c_n}^*]$ and $[D_n^*]$ in the sampling process. Specifically, we sample a novel molecule with random molecule indices i, j uniformly sampled from

[2] Our method is also applicable to any future text-to-molecule models that represent $\mathbf{x}_n$ as graphs or 3D point clouds by replacing $\mathcal{L}_{\texttt{CE}}$ with an appropriate objective to reconstruct $\mathbf{x}_n$.

Table 2. Quantitative results of the generated molecules on the three datasets (HIV, BBBP, BACE) in the MoleculeNet benchmark (Wu et al., 2018). We mark in Grammar if the method explicitly exploits the grammar of molecular data and thus yields a high Valid. score. The Active. score is averaged over three independently pre-trained classifiers. We report the results using the 500 non-overlapping generated molecules to the training dataset. We set the highest score in bold. ↑ and ↓ indicate higher and lower values are better, respectively.

Dataset	Method	Class	Grammar	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
HIV	GDSS (Jo et al., 2022)	Graph	✗	0.0	34.1	0.080	69.4	**100**	**100**
	DiGress (Vignac et al., 2023)	Graph	✗	0.0	26.2	0.067	17.8	**100**	**100**
	JT-VAE (Jin et al., 2018)	Fragment	✓	0.0	38.8	0.221	**100**	25.4	**100**
	PS-VAE (Kong et al., 2022)	Fragment	✓	3.7	21.8	0.053	**100**	91.4	**100**
	MiCaM (Geng et al., 2023)	Fragment	✓	3.4	20.4	0.037	**100**	81.6	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	3.3	29.7	0.064	30.0	**100**	**100**
	STGG (Ahn et al., 2022)	SMILES	✓	1.6	20.2	0.033	**100**	95.8	**100**
	HI-Mol (Ours)	SMILES	✗	**11.4**	19.0	**0.019**	60.6	94.1	**100**
	HI-Mol (Ours)	SMILES	✓	**11.4**	**16.6**	**0.019**	**100**	95.6	**100**
BBBP	GDSS (Jo et al., 2022)	Graph	✗	0.0	35.7	0.065	88.4	99.2	**100**
	DiGress (Vignac et al., 2023)	Graph	✗	8.2	17.4	0.033	43.8	94.6	**100**
	JT-VAE (Jin et al., 2018)	Fragment	✓	80.6	37.4	0.202	**100**	10.8	**100**
	PS-VAE (Kong et al., 2022)	Fragment	✓	84.9	17.3	0.039	**100**	91.6	**100**
	MiCaM (Geng et al., 2023)	Fragment	✓	82.0	14.3	0.021	**100**	89.4	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	88.8	20.2	0.026	54.0	**100**	**100**
	STGG (Ahn et al., 2022)	SMILES	✓	89.1	14.4	0.019	99.8	95.8	**100**
	HI-Mol (Ours)	SMILES	✗	94.4	11.2	0.011	78.8	92.9	**100**
	HI-Mol (Ours)	SMILES	✓	**94.6**	**10.7**	**0.009**	**100**	94.2	**100**
BACE	GDSS (Jo et al., 2022)	Graph	✗	9.1	66.0	0.205	73.4	**100**	**100**
	DiGress (Vignac et al., 2023)	Graph	✗	21.1	26.7	0.102	16.4	**100**	**100**
	JT-VAE (Jin et al., 2018)	Fragment	✓	40.4	49.1	0.304	**100**	13.0	**100**
	PS-VAE (Kong et al., 2022)	Fragment	✓	57.3	30.2	0.111	**100**	75.6	**100**
	MiCaM (Geng et al., 2023)	Fragment	✓	56.2	18.5	0.060	**100**	64.2	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	79.0	21.7	0.066	38.0	**100**	**100**
	STGG (Ahn et al., 2022)	SMILES	✓	42.9	17.6	0.053	**100**	94.8	**100**
	HI-Mol (Ours)	SMILES	✗	**81.0**	16.4	0.052	71.0	69.9	**100**
	HI-Mol (Ours)	SMILES	✓	80.4	**14.0**	**0.039**	**100**	74.4	**100**

$[1, N]$ and a coefficient λ drawn from a pre-defined prior distribution $p(\lambda)$ (see Appendix A for our choice of $p(\lambda)$):

$$(\bar{\mathbf{i}}, \bar{\mathbf{d}}) := \lambda(\mathbf{i}_{c_i}, \mathbf{d}_i) + (1-\lambda)(\mathbf{i}_{c_j}, \mathbf{d}_j),$$
$$\mathbf{x} := f(\text{“A similar chemical of } [S^*][\bar{I}^*][\bar{D}^*]\text{”}), \quad (2)$$

where $[\bar{I}^*]$, $[\bar{D}^*]$ indicate that we pass interpolated token embeddings $\bar{\mathbf{i}}, \bar{\mathbf{d}}$ to f, respectively, and $c_n \in [1, K]$ is an index of the intermediate token of a given molecule $\mathbf{x}_n$, i.e., an intermediate token index that minimizes the training objective in Eq. (1).[3] This additional consideration of low-level tokens $\{[I_k^*]\}_{k=1}^K, \{[D_n^*]\}_{n=1}^N$ (as well as $[S^*]$) encourages the sampling process to exploit the knowledge from the molecular dataset extensively, mitigating the issue of scarcity of molecules that lie in our desired molecule distribution and thus enables to generate high-quality molecules. We provide qualitative analysis of our sampling scheme in Appendix K.

[3]We simply set the number of clusters K to 10 in our experiments. Please see Appendix G for the analysis of K.

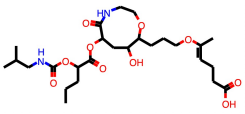
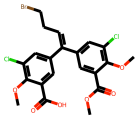
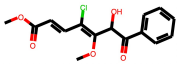
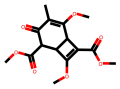
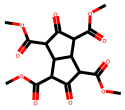
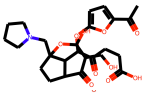
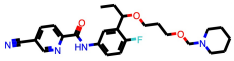
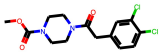
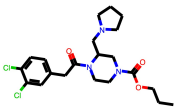
4. Experiments

We extensively verify the superiority of HI-Mol by considering various data-efficient molecular generation scenarios. In Section 4.1, we explain our experimental setup, e.g., datasets and metrics. In Section 4.2, we present our main molecular generation results on MoleculeNet and QM9 as well as the applicability of our generated molecules in the low-shot molecular property prediction tasks. In Section 4.3, we conduct some analysis and an ablation study to validate the effect of each component of our method. We present the application of HI-Mol on molecular optimization in Appendix F. We provide further ablation study and additional experimental results in Appendix G and H, respectively.

4.1. Experimental Setup

Datasets. Due to the lack of benchmarks designed particularly for data-efficient molecular generation, we propose to use the following datasets for evaluating molecular generation methods under our problem setup. First, we consider three datasets in the MoleculeNet (Wu et al., 2018) benchmark (originally designed for activity detection): HIV,

Table 3. Qualitative results of the generated molecules on the two datasets (HIV, BBBP) of the MoleculeNet benchmark (Wu et al., 2018). We visualize the generated molecules from each method that has the maximum Tanimoto similarity with a given anchor molecule. We report the similarity below each visualization of the generated molecule. We set the highest similarity in bold.

Dataset	DiGress (Vignac et al., 2023)	MiCaM (Geng et al., 2023)	STGG (Ahn et al., 2022)	**HI-Mol (Ours)**	Train
HIV	 0.154	 0.146	 0.157	 **0.326**	
BBBP	0.238	 0.247	 0.246	 **0.505**	

BBBP, and BACE, which have a significantly smaller number of molecules than popular molecular generation benchmarks (Sterling & Irwin, 2015; Polykovskiy et al., 2020). For example, BACE includes only 691 active molecules. Using the active molecules in each dataset, we construct tasks to generate novel molecules that share the chemical concept, e.g., blood-brain membrane permeability for BBBP. We also utilize these datasets to evaluate the quality of the generated molecules in low-shot molecular property prediction tasks.

Moreover, we utilize the QM9 dataset (Ramakrishnan et al., 2014) for our experiments to show the data-efficiency of HI-Mol. This dataset consists of more than 100k molecules, and thus has become a popular benchmark to evaluate large-scale molecular generation frameworks. Here, we train our method with an extremely small subset of the entire QM9 training split, e.g., 2% and 10%, whereas other baseline methods are trained on the entire training split. We provide more details about the datasets in Appendix B.

Evaluation setup. We consider six metrics that represent diverse aspects which are critical to the evaluation of the generated molecules, e.g., similarity to the target distribution, uniqueness, and novelty. We incorporate some well-known metrics, such as those used in Jo et al. (2022), as well as introducing a new metric "Active ratio":

- **Active ratio**[4] **(Active.)**: Our proposed metric, measuring the ratio of the valid generated molecules that are active, i.e., satisfying the target concept for each task.
- **Fréchet ChemNet Distance (FCD**; Preuer et al., 2018): Metric for measuring the distance between the source and the target distribution using pre-trained ChemNet.
- **Neighborhood Subgraph Pairwise Distance Kernel MMD (NSPDK**; Costa & De Grave, 2010): Another metric for measuring the gap between source and the target distributions, based on algorithmic computation using graph-based representations of molecules.
- **Validity (Valid.)**: The ratio of the generated molecules that have the chemically valid structure.
- **Uniqueness (Unique.)**: Diversity of the generated molecules based on the ratio of different samples over total valid molecules earned from the generative model.
- **Novelty**: Fraction of the valid molecules that are not included in the training set.

Baselines. We mainly consider the following recently proposed molecular generation methods for evaluation: GDSS (Jo et al., 2022), DiGress (Vignac et al., 2023), DEG (Guo et al., 2022), JT-VAE (Jin et al., 2018), PS-VAE (Kong et al., 2022), MiCaM (Geng et al., 2023), CRNN (Segler et al., 2018), and STGG (Ahn et al., 2022). For evaluation on the QM9 dataset (Ramakrishnan et al., 2014), we also consider GraphAF (Shi et al., 2020), GraphDF (Luo et al., 2021), MoFlow (Zang & Wang, 2020), EDP-GNN (Niu et al., 2020), and GraphEBM (Liu et al., 2021), following the recent works (Jo et al., 2022; Luo et al., 2022). We provide more details of the baselines in Appendix D.

4.2. Main Results

Generation on MoleculeNet. Table 2 summarizes the quantitative results of the generated molecules on the HIV, BBBP, and BACE datasets in the MoleculeNet benchmark (Wu et al., 2018). Our method consistently outperforms other generation methods in terms of Active ratio, FCD, and NSPDK scores on all three datasets. We note that the improvements in these scores are particularly crucial for the

[4]For reliable evaluation with our metric, we avoid the overlap between the generated molecules and the training data used for generation methods by ignoring the molecule if it is contained in this dataset. Hence, the Novelty score is 100 for all MoleculeNet experiments since all samples are different from the training set (see Table 2 for an example). We provide the detailed description of this metric in Appendix C.

Table 4. Quantitative results of the generated molecules on the QM9 dataset (Ramakrishnan et al., 2014). We mark in Grammar if the method explicitly exploits the grammar of molecular data and thus yields a high Valid. score. Following the setup of Jo et al. (2022), we report the results using 10,000 sampled molecules. We denote the scores drawn from Luo et al. (2022) and Ahn et al. (2022) with (*) and (†), respectively. We mark (-) when the score is not available in the literature. We set the highest score in bold. ↑ and ↓ indicate higher and lower values are better, respectively. For our method, we report the ratio of the number of samples of the dataset used for training.

Method	Class	Grammar	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
CG-VAE† (Liu et al., 2018)	Graph	✓	1.852	-	**100**	98.6	94.3
GraphAF (Shi et al., 2020)	Graph	✗	5.268	0.020	67	94.5	88.8
MoFlow (Zang & Wang, 2020)	Graph	✗	4.467	0.017	91.4	98.7	94.7
EDP-GNN (Niu et al., 2020)	Graph	✗	2.680	0.005	47.5	**99.3**	86.6
GraphDF (Luo et al., 2021)	Graph	✗	10.82	0.063	82.7	97.6	**98.1**
GraphEBM (Liu et al., 2021)	Graph	✗	6.143	0.030	8.22	97.8	97.0
GDSS (Jo et al., 2022)	Graph	✗	2.900	0.003	95.7	98.5	86.3
GSDM* (Luo et al., 2022)	Graph	✗	2.650	0.003	99.9	-	-
STGG† (Ahn et al., 2022)	SMILES	✓	0.585	-	**100**	95.6	69.8
HI-Mol (Ours; 2%)	SMILES	✓	0.430	**0.001**	**100**	76.1	75.6
HI-Mol (Ours; 10%)	SMILES	✓	**0.398**	**0.001**	**100**	88.3	73.2

Table 5. Average ΔROC-AUC of the low-shot property prediction tasks in the datasets in the MoleculeNet (Wu et al., 2018) benchmark. The results are averaged over 20 random seeds.

Dataset	Method	16-shot	32-shot
HIV	DiGress (Vignac et al., 2023)	-2.30	-2.67
	MiCaM (Geng et al., 2023)	1.02	0.69
	STGG (Ahn et al., 2022)	0.53	-0.47
	HI-Mol (Ours)	**2.35**	**2.16**
BBBP	DiGress (Vignac et al., 2023)	1.73	0.97
	MiCaM (Geng et al., 2023)	1.91	1.78
	STGG (Ahn et al., 2022)	1.85	1.76
	HI-Mol (Ours)	**2.73**	**2.64**
BACE	DiGress (Vignac et al., 2023)	-0.60	-0.91
	MiCaM (Geng et al., 2023)	-0.65	-1.11
	STGG (Ahn et al., 2022)	2.34	2.01
	HI-Mol (Ours)	**3.53**	**3.39**

deployment of the molecular generation method. For example, the superior Active ratio of HI-Mol, e.g., 3.7 → 11.4 on the HIV dataset, indicates that the generated molecules are more likely to exhibit the desired activeness on our target task. Our method also significantly improves the FCD metric by 20.2 → 19.0 and the NSPDK metric by 0.033 → 0.019 on the HIV dataset. These improvements highlight the effectiveness of HI-Mol in generating more faithful molecules that lie in the target distribution. We provide qualitative results in Table 3 by visualizing some of the generated molecules from each dataset. We observe that the molecules generated by HI-Mol capture several crucial common substructures, e.g., many ester groups, while introducing the novel components, e.g., 4-membered ring, due to our unique hierarchical inversion framework.

We also propose a simple algorithm to modify the generated invalid SMILES strings by correcting invalid patterns[5] without a computational overhead. By applying this modification algorithm, we convert an invalid SMILES string to a valid one that represents a valid molecule, therefore, the Validity score becomes 100 in this case. In particular, the molecules from the modified SMILES further improve the overall metrics, e.g., FCD by 19.0 → 16.6 and 11.2 → 10.7 in the HIV and the BBBP dataset, respectively. This indicates that the modified SMILES indeed represent molecules from the desired low-shot molecule distribution and further highlights the superior quality of our generated molecules.

Generation on QM9. In Table 4, we report the quantitative results of the generated molecules from each method. Here, we train our method with a limited portion of data, e.g., 2% and 10%, and then compare the results with the baselines that are trained on the entire dataset. Our model shows strong data-efficiency : only with a 2% subset of the training data, our method already outperforms the state-of-the-art baseline, STGG (Ahn et al., 2022), by 0.585 → 0.430 in FCD. Utilizing a 10% subset further improves the performance of HI-Mol, reducing the FCD by 0.430 → 0.398. In particular, compared with STGG, HI-Mol not only improves the FCD score but also shows a better Novelty score, which validates the capability of HI-Mol to find unseen novel molecules from the desired target distribution.

For an extensive comparison with the baselines which show high Uniqueness and Novelty scores, e.g., GDSS (Jo et al., 2022), we perform an additional comparison after we adjust

[5]For example, we modify the invalid SMILES caused by the unclosed ring, e.g., `C1CCC` → `CCCCC`. Please see Appendix I for the detailed algorithm. We mark in the Grammar column in Table 2 and 4 when modification is applied for evaluation.

Table 6. Ablation of the components of hierarchical textual inversion on the HIV dataset in the MoleculeNet (Wu et al., 2018) benchmark.

Inversion	Inverted tokens	Grammar	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
✗	-	✗	0.0	65.3	0.450	0.4	**100**	**100**
✓	$[S^*]$	✗	0.0	64.2	0.448	0.4	**100**	**100**
✓	$[S^*][D^*]$	✗	10.2	20.3	0.021	60.0	89.3	**100**
✓	$[S^*][I^*][D^*]$	✗	**11.4**	19.0	**0.019**	60.6	94.1	**100**
✓	$[S^*][I^*][D^*]$	✓	**11.4**	**16.6**	**0.019**	**100**	95.6	**100**

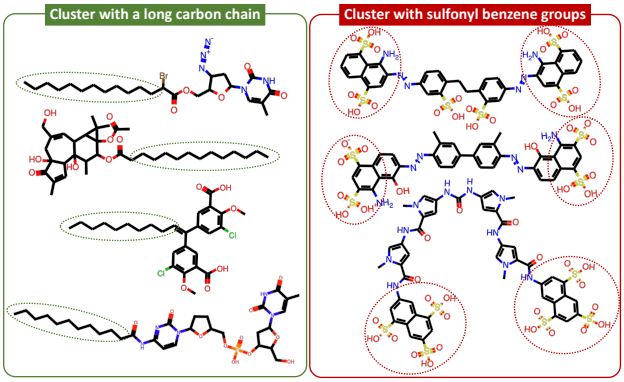


Figure 2. Visualizations of molecules in two different clusters obtained from the unsupervised clustering objective with the intermediate tokens in Eq. (1) on the HIV dataset (Wu et al., 2018).

the Uniqueness and Novelty scores of our method to 100; this setup allows us to perform a fair comparison in the FCD score with these methods. Here, we adjust the sampling strategy slightly; we ignore the generated molecules which have an overlap with the training molecules and the already generated molecules. Even in this case, HI-Mol achieves an FCD of 0.601, which outperforms all these baselines. We provide detailed results and discussion in Appendix J.

Low-shot molecular property prediction. We show that the molecules generated by HI-Mol can be utilized to improve the performance of classifiers for low-shot molecular property prediction. Here, we collect both active and inactive low-shot molecules for each dataset (HIV, BBBP, and BACE) from the MoleculeNet benchmark (Wu et al., 2018). We separately train molecular generative models for active and inactive molecules, and then generate molecules from the models. In Table 5, we report ΔROC-AUC scores[6] from each method. We find that HI-Mol consistently shows the superior ΔROC-AUC scores in various low-shot property prediction tasks. This demonstrates the efficacy of HI-Mol to learn the common concept, i.e., activeness and inactiveness, of each molecular property prediction task even with a limited number of molecules. In practical scenarios, where the label information is hard to achieve, our HI-Mol indeed plays an important role in improving the classifier. We provide experimental details in Appendix N.

[6]This score is calculated by the improvement in the ROC-AUC score when the generated molecules are additionally added to the original low-shot training data; higher is better.

Extremely limited data regime. Since our model exploits the power of large molecular language models by designing a molecule-specialized textual inversion scheme, one can expect our model to be beneficial in extremely limited data regimes compared with prior methods. To verify this, we conduct an experiment using only subset of the HIV dataset and report its quantitative result in Table 7. Even with this situation, HI-Mol still outperforms prior state-of-the-art molecular generation methods, e.g., our method improves FCD as 39.2 → 34.8 when trained with 30 samples.

4.3. Analysis

Effect of intermediate tokens. Recall that we have introduced intermediate tokens $\{[I_k^*]\}_{k=1}^K$ in our hierarchical textual inversion framework, which are selected in an unsupervised manner during the inversion to learn some of the cluster-wise features included in given molecules (see Eq. (1)). To validate the effect of our text token design, we visualize the clustering results in Figure 2 by providing groups of the molecules that are assigned to the same intermediate token. As shown in this figure, molecules are well grouped according to their common substructures, e.g., a long carbon chain or sulfonyl benzene groups. Such a learning of cluster-wise low-level semantics is indeed beneficial in molecular generation, since molecules often share the same chemical concept, e.g., blood-membrane permeability, even when they have large structural differences.

Ablation on hierarchical textual inversion. To validate the effectiveness of each component in our HI-Mol framework, we compare the results where some components are excluded from the overall framework. Specifically, we compare the generation performance of the following setups: (1) not using the inversion technique; we train the text-to-molecule model with the molecule-description pairs, (2) using the shared token $[S^*]$ only, (3) using $[S^*]$ and the detail tokens $[D_n^*]$, (4) using all three types of tokens, and (5) applying the additional modification algorithm. Note that for (1) and (2), it is impossible to apply our interpolation-based sampling; instead, we use temperature sampling with temperature $\tau = 2.0$. We provide this result in Table 6.

Table 7. Results of molecular generation on subsets of the HIV dataset (Wu et al., 2018). We generate the same number of molecules as the number of the training samples. Due to the large training cost, we report the score of DEG (Guo et al., 2022) only for 30 samples.

# Samples	Method	Class	Grammar	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
30	DEG (Guo et al., 2022)	Graph	✓	3.3	39.2	0.105	**100**	**100**	**100**
	STGG (Ahn et al., 2022)	SMILES	✓	0.0	41.5	0.110	**100**	67	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	0.0	40.0	0.121	80	71	**100**
	HI-Mol (Ours)	SMILES	✗	**8.3**	**34.8**	**0.103**	80	75	**100**
150	STGG (Ahn et al., 2022)	SMILES	✓	1.3	28.2	0.054	**100**	90	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	1.3	30.1	0.063	50	84	**100**
	HI-Mol (Ours)	SMILES	✗	**8.3**	**22.1**	**0.038**	64	**91**	**100**
500	STGG (Ahn et al., 2022)	SMILES	✓	1.3	22.8	0.041	**100**	74	**100**
	CRNN (Segler et al., 2018)	SMILES	✗	2.7	30.0	0.064	51	**100**	**100**
	HI-Mol (Ours)	SMILES	✗	**10.3**	**20.8**	**0.020**	63	91	**100**

First, we find that (1) the naïve training and (2) the inversion with a single shared token (Gal et al., 2022) do not show reasonable performance, i.e., they achieve only 0.4% Validity. In (3) and (4), introducing low-level tokens in the inversion framework significantly improves the generation quality by learning the low-level features in molecules. Finally, (5) the modification algorithm converts an invalid generated SMILES into a valid one that lies in our target distribution. We provide additional ablation results in Appendix G.

5. Conclusion

We propose HI-Mol, a data-efficient molecular generation framework that utilizes a molecule-specialized textual inversion scheme. Specifically, we propose to capture the hierarchical information of molecular data in the inversion stage, and use it to sample novel molecules. We hope our method initiates under-explored but crucial research direction in the data-efficient generation of molecules.

Limitation and future work. In this work, we apply our novel hierarchical textual inversion scheme to the molecular language model (Edwards et al., 2022), where developing such a model is a very recently considered research direction. An important future work would be improving the large-scale molecular language models themselves, e.g., the breakthroughs in the image domain (Rombach et al., 2022), which will allow more intriguing applications of our HI-Mol framework, such as composition (see Appendix H).

Impact Statement

This work will facilitate research in molecular generation, which can speed up the development of many important generation tasks such as finding drugs for a specific organ and disease when the hit molecules are rarely known. However, malicious use of well-learned molecular generative model poses a potential threat of creating hazardous molecules, such as toxic chemical substances. It is an important research direction to prevent malicious usages of generative models (Achiam et al., 2023). On the other hand, molecular generation is also essential for generating molecules to defend against harmful substances, so the careful use of our work, HI-Mol, can lead to more positive effects.

Acknowledgements

We thank all the anonymous reviewers for their insightful feedbacks and discussions. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2019-II190075, Artificial Intelligence Graduate School Program(KAIST), No.RS-2021-II212068, Artificial Intelligence Innovation Hub) and Mogam institute for Biomedical Research and GC Biopharma.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ahn, S., Chen, B., Wang, T., and Song, L. Spanning tree-based graph generation for molecules. In *International Conference on Learning Representations*, 2022.

Alexander, N., Woetzel, N., and Meiler, J. bcl:: Cluster: A method for clustering biological molecules coupled with visualization in the pymol molecular graphics system. In *2011 IEEE 1st international conference on computational advances in bio and medical sciences (ICCABS)*, pp. 13–18. IEEE, 2011.

Altae-Tran, H., Ramsundar, B., Pappu, A. S., and Pande, V. Low data drug discovery with one-shot learning. *ACS central science*, 3(4):283–293, 2017.

Bagal, V., Aggarwal, R., Vinod, P., and Priyakumar, U. D. Molgpt: molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 62(9):2064–2076, 2021.

Bongini, P., Bianchini, M., and Scarselli, F. Molecular generative graph neural networks for drug discovery. *Neurocomputing*, 450:242–252, 2021.

Brown, N., Fiscato, M., Segler, M. H., and Vaucher, A. C. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

Christofidellis, D., Giannone, G., Born, J., Winther, O., Laino, T., and Manica, M. Unifying molecular and textual representations via multi-task language modelling. *arXiv preprint arXiv:2301.12586*, 2023.

Cohen, N., Gal, R., Meirom, E. A., Chechik, G., and Atzmon, Y. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *European Conference on Computer Vision*, pp. 558–577. Springer, 2022.

Coley, C. W. Defining and exploring chemical spaces. *Trends in Chemistry*, 3(2):133–145, 2021.

Costa, F. and De Grave, K. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 255–262. Omnipress; Madison, WI, USA, 2010.

Dai, H., Tian, Y., Dai, B., Skiena, S., and Song, L. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*, 2018.

Dao, T., Fu, D., Ermon, S., Rudra, A., and Ré, C. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

Drew, K. L., Baiman, H., Khwaounjoo, P., Yu, B., and Reynisson, J. Size estimation of chemical space: how big is it? *Journal of Pharmacy and Pharmacology*, 64(4): 490–495, 2012.

Drews, J. Drug discovery: a historical perspective. *science*, 287(5460):1960–1964, 2000.

Edwards, C., Lai, T., Ros, K., Honke, G., Cho, K., and Ji, H. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 375–413, 2022.

Flam-Shepherd, D., Zhu, K., and Aspuru-Guzik, A. Language models can learn complex molecular distributions. *Nature Communications*, 13(1):3293, 2022.

Fu, T., Gao, W., Xiao, C., Yasonik, J., Coley, C. W., and Sun, J. Differentiable scaffolding tree for molecular optimization. *arXiv preprint arXiv:2109.10469*, 2021.

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-Or, D. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

Gao, W. and Coley, C. W. The synthesizability of molecules proposed by generative models. *Journal of chemical information and modeling*, 60(12):5714–5723, 2020.

Gao, W., Fu, T., Sun, J., and Coley, C. Sample efficiency matters: a benchmark for practical molecular optimization. *Advances in Neural Information Processing Systems*, 35:21342–21357, 2022.

Geng, Z., Xie, S., Xia, Y., Wu, L., Qin, T., Wang, J., Zhang, Y., Wu, F., and Liu, T.-Y. De novo molecular generation via connection-aware motif mining. *arXiv preprint arXiv:2302.01129*, 2023.

Gómez-Bombarelli, R., Aguilera-Iparraguirre, J., Hirzel, T. D., Duvenaud, D., Maclaurin, D., Blood-Forsythe, M. A., Chae, H. S., Einzinger, M., Ha, D.-G., Wu, T., et al. Design of efficient molecular organic light-emitting diodes by a high-throughput virtual screening and experimental approach. *Nature materials*, 15(10):1120–1127, 2016.

Guo, M., Thost, V., Li, B., Das, P., Chen, J., and Matusik, W. Data-efficient graph grammar learning for molecular generation. *arXiv preprint arXiv:2203.08031*, 2022.

Hamdia, K. M., Ghasemi, H., Bazi, Y., AlHichri, H., Alajlan, N., and Rabczuk, T. A novel deep learning based method for the computational material design of flexoelectric nanostructures with topology optimization. *Finite Elements in Analysis and Design*, 165:21–30, 2019.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Hoogeboom, E., Satorras, V. G., Vignac, C., and Welling, M. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

Hughes, J. P., Rees, S., Kalindjian, S. B., and Philpott, K. L. Principles of early drug discovery. *British journal of pharmacology*, 162(6):1239–1249, 2011.

Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768, 2012.

Jensen, J. H. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical science*, 10(12):3567–3572, 2019.

Jin, W., Barzilay, R., and Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.

Jin, W., Barzilay, R., and Jaakkola, T. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*, pp. 4839–4848. PMLR, 2020.

Jo, J., Lee, S., and Hwang, S. J. Score-based generative modeling of graphs via the system of stochastic differential equations. In *International Conference on Machine Learning*, pp. 10362–10383. PMLR, 2022.

Kajino, H. Molecular hypergraph grammar with its application to molecular optimization. In *International Conference on Machine Learning*, pp. 3183–3191. PMLR, 2019.

Kong, X., Huang, W., Tan, Z., and Liu, Y. Molecule generation by principal subgraph mining and assembling. *Advances in Neural Information Processing Systems*, 35: 2550–2563, 2022.

Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. Grammar variational autoencoder. In *International conference on machine learning*, pp. 1945–1954. PMLR, 2017.

Liu, M., Yan, K., Oztekin, B., and Ji, S. Graphebm: Molecular graph generation with energy-based models. *arXiv preprint arXiv:2102.00546*, 2021.

Liu, Q., Allamanis, M., Brockschmidt, M., and Gaunt, A. Constrained graph variational autoencoders for molecule design. *Advances in neural information processing systems*, 31, 2018.

Liu, S., Nie, W., Wang, C., Lu, J., Qiao, Z., Liu, L., Tang, J., Xiao, C., and Anandkumar, A. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023.

Luo, T., Mo, Z., and Pan, S. J. Fast graph generative model via spectral diffusion. *arXiv preprint arXiv:2211.08892*, 2022.

Luo, Y., Yan, K., and Ji, S. Graphdf: A discrete flow model for molecular graph generation. In *International Conference on Machine Learning*, pp. 7192–7203. PMLR, 2021.

Nam, J., Tack, J., Lee, K., Lee, H., and Shin, J. STUNT: Few-shot tabular learning with self-generated tasks from unlabeled tables. In *International Conference on Learning Representations*, 2023.

Niu, C., Song, Y., Song, J., Zhao, S., Grover, A., and Ermon, S. Permutation invariant graph generation via score-based generative modeling. In *International Conference on Artificial Intelligence and Statistics*, pp. 4474–4484. PMLR, 2020.

Noguchi, A. and Harada, T. Image generation from small datasets via batch statistics adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2750–2758, 2019.

Pei, Q., Zhang, W., Zhu, J., Wu, K., Gao, K., Wu, L., Xia, Y., and Yan, R. Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. *arXiv preprint arXiv:2310.07276*, 2023.

Polykovskiy, D., Zhebrak, A., Sanchez-Lengeling, B., Golovanov, S., Tatanov, O., Belyaev, S., Kurbanov, R., Artamonov, A., Aladinskiy, V., Veselov, M., et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 11:565644, 2020.

Preuer, K., Renz, P., Unterthiner, T., Hochreiter, S., and Klambauer, G. Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. *Journal of chemical information and modeling*, 58(9):1736–1741, 2018.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Ramakrishnan, R., Dral, P. O., Rupp, M., and Von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022.

Schneider, G. and Fechner, U. Computer-based de novo design of drug-like molecules. *Nature Reviews Drug Discovery*, 4(8):649–663, 2005.

Segler, M., Kogej, T., Tyrchan, C., and Waller, M. Generating focused molecule libraries for drug discovery with recurrent neural networks. acs cent sci 4 (1): 120–131. *arXiv preprint arXiv:1701.0132*, 9, 2018.

Shi, C., Xu, M., Zhu, Z., Zhang, W., Zhang, M., and Tang, J. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*, 2020.

Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Stanley, M., Bronskill, J. F., Maziarz, K., Misztela, H., Lanini, J., Segler, M., Schneider, N., and Brockschmidt, M. Fs-mol: A few-shot learning dataset of molecules. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Sterling, T. and Irwin, J. J. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.

Tagade, P. M., Adiga, S. P., Pandian, S., Park, M. S., Hariharan, K. S., and Kolake, S. M. Attribute driven inverse materials design using deep learning bayesian framework. *npj Computational Materials*, 5(1):127, 2019.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023.

Wang, Y., Wu, C., Herranz, L., Van de Weijer, J., Gonzalez-Garcia, A., and Raducanu, B. Transferring gans: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 218–234, 2018.

Wang, Y., Wang, W., Liang, Y., Cai, Y., and Hooi, B. Mixup for node and graph classification. In *Proceedings of the Web Conference 2021*, pp. 3663–3674, 2021.

Wei, Y., Zhang, Y., Ji, Z., Bai, J., Zhang, L., and Zuo, W. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848*, 2023.

Weininger, D. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

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

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

Xu, Y., Lin, K., Wang, S., Wang, L., Cai, C., Song, C., Lai, L., and Pei, J. Deep learning for molecular generation. *Future medicinal chemistry*, 11(6):567–597, 2019b.

Xue, D., Gong, Y., Yang, Z., Chuai, G., Qu, S., Shen, A., Yu, J., and Liu, Q. Advances and challenges in deep generative models for de novo molecule generation. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 9(3):e1395, 2019.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., and Shen, Y. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

Yu, S., Sohn, K., Kim, S., and Shin, J. Video probabilistic diffusion models in projected latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18456–18466, 2023.

Zang, C. and Wang, F. Moflow: an invertible flow model for generating molecular graphs. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 617–626, 2020.

Zhang, Z., Liu, Q., Lee, C.-K., Hsieh, C.-Y., and Chen, E. An equivariant generative framework for molecular graph-structure co-design. *Chemical Science*, 14(31): 8380–8392, 2023.

Zhu, Y., Ouyang, Z., Liao, B., Wu, J., Wu, Y., Hsieh, C.-Y., Hou, T., and Wu, J. Molhf: A hierarchical normalizing flow for molecular graph generation. *arXiv preprint arXiv:2305.08457*, 2023.

Appendix: Data-Efficient Molecular Generation with Hierarchical Textual Inversion

A. Method Details

We utilize a recently introduced text-to-molecule model, MolT5-Large-Caption2Smiles (Edwards et al., 2022) in our HI-Mol framework.[7] This model is constructed upon a text-to-text model, T5 (Raffel et al., 2020), and molecular information is injected by additional training with both unpaired SMILES (Weininger, 1988) string and caption-SMILES paired dataset. We update the token embeddings and the linear heads, while freezing other parameters. Our experiment is conducted for 1,000 epochs using a single NVIDIA GeForce RTX 3090 GPU with a batch size of 4. We use AdamW optimizer with $\epsilon = 1.0 \times 10^{-8}$ and let the learning rate 0.3 with linear scheduler. We clip gradients with the maximum norm of 1.0. We update the assigned cluster c_n of each molecule for the first 5 epochs following Eq. (1). For interpolation-based sampling, we choose a uniform distribution $p(\lambda)$, (i.e., $p(\lambda) := \mathcal{U}(l, 1-l)$), where λ controls relative contributions of interpolated token embeddings. We set $l = 0.0$ on the datasets in MoleculeNet benchmark (Wu et al., 2018), and $l = 0.3$ on the QM9 dataset (Ramakrishnan et al., 2014).

B. Datasets

MoleculeNet dataset. We perform generation experiments on single-task datasets, HIV, BBBP, and BACE, from MoleculeNet (Wu et al., 2018) benchmark. For each dataset, molecules are labeled with 0 or 1, based on its activeness of the target property:

- *HIV* consists of molecules and its capability to prevent HIV replication.
- *BBBP* consists of molecules and whether each compound is permeable to the blood-brain barrier.
- *BACE* consists of molecules and its binding results for a set of inhibitors of β-secretase-1.

We collect active (e.g., label-1) molecules to train molecular generative models. We utilize a common splitting scheme for MoleculeNet dataset, *scaffold split* with split ratio of train:valid:test = 80:10:10 (Wu et al., 2018). We emphasize that such *scaffold split* is widely considered in molecular generation domain (Ahn et al., 2022). Additional statistics for datasets on MoleculeNet are provided in Table 8.

Table 8. MoleculeNet downstream classification dataset statistics

Dataset	HIV	BBBP	BACE
Number of molecules	41,127	2,039	1,513
Number of active molecules	1,443	1,567	691
Avg. Node	25.51	24.06	34.08
Avg. Degree	54.93	51.90	73.71

QM9 dataset. We perform generation experiments on the QM9 dataset (Ramakrishnan et al., 2014), which is a widely adopted to benchmark molecular generation methods. This dataset consists of 133,885 small orginic molecules. We follow the dataset splitting scheme of (Ahn et al., 2022) and randomly subset the training split with 2%, 5%, 10%, 20% and 50% ratio for training our HI-Mol.

[7] https://huggingface.co/laituan245/molt5-large-caption2smiles

C. Evaluation Metrics

We mainly utilize 6 metrics to incorporate diverse aspects for evaluation of the generated molecules. We adopt 5 metrics (FCD, NSPDK, Validity, Uniqueness, Novelty) used in (Jo et al., 2022):

- **Fréchet ChemNet Distance (FCD)** (Preuer et al., 2018) evaluates the distance between the generated molecules and test molecules using the activations of the penultimate layer of the ChemNet, similar to popular Fréchet inception distance (FI) used in image domain (Heusel et al., 2017):

$$\texttt{FCD} \coloneqq \|m - m_g\|_2^2 + \mathrm{Tr}\big(C + C_g - 2(CC_g)^{1/2}\big), \tag{3}$$

  where m, C are the mean and covariance of the activations of the test molecules, and m_g, C_g are the mean and covariance of the activations of the generated molecules.

- **Neighborhood Subgraph Pairwise Distance Kernel MMD (NSPDK)** (Costa & De Grave, 2010) calculates the maximum mean discrepancy between the generated molecules and test molecules. We follow the evaluation protocol in (Jo et al., 2022), to incorporate both atom and bond features.
- **Validity (Valid.)** is the ratio of the generated molecules that does not violate chemical validity, e.g., molecules that obey the valency rule.
- **Uniqueness (Unique.)** is the ratio of different samples over total valid generated molecules.
- **Novelty** is the ratio of valid generated molecules that are not included in the training set.

We introduce an additional metric (Active ratio) to evaluate how the generated molecules are likely to be active, e.g., label-1 on our target property:

- **Active ratio (Active.)** is the ratio of the valid generated molecules that are active.

We utilize pre-trained classifiers to measure the activeness of the generated molecules. To be specific, we train a graph isomorphism network (GIN, Xu et al., 2019a) with the entire training split, e.g., contains both active (label-1) and inactive (label-0) molecules, of each dataset in the MoleculeNet benchmark (Wu et al., 2018). We train 5-layer GIN with a linear projection layer for 100 epochs with Adam optimizer, a batch size of 256, a learning rate of 0.001, and a dropout ratio of 0.5. We select the classifier of the epoch with the best validation accuracy. The accuracies of the pre-trained classifier on the validation split are 98.2%, 86.3%, and 86.1%, respectively. We calculate Active ratio by the ratio of the generated molecules that this classifier classifies as label-1.

D. Baselines

In this paper, we compare our method with an extensive list of baseline methods in the literature of molecular generation. We provide detailed descriptions of the baselines we considered:

- **GDSS** (Jo et al., 2022) proposes a diffusion model for graph structure, jointly learning both node and adjacency space by regarding each attributes as continuous values.
- **DiGress** (Vignac et al., 2023) proposes a discrete diffusion process for graph structure to properly consider categorical distributions of node and edge attributes.
- **DEG** (Guo et al., 2022) suggests constructing molecular grammars from automatically learned production rules for data-efficient generation of molecules. Due to the high computational complexity of the grammar construction, this method can only be applied to structurally similar molecules, e.g., monomers or chain-extenders, with an extremely limited number of molecules ($\sim$100 molecules with high structural similarity). Nevertheless, we compare with this method in the extremely limited data regime of Appendix H.
- **JT-VAE** (Jin et al., 2018) proposes a variational auto-encoder that represents molecules as junction trees, regarding motifs of molecules as the nodes of junction trees.
- **PS-VAE** (Kong et al., 2022) utilizes a principal subgraph as a building block of molecules and generates molecules via merge-and-update subgraph extraction.
- **MiCaM** (Geng et al., 2023) introduces a connection-aware motif mining method to model the target distribution with the automatically discovered motifs.
- **CRNN** (Segler et al., 2018) builds generative models of SMILES strings with recurrent decoders.
- **STGG** (Ahn et al., 2022) introduces a spanning tree-based molecule generation which learns the distribution of intermediate molecular graph structure with tree-constructive grammar.
- **GraphAF** (Shi et al., 2020) proposes an auto-regressive flow-based model for graph generation.
- **GraphDF** (Luo et al., 2021) introduces an auto-regressive flow-based model with discrete latent variables.
- **MoFlow** (Zang & Wang, 2020) utilizes a flow-based model for one-shot molecular generation.
- **EDP-GNN** (Niu et al., 2020) proposes a one-shot score-based molecular generative model, utilizing a discrete-step perturbation procedure of node and edge attributes.
- **GraphEBM** (Liu et al., 2021) introduces a one-shot energy-based model to generate molecules by minimizing energies with Langevin dynamics.
- **GSDM** (Luo et al., 2022) is a follow-up work of GDSS (Jo et al., 2022), suggesting to consider the spectral values of adjacency matrix instead of adjacency matrix itself.
- **CG-VAE** (Liu et al., 2018) proposes a recursive molecular generation framework that generates molecules satisfying the valency rules by masking out the action space.

E. Results with Other Molecular Language Models

Table 9. Quantitative results of generated molecules with HI-Mol varying the molecular language models.

Dataset	Model	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
HIV	MolT5 (Edwards et al., 2022)	**11.4**	**16.6**	**0.019**	**100**	95.6	**100**
	ChemT5 (Christofidellis et al., 2023)	10.8	16.8	**0.019**	**100**	98.6	**100**
	BioT5 (Pei et al., 2023)	10.1	16.9	0.023	**100**	**99.4**	**100**
BBBP	MolT5 (Edwards et al., 2022)	**94.6**	**10.7**	**0.009**	**100**	94.2	**100**
	ChemT5 (Christofidellis et al., 2023)	93.2	10.8	0.011	**100**	96.6	**100**
	BioT5 (Pei et al., 2023)	92.8	11.4	0.013	**100**	**99.0**	**100**
BACE	MolT5 (Edwards et al., 2022)	80.4	14.0	0.039	**100**	74.4	**100**
	ChemT5 (Christofidellis et al., 2023)	**83.1**	**13.6**	**0.036**	**100**	87.0	**100**
	BioT5 (Pei et al., 2023)	82.4	14.3	0.038	**100**	**98.6**	**100**

In Table 9, we show the experimental results of our HI-Mol framework based on varying molecular language models. We utilize Large-Caption2Smiles (MolT5, Edwards et al., 2022), Text+Chem T5-augm (ChemT5, Christofidellis et al., 2023), and BioT5 (BioT5, Pei et al., 2023). The results show that the performance of HI-Mol is consistent across various molecular language models, i.e., HI-Mol framework reliably generates high quality molecules that lie in the desired target distribution.

F. Offline Molecular Property Optimization

Table 10. Results of molecular property maximization task. We report the top-3 property scores denoted by `1st`, `2nd`, and `3rd`. The baseline scores are drawn from Ahn et al. (2022).

			PlogP		
Method	Class	Offline	`1st`	`2nd`	`3rd`
GVAE (Kusner et al., 2017)	SMILES	✓	2.94	2.89	2.80
SD-VAE (Dai et al., 2018)	Syntax Tree	✓	4.04	3.50	2.96
JT-VAE (Jin et al., 2018)	Fragment	✗	5.30	4.93	4.49
MHG-VAE (Kajino, 2019)	Fragment	✗	5.56	5.40	5.34
GraphAF (Shi et al., 2020)	Graph	✗	12.23	11.29	11.05
GraphDF (Luo et al., 2021)	Graph	✗	13.70	13.18	13.17
STGG (Ahn et al., 2022)	SMILES	✓	23.32	18.75	16.50
HI-Mol (Ours; 1%)	SMILES	✓	**24.67**	**21.72**	**20.73**

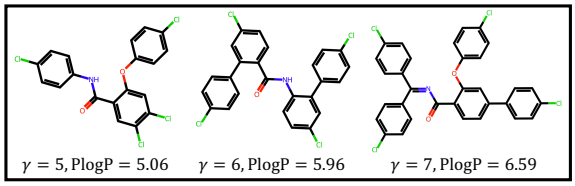


Figure 3. Visualization of the generated molecules with the specific condition γ. The maximum PLogP among the training molecules is 4.52.

In this section, we show the applicability of our HI-Mol in molecular optimization, mainly following the experimental setup of Ahn et al. (2022). Specifically, we consider the offline[8] molecular property optimization task on the penalized octanol-water partition coefficient (PlogP). We train a conditional molecular generative model $p_{\text{model}}(\mathbf{x}|\gamma)$ under the HI-Mol framework where γ denotes the PlogP value. Then, we sample with a high γ to generate molecules with high PLogP. In Table 10, our HI-Mol generates molecules with high PLogP even when trained with only 1% of the entire training dataset. Here, we remark that solely maximizing the molecular property (such as PLogP) may generate unrealistic molecules (Ahn et al., 2022), e.g., unstable or hard-to-synthesize (see Appendix M). To address this and highlight the practical application of our HI-Mol framework, we further show the model's capability to generate molecules with the desired PLogP. In Figure 3, HI-Mol generates realistic molecules with the target PLogP, even when the desired condition γ is unseen in the training molecules. The overall results show that our HI-Mol exhibits a huge potential for real-world scenarios where we aim to generate molecules with a specific target property.

[8]While some online optimization algorithms show promising performances (Jensen, 2019; Fu et al., 2021), they require the specific value of the relevant property for the intermediate molecules in the learning process. This additional cost often limits the practical application of online algorithms since the calculation of the property sometimes requires high experimental costs (Gao et al., 2022).

G. Ablation Study

Table 11. Ablation on the text prompts for interpolation-based sampling on the 2% subset of QM9.

Generation prompt	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
The molecule is a $[S^*][I^*_{c_n}][D^*_n]$	**0.210**	**0.001**	**92.2**	61.4	47.5
The molecule is similar to $[S^*][I^*_{c_n}][D^*_n]$	0.234	**0.001**	91.1	63.4	50.6
A similar molecule of $[S^*][I^*_{c_n}][D^*_n]$	0.271	**0.001**	91.5	65.0	52.6
The chemical is similar to $[S^*][I^*_{c_n}][D^*_n]$	0.437	0.002	90.2	75.5	72.4
A similar chemical of $[S^*][I^*_{c_n}][D^*_n]$	0.434	**0.001**	90.7	**75.8**	**73.5**

Table 12. Ablation of hierarchical textual inversion on the HIV dataset in the MoleculeNet (Wu et al., 2018) benchmark.

Inverted tokens	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
$[D^*]$	5.4	21.7	0.026	**100**	88.8	**100**
$[S^*][I^*][D^*]$	**11.4**	**16.6**	**0.019**	**100**	**95.6**	**100**

Table 13. Ablation on the number of clusters K in Eq. (1) on the 2% subset of QM9.

K	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
0	0.486	0.002	93.8	70.8	72.3
1	0.474	0.002	87.0	72.9	72.0
3	0.455	0.002	88.9	76.5	71.1
5	0.443	**0.001**	88.0	77.0	73.2
10	0.434	**0.001**	**90.7**	75.8	73.5
20	**0.430**	**0.001**	87.9	**77.3**	73.8
30	0.436	**0.001**	88.9	77.2	**73.9**
2,113	0.443	**0.001**	86.2	75.4	72.6

Effect of prompt. In Table 11, we show the ablation results on the generation prompt for embedding interpolation-based sampling. We observe that we obtain low FCD and NSPDK scores when we use a prompt similar to the training prompt. However, such choices yield low Novelty scores, generating the many molecules contained in the training samples. The prompt we utilize generates more novel molecules while preserving the state-of-the-art FCD and NSPDK scores.

Effect of hierarchical textual inversion. In Table 12, we show the importance of our hierarchical textual inversion. Specifically, we compare using a single hierarchy, i.e., detail tokens (using $[D^*]$), and our multi-level hierarchy (using $[S^*][I^*][D^*]$). The results show that our multi-level textual inversion strategy is highly useful to generate faithful molecules from the desired distribution.

Effect of K. In Table 13, we report the quantitative results of the following cases. First, we consider our proposed design with varying K from 3 to 30. In addition, we consider three other designs that do not contain intermediate tokens to verify the effect of them: (a) $[S^*_1][D^*_n]$ that the intermediate tokens are removed, i.e., K=0, (b) $[S^*_1][S^*_2][D^*_n]$ that the intermediate tokens are replaced with a shared token $[S^*_2]$, i.e., K=1, and (c) $[S^*][D^*_{1,n}][D^*_{2,n}]$ that the intermediate tokens are replaced with a detail token $[D^*_{1,n}]$, i.e., K=2,113. The results exhibit that the intermediate tokens are indeed crucial for the performance, given that the performance $10 \leq K \leq 30$ is much better than (a), (b) and (c). We find that the overall performance is rather degraded with K=2,113 compared to K=10, 20, and 30. We hypothesis that this is because the sharing of the coarse-grained common features (i.e., intermediate tokens) serves to regularize the fine-grained features (i.e., detail tokens) which are biased toward a single molecule in the embedding interpolation-based sampling. We also remark that we did not put much effort on tuning K, e.g., K=20 improves FCD as $0.434 \rightarrow 0.430$ from K=10.

H. Additional Experiments

Table 14. Generated molecules from HI-Mol with compositional prompt. We invert 4 aromatic molecules (top row) with the prompt "The molecule is a $[S^*][D_i^*]$". With learned embeddings of $[S^*]$ and $[D_i^*]$, we generate molecules (bottom row) with "The molecule is a boron compound of $[S^*][\bar{D}^*]$". We circle the substructures which indicate that the generated molecules indeed satisfy the condition of the given language prompt.

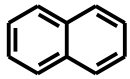
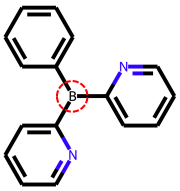
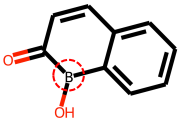
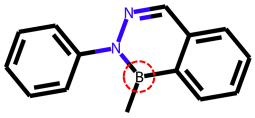
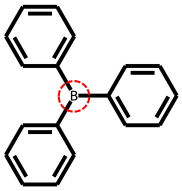
Input molecules for inversion			
			
The molecule is a $[S^*][D_i^*]$			
Generated molecules			
			
The molecule is a boron compound of $[S^*][\bar{D}^*]$			

Table 15. Molecular generation results on (1) learning several concepts (the first row) and (2) learning an underlying concept among diverse molecules (the second row).

	MiCaM	STGG	GSDM	HI-Mol (Ours)
Success ratio (%)	18.2	33.2	0.0	**52.0**
Average QED	0.555	0.558	0.090	**0.581**

Compositionality. In Table 14, we explore the compositionality of the learned token embeddings from HI-Mol. We learn the common features of 4 aromatic molecules,[9] e.g., naphthalene, pyrrole, benzene, and pyridine, via textual inversion. Then, we generate molecules with an additional condition via language prompt. We observe that the generated molecules both satisfy (1) the learned common concept of aromatic molecules and (2) the additional conditions from the language prompt. Although our current molecular language model (Edwards et al., 2022) shows some interesting examples of composition between natural language and the learned concept, we strongly believe that future advances in molecular language models will provide more intriguing examples in this application.

Learning complex molecular concepts. In this section, we explore the ability of HI-Mol to learn more complex molecular concepts. We conduct two kinds of experiments. Firstly, we impose several target concepts for molecular generation. We collect 300 molecules from GuacaMol (Brown et al., 2019) which satisfy QED>0.5, SA>2.5, and GSK3B>0.3.[10] With these molecules, we check whether the generative models can learn to model several molecular concepts. We report the ratio of the generated molecules that satisfy the aforementioned condition, e.g., QED>0.5, SA>2.5, and GSK3B>0.3, as the Success ratio in Table 15. Our HI-Mol shows superior results on learning several concepts, e.g., $33.2 \rightarrow 52.0$, compared to the most competitive baseline, STGG (Ahn et al., 2022). Secondly, we explore whether HI-Mol can learn the "underlying" molecular property, e.g., QED, among structurally diverse molecules. We curate 329 molecules in the QM9 dataset (Ramakrishnan et al., 2014) where (a) each molecule in this subset has a Tanimoto similarity of no higher than 0.4 with any other molecule in the subset and (b) all the molecules in this subset have a high QED ratio greater than 0.6. The average QED in Table 15 shows that HI-Mol generates molecules with high QED even when the training molecules are structurally largely different, i.e., HI-Mol indeed learns the underlying molecular concept.

[9] These molecules share several chemical properties such as resonance and planar structure.

[10] QED, SA, and GSK3B measure the drug-likeness, synthesizability, activity to GSK3B, respectively.

Table 16. Comparison with pre-trained model of STGG (Ahn et al., 2022) on the HIV dataset.

Method	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
STGG (from scratch)	1.6	20.2	0.033	**100**	**95.8**	**100**
STGG (fine-tuned)	3.6	20.0	0.030	**100**	87.1	**100**
HI-Mol (Ours)	**11.4**	**16.6**	**0.019**	**100**	95.6	**100**

Comparison with pre-trained model. In Table 16, we report the performance of the baseline method by fine-tuning the pre-trained baseline model. Specifically, we fine-tune the model of STGG (Ahn et al., 2022) pre-trained with the ZINC250k dataset (Irwin et al., 2012) on the HIV dataset (Wu et al., 2018). We observe that HI-Mol still achieves significantly better performance in overall metrics, e.g., 20.0 → 16.6 and 0.030 → 0.019 in FCD and NSPDK, respectively.

I. Modification Algorithm

Algorithm 1 Modification algorithm for an invalid SMILES string

Input: An invalid SMILES string
Output: A modified SMILES string

while *exist a branch closing token token prior to a branch opening token* **do**
  Remove the corresponding branch closing token. `// ``CC)CCC'' to ``CCCCC''`
while *exist an unclosed branch opening token* **do**
  Add the the branch closing token at the end of the string. `// ``CC(CCC'' to ``CC(CCC)''`
while *exist an unclosed ring opening token* **do**
  Remove the ring opening token. `// ``CC1CCC'' to ``CCCCC''`
while *exist an atom that exceeds the valency* **do**
  Randomly drop a branch to satisfy the valency. `// ``C#C(=CC)C to ``C#CC''`
while *exist a ring with less than 3 atoms* **do**
  Remove the ring opening/closing token. `// ``CC1C1 to ``CCC''`

J. Details on QM9 Experiments

Table 17. Qualitative results for molecular generation varying the data ratio on QM9.

Ratio (%)	Method	Grammar	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
2	GDSS (Jo et al., 2022)	✗	22.953	0.455	99.8	1.2	72.2
	STGG (Ahn et al., 2022)	✓	0.715	0.002	**100**	**88.0**	42.1
	HI-Mol (Ours)	✗	0.434	**0.001**	90.7	75.8	73.5
	HI-Mol (Ours)	✓	0.430	**0.001**	**100**	76.1	**75.6**
5	GDSS (Jo et al., 2022)	✗	17.013	0.066	97.2	25.5	44.2
	STGG (Ahn et al., 2022)	✓	0.665	**0.001**	**100**	**95.8**	63.0
	HI-Mol (Ours)	✗	0.412	**0.001**	89.4	85.8	70.4
	HI-Mol (Ours)	✓	0.410	**0.001**	**100**	86.4	**72.4**
10	GDSS (Jo et al., 2022)	✗	17.170	0.067	98.0	22.8	36.6
	STGG (Ahn et al., 2022)	✓	0.603	0.002	**100**	**99.4**	63.5
	HI-Mol (Ours)	✗	0.400	0.002	87.6	87.6	71.2
	HI-Mol (Ours)	✓	0.398	**0.001**	**100**	88.3	**73.2**
20	GDSS (Jo et al., 2022)	✗	7.345	0.025	94.2	82.3	67.6
	STGG (Ahn et al., 2022)	✓	0.599	**0.001**	**100**	**99.4**	64.3
	HI-Mol (Ours)	✗	0.384	**0.001**	86.7	87.8	70.0
	HI-Mol (Ours)	✓	0.383	**0.001**	**100**	88.7	**71.8**
50	GDSS (Jo et al., 2022)	✗	3.564	0.008	96.0	96.6	80.1
	STGG (Ahn et al., 2022)	✓	0.592	**0.001**	**100**	**99.2**	**70.6**
	HI-Mol (Ours)	✗	**0.372**	**0.001**	88.7	87.7	68.8
	HI-Mol (Ours)	✓	**0.372**	**0.001**	**100**	88.5	70.5

Table 18. Comparison with the baseline with high Novelty via resampling strategy on QM9.

Method	Resampling ratio	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
GDSS (Jo et al., 2022)	1.0	2.900	0.003	95.7	98.5	86.3
HI-Mol (Ours; 2%)	1.9	**0.601**	**0.002**	**100**	**100**	**100**

In Table 17, we report experimental results varying the data ratio from 2% to 50%. In particular, when we use 50% of the training data the performance improves further by 0.430 → 0.372 (compared to using 2% of training data), i.e., our HI-Mol better learns molecule distribution when more molecules are available for training.

We note that there is a fundamental trade-off between FCD and Novelty. If the generated molecules have many overlaps with training molecules, i.e., low Novelty, the FCD score improves, i.e., decreases, since the generated molecules are more likely to follow the target distribution. Therefore, it is crucial to compare FCD under a similar Novelty score. Therefore, in Table 18, we report the generation results with the resampling strategy, i.e., we sample molecules until we have 10,000 molecules with Validity, Uniqueness, and Novelty scores as 100 and we reject samples that violate these scores. We denote the relative ratio of the total sampling trial (including the rejected ones) as Resampling ratio. Here, we remark that such resampling process does not incur much computational cost, e.g., only 1.8 sec for a sample (see Appendix L for analysis of time complexity). The result shows that HI-Mol generates high-quality novel molecules from our desired target distribution.

K. Analysis of Interpolation-based Sampling

Table 19. Generated molecules from HI-Mol with varying λ in Eq. (2). Samples are generated with the prompt "A similar chemical of $[S^*][\bar{I}^*][\bar{D}^*]$". The columns $[D_i^*]$ and $[D_j^*]$ denote molecules in the HIV dataset (Wu et al., 2018) whose token embeddings are interpolated for each row.

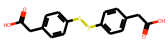
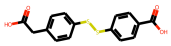
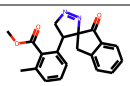
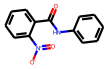
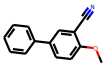
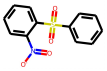
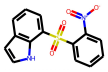
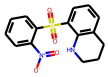
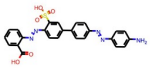
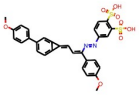
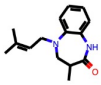
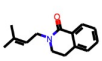
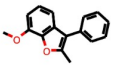
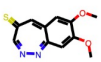
$[D_i^*]$	A similar chemical of $[S^*][\bar{I}^*][\bar{D}^*]$					$[D_j^*]$
						
	$\lambda = 0.0$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$	$\lambda = 1.0$	
						
	$\lambda = 0.0$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$	$\lambda = 1.0$	

Table 20. Generated molecules from HI-Mol with varying λ in Eq. (2). We interpolate a single-level token, e.g., "A similar chemical of $[S^*][\bar{I}^*][D^*]$" and "A similar chemical of $[S^*][I^*][\bar{D}^*]$".

A similar chemical of $[S^*][\bar{I}^*][D^*]$				
				
$\lambda = 0.0$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$	$\lambda = 1.0$
A similar chemical of $[S^*][I^*][\bar{D}^*]$				
				
$\lambda = 0.0$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$	$\lambda = 1.0$

Note that our sampling is based on the interpolation of two different token embeddings with different values of $\lambda \sim p(\lambda)$. In Table 19, we provide how the generated molecules are changed with different values of λ. With varying λ, one can observe that the generated molecules (1) maintain some original important low-level semantics and (2) introduce some novel aspects distinct from both original semantics. For example, $\lambda = 0.7$ in the first row of Table 19 introduces a new 4-membered ring system while preserving the phosphorous-sulfur double bond structure of the original features in $[D_j^*]$. This observation exhibits that our embedding space models the manifold of underlying target distribution effectively, enabling data-efficient sampling from the target distribution. We also provide the generated samples from different hierarchies. Interpolating intermediate tokens (see the first row of Table 20) change the low-level semantics, i.e., size of molecules, of the generated molecules and interpolating detail (see the second row of Table 20) tokens change the high-level features, i.e., insertion of a single atom, of the generated molecules.

L. Complexity

Table 21. Time and space complexity of each molecular generative method.

	JT-VAE	PS-VAE	MiCaM	STGG	CRNN	GDSS	GSDM	DiGress	HI-Mol (Ours)
Time complexity (s)	4.8	0.1	0.9	0.7	0.5	71.2	2.0	9.1	1.8
Space complexity (GB)	0.4	1.2	1.6	2.1	0.4	1.2	1.1	1.5	4.8

In Table 21, we provide the time and space complexity to generate a molecule via various molecular generative models. For time complexity, measured with a single RTX 3090 GPU, HI-Mol takes about 1.8 seconds to sample a single molecule, while other methods, e.g., GDSS and DiGress, require more time due to denoising diffusion steps. For memory complexity, HI-Mol requires 4.8GB of GPU VRAM space due to the usage of the large model. We believe that reducing this space for large language models, e.g., through Dao et al. (2022), will be an interesting future direction.

M. Discussion on Molecular Optimization

In Table 10, we have shown the usefulness of our HI-Mol to maximize the PLogP value of the generated molecules. While this evaluation setup for molecular optimization is a common and popular choice in molecular domain (Jin et al., 2018; Shi et al., 2020; Luo et al., 2021; Ahn et al., 2022), some prior works have noted that solely maximizing the PLogP value may yield unstable or hard-to-synthesize molecules (Gao & Coley, 2020; Coley, 2021; Ahn et al., 2022). In Figure 4, we show the visualizations of the optimized molecules with the highest PLogP values. Similar to the most competitive baseline, STGG (Ahn et al., 2022), our optimized molecules contain a large number of atoms, and thus relatively hard to synthesize. Although these results show that our HI-Mol effectively learns to incorporate the condition PLogP in a data-efficient manner, it would be an important research direction to develop an evaluation framework for molecular optimization that takes into account the "realistic-ness", e.g., stability and synthesizability, of the molecules.

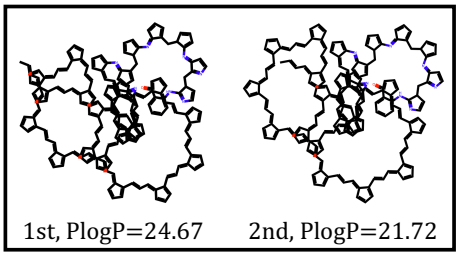


Figure 4. Visualizations of the generated molecules with $\gamma = 50$. The maximum PLogP among the training molecules is 4.52.

N. Details on Low-shot Molecular Property Prediction

Table 22. Results on low-shot classification on the MoleculeNet benchmark. We report the average and 95% confidence interval of the test ROC-AUC scores within 20 random seeds.

Dataset	Method	16-shot	32-shot
HIV	DiGress (Vignac et al., 2023)	-2.30±3.50	-2.67±3.15
	MiCaM (Geng et al., 2023)	1.02±3.29	0.69±2.09
	STGG (Ahn et al., 2022)	0.53±2.79	-0.47±2.36
	HI-Mol (Ours)	**2.35**±2.71	**2.16**±1.64
BBBP	DiGress (Vignac et al., 2023)	1.73±1.53	0.97±1.99
	MiCaM (Geng et al., 2023)	1.91±2.13	1.78±1.98
	STGG (Ahn et al., 2022)	1.85±1.83	1.76±1.72
	HI-Mol (Ours)	**2.73**±2.01	**2.64**±1.75
BACE	DiGress (Vignac et al., 2023)	-0.60±2.88	-0.91±1.82
	MiCaM (Geng et al., 2023)	-0.65±3.17	-1.11±2.95
	STGG (Ahn et al., 2022)	2.34±2.15	2.01±1.45
	HI-Mol (Ours)	**3.53**±1.57	**3.39**±1.80

Table 23. Comparison with latent mixup (Wang et al., 2021) in the low-shot classification task. We report ΔROC-AUC averaged over 20 random seeds.

32-Shot	HIV	BBBP	BACE
Latent mixup (Wang et al., 2021)	0.55	1.27	0.52
HI-Mol (Ours)	**2.16**	**2.64**	**3.39**

Low-shot (or few-shot) prediction tasks are one of the important applications for industrial deployments (Nam et al., 2023), and we have shown our HI-Mol's capability to be beneficial to these tasks. In Table 22, we report the full results of low-shot molecular property prediction experiments with averages and 95% confidence intervals. With randomly sampled low-shot molecules from the train split (used in our main experiments of Table 2), we generate $\times 3$ number of valid molecules via generative models, e.g., we generate 96 molecules for 32-shot experiments. For the classifier, we utilize the 5-layer GIN (Xu et al., 2019a) from You et al. (2020), which is pre-trained with unlabeled molecules via self-supervised contrastive learning. We fine-tune this model for 100 epochs by introducing a linear projection head for each dataset. We use Adam optimizer with a learning rate of 0.0001 and no weight decay. The results are calculated based on the test ROC-AUC score of the epoch with the best validation ROC-AUC score. Specifically, we consider two scenarios: (1) training the classifier with only the low-shot molecules and (2) training the classifier with both the original low-shot molecules and the generated molecules via the molecular generative model. We report ΔROC-AUC score, calculated by the subtraction of the ROC-AUC score of (1) from (2). In Table 23, we additionally compare with conventional latent mixup strategy (Wang et al., 2021). They directly use the interpolated latent embeddings (and corresponding interpolated labels) as inputs, which mostly do not become real data. However, we generate "new molecules" (rather than just latent embeddings) based on this embedding and use it as real data to train a classifier for a molecular prediction task. For the latent mixup, we train the classifier using given molecules and interpolated latent embeddings (and labels) using uniformly sampling coefficient λ from a range of [0, 1] (which is the same with the choice of λ in our method). As shown in the table, our method indeed shows a significantly better ΔROC-AUC compared to latent mixup.