Data-Efficient Molecular Generation with Hierarchical Textual Inversion

Seojin Kim[1] **Jaehyun Nam**[1] **Sihyun Yu**[1] **Younghoon Shin**[2] **Jinwoo Shin**[1]
Korea Advanced Institute of Science and Technology (KAIST)
Korea University
{osikjs,jaehyun.nam,sihyun.yu,jinwoos}@kaist.ac.kr
yhoon99@korea.ac.kr

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 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 a recent textual inversion technique in the visual domain that achieves data-efficient generation via simple optimization of a new single text token of a text-to-image generative model. However, we find that its naïve adoption fails for molecules due to their complicatedly structured nature. Hence, we propose a hierarchical textual inversion scheme based on introducing low-level tokens that are selected differently per molecule in addition to the original single text token in textual inversion to learn common concepts. We then generate molecules using a pre-trained text-to-molecule model by interpolating the low-level tokens. 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 efficacy of HI-Mol in various applications, including molecular optimization and low-shot molecular property prediction.

1 Introduction

Finding novel molecules has been a fundamental yet crucial problem in chemistry [1, 2] due to its strong relationship in achieving important applications, such as drug discovery [3, 4]. However, generating molecules poses a challenge due to their highly structured nature and the vast size of the input space [5]. To tackle this issue, several works have considered training deep generative models to learn the molecule distribution using large molecular datasets [6, 7]. This is inspired by the recent advances of generative models in other domains, e.g., images and videos [8, 9], in learning large and complex data distribution. Intriguingly, such deep molecular generative methods have demonstrated reasonable performance [6, 10, 11] on the large-scale molecular generation benchmarks [12, 13] in finding chemically valid and novel molecules, showing great potential to solve the challenge.

Unfortunately, existing molecular generation frameworks often fail in limited data regimes [14]. This restricts the deployment of existing approaches to practical scenarios, because task-related molecular data for the real-world applications are mostly insufficient to train molecular generative models. For example, the drug-likeness of each candidate molecule should be verified through years of extensive wet experiments and clinical trials [15, 16]. This time-consuming and labor-intensive data acquisition process of new task-related molecules [17] limits the number of available training data for a model to learn the desired molecule distribution. Thus, it is often crucial to develop a *data-efficient molecular generation* framework, yet this direction has been overlooked in deep molecular generation [14].

The NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development (AI4D3 2023), New Orleans, LA, USA, 2023.

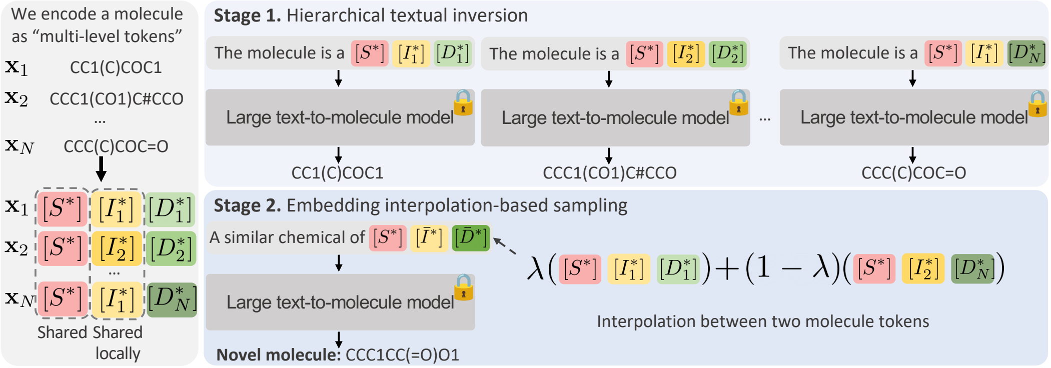

We encode a molecule as "multi-level tokens"
$\mathbf{x}_1$ CC1(C)COC1
$\mathbf{x}_2$ CCC1(CO1)C#CCO
...
$\mathbf{x}_N$ CCC(C)COC=O
$\mathbf{x}_1$ $[S^*]$ $[I_1^*]$ $[D_1^*]$
$\mathbf{x}_2$ $[S^*]$ $[I_2^*]$ $[D_2^*]$
...
$\mathbf{x}_N$ $[S^*]$ $[I_1^*]$ $[D_N^*]$
Shared Shared locally
Stage 1. Hierarchical textual inversion
The molecule is a $[S^*]$ $[I_1^*]$ $[D_1^*]$
Large text-to-molecule model
CC1(C)COC1
The molecule is a $[S^*]$ $[I_2^*]$ $[D_2^*]$
Large text-to-molecule model
CCC1(CO1)C#CCO
...
The molecule is a $[S^*]$ $[I_1^*]$ $[D_N^*]$
Large text-to-molecule model
CCC(C)COC=O
Stage 2. Embedding interpolation-based sampling
A similar chemical of $[S^*]$ $[\bar{I}^*]$ $[\bar{D}^*]$
Large text-to-molecule model
Novel molecule: CCC1CC(=O)O1
$\lambda([S^*][I_1^*][D_1^*]) + (1-\lambda)([S^*][I_2^*][D_N^*])$
Interpolation between two molecule tokens

Figure 1: Overview of our HI-Mol framework. (1) Hierarchical textual inversion: we encode the features of molecules as multi-level token embeddings. (2) Embedding interpolation-based sampling: we sample novel molecules using interpolation of low-level token embeddings.

Meanwhile, recent works in text-to-image generation have explored the problem of low-shot (or personalized) generation using the power of large pre-trained models trained on a massive amount of data [18, 19]. In particular, Gal et al., [20] achieve this by proposing a *textual inversion* using pre-trained text-to-image diffusion models—given a small set of images, they show that common concepts among them can be learned effectively by optimizing a single text token embedding under the frozen generative model, where the learned token can be used for the desired generation.

Considering the recent success of large-scale pre-trained text-to-molecule models [21], what we ask in this paper is: *can textual inversion be exploited to enable data-efficient molecular generation with pre-trained text-to-molecule models?* However, we find that naïve adoption of textual inversion methods fails to achieve the goal, due to the highly complicated and structured nature of molecules. To enjoy the effectiveness of textual inversion for our problem of data-efficient molecular generation, we suggest considering the unique aspects of molecular data carefully in its adoption.

Contribution. We introduce a novel data-efficient molecular generation framework, coined **H**ierarchical textual **I**nversion for **Mol**ecular generation (**HI-Mol**). Specifically, HI-Mol is composed of two components (see Figure 1 for the overall illustration):

- *Hierarchical textual inversion:* We propose a molecule-specialized textual inversion framework to capture the hierarchical information of molecules [22]. In contrast to textual inversion for the visual domain that optimizes a single shared token on given data, we design multi-level tokens for the inversion so that some of the low-level tokens are selected differently per molecule. Thus, the shared token learns the common concept among molecules and low-level tokens learn molecule-specific features. This low-level token selection does not require any specific knowledge of each molecule and can be achieved completely in an unsupervised manner.
- *Embedding interpolation-based sampling:* We present a molecule sampling scheme which utilizes the multi-level tokens optimized in the inversion stage. Our main idea is to utilize low-level tokens in addition to the shared token for molecular generation. In particular, we consider using the interpolation of two different low-level token embeddings for generation. The mixing approach is designed to utilize the information of given molecules extensively, and thus to effectively alleviate the issue of the limited number of molecules that lie in the target distribution.

We extensively evaluate our method by designing several data-efficient molecular generation tasks. In the HIV dataset [23], measured by Frechet ChemNet Distance (FCD [24]; lower is better) and Neighborhood Subgraph Pairwise Distance Kernel MMD (NSPDK [25]; lower is better) metrics, our method improves the prior arts as 20.2 → 16.6, and 0.033 → 0.019, respectively. Our method also achieves much better active ratio (higher is better) by improving the previous state-of-the-art as 3.7 → 11.4. We also demonstrate the strong data-efficiency of HI-Mol. For instance, on QM9 [12], our method already outperforms the previous state-of-the-art methods, e.g., STGG [6] by 0.585 → 0.434 in FCD, with 50× less training data. We also validate the superiority of HI-Mol on the molecular optimization for penalized octanol-water partition coefficient on the ZINC dataset [26] and the low-shot molecular property prediction on the datasets in the MoleculeNet benchmark [23].

2 Related work

Molecular generation. Most molecular generation methods fall into three categories based on different representations of molecules. First, there exist many attempts [7, 14, 27, 28, 29, 30, 31, 32, 33] to formalize molecular generation as a graph generation problem by representing each molecule as an attributed graph. Next, there are several fragment-based methods [10, 11, 34], which define a dictionary of chemically meaningful fragments, e.g., functional groups. Each molecule is represented as a tree structure of dictionary elements and the distribution of connected fragments is then modeled. Finally, there are many approaches [6, 35, 36] that utilize the Simplified Molecular-Input Line-Entry System [SMILES, 37] representation to write molecules as strings and learn the distribution in this string space. Among them, some recent works have tried to train large-scale text-to-molecule models; they observe that fine-tuning large language models in natural language domain [38] using molecular data interpreted as SMILES representation can result in good text-to-molecule models [21, 39]. Our method takes the string-based approach based on the utilization of recent large-scale text-to-molecule models that use SMILES representation, where we carefully design a hierarchical textual inversion method for molecules to tackle under-explored data-efficient molecular generation.

Low-shot generation. There have been substantial efforts in the generative model literature to design a low-shot generation framework for generating new samples from a given small number of data. In particular, in the image domain, many approaches have proposed some adaptation or fine-tuning methods of the pre-trained generative models [18, 40, 41, 42], mostly focusing on generative adversarial networks [GAN, 43]. Despite their efforts, exploiting knowledge from pre-trained generative models for a low-shot generation had remained a challenge, in contrast to the great progress in other tasks, e.g., low-shot classification [44]. Intriguingly, recent works on large-scale text-to-image diffusion models [21, 39] have surprisingly resolved this challenge, even enabling "personalization" of the model at a few in-the-wild images through very simple optimization schemes that update only a few parameters [19, 20, 45]. In particular, textual inversion [20] exhibits that the personalization of large-scale text-to-image diffusion models can be achieved even with a simple optimization of an additional single 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 method is under-explored despite its practical importance [46, 14]. Our method tackle this problem by designing a molecule-specific textual inversion method using the recent large-scale text-to-molecule models. Specifically, due to our unique motivation to consider "hierarchy" of molecular structures [22], our method effectively learns the distribution of diverse molecular structures of low-shot molecules, while the applications of prior works, e.g., Guo et al., [14], are limited to structurally similar molecules such as monomers and chain-extenders.

3 HI-Mol: Hierarchical textual inversion for molecular generation

In this section, we explain our method, coined HI-Mol, in detail. In Section 3.1, we provide a brief overview of our problem of interest and the main idea to solve the challenge. In Section 3.2, we provide descriptions of textual inversion and molecular language models to explain our method. In Section 3.3, we provide a component-by-component description of our method.

3.1 Problem description and overview

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

Overview. To solve this problem, we take the recent approach of textual inversion [20] from the text-to-image diffusion model literature—a simple yet powerful technique in low-shot image generation that learns a common concept in given images as a token in text embedding space. Similarly, we aim to learn the common concepts of molecules as text tokens and use them for our target of data-efficient generation. However, exploiting this approach for our goal faces several challenges, mainly due to the unique characteristics of molecules differentiated from images. First, it is yet overlooked *which* of the large-scale model for molecules is beneficial to achieve textual inversion for the given molecules, like the success of text-to-image diffusion models in achieving successful inversion in the image domain. Moreover, molecules have a very different structural nature from images—unlike images, molecules with similar semantics often have entirely different structures (see Figure 2), making it difficult to simply learn the common concept as a single text token. Our contribution lies in resolving these challenges by adopting molecule-specific priors into the molecular generation framework to enjoy the power of textual inversion techniques to achieve data-efficient molecular generation.

3.2 Preliminaries

Textual inversion. Recent text-to-image generation methods have proposed textual inversion [20], which aims to learn a common concept $\mathbf{c}$, i.e., the distribution $p(\mathbf{x}|\mathbf{c})$, from a 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 token $[S^*]$ shared among images to learn $\mathbf{c}$ using a pre-trained frozen text-to-image diffusion 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 images with the exact same training objective that is used for training $f_{\texttt{t2i}}$.

Molecular language model. Following the recent progress in large language models [38, 47, 48], there exist several attempts to train molecular language models [39, 49, 50, 51]. Specifically, these works exploit popular language model architectures to have pre-trained models for molecules, based on the SMILES [37] representation $\texttt{SMILES}(\mathbf{x})$ that interprets a given molecule $\mathbf{x}$ as a string. In particular, MolT5 [21] proposes to fine-tune a pre-trained large-scale text-to-text language model, T5 [38], with large-scale molecular SMILES representations and text 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 on text-to-molecule generation tasks. Inspired by its success, we use the Large-Caption2Smiles model trained with this MolT5 approach to design molecule-specific textual inversion framework for our goal of data-efficient molecular generation.

3.3 Detailed description of HI-Mol

Hierarchical textual inversion. We first propose a molecule-specific textual inversion to learn the desired molecular distribution. Unlike prior textual inversion 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 $K < N$). Such intermediate and detail tokens are selected differently for each molecule and learn cluster-wise features and molecule-wise features, respectively, enabling to capture the hierarchical, e.g., high-level and low-level, semantics of the molecular dataset.

To learn these hierarchical tokens, we consider a frozen text-to-molecule model f, e.g., Large-Caption2Smiles [21], to apply our proposed hierarchical textual inversion objective. Specifically, we optimize θ by minimizing the following objective on the given molecular dataset $\mathcal{M}$:

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

where $\mathcal{L}_{\texttt{CE}}$ denotes cross-entropy loss and $\texttt{SMILES}(\mathbf{x}_n)$ is a SMILES [37] string interpretation of $\mathbf{x}_n$. Thus, after training, each $\mathbf{x}_n$ is interpreted as three tokens $[S^*][I_{c_n}^*][D_n^*]$, where each intermediate token index $c_n \in [1, K]$ (for $1 \leq n \leq N$) is chosen during optimization to minimize the training objective $\mathcal{L}$. Note that the selection of $[I_k^*]$ is achieved in an unsupervised manner so that it does not require any specific information about each of the given molecules. Intriguingly, we find this simple selection scheme of $[I_k^*]$ can learn some of the informative cluster-wise features although we have not injected any prior knowledge of a 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 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 concept, e.g., ability on the blood-brain membrane permeability [23]. 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 chemistry literature highlighting that molecular data can be clustered hierarchically [22].

Table 1: Quantitative results of the generated molecules on the three datasets (HIV, BBBP, BACE) in the MoleculeNet benchmark [23]. 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 compute and 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) for each metric.

Dataset	Method	Class	Grammar	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
HIV	GDSS [7]	Graph	✗	0.0	34.1	0.080	69.4	**100**	**100**
	DiGress [33]	Graph	✗	0.0	26.2	0.067	17.8	**100**	**100**
	JT-VAE [10]	Fragment	✓	0.0	38.8	0.221	**100**	25.4	**100**
	PS-VAE [34]	Fragment	✓	3.7	21.8	0.053	**100**	91.4	**100**
	MiCaM [52]	Fragment	✓	3.4	20.4	0.037	**100**	81.6	**100**
	CRNN [3]	SMILES	✗	3.3	29.7	0.064	30.0	**100**	**100**
	STGG [6]	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 [7]	Graph	✗	0.0	35.7	0.065	88.4	99.2	**100**
	DiGress [33]	Graph	✗	8.2	17.4	0.033	43.8	94.6	**100**
	JT-VAE [10]	Fragment	✓	80.6	37.4	0.202	**100**	10.8	**100**
	PS-VAE [34]	Fragment	✓	84.9	17.3	0.039	**100**	91.6	**100**
	MiCaM [52]	Fragment	✓	82.0	14.3	0.021	**100**	89.4	**100**
	CRNN [3]	SMILES	✗	88.8	20.2	0.026	54.0	**100**	**100**
	STGG [6]	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 [7]	Graph	✗	9.1	66.0	0.205	73.4	**100**	**100**
	DiGress [33]	Graph	✗	21.1	26.7	0.102	16.4	**100**	**100**
	JT-VAE [10]	Fragment	✓	40.4	49.1	0.304	**100**	13.0	**100**
	PS-VAE [34]	Fragment	✓	57.3	30.2	0.111	99.8	75.6	**100**
	MiCaM [52]	Fragment	✓	56.2	18.5	0.060	**100**	64.2	**100**
	CRNN [3]	SMILES	✗	79.0	21.7	0.066	38.0	**100**	**100**
	STGG [6]	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**

Embedding interpolation-based sampling. Given the learned distribution from hierarchical textual inversion, we propose a strategy to sample novel molecules from the distribution. One can consider similar sampling schemes used in existing textual inversion for images. For example, $[S^*]$ can be used for generating sample from our target distribution by putting text prompts including $[S^*]$, e.g., "A similar chemical of $[S^*]$", into the molecular language model f. However, we find that such a simple strategy does not work well in molecular generation (see Table 6), which might be due to the complex structured nature of molecules, and relatively less description-molecule pairs for training molecular language models than training large-scale text-to-image generative models.

To alleviate this issue, we propose to utilize the learned 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 intermediate tokens and detail tokens, i.e., we incorporate the hierarchy information of the molecules, which is obtained in our textual inversion, in the sampling process. Specifically, we sample a novel molecule with random molecule indices i, j sampled uniformly from $[1, \dots, N]$ and a coefficient λ drawn from a pre-defined prior distribution $p(\lambda)$ (see Appendix A for our choice of $p(\lambda)$):

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

where $[\bar{I}^*]$, $[\bar{D}^*]$ indicate that we pass interpolated 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 Eq. (1).[1] This additional consideration of low-level tokens $\{[I_k^*]\}_{k=1}^K, \{[D_n^*]\}_{n=1}^N$ (as well as $[S^*]$) encourages the sampling to exploit the knowledge from given molecular dataset extensively, mitigating the issue of scarcity of target molecules that lie in the distribution we want to learn and thus enables generating high-quality molecules. We provide qualitative analysis on our embedding interpolation-based sampling scheme in Appendix I.

[1] We simply set the number of clusters K, as 10 in our experiments. Please see Appendix E for details.

Table 2: Qualitative results of the generated molecules on the two datasets (HIV, BBBP) of the MoleculeNet benchmark [23]. 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.

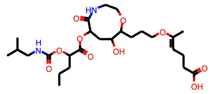
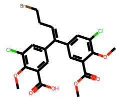
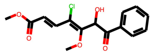
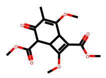
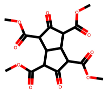
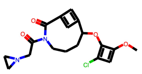
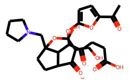
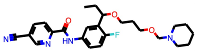
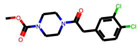
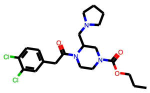
Dataset	DiGress [33]	MiCaM [52]	STGG [6]	**HI-Mol (Ours)**	Train
HIV	 0.154	 0.146	 0.157	 **0.326**	
BBBP	 0.238	 0.247	 0.246	 **0.505**	

Table 3: Quantitative results on the QM9 dataset [12]. 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 [7], we report the results using 10,000 sampled molecules. We denote the scores drawn from [32] and [6] 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 each metric. 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† [36]	Graph	✓	1.852	-	**100**	98.6	94.3
GraphAF [27]	Graph	✗	5.268	0.020	67	94.5	88.8
MoFlow [28]	Graph	✗	4.467	0.017	91.4	98.7	94.7
EDP-GNN [29]	Graph	✗	2.680	0.005	47.5	**99.3**	86.6
GraphDF [30]	Graph	✗	10.82	0.063	82.7	97.6	**98.1**
GraphEBM [31]	Graph	✗	6.143	0.030	8.22	97.8	97.0
GDSS [7]	Graph	✗	2.900	0.003	95.7	98.5	86.3
GSDM* [32]	Graph	✗	2.650	0.003	99.9	-	-
STGG† [6]	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

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. In Section 4.2, we present our main molecular generation results on MoleculeNet and QM9. In Section 4.4, we conduct some analysis and an ablation study to validate the effect of components of our method. In Section 4.3, we present results on additional applications, i.e., optimization and low-shot property prediction. We provide further ablation study and additional experimental results in Appendix E and F, respectively.

4.1 Experimental setup

Datasets. Given the lack of benchmarks designed specifically for data-efficient molecular generation, we propose to use the following datasets for evaluating molecular generation methods under our problem setup. First, we consider the three datasets in the MoleculeNet [23] benchmark (originally designed for activity detection), HIV, BBBP, and BACE, which have a significantly small number of molecules than popular molecular generation benchmarks [53, 54], e.g., BACE includes only 691 active molecules. Considering only the active molecules in each dataset, we construct tasks to generate novel molecules, where they should share the same chemical concept, e.g., drug-likeness on the HIV disease or blood-brain membrane permeability, of the given dataset.

Moreover, we also utilize the QM9 dataset [12] for our experiments to show the data-efficiency of HI-Mol. Specifically, we train our method with an extremely small subset of the entire QM9

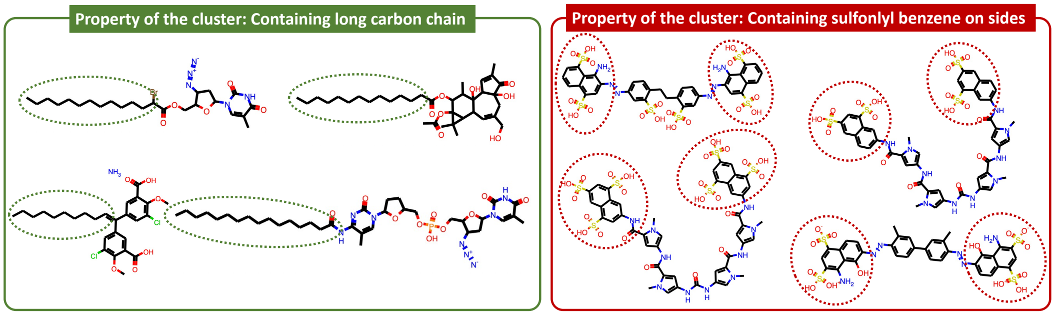

Property of the cluster: Containing long carbon chain
Property of the cluster: Containing sulfonyl benzene on sides

Figure 2: Visualizations of molecules in two clusters obtained from the unsupervised clustering in Eq. (1) on the HIV dataset [23]. We mark the common substructures as dotted lines.

training split, e.g., 2%, where other baseline methods are trained with the whole training split (105k molecules). We provide more detailed explanation about the datasets in Appendix B.

Evaluation setup. To evaluate the quality of the generated molecules, we consider six metrics that represent diverse aspects critical to the evaluation of the generated molecules, e.g., similarity to the target molecules, uniqueness, novelty. Our evaluation incorporates some well-known metrics, such as those used in [7], as well as introducing a new metric "Active ratio":

- **Fréchet ChemNet Distance (FCD)** [24]: Metric for measuring the distance between the source distribution and the target distribution using pre-trained ChemNet.
- **Neighborhood Subgraph Pairwise Distance Kernel MMD (NSPDK)** [25]: 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 evaluating 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.
- **Active ratio**[2] **(Active.)**: Our proposed metric, measuring the ratio of the valid generated molecules that are active, i.e., satisfying the target property for the relevant task. See Appendix C for details.

Baselines. We mainly consider the following methods for evaluation: GDSS [7], DiGress [33], DEG [14], JT-VAE [10], PS-VAE [34], MiCaM [52], CRNN [3], and STGG [6]. For evaluation on QM9, we also consider GraphAF [27], GraphDF [30], MoFlow [28], EDP-GNN [29], and GraphEBM [31], following the recent works [7, 32]. We provide more details of the baselines in Appendix D.

4.2 Main results

Generation on MoleculeNet. Table 1 summarizes the quantitative results of the generated molecules on the HIV, BBBP, and BACE datasets in the MoleculeNet benchmark [23]. 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 of these scores are particularly crucial for the deployment of the molecular generation method. For example, the superior Active ratio and FCD of HI-Mol, e.g., 3.7 → 11.4 and 20.2 → 19.0 on the HIV dataset, respectively, indicate the effectiveness of HI-Mol in generating more faithful molecules which lies in the target distribution. We provide qualitative results in Table 2 by providing some of the generated molecules from the each dataset. One can observe that the generated molecules of HI-Mol capture several crucial common substructures, e.g., many ester groups, while introducing the novel components, e.g., 4-membered ring.

[2] 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 1 for an example). We only consider valid generated molecules to calculate this score.

Table 4: Results of PLogP maximization task. We report the top-3 property scores denoted by `1st`, `2nd`, and `3rd`. The baseline scores are drawn from [6].

	PlogP		
Method	`1st`	`2nd`	`3rd`
GVAE [55]	2.94	2.89	2.80
SD-VAE [56]	4.04	3.50	2.96
JT-VAE [10]	5.30	4.93	4.49
MHG-VAE [57]	5.56	5.40	5.34
GraphAF [27]	12.23	11.29	11.05
GraphDF [30]	13.70	13.18	13.17
STGG [6]	23.32	18.75	16.50
HI-Mol (Ours; 1%)	**24.67**	**21.72**	**20.73**

Table 5: Average ΔROC-AUC of the low-shot property prediction tasks with 20 random seeds.

Dataset	Method	16-shot	32-shot
HIV	DiGress [33]	-2.30	-2.67
	MiCaM [52]	1.02	0.69
	STGG [6]	0.53	-0.47
	HI-Mol (Ours)	**2.35**	**2.16**
BBBP	DiGress [33]	1.73	0.97
	MiCaM [52]	1.91	1.78
	STGG [6]	1.85	1.76
	HI-Mol (Ours)	**2.73**	**2.64**
BACE	DiGress [33]	-0.60	-0.91
	MiCaM [52]	-0.65	-1.11
	STGG [6]	2.34	2.01
	HI-Mol (Ours)	**3.53**	**3.39**

We also propose a simple algorithm to modify the generated invalid SMILES by correcting invalid patterns[3] without a computational overhead. By applying this algorithm, we convert all invalid SMILES to valid ones, i.e., Validity becomes 100. In particular, the modified molecules further improves the overall metrics, e.g., FCD by 19.0 → 16.6 and 11.2 → 10.7 in the HIV and BBBP dataset, respectively. This indicates the modified SMILES indeed represent molecules from our desired distribution and further highlights the superior quality of our generated molecules.

Generation on QM9. In Table 3, we report the quantitative results on QM9 [12]. 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 with 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 [6], 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 novel molecules from the target distribution. We provide further experimental results in Appendix G.

4.3 Applications of HI-Mol

Molecular optimization. We demonstrate the effectiveness of HI-Mol in molecular optimization, mainly following the experimental setup of [6]. We train a conditional molecular generative model $p_{\text{model}}(\mathbf{x}|\gamma)$ under the HI-Mol framework where γ denotes the penalized octanol-water partition coefficient (PLogP). Then, we sample with a high γ to obtain the molecules with high PLogP. In Table 4, our HI-Mol generates molecules with considerably 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 [6], e.g., unstable or hard-to-synthesize (see Appendix K). 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 the real-world scenarios where we aim to generate molecules with a specific target property.

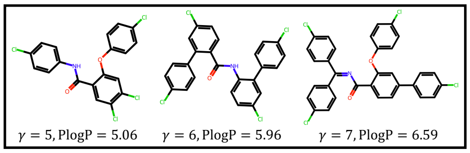

$\gamma = 5$, PlogP = 5.06
$\gamma = 6$, PlogP = 5.96
$\gamma = 7$, PlogP = 6.59

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

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

Table 6: Ablation of the components of hierarchical textual inversion on the QM9 dataset [12] with 2% subset. We report the results using 10,000 sampled molecules.

Training prompt	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
The molecule is a $[S^*]$	7.913	0.041	**96.2**	19.3	39.5
The molecule is a $[S^*][D^*_n]$	0.486	0.002	93.8	70.8	72.3
The molecule is a $[S^*][I^*_{c_n}][D^*_n]$	**0.434**	**0.001**	90.7	**75.8**	**73.5**

Low-shot molecular property prediction. In this section, we show that the generated molecules by HI-Mol can be utilized to improve the performance of classifiers for the low-shot molecular property prediction. Here, we collect low-shot molecules by a random subset from the MoleculeNet benchmark [23] and generate molecules via molecular generative models for each label. Then, we train the classifier with both (1) the original low-shot molecules and (2) the generated molecules via the molecular generative model. In Table 5, we report the ΔROC-AUC [4] score for each generative model. The results show that our HI-Mol consistently outperforms the prior methods in various low-shot molecular property prediction tasks. This verifies the superior ability of HI-Mol to learn the concept, e.g., activeness and in-activeness, of each label information with a limited number of molecules. In practical scenarios, where the label information is hard to achieve, our HI-Mol can indeed play an important role to improve the classifier. We provide experimental details in Appendix L.

4.4 Analysis

Effect of intermediate tokens. Recall that we have introduced intermediate text tokens $\{[I^*_k]\}^K_{k=1}$, which are selected in an unsupervised manner during the hierarchical textual inversion to learn some of the cluster-wise properties included in given molecules. To validate the effect of our text token design, we visualize the clustering results in Figure 2 by providing groups of the molecules that have 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 concept, e.g., molecular property, even when they have large structural difference.

Ablation on hierarchical tokens. To validate the effect of each token in our proposed hierarchical textual inversion, we perform an ablation study by comparing the results with our method where some of the tokens are excluded from the overall framework. Specifically, we compare the generation performance of the following three variants: (1) using the shared token $[S^*]$ only, (2) using $[S^*]$ and the detail tokens $[D^*_n]$, and (3) using all three types of tokens (HI-Mol). Note that for (1), it is impossible to apply our interpolation-based sampling; hence, we use temperature sampling instead based on the categorical distribution from a molecular language model with temperature $\tau = 2.0$. We provide this result in Table 6: as shown in this table, introducing each of the additional tokens successively improves most of the metrics, while maintaining the Validity metric as well.

5 Conclusion

We propose a new framework for data-efficient molecular generation, called Hierarchical textual Inversion for Molecular generation (HI-Mol). Specifically, we derive a molecule-specialized textual inversion scheme and corresponding molecule sampling procedure using a recent large-scale molecular language model. Extensive experiments show the effectiveness of our framework across various datasets, especially in achieving data-efficiency and having the capability to generate molecules with our desired distribution. We hope our work initiates under-explored but crucial research direction of exploiting large molecular models toward the data-efficient generation of molecules.

Future work and limitation. In this work, we apply our novel textual inversion scheme to the molecular language model [21], 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 [8], which will allow more intriguing applications of HI-Mol, such as composition (see Appendix F).

[4] This score is calculated by the improvement of the ROC-AUC score when the generated molecules are additionally added to the training data; higher is better.

References

[1] Youjun Xu, Kangjie Lin, Shiwei Wang, Lei Wang, Chenjing Cai, Chen Song, Luhua Lai, and Jianfeng Pei. Deep learning for molecular generation. *Future medicinal chemistry*, 2019.

[2] Dongyu Xue, Yukang Gong, Zhaoyi Yang, Guohui Chuai, Sheng Qu, Aizong Shen, Jing Yu, and Qi Liu. Advances and challenges in deep generative models for de novo molecule generation. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 2019.

[3] Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 2018.

[4] Pietro Bongini, Monica Bianchini, and Franco Scarselli. Molecular generative graph neural networks for drug discovery. *Neurocomputing*, 2021.

[5] Kurt LM Drew, Hakim Baiman, Prashanna Khwaounjoo, Bo Yu, and Jóhannes Reynisson. Size estimation of chemical space: how big is it? *Journal of Pharmacy and Pharmacology*, 2012.

[6] Sungsoo Ahn, Binghong Chen, Tianzhe Wang, and Le Song. Spanning tree-based graph generation for molecules. In *International Conference on Learning Representations*, 2022.

[7] Jaehyeong Jo, Seul Lee, and Sung Ju Hwang. Score-based generative modeling of graphs via the system of stochastic differential equations. In *International Conference on Machine Learning*. PMLR, 2022.

[8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[9] Sihyun Yu, Kihyuk Sohn, Subin Kim, and Jinwoo Shin. Video probabilistic diffusion models in projected latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[10] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*. PMLR, 2018.

[11] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*. PMLR, 2020.

[12] Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 2014.

[13] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, Artur Kadurin, Simon Johansson, Hongming Chen, Sergey Nikolenko, Alan Aspuru-Guzik, and Alex Zhavoronkov. Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. *Frontiers in Pharmacology*, 2020.

[14] Minghao Guo, Veronika Thost, Beichen Li, Payel Das, Jie Chen, and Wojciech Matusik. Data-efficient graph grammar learning for molecular generation. *arXiv preprint arXiv:2203.08031*, 2022.

[15] Jurgen Drews. Drug discovery: a historical perspective. *science*, 2000.

[16] James P Hughes, Stephen Rees, S Barrett Kalindjian, and Karen L Philpott. Principles of early drug discovery. *British journal of pharmacology*, 2011.

[17] Megan Stanley, John F Bronskill, Krzysztof Maziarz, Hubert Misztela, Jessica Lanini, Marwin Segler, Nadine Schneider, and Marc Brockschmidt. 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.

[18] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022.

[19] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848*, 2023.

[20] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

[21] Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

[22] Nathan Alexander, Nils Woetzel, and Jens Meiler. 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)*. IEEE, 2011.

[23] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 2018.

[24] Kristina Preuer, Philipp Renz, Thomas Unterthiner, Sepp Hochreiter, and Gunter Klambauer. Fréchet chemnet distance: a metric for generative models for molecules in drug discovery. *Journal of chemical information and modeling*, 2018.

[25] Fabrizio Costa and Kurt De Grave. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*. Omnipress; Madison, WI, USA, 2010.

[26] John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 2012.

[27] Chence Shi, Minkai Xu, Zhaocheng Zhu, Weinan Zhang, Ming Zhang, and Jian Tang. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382*, 2020.

[28] Chengxi Zang and Fei Wang. Moflow: an invertible flow model for generating molecular graphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.

[29] Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020.

[30] Youzhi Luo, Keqiang Yan, and Shuiwang Ji. Graphdf: A discrete flow model for molecular graph generation. In *International Conference on Machine Learning*. PMLR, 2021.

[31] Meng Liu, Keqiang Yan, Bora Oztekin, and Shuiwang Ji. Graphebm: Molecular graph generation with energy-based models. *arXiv preprint arXiv:2102.00546*, 2021.

[32] Tianze Luo, Zhanfeng Mo, and Sinno Jialin Pan. Fast graph generative model via spectral diffusion. *arXiv preprint arXiv:2211.08892*, 2022.

[33] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023.

[34] Xiangzhe Kong, Wenbing Huang, Zhixing Tan, and Yang Liu. Molecule generation by principal subgraph mining and assembling. In *Advances in Neural Information Processing Systems*, 2022.

[35] Rafael Gómez-Bombarelli, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, David Duvenaud, Dougal Maclaurin, Martin A Blood-Forsythe, Hyun Sik Chae, Markus Einzinger, Dong-Gwang Ha, Tony Wu, et al. Design of efficient molecular organic light-emitting diodes by a high-throughput virtual screening and experimental approach. *Nature materials*, 2016.

[36] Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander Gaunt. Constrained graph variational autoencoders for molecule design. *Advances in neural information processing systems*, 2018.

[37] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 1988.

[38] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 2020.

[39] Dimitrios Christofidellis, Giorgio Giannone, Jannis Born, Ole Winther, Teodoro Laino, and Matteo Manica. Unifying molecular and textual representations via multi-task language modelling. *arXiv preprint arXiv:2301.12586*, 2023.

[40] Yaxing Wang, Chenshen Wu, Luis Herranz, Joost Van de Weijer, Abel Gonzalez-Garcia, and Bogdan Raducanu. Transferring gans: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[41] Atsuhiro Noguchi and Tatsuya Harada. Image generation from small datasets via batch statistics adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[42] Esther Robb, Wen-Sheng Chu, Abhishek Kumar, and Jia-Bin Huang. Few-shot adaptation of generative adversarial networks. *arXiv preprint arXiv:2010.11943*, 2020.

[43] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

[44] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019.

[45] Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *European Conference on Computer Vision*. Springer, 2022.

[46] Han Altae-Tran, Bharath Ramsundar, Aneesh S Pappu, and Vijay Pande. Low data drug discovery with one-shot learning. *ACS central science*, 2017.

[47] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020.

[48] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[49] Sheng Wang, Yuzhi Guo, Yuhong Wang, Hongmao Sun, and Junzhou Huang. Smiles-bert: large scale unsupervised pre-training for molecular property prediction. In *Proceedings of the 10th ACM international conference on bioinformatics, computational biology and health informatics*, 2019.

[50] Benedek Fabian, Thomas Edlich, Héléna Gaspar, Marwin Segler, Joshua Meyers, Marco Fiscato, and Mohamed Ahmed. Molecular representation learning with language models and domain-relevant auxiliary tasks. *arXiv preprint arXiv:2011.13230*, 2020.

[51] Viraj Bagal, Rishal Aggarwal, PK Vinod, and U Deva Priyakumar. Molgpt: molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 2021.

[52] Zijie Geng, Shufang Xie, Yingce Xia, Lijun Wu, Tao Qin, Jie Wang, Yongdong Zhang, Feng Wu, and Tie-Yan Liu. De novo molecular generation via connection-aware motif mining. *arXiv preprint arXiv:2302.01129*, 2023.

[53] Teague Sterling and John J Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 2015.

[54] Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, et al. Molecular sets (moses): a benchmarking platform for molecular generation models. *Frontiers in pharmacology*, 2020.

[55] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *International conference on machine learning*. PMLR, 2017.

[56] Hanjun Dai, Yingtao Tian, Bo Dai, Steven Skiena, and Le Song. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*, 2018.

[57] Hiroshi Kajino. Molecular hypergraph grammar with its application to molecular optimization. In *International Conference on Machine Learning*. PMLR, 2019.

[58] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 2017.

[59] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

[60] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 2019.

[61] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 2022.

[62] Wenhao Gao and Connor W Coley. The synthesizability of molecules proposed by generative models. *Journal of chemical information and modeling*, 2020.

[63] Connor W Coley. Defining and exploring chemical spaces. *Trends in Chemistry*, 2021.

[64] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 2020.

Appendix: Data-Efficient Molecular Generation with Hierarchical Textual Inversion

A Method details

We utilize a recently introduced text-to-molecule model, Molt5-Large-Caption2Smiles [21] in our HI-Mol framework.[5] This model is constructed upon a text-to-text model, T5 [38], and molecular information is injected by additional training with both unpaired SMILES [37] string and caption-SMILES paired dataset. 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 maximum norm of 1.0. We update the assigned cluster c_n of each molecules for the first 5 epochs following Eq. (1). For interpolation-based sampling, we choose a uniform distribution $p(\lambda)$, (i.e., $p(\lambda) \coloneqq \mathcal{U}(l, 1-l)$), where λ controls relative contributions of interpolated token embeddings. We set $l = 0.0$ on the datasets in MoleculeNet benchmark [23], and $l = 0.3$ on the QM9 dataset [12].

B Datasets

MoleculeNet dataset. We perform generation experiments on single-task datasets, HIV, BBBP, and BACE, from MoleculeNet [23] 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 [23]. We emphasize that such *scaffold split* is widely considered in molecular generation domain [6]. Additional statistics for datasets on MoleculeNet are provided in Table 7.

Table 7: 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 [12], 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 [6] and randomly subset the training split with 2%, 5%, 10%, and 20% ratio for training our HI-Mol.

[5] 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 [7]:

- **Fréchet ChemNet Distance (FCD)** [24] 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 [58]:

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

  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)** [25] calculates the maximum mean discrepancy between the generated molecules and test molecules. We follow the evaluation protocol in [7], 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) [59] with the entire training split, e.g., contains both active (label-1) and inactive (label-0) molecules, of each dataset in the MoleculeNet benchmark [23]. 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** [7] proposes a diffusion model for graph structure, jointly learning both node and adjacency space by regarding each attributes as continuous values.
- **DiGress** [33] proposes a discrete diffusion process for graph structure to properly consider categorical distributions of node and edge attributes.
- **DEG** [14] suggests to construct 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 the 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 F.
- **JT-VAE** [10] proposes a variational auto-encoder that represents molecules as junction trees, regarding motifs of molecules as the nodes of junction trees.
- **PS-VAE** [34] utilizes a principal subgraph as a building block of molecules and generates molecules via merge-and-update subgraph extraction.
- **MiCaM** [52] introduces a connection-aware motif mining method to model the target distribution with the automatically discovered motifs.
- **CRNN** [3] builds generative models of SMILES strings with recurrent decoders.
- **STGG** [6] introduces a spanning tree-based molecule generation which learns the distribution of intermediate molecular graph structure with tree-constructive grammar.
- **GraphAF** [27] proposes an auto-regressive flow-based model for graph generation.
- **GraphDF** [30] introduces an auto-regressive flow-based model with discrete latent variables.
- **MoFlow** [28] utilizes a flow-based model for one-shot molecular generation.
- **EDP-GNN** [29] proposes a one-shot score-based molecular generative model, utilizing a discrete-step perturbation procedure of node and edge attributes.
- **GraphEBM** [31] introduces a one-shot energy-based model to generate molecules by minimizing energies with Langevin dynamics.
- **GSDM** [32] is a follow-up work of GDSS [7], suggesting to consider the spectral values of adjacency matrix instead of adjacency matrix itself.
- **CG-VAE** [36] proposes a recursive molecular generation framework that generates molecules satisfying the valency rules by masking out the action space.

E Ablation study

Table 8: 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 9: Ablation on the hierarchical tokens on the 2% subset of QM9.

Training prompt	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
The molecule is a $[S^*_1][S^*_2][S^*_3]$	6.529	0.032	**96.6**	21.4	37.2
The molecule is a $[S^*_1][S^*_2][D^*_n]$	0.474	0.002	87.0	72.9	72.0
The molecule is a $[S^*_1][I^*_{c_n}][D^*_n]$	**0.434**	**0.001**	90.7	**75.8**	**73.5**

Table 10: Ablation on the number of clusters K in Eq. (1) on the 2% subset of QM9.

K	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
0	0.210	0.001	92.2	61.4	47.5
1	0.486	0.002	**93.8**	70.8	72.3
3	0.474	0.002	87.0	72.9	72.0
5	0.455	0.002	88.9	76.5	71.1
10	0.443	**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**

Effect of prompt. In Table 8, 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 score, 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 tokens. In Table 9, we additionally conduct an ablation study on the effect of the hierarchical tokens. We compare our design with different choice of hierarchy: (1) utilization of only shared tokens, and (2) utilization of shared and detail tokens (without intermediate tokens). For (1), we use temperature sampling instead based on the categorical distribution from a molecular language model with temperature $\tau = 2.0$ since it is impossible to apply our interpolation-based sampling. The results show that consideration of each shared, intermediate, and detail tokens is indeed important for improving the quality measured with various metrics.

Effect of K. In Table 10, we report the quantitative results of the following three cases. First, we consider our proposed design with varying K from 3 to 30. IN addition, we consider two 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 and (b) $[S^*_1][S^*_2][D_n*]$ that the intermediate tokens are replaced with a shared token $[S^*_2]$, i.e., K=1. The results exhibit that the intermediate tokens are indeed crucial for the performance, given that the performance $3 \leq K \leq 30$ is much better than (a) and (b). 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.

F Additional experiments

Table 11: 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.

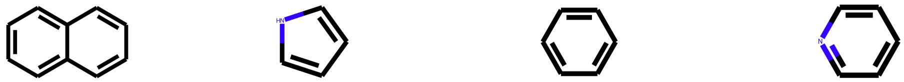
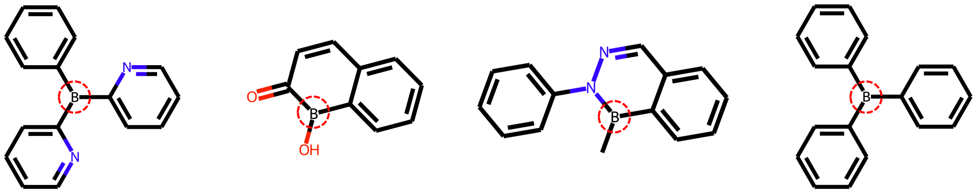
Input molecules for inversion

The molecule is a $[S^*][D_i^*]$
Generated molecules

The molecule is a boron compound of $[S^*][\bar{D}^*]$

Table 12: 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 11, we explore the compositionality of the learned token embeddings from HI-Mol. We learn the common features of 4 aromatic molecules[6], 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.

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 [60] which satisfy QED>0.5, SA>2.5, and GSK3B>0.3.[7] 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 12. Our HI-Mol shows superior results on learning several concepts, e.g., 33.2 → 52.0, compared to the most competitive baseline, STGG [6]. 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 [12] 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 12 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.

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

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

Table 13: Quantitative results of the few-shot generation experiments on subsets of the HIV dataset [23]. 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 [14] only for 30 samples.

# Samples	Method	Class	Grammar	Active. ↑	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
30	DEG [14]	Graph	✓	3.3	39.2	0.105	**100**	**100**	**100**
	STGG [6]	SMILES	✓	0.0	41.5	0.110	**100**	67	**100**
	CRNN [3]	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 [6]	SMILES	✓	1.3	28.2	0.054	**100**	90	**100**
	CRNN [3]	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 [6]	SMILES	✓	1.3	22.8	0.041	**100**	74	**100**
	CRNN [3]	SMILES	✗	2.7	30.0	0.064	51	**100**	**100**
	HI-Mol (Ours)	SMILES	✗	**10.3**	**20.8**	**0.020**	63	91	**100**

Table 14: Comparison with pre-trained model of STGG [6] 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**

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 a subset of the HIV dataset and report its quantitative result in Table 13. 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.

Comparison with pre-trained model. In Table 14, we report the performance of the baseline method by fine-tuning the pre-trained baseline model. Specifically, we fine-tune the model of STGG [6] pre-trained with the ZINC250k dataset [26] on the HIV dataset [23]. 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.

G Details on QM9 experiments

Table 15: Qualitative results for molecular generation varying the data ratio on QM9.

Ratio (%)	Grammar	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
2	✗	0.434	**0.001**	90.7	75.8	**73.5**
	✓	0.430	**0.001**	**100**	76.1	75.6
5	✗	0.412	**0.001**	89.4	85.8	70.4
	✓	0.410	**0.001**	**100**	86.4	72.4
10	✗	0.400	0.002	87.6	87.6	71.2
	✓	0.398	**0.001**	**100**	88.3	73.2
20	✗	0.384	**0.001**	86.7	87.8	70.0
	✓	**0.383**	**0.001**	**100**	**88.7**	71.8

Table 16: Comparison with the baseline with high Novelty via resampling strategy on QM9.

Method	Resampling ratio	FCD ↓	NSPDK ↓	Valid. ↑	Unique. ↑	Novelty ↑
GDSS [7]	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 15, we report additional experimental results varying the data ratio from 2% (2,113 molecules) to 20% (21,126 molecules). In particular, when we use 20% of the training data the performance improves further by 0.430 → 0.383 (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 16, 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 J for analysis on time complexity). The result shows that HI-Mol generates high-quality novel molecules from our desired target distribution.

H 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’’
```

I Analysis on interpolation-based sampling

Table 17: 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 [23] whose token embeddings are interpolated for each row.

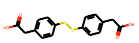
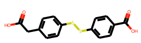
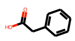
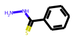
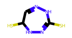
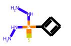
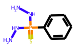
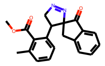
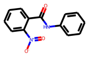
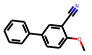
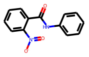
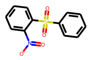
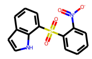
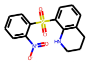
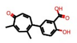
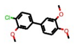
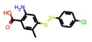
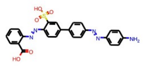
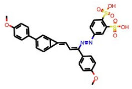
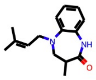
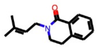
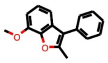
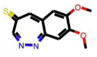
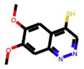
$[D_i^*]$	A similar chemical of $[S^*][\bar{I}^*][\bar{D}^*]$					$[D_j^*]$
	 $\lambda = 0.0$	 $\lambda = 0.3$	 $\lambda = 0.5$	 $\lambda = 0.7$	 $\lambda = 1.0$	
	 $\lambda = 0.0$	 $\lambda = 0.3$	 $\lambda = 0.5$	 $\lambda = 0.7$	 $\lambda = 1.0$	

Table 18: 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 17, 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 17 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 18 change the low-level semantics, i.e., size of molecules, of the generated molecules and interpolating detail (see the second row of Table 18) tokens change the high-level features, i.e., insertion of a single atom, of the generated molecules.

J Complexity

Table 19: 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 19, 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 large model. We believe that reducing this space for large language models, e.g., through [61] will be an interesting future direction.

Table 20: 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 [33]	-2.30±3.50	-2.67±3.15
	MiCaM [52]	1.02±3.29	0.69±2.09
	STGG [6]	0.53±2.79	-0.47±2.36
	HI-Mol (Ours)	**2.35**±2.71	**2.16**±1.64
BBBP	DiGress [33]	1.73±1.53	0.97±1.99
	MiCaM [52]	1.91±2.13	1.78±1.98
	STGG [6]	1.85±1.83	1.76±1.72
	HI-Mol (Ours)	**2.73**±2.01	**2.64**±1.75
BACE	DiGress [33]	-0.60±2.88	-0.91±1.82
	MiCaM [52]	-0.65±3.17	-1.11±2.95
	STGG [6]	2.34±2.15	2.01±1.45
	HI-Mol (Ours)	**3.53**±1.57	**3.39**±1.80

K Discussion on molecular optimization

In Table 4, 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 [6, 10, 27, 30], some prior works have noted that solely maximizing the PLogP value may yield unstable or hard-to-synthesize molecules [6, 62, 63]. In Figure 4, we show the visualizations of the optimized molecules with the highest PLogP values. Similar to the most competitive baseline, STGG [6], 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.

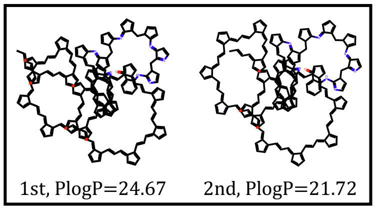


Figure 4: Visualizations of the generated molecules with $\gamma = 50$. The maximum PLogP among the training molecules is 4.52.

L Details of low-shot molecular property prediction

In Table 20, 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 1), we generate ×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 [59] from [64], 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).

M Broader impact

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. 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.