An Efficient Tokenization for Molecular Language Models

Seojin Kim, Jaehyun Nam, Jinwoo Shin
Korea Advanced Institute of Science and Technology (KAIST)
{osikjs,jaehyun.nam,jinwoos}@kaist.ac.kr

Abstract

Recently, molecular language models have shown great potential in various chemical applications, e.g., drug-discovery. These models adapt auto-regressive language models to molecular data by considering molecules as sequences of atoms, where each atom is mapped to individual tokens of the language models. However, such atom-level tokenizations limit the models' ability to capture the global structural context of molecules. To tackle this issue, we propose a novel molecular language model, coined ***C**ontext-**A**ware **M**olecular **T5*** (**CAMT5**). Inspired by the importance of the substructure-level contexts, e.g., ring systems, in understanding molecules, we introduce substructure-level tokenization for molecular language models. Specifically, we construct a tree structure for each molecule whose nodes correspond to important substructures, i.e., motifs. Then, we train our CAMT5 by considering a molecule as a sequence of motif tokens, whose order is determined by a tree-search algorithm. Under the proposed motif token space, one can incorporate chemical context with a significantly shorter token length (than atom-level tokenizations), which is useful for mitigating the issues during the auto-regressive molecular generation, e.g., error propagation. In addition, CAMT5 guarantees to generate a valid molecule with non-degeneracy, i.e., no ambiguity in the meaning of each token, which is also overlooked in previous models. Extensive experiments demonstrate the effectiveness of CAMT5 in the text-to-molecule generation task. Finally, we also propose a simple strategy of ensemble that can aggregate the outputs of molecular language models of different tokenizations, e.g., SMILES, SELFIES and ours, further boosting the quality of the generated molecules.

1 Introduction

Discovering molecules that match desired language descriptions is a long-standing goal in chemistry since it is an essential ingredient for practical deployments like drug-discovery and material design [1, 2, 3]. However, achieving such text-to-molecule generation poses a challenge due to the different structural modalities of language and molecules. To address this challenge, researchers have explored the fine-tuning of auto-regressive language models, with additional molecular data [4, 5], which is inspired by the recent success of language models in leveraging various domain knowledge including chemical concepts [6, 7]. Specifically, they treat each molecule as a sequence of tokens based on using string representations of molecules such as SMILES [8] and SELFIES [9]. Intriguingly, they show that these molecule-aware language models, i.e., molecular language models, can be obtained by learning the text-conditional molecule distribution, considering atoms of molecules as tokens of the language models [10, 11].

However, it is yet underexplored *which* tokenization strategy for a molecule is more effective for molecular language models. Previous state-of-the-art molecular language models [10, 11] have proposed to use atom-level token space, i.e., each atom is represented by a single token within the

AI for New Drug Modalities at NeurIPS 2024.

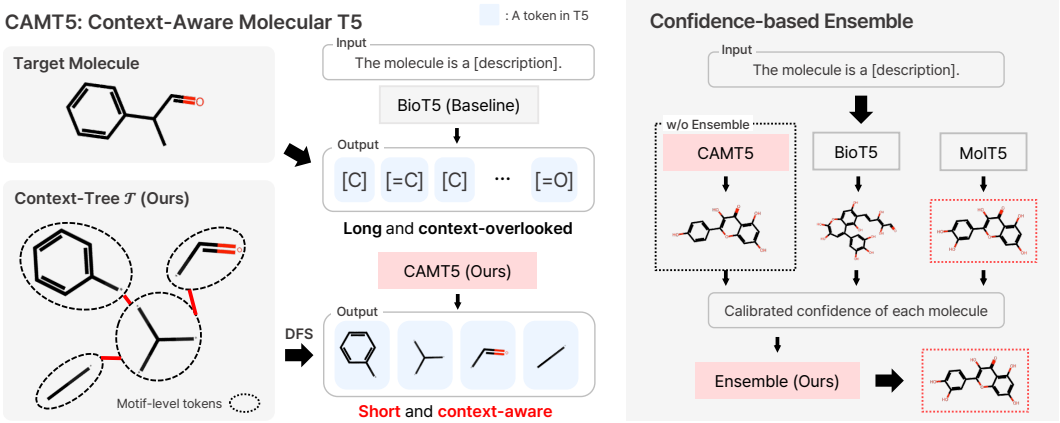


Figure 1: An overview of our proposed method. (1) Context-Aware Molecular T5 (CAMT5): we train molecular language models with motif-level token space. (2) Confidence-based Ensemble: we propose a simple ensemble strategy to further improve the generation quality of our model.

token space of molecular language models [4, 5, 10, 11, 12]. Even though they show remarkable performance as pioneering efforts, such atom-level tokenizations limit the models' ability to learn the crucial global contextual patterns in molecules, only focusing on local connectivities [13, 14, 15, 16]. For example, they consider the carbon atoms in a cyclohexyl group and aliphatic carbon chains to be the same token, despite their distinct structural context, e.g., a ring structure. In addition, such strategies represent a molecule as a long sequence of atom-by-atom tokens; this may disturb the desired text-to-molecule generation since auto-regressive language models often suffer from dealing with long sequences, e.g., error propagation [17, 18]. This leads to the question of *how to tokenize molecules in a context-preserving manner to train molecular language models more effectively.*

To answer this, we draw inspiration from the following chemical prior—the structural context of molecules is more effectively captured through their substructure-level, i.e., motif-level, characteristics rather than atom-level attributes [19, 20, 21]. Consequently, we hypothesize that the molecular language models can benefit from regarding a motif as a single token to incorporate various motif-level structural contexts in an efficient manner with a reduced number of tokens. To this end, we propose a new concept, i.e., motif-level token, in the token space of the molecular language models.

Contribution. We introduce a novel chemistry-inspired molecular language model coined **C**ontext-**A**ware **M**olecular **T5** (**CAMT5**). Here, we propose to use motif-level tokens to efficiently and effectively capture the structural context of molecules in molecular language models. Specifically, we first construct a tree of motifs from a molecule, coined Context-Tree, treating each motif as a token of our model. We then train our CAMT5 by regarding each molecule as a sequence of motif tokens whose order is determined by a tree-search algorithm on the Context-Tree (see Figure 1).

In particular, we carefully design the motif-level tokenization for CAMT5 to alleviate two drawbacks in tokenization used in the previous molecular language models. First, CAMT5 always generates a *valid* molecule, while MolT5 [10] often generates a *invalid* sequence of tokens that do not correspond to a molecule. Secondly, each of our motif-level tokens has a *unique* meaning, while some of the tokens in BioT5 [11] have *multiple* meaning, e.g., both an atom and the number of atoms in a ring are represented with a single token [9], resulting ambiguities to the model.

Finally, we also introduce a simple ensemble strategy to aggregate the outputs of molecular language models of different tokenizations, for further enhancing the performance of CAMT5 with help of other molecular language models. To this end, we first define the *confidence* of each molecular language model as a criterion to evaluate the generated molecules. To compare the confidences between different models, we propose to calibrate the confidence of each model based on the token length of the generated molecule. We then suggest selecting the molecule that achieves the highest calibrated confidence score as the output of the ensemble with respect to the given descriptions. This ensemble strategy allows us to fully leverage the advantages of each molecular language model, which results in the selection of more faithful molecules.

Table 1: Comparison of the token space in molecular language models. We mark Validity if a sequence of tokens always represents a valid molecule, and we mark Non-degeneracy if a single token corresponds to a unique molecular meaning.

Method	Validity	Non-degeneracy
MolT5 [10]	✗	✓
BioT5 [11]	✓	✗
CAMT5 (Ours)	✓	✓

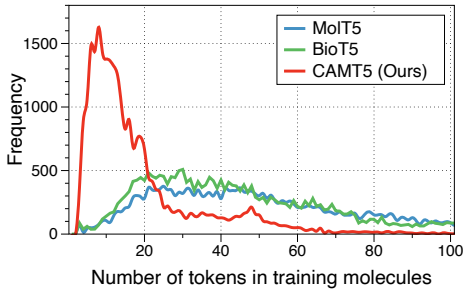


Figure 2: Distribution of tokens for molecular language models in the ChEBI-20 dataset.

We verify the effectiveness of our method on the popular ChEBI-20 [22] and PCDes [23] benchmarks. In ChEBI-20, CAMT5 improves the ratio of molecules that exactly matches the description (Exact; higher is better) by 21.7 $\rightarrow$ 26.6, compared to the previous best-performing baseline. Moreover, CAMT5 generates more faithful molecules that are similar to the targets, i.e., 0.796 $\rightarrow$ 0.826, 0.725 $\rightarrow$ 0.766, and 0.593 $\rightarrow$ 0.645 in MACCS, RDK, and Morgan FTS (higher is better), respectively. We also show that our confidence-based ensemble strategy further improves the performance, improving the Exact metric by 26.6 $\rightarrow$ 30.3. We also demonstrate the various applications of our CAMT5, such as data-efficient molecular generation [24] and molecule modificaiton.

2 Related work

Molecular language models. Inspired by the recent success in auto-regressive language models [6, 7], there have been several attempts to adapt these language models to achieve molecule-aware language models, i.e., molecular language models [4, 5, 10, 11, 12, 22]. Specifically, they fine-tune existing language models, e.g., T5 [6], with molecular data by treating molecules as sequences of tokens. In particular, MolT5 [10] employs the widely used SMILES [8] representation to convert a molecule into tokens of molecular language models. However, this model often generates *invalid* token sequences that violate the grammar and, therefore, do not correspond to valid molecules. To alleviate this issue, BioT5 [11] proposes to use SELFIES [9], a representation guaranteed to generate valid molecules. However, SELFIES introduces ambiguities, i.e., *degeneracy*, in the meanings of tokens, leading to sub-optimal performance in modeling the token distribution. For example, the '[0]' token can be interpreted completely differently: an oxygen atom or an indicator of a ring system comprising six atoms preceding this token. To overcome the limitations of the token spaces in previous molecular language models, we carefully design the token space of CAMT5 with (1) guaranteed *validity* of the generated molecules with (2) *non-degeneracy* in the meanings of tokens.

Context-aware molecule learning. Recent studies in the molecular domain have explored the concept of *context-aware* learning of molecules. For example, [13, 14, 21] learn chemistry-friendly molecule embeddings by leveraging motif-level context in self-supervised learning frameworks, and [25, 26] approximate 3D conformers of molecules while preserving the geometric structural context of motifs. A notable approach in this line of work is context-aware molecular generation [19, 20, 27, 28]. Specifically, they learn the distribution of motifs rather than learning the distribution of atoms. Intriguingly, they show superior performance in generating molecules from the learned molecule distribution, due to the incorporation of contextual patterns in the motifs of the molecules. However, recent molecular language models still rely on learning the atom-level token space [10, 11], which limits the incorporation of the structural context of molecules. In contrast to these works, we aim to develop a context-aware molecular language model based on the motif-level token space.

3 Method

In Section 3.1, we explain an overview of our problem. In Section 3.2, we provide the description of CAMT5, our proposed context-aware molecular language model. In Section 3.3, we describe our confidence-based ensemble strategy.

3.1 Problem description

We formulate our problem of *text-to-molecule generation* as follows. Our goal is to train a molecular language model f_θ so that $f_\theta(\mathbf{x}) = \mathbf{m}$, where $\mathbf{x}$ is a text description of the desired molecule and $\mathbf{m}$ is the corresponding molecule (see Table 3 for an example). Recent studies [10, 11] have shown that such f_θ can be trained with description-molecule pairs $\{\mathbf{x}_k, \mathbf{m}_k\}_{k=1}^N$ with the following objective:

$$\mathcal{L}(\theta; \mathbf{x}_k, \mathbf{m}_k) \coloneqq \mathcal{L}_{\texttt{CE}}\Big(f_\theta(\mathbf{x}_k), \mathbf{m}_k\Big), \tag{1}$$

where $\mathcal{L}_{\texttt{CE}}$ denotes cross-entropy loss, and $\mathbf{x}_k$ and $\mathbf{m}_k$ denote the k-th text description and the corresponding tokenized molecule in the token space of the molecular language model, respectively.

Here, the choice of tokenization strategy for $\mathbf{m}_k$ plays a crucial role in training an effective f_θ [11], since the sequence of tokens has to reflect the structural context of the original molecule. However, previous molecular language models overlook such importance, relying only on the local connectivity of atoms based on the atom-level tokenization, e.g., SMILES [8] and SELFIES [9]. Furthermore, they represent a molecule with a long sequence of tokens based on the individual atoms, and this may disturb the desired text-to-molecule generation since auto-regressive language models often suffer from dealing with long sequences, e.g., error propagation [17, 18]. Our contribution lies in resolving such challenges by incorporating the substructure-level contextual patterns into the token space of molecular language models to efficiently represent a molecule in a context-aware manner.

3.2 CAMT5: Context-Aware Molecular T5

Context-aware molecule tokenization. We propose to construct the molecule token space of CAMT5 to efficiently reflect the structural context of molecules. To this end, we consider chemically meaningful fragments, i.e., motifs, as individual tokens, in contrast to previous methods based on atom-level tokens [10, 11]. Specifically, we consider the following set of atoms, i.e., a motif, as a single token: (1) atoms forming a ring structure and (2) atoms connected by a non-single bond (see Figure 1). Such atoms are rigidly bound to each other and represent an important structural context, such as resonance [29]. An atom not associated with (1) and (2) is considered as a single motif.

We then propose to represent a molecule as a sequence of motif-level tokens, based on the order of the tree-search algorithm on a tree of motifs. Consider a molecule graph $G = (V, E)$ with the set of atoms V and edges E. We construct $\mathcal{T}(G) = (\mathcal{V}, \mathcal{E})$, namely Context-Tree, where $\mathcal{V} = \{M_i\}_{i=1}^n$ is the set of n motifs with $M_i = (V_i, E_i)$, and $\mathcal{E}$ is the set of bonds between motifs. Here, $\mathcal{T}(G)$ efficiently preserves all the information of the original molecule graph G, i.e., $V = \cup V_i$ and $E = \cup_i E_i \cup \mathcal{E}$, with context-enriched nodes by replacing atom-level nodes V with motif-level nodes $\mathcal{V}$, satisfying $|\mathcal{V}| \leq |V|$. Consequently, we obtain the sequence of motif tokens by enumerating $\mathcal{V}$ based on the order of the depth-first-search (DFS) algorithm, i.e., $\mathbf{m}_{\text{CAMT5}} = [M_1, ..., M_n]$. We then train our molecular language model f_{CAMT5} with $\{\mathbf{x}_k, \mathbf{m}_{\text{CAMT5},k}\}_{k=1}^N$ using the training objective in Eq. (1). Note that our method ensures the (1) *validity* of the generated token sequences since we do not introduce tokens that should appear as a pair, c.f., the branch tokens '(' and ')' in SMILES [8]. Also, our tokens are (2) *non-degenerate* by construction; a single token represents only a single motif, c.f., '[O]' as an oxygen atom or an indicator of a ring system comprising six atoms preceding this token in SELFIES [9]. We provide further details about our tokenization strategy in Appendix A.

Our context-enriched tokenization plays a crucial role in discriminating the atoms with different structural contexts. For example, the aromatic carbon atoms in phenyl group (represented as [C][= C][C][= C][C][= C][C][= C][Ring1][= Branch1] in BioT5 [11]) and the aliphatic carbon atoms (represented as [C][C][C][C][C][C]) are completely different in chemical context, due to the resonance and the ring structure. However, previous molecular language models do not distinguish the difference between them, regarding both carbons as the same [C] token. Our CAMT5 alleviates this issue by assigning different tokens for the entire phenyl groups and the carbons in aliphatic carbons.

Pre-training and fine-tuning. We follow the common pre-training and fine-tuning strategies in previous molecular language models [4, 5, 10, 11, 22]. Specifically, we build our molecular language model based on T5 [6] language model. We first pre-train the language models with text corpus (Colossal Clean Crawled Corpus [6]) and molecule corpus (ZINC-15 [30]). To effectively incorporate such unpaired data for each domain, we use the masked language modeling objective introduced in the original T5 paper, which is also utilized in previous molecular language models [10, 11]. We then fine-tune the models with the description-molecule paired dataset based on the objective in Eq. (1).

Table 2: Quantitative results of the text-to-molecule generation task in the CheBI-20 [22] and PCDes [23] benchmarks. `small` and `base` denote that the model is derived from T5-small and T5-base [6], respectively. We highlight the best score in bold.

Method	Representation	Exact ↑	MACCS ↑	RDK ↑	Morgan ↑	Valid. ↑
Results on the CheBI-20 benchmark.						
MolT5$_{\texttt{small}}$ [10]	SMILES [8]	14.4	0.636	0.584	0.498	0.80
BioT5$_{\texttt{small}}$ [11]	SELFIES [9]	17.7	0.766	0.691	0.547	**1.0**
CAMT5$_{\texttt{small}}$ (Ours)	**Context-Tree (Ours)**	**19.7**	**0.796**	**0.732**	**0.600**	**1.0**
MolT5$_{\texttt{base}}$ [10]	SMILES [8]	19.2	0.672	0.623	0.546	0.81
BioT5$_{\texttt{base}}$ [11]	SELFIES [9]	21.7	0.796	0.725	0.593	**1.0**
CAMT5$_{\texttt{base}}$ (Ours)	**Context-Tree (Ours)**	**26.6**	**0.826**	**0.766**	**0.645**	**1.0**
Results on the PCDes benchmark.						
MolT5$_{\texttt{small}}$ [10]	SMILES [8]	2.6	0.446	0.401	0.270	0.76
BioT5$_{\texttt{small}}$ [11]	SELFIES [9]	2.6	0.594	0.533	0.338	**1.0**
CAMT5$_{\texttt{small}}$ (Ours)	**Context-Tree (Ours)**	**3.2**	**0.615**	**0.558**	**0.364**	**1.0**
MolT5$_{\texttt{base}}$ [10]	SMILES [8]	4.7	0.503	0.448	0.320	0.78
BioT5$_{\texttt{base}}$ [11]	SELFIES [9]	3.2	0.600	0.537	0.348	**1.0**
CAMT5$_{\texttt{base}}$ (Ours)	**Context-Tree (Ours)**	**5.2**	**0.644**	**0.582**	**0.397**	**1.0**

3.3 Confidence-based ensemble of molecular language models

We propose a simple ensemble method to further improve the generation quality of our CAMT5, using other molecular language models with different tokenizations, e.g., MolT5 [10], BioT5 [11]. Here, we note that traditional ensemble strategies, e.g., majority voting, are often not applicable in molecular language models due to the large and complicated molecule space. For example, each of the molecular language models in Table 2 generates different molecules for 77.5% of the language descriptions given in the dataset, in which case majority voting is not possible.

To tackle this issue, we suggest to use the *confidence* of each molecule as a proxy for the quality measure. Let f_i be the i-th molecular language model and $\mathbf{m}_i = [T_1, ..., T_{K_i}]$ be the generated K_i tokens from f_i with respect to the given description $\mathbf{x}$. Then, we define the confidence-based ensemble $f_{\texttt{Ensemble}}$ of the molecular language models $\{f_1, ..., f_n\}$ as follows:

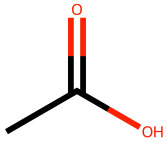
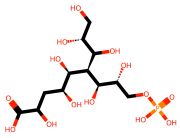
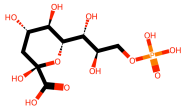
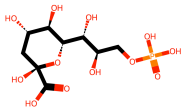
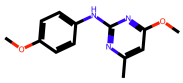
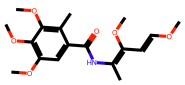
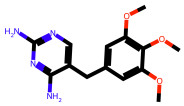
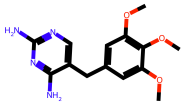
$$C_\alpha(\mathbf{m}_i; f_i, \mathbf{x}) = \frac{\sum_{j=1}^{K_i} \texttt{log} P_{f_i}([T_j]|\mathbf{x}, [T_1 ..., T_{j-1}])}{K_i^\alpha} = -K_i^{1-\alpha} \mathcal{L}_{\texttt{CE}}(f_i(\mathbf{x}), \mathbf{m}_i), \tag{2}$$

$$f_{\texttt{Ensemble}}(\mathbf{x}) = \mathbf{m}_k, \text{ where } k = \texttt{argmax}_i \, C_\alpha(\mathbf{m}_i; f_i, \mathbf{x}). \tag{3}$$

A natural way to calculate the confidence of $\mathbf{m}_i$ is using the average log-likelihood of each token, which corresponds to $\alpha = 1.0$ in Eq. (2). However, we find that such a naïve choice of α leads to sub-optimal performance in $f_{\texttt{Ensemble}}$, since the *scale* of $C_{1.0}$ is different across the molecular language models. Specifically, we find that $P_{f_i}([T_j]|\mathbf{x}, [T_1, ..., T_{j-1}]) \approx 1$ when j is large, e.g., MolT5 [10] and BioT5 [11] often become over-confident after generating the first few tokens so that mistakenly assigns high $C_{1.0}$ for $\mathbf{m}_i$ because of their long token length (see Figure 2). To alleviate this, we suggest to *calibrate* the average log-likelihood by a factor of $\alpha \in [0, 1]$ to align the confidence scale of each model, which turns out to be crucial for achieving an effective $f_{\texttt{Ensemble}}$ (see Table 4). The specific value of α is determined by the value that achieves the best Exact score in the validation set (see Appendix A for detailed explanation).

We note that this ensemble strategy is particularly useful in practical scenarios. Previously, people simply chose the best-performing model among the existing molecular language models, ignoring other on-average underperforming models. However, when the selected model is not *confident* in a certain text description, other models may provide more confident alternatives. In this case, our confidence-based ensemble strategy can be applied to further improve the performance of the best-performing model, i.e., CAMT5, with the help of other models, i.e., MolT5 and BioT5.

Table 3: Qualitative results of the text-to-molecule generation task in the CheBI-20 [22] (the first row) and PCDes [23] (the second low) benchmarks. For each model, we visualize the generated molecules with respect to the given description. We report the RDK score between the generated and ground truth molecules below each visualization. We set the highest score in bold.

Description	$\text{MolT5}_{\text{base}}$	$\text{BioT5}_{\text{base}}$	$\textbf{CAMT5}_{\text{base}}$ **(Ours)**	Target
The molecule is a ketoaldonic acid phosphate that is 3-deoxy-D-glycero-beta-D-galacto-nonulosonic acid...	 RDK: 0.19	 RDK: 0.61	 RDK: **1.00**	
It is an aminopyrimidine antibiotic whose structure consists of pyrimidine 2,4-diamine...	 RDK: 0.38	 RDK: 0.44	 RDK: **1.00**	

4 Experiments

We verify the effectiveness of our CAMT5 by conducting comprehensive experiments. In Section 4.1, we explain our experimental setups, such as datasets and evaluation metrics. In Section 4.2, we present the text-to-molecule generation results on the ChEBI-20 and PCDes benchmarks. In Section 3.3, we present the results of our confidence-based ensemble strategy. In Section 4.4, we apply our CAMT5 in various downstream tasks, including data-efficient molecular generation and molecule modification.

4.1 Experimental setup

Baselines. A few works have introduced molecule representations for molecular language models. Specifically, MolT5 [10] utilizes SMILES [8] representation, and BioT5 [11] suggests to use SELFIES [11] representation. We extensively compare our CAMT5 with these works.

Datasets. We evaluate the text-to-molecule generation performance of molecular language models in two popular benchmarks, ChEBI-20 [22] and PCDes [23]. The ChEBI-20 dataset consists of 33,008 description-molecule pairs, which are separated by 26,407/3,301/3,300 pairs as train/validation/test splits [11]. The PCDes dataset contains more challenging 15,000 description-molecule pairs, which are separated by 10,500/1,500/3,000 pairs as train/validation/test splits [23]. They are both derived from the qualified description-molecule pairs from the open-sourced PubChem database [31], where each text description describes the structure and the chemical properties of the corresponding molecule. We provide the more information about the datasets in Appendix B.

Training setup. Previous molecular language models, e.g., MolT5 [10] and BioT5 [11], are trained with different configurations, e.g., pre-training datasets,[1] which limits the genuine comparison with their proposed token space. To alleviate this issue, we have aligned the pre-training and fine-tuning configurations of each molecular language model. Specifically, we use publically available uni-modal datasets, i.e., Colossal Clean Crawled Corpus (C4) [6] for the text corpus and ZINC-15 [30] for the molecule corpus, to pre-train the baselines and our models. We provide a further description of the training configurations in Appendix A.

Metrics. For an extensive evaluation of text-to-molecule generation, we utilize various metrics which reflect the quality of the generated molecules, e.g., similarity to the target molecule. We provide the details of the metric as follows:

- **Exact**: The percentage of the generated molecules that exactly match with the target molecule.

[1]For example, BioT5 [11] utilized additional pre-training datasets compared to MolT5 [22], but they have not released the datasets in public.

Table 4: Quantitative results of our confidence-based ensemble in the CheBI-20 [22] and PCDes [23] benchmarks. `small` and `base` denote that the model is derived from T5-small and T5-base [6], respectively. Ensemble denotes the model $f_{\texttt{Ensemble}}$, which is constructed from {MolT5, BioT5, CAMT5} with average log-likelihood confidence, i.e., $\alpha = 1.0$ (see Eq. (2)). Calibration denotes that we use the calibrated confidence by setting α as the value that achieves the best Exact score in the validation set. We highlight the best score in bold.

Method	Exact ↑	MACCS ↑	RDK ↑	Morgan ↑	Valid. ↑
Results on the CheBI-20 benchmark.					
CAMT5$_{\texttt{small}}$	19.7	0.796	0.732	0.600	**1.0**
+ Ensemble	23.2	0.805	0.744	0.622	**1.0**
+ Calibration	**23.7**	**0.813**	**0.753**	**0.632**	**1.0**
CAMT5$_{\texttt{base}}$	26.6	0.826	0.766	0.645	**1.0**
+ Ensemble	29.9	0.829	0.773	0.661	**1.0**
+ Calibration	**30.3**	**0.837**	**0.783**	**0.672**	**1.0**
Results on the PCDes benchmark.					
CAMT5$_{\texttt{small}}$	3.2	0.615	0.558	0.364	**1.0**
+ Ensemble	3.8	0.617	0.558	0.375	**1.0**
+ Calibration	**4.0**	**0.624**	**0.566**	**0.383**	**1.0**
CAMT5$_{\texttt{base}}$	5.2	0.644	0.582	0.397	**1.0**
+ Ensemble	5.7	0.650	0.584	0.407	**1.0**
+ Calibration	**6.0**	**0.655**	**0.591**	**0.414**	**1.0**

- **MACCS/RDK/Morgan FTS (MACCS/RDK/Morgan)**: Metrics that measure the fingerprint-level similarity between the generated molecule and the target molecule. MACCS [32], RDK [33], and Morgan [34] fingerprints are used. We report the average score for each metric; if the generated token sequence does not represent a valid molecule, we set this score as 0.
- **Validity (Valid.)**: The ratio of the generated token sequences which represent a valid molecule.[2]

4.2 Main experiments

Table 2 summarizes the quantitative results of the text-to-molecule generation tasks in the ChEBI-20 [22] and the PCDes [23] benchmarks. In both benchmarks, our method consistently outperforms the baseline models by generating desirable molecules corresponding to the text description. In ChEBI-20, CAMT5$_{\texttt{base}}$ significantly improves the Exact score of the best-performing baseline, BioT5$_{\texttt{base}}$, by 21.7 $\rightarrow$ 26.6, which highlights the superiority of our molecule tokenization scheme. Also, the improvements in the fingerprint similarity-based scores, e.g., 0.593 $\rightarrow$ 0.645 in Morgan FTS, demonstrate the usefulness of CAMT5 in capturing the substructure-level semantics of molecules. Notably, CAMT5$_{\texttt{small}}$ (80M parameters) already outperforms BioT5$_{\texttt{base}}$ (250M parameters) in several metrics, e.g., 0.725 $\rightarrow$ 0.732 in RDK FTS, with only a third of the model size. Our CAMT5 also shows its effectiveness in the more challenging PCDes benchmark, e.g., 4.7 $\rightarrow$ 5.2 in Exact and 0.537 $\rightarrow$ 0.582 in RDK. In Table 3, we provide visualizations of the generated molecules. We observe that our CAMT5 effectively generates molecules that contain crucial motifs of the target molecules, e.g., phosphorus acid and hydropyran, and this further demonstrates the importance of our motif-level tokenization scheme in CAMT5. We provide additional experimental results in Appendix C.

4.3 Results on confidence-based ensemble

In Table 4, we report the quantitative results of the generated molecules from our confidence-based ensemble model introduced in Section 3.3. In this experiment, we construct an ensemble model $f_{\texttt{Ensemble}}$ in Eq. (3) based on the molecular language models {MolT5, BioT5, CAMT5} for each model size, i.e., `small` and `base`. When calibration is not used, we set $\alpha = 1.0$, i.e., the confidence score becomes the average log-likelihood of the generated tokens (see Eq. (2)). When calibration

[2] In BioT5 [11] and CAMT5 (Ours), Validity is guaranteed to be 1.0 due to the characteristics of the used representation, SELFIES [9] and Context-Tree (Ours), respectively.

Table 5: Qualitative results of the confidence-based ensemble in the CheBI-20 [22] (the first row) and PCDes [23] (the second low) benchmarks. We visualize the cases that other models, i.e., MolT5 and BioT5, help our CAMT5 through $f_{\texttt{Ensemble}}$ when the confidence (maximally 0.00) of our generated model is relatively low. We report the confidence and the RDK score between the generated and ground truth molecules below each visualization. Here, the molecule with the highest confidence is selected as the output of $f_{\texttt{Ensemble}}$ (see Eq. (3)). We set the highest score in bold.

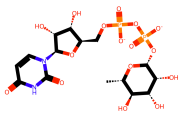
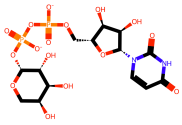
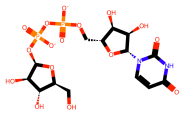
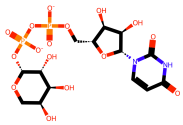
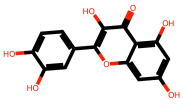
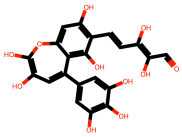
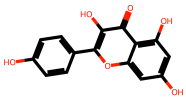
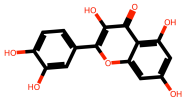
Description	MolT5$_{\texttt{base}}$	BioT5$_{\texttt{base}}$	**CAMT5$_{\texttt{base}}$ (Ours)**	Target
The molecule is a nucleotide-sugar oxoanion resulting from removal of two protons from diphosphate...	 RDK: 0.97 Confidence: -0.04	 RDK: **1.00** Confidence: **-0.01**	 RDK: 0.98 Confidence: -0.11	
It appears as yellow needles or yellow powder. Converts to anhydrous form at 203-207Â°F...	 RDK: **1.00** Confidence: **0.00**	 RDK: 0.30 Confidence: -0.16	 RDK: 0.94 Confidence: -0.11	

Table 6: Quantitative results of the data-efficient molecular generation on the HIV dataset in the MoleculeNet benchmark [35]. Following [24], we provide the results based on the 500 non-overlapping generated molecules to the training dataset. We set the highest score in bold. ↑ and ↓ denote higher and lower values are better, respectively.

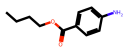
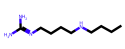
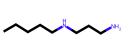
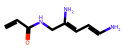
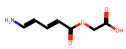
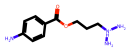
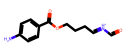
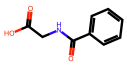
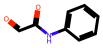
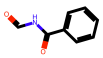
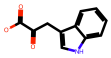
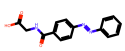
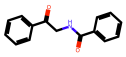
Method	Active ↑	FCD ↓	NSPDK ↓	Valid ↑	Unique ↑	Novelty ↑
MolT5$_{\texttt{base}}$ [10]	6.4	20.8	0.053	73.4	72.2	**100**
BioT5$_{\texttt{base}}$ [11]	6.0	21.2	0.034	**100**	**73.6**	**100**
CAMT5$_{\texttt{base}}$ (Ours)	**8.8**	**20.0**	**0.029**	**100**	68.7	**100**

is used, we find α which leads to the best Exact score in the validation set. Firstly, our ensemble improves CAMT5 in overall metrics due to our confidence-based molecule selection strategy; the quality of the generated molecule is closely correlated with the log-likelihood of the molecular language models. In particular, our ensemble achieves a notable improvement in the Exact score, e.g., 26.6 → 29.9 in the ChEBI-20 benchmark. Applying our calibration technique further improves the quality of the generated molecules, e.g., 0.773 → 0.783 in RDK FTS, by alleviating the over-confidence issue in the long token sequences. In Table 5, we provide some examples where our CAMT5 is not quite confident in its output, and other models, i.e., MolT5 and BioT5, generate more confident molecules. In this case, the ensemble model selects the generated molecules generated by MolT5 or BioT5, which are indeed more similar to the target molecules. In summary, on-average underperforming models, i.e., MolT5 and BioT5, can help the best-performing model, i.e., CAMT5, through the confidence-based selection strategy of our ensemble model.

4.4 Applications of CAMT5

Data-efficient molecular generation. We explore the applicability of our CAMT5 in data-efficient molecular generation, which is an important application of molecular language models in practical scenarios; the collection of task-relevant molecular data is expensive. Specifically, we adapt molecular language models to learn the distribution of 1,232 active molecules of the HIV dataset in the MoleculeNet benchmark [35] via HI-Mol framework [24], and then generate molecules from the learned distribution. As shown in Table 6, our CAMT5 outperforms the baseline models in Active., FCD, and NSPDK, demonstrating the effectiveness of our CAMT5 in learning the underlying distribution of low-shot molecules. We believe that the key success of our CAMT5 is to better capture

Table 7: Experimental results of the molecule modification. We visualize the generated molecules with respect to the prompt with an additional condition, i.e., solubility in water. We report the LogP score below each visualization. Molecules with lower LogP values are more soluble in water. For each model, we report the top-2 molecules that match the property description among the 100 molecules, generated by temperature sampling with $\tau = 2.0$.

Query	$\text{MolT5}_{\texttt{base}}$		$\text{BioT5}_{\texttt{base}}$		**$\text{CAMT5}_{\texttt{base}}$ (Ours)**	
Prompt: "The molecule is an amino acid ester ... Make it *soluble* in water." (Lower LogP is better)						
 LogP: 2.22	 LogP: 0.43	 LogP: 1.12	 LogP: -0.39	 LogP: -0.36	 LogP: **-1.55**	 LogP: -0.48
Prompt: "The molecule is an N-acylglycine... Make it *insoluble* in water." (Higher LogP is better)						
 LogP: 0.50	 LogP: 0.82	 LogP: 0.57	LogP: 1.82	 LogP: 1.70	 LogP: **2.92**	 LogP: 2.30

the global context of molecules through motif-level tokenization, which is also crucial in learning the features among low-shot molecules. We provide the details of the metrics in Appendix A.

Molecule modification. We demonstrate the applicability of our CAMT5 in *modifying* molecules. Consider a molecular language model f, where $f(\mathbf{x}) = \mathbf{m}$ with a molecule description $\mathbf{x}$ and the corresponding molecule $\mathbf{m}$. We examine a scenario where the description $\mathbf{x}$ is slightly modified to $\mathbf{x}'$ by adding an additional propmt, such as $\mathbf{x}' = \mathbf{x} +$ "Make it *insoluble* in water". Here, the resulting molecule $\mathbf{m}' = f(\mathbf{x}')$ is expected to (1) maintain the structural similarity to $\mathbf{m}$ and (2) capture the additional prompt in $\mathbf{x}'$. Although researchers have investigated the modification of molecules based on numerical properties [36, 37], the exploration of modifications based on text descriptions is yet under-explored despite its potential in practical applications.

In Table 7, we consider the descriptions in the ChEBI-20 test set where $\text{MolT5}_{\texttt{base}}$, $\text{BioT5}_{\texttt{base}}$, and $\text{CAMT5}_{\texttt{base}}$ each generate the same molecule as shown in the Query column. We then generate molecules with the prompt, "Make it *soluble/insoluble* in water.", in addition to the original description based on temperature sampling with $\tau = 2.0$. Among 100 generated molecules, we show the top-2 molecules that match the additional prompt, i.e., molecules with the lowest/highest LogP for the first/second row, respectively. The results demonstrate that our CAMT5 achieves superior modification ability by (1) preserving the crucial substructures of the original molecule in the Query column, e.g., the aniline structure in the first row, and (2) effectively incorporating the additional prompts (see LogP values). We hypothesize that the improvement is due to the our unique motif-level tokenization strategy; this is useful to preserve the motifs in the modified molecules and motifs are more closely related to the molecular properties, e.g., solubility in water, than individual atoms.

5 Conclusion

We propose CAMT5, a chemical context-aware molecular language model. Specifically, we propose to utilize motif-level tokenization to better understand the chemical structural context. In addition, we propose a confidence-based ensemble strategy to further improve the generation quality of CAMT5. Extensive experiments demonstrate the effectiveness of our tokenization scheme and the ensemble strategy in improving the performance of text-to-molecule generation and several applications.

Limitation and future work. In this work, we mainly focus on improving the token space of molecular language models, which is crucial yet under-explored problem in molecular language models. An interesting future direction would be applying our tokenization to advanced training strategies for molecular langague models, e.g., leveraging pseudo-data [4] and multi-task language modeling [5], which are originally based on the previous tokenization schemes, e.g., SMILES [8]. We believe that those works will further benefit from our carefully designed context-aware tokenzation.

References

[1] Bing Su, Dazhao Du, Zhao Yang, Yujie Zhou, Jiangmeng Li, Anyi Rao, Hao Sun, Zhiwu Lu, and Ji-Rong Wen. A molecular multimodal foundation model associating molecule graphs with natural language. *arXiv preprint arXiv:2209.05481*, 2022.

[2] Haisong Gong, Qiang Liu, Shu Wu, and Liang Wang. Text-guided molecule generation with diffusion language model. *arXiv preprint arXiv:2402.13040*, 2024.

[3] Jiatong Li, Yunqing Liu, Wenqi Fan, Xiao-Yong Wei, Hui Liu, Jiliang Tang, and Qing Li. Empowering molecule discovery for molecule-caption translation with large language models: A chatgpt perspective. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[4] Yuhan Chen, Nuwa Xi, Yanrui Du, Haochun Wang, Jianyu Chen, Sendong Zhao, and Bing Qin. From artificially real to real: Leveraging pseudo data from large language models for low-resource molecule discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21958–21966, 2024.

[5] Dimitrios Christofidellis, Giorgio Giannone, Jannis Born, Ole Winther, Teodoro Laino, and Matteo Manica. Unifying molecular and textual representations via multi-task language modelling. In *International Conference on Machine Learning*, pages 6140–6157. PMLR, 2023.

[6] 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. *Journal of machine learning research*, 21(140):1–67, 2020.

[7] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

[8] David Weininger. 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.

[9] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.

[10] Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. *arXiv preprint arXiv:2204.11817*, 2022.

[11] Qizhi Pei, Wei Zhang, Jinhua Zhu, Kehan Wu, Kaiyuan Gao, Lijun Wu, Yingce Xia, and Rui Yan. Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. *arXiv preprint arXiv:2310.07276*, 2023.

[12] Zequn Liu, Wei Zhang, Yingce Xia, Lijun Wu, Shufang Xie, Tao Qin, Ming Zhang, and Tie-Yan Liu. Molxpt: Wrapping molecules with text for generative pre-training. *arXiv preprint arXiv:2305.10688*, 2023.

[13] Seojin Kim, Jaehyun Nam, Junsu Kim, Hankook Lee, Sungsoo Ahn, and Jinwoo Shin. Fragment-based multi-view molecular contrastive learning. In *Workshop on"Machine Learning for Materials"ICLR 2023*, 2023.

[14] Kha-Dinh Luong and Ambuj K Singh. Fragment-based pretraining and finetuning on molecular graphs. *Advances in Neural Information Processing Systems*, 36, 2024.

[15] Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z Li. Mole-bert: Rethinking pre-training graph neural networks for molecules. In *The Eleventh International Conference on Learning Representations*, 2022.

[16] Zhiyuan Liu, Yaorui Shi, An Zhang, Enzhi Zhang, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. Rethinking tokenizer and decoder in masked graph modeling for molecules. *Advances in Neural Information Processing Systems*, 36, 2024.

[17] Lijun Wu, Xu Tan, Di He, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. Beyond error propagation in neural machine translation: Characteristics of language also matter. *arXiv preprint arXiv:1809.00120*, 2018.

[18] Minh Lê and Antske Fokkens. Tackling error propagation through reinforcement learning: A case of greedy dependency parsing. *arXiv preprint arXiv:1702.06794*, 2017.

[19] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pages 2323–2332. PMLR, 2018.

[20] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*, pages 4839–4848. PMLR, 2020.

[21] Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee. Motif-based graph self-supervised learning for molecular property prediction. *Advances in Neural Information Processing Systems*, 34:15870–15882, 2021.

[22] Carl Edwards, ChengXiang Zhai, and Heng Ji. Text2mol: Cross-modal molecule retrieval with natural language queries. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 595–607, 2021.

[23] Zheni Zeng, Yuan Yao, Zhiyuan Liu, and Maosong Sun. A deep-learning system bridging molecule structure and biomedical text with comprehension comparable to human professionals. *Nature communications*, 13(1):862, 2022.

[24] Seojin Kim, Jaehyun Nam, Sihyun Yu, Younghoon Shin, and Jinwoo Shin. Data-efficient molecular generation with hierarchical textual inversion. *arXiv preprint arXiv:2405.02845*, 2024.

[25] Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation. *Advances in Neural Information Processing Systems*, 35:24240–24253, 2022.

[26] Thomas Seidel, Christian Permann, Oliver Wieder, Stefan M Kohlbacher, and Thierry Langer. High-quality conformer generation with conforge: Algorithm and performance assessment. *Journal of Chemical Information and Modeling*, 63(17):5549–5570, 2023.

[27] Xiangzhe Kong, Wenbing Huang, Zhixing Tan, and Yang Liu. Molecule generation by principal subgraph mining and assembling. *Advances in Neural Information Processing Systems*, 35:2550–2563, 2022.

[28] 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. In *International Conference on Learning Representations*, 2023.

[29] Eric V Anslyn and Dennis A Dougherty. *Modern physical organic chemistry*. University science books, 2006.

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

[31] Yanli Wang, Jewen Xiao, Tugba O Suzek, Jian Zhang, Jiyao Wang, and Stephen H Bryant. Pubchem: a public information system for analyzing bioactivities of small molecules. *Nucleic acids research*, 37(suppl_2):W623–W633, 2009.

[32] Joseph L Durant, Burton A Leland, Douglas R Henry, and James G Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of chemical information and computer sciences*, 42(6):1273–1280, 2002.

[33] Nadine Schneider, Roger A Sayle, and Gregory A Landrum. Get your atoms in order an open-source implementation of a novel and robust molecular canonicalization algorithm. *Journal of chemical information and modeling*, 55(10):2111–2120, 2015.

[34] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[35] 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*, 9(2):513–530, 2018.

[36] Ziqi Chen, Martin Renqiang Min, Srinivasan Parthasarathy, and Xia Ning. A deep generative model for molecule optimization via one fragment modification. *Nature machine intelligence*, 3(12):1040–1049, 2021.

[37] Yiheng Zhu, Jialu Wu, Chaowen Hu, Jiahuan Yan, Tingjun Hou, Jian Wu, et al. Sample-efficient multi-objective molecular optimization with gflownets. *Advances in Neural Information Processing Systems*, 36, 2024.

[38] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[39] 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*, 58(9):1736–1741, 2018.

[40] Fabrizio Costa and Kurt De Grave. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*, pages 255–262. Omnipress; Madison, WI, USA, 2010.

[41] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Appendix: An Efficient Tokenization for Molecular Language Models

A Experimental details

Details on context-aware tokenization. For each motif-level token M_i, there may exist several $v \in V_i$ where $(u, v) \in \mathcal{E}$ for some $u \in V$, i.e., a single motif which is connected to several motifs in $\mathcal{T}$ (see the second token of CAMT5 in Figure 1 for an example). In this case, we additionally store the used order of such v's based on the DFS algorithm within each token. We utilize this order when converting the sequence of tokens to a molecule. For a given sequence of tokens, we convert the sequence to a molecule by the exactly inverse consequences of the construction of the token sequences. Here, the number of children of each token is the number of aformentioned v's. If there exist unvisited v's after the conversion, we simply ignore them, i.e., we consider them to be connected to a hydrogen atom, not to other motif tokens. The number of motif tokens introduced in our CAMT5 is 15,230 in the ChEBI-20 and PCDes benchmarks.

Details on confidence-based ensemble. In our confidence-based ensemble strategy, α in Eq. (2) plays a role in calibrating the confidences of molecules from molecular language models with different tokenization strategy, i.e., different token lengths. Specifically, we find $\alpha \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ which leads to the best Exact score in the validation set. When we select the maximum confidence molecule through $f_{\texttt{Ensemble}}$, we exclude invalid molecules, i.e., if an invalid molecule achieves the maximum confidence, we select the second-maximum confidence molecule. In practice, it does not incur additional costs since one can directly check the validity of a token sequence.

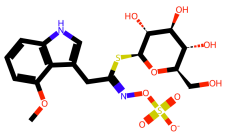
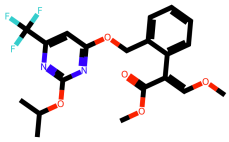
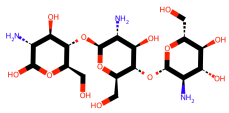
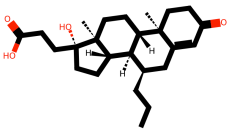
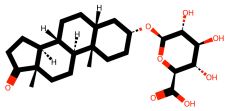
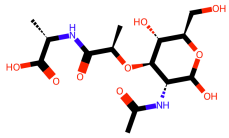
Details on pre-traning and fine-tuning. We pre-train each molecular language model, i.e., MolT5, BioT5, and CAMT5, with a text corpus (Colossal Clean Crawled Corpus [6]) and a molecule corpus (ZINC-15 [30]). To effectively incorporate such unpaired data for each domain, we use the masked language modeling objective, i.e., replace corrupted spans [6]. We pre-train each model for 100k steps with a batch size of 128, using the cosine learning rate scheduler with the base learning rate of $1e^{-3}$ and the warmup steps of 1k based on the adamw optimizer. We fine-tune each model with description-molecule data pairs in the ChEBI-20 [22] and the PCDes [23] benchmarks based on the objective in Eq. (1) with the molecule token representation of each model. We fine-tune the models in 50k steps with the batch size of 48, using a constant learning rate at the rate of $5e^{-4}$ based on clipping the gradient by 30.0.

Metrics in data-efficient molecualr generation. We use six metrics to evaluate the data-efficient molecular generation [24]. We evaluate the quality of 500 generated sample from the prior distribution $p(\lambda) = \mathcal{U}(-0.3, 0.7)$ where $\mathcal{U}$ denotes the uniform distribution. Active denotes the ratio of the *active* molecules that achieve the desired property. We use pre-trained classifier on the HIV dataset with 5-layer GIN [38]. FCD [39] denotes the Fréchet distance which measures the distance between the source distribution and the target distribution based on ChemNet. NSPDK [40] also measures the distance between the source distribution and the target distribution based on an algorithmic computation. Valid is the ratio of the generated token sequences that represent valid molecules. Unique is the ratio of different generated molecules among the valid molecules. Novelty is the ratio of valid molecules that are not in the training set. In our experiments, Novelty is always 100, since we only consider the generated molecules that do not overlap with the training data, for a reliable measure in Active score, which is suggested in [24].

Computing resources. In our experiments, we use Intel(R) Xeon(R) Gold 6426Y CPU @ 2.50GHz and A6000 48GB GPUs.

B Dataset details

Table 8: Visualizations of description-molecule pairs in ChEBI-20 [22] and PCDes [23].

ChEBI-20		PCDes	
The molecule is an indolylmethylglucosinolate that is the conjugate base of 4-methoxyglucobrassicin, obtained by deprotonation of the sulfo group. It is a conjugate base of a 4-methoxyglucobrassicin.		It is a member of pyrimidines, an organofluorine acaricide, a methyl ester, an enoate ester and an enol ether. It has a role as a mitochondrial cytochrome-bc1 complex inhibitor.	
The molecule is an amino trisaccharide comprising of three 2-amino-2-deoxy-D-glucopyranose units joined by beta-(1->4) linkages. It has a role as a marine metabolite and a eukaryotic metabolite.		It is a spironolactone derivative and a potent aldosterone antagonist on mineralocorticoid biosynthesis with diuretic activity . As an aldosterone antagonist, it may inhibit sodium resorption in the collecting duct and may eventually lead to diuresis.	
The molecule is a steroid glucosiduronic acid. It has a role as a human metabolite and a mouse metabolite. It derives from a 3alpha-hydroxy-5beta-androstan-17-one.		It is an L-alanine derivative consisting of an N-acetyl-D-muramoyl group attached to L-alanine via an amide linkage. It is a glyco-amino acid and a L-alanine derivative. It is a conjugate acid of a N-acetyl-D-muramoyl-L-alaninate.	

We evaluate our CAMT5 in the text-to-molecule generation tasks of the ChEBI-20 [22] and PCDes [23] benchmarks, which consist of description-molecule pairs from the open-sourced PubChem database [31]. In Table 8, we visualize some description-molecule pairs of each benchmark.

C Additional results

Table 9: Quantitative results of the text-to-molecule generation task in the CheBI-20 [22] benchmark. `large` denotes that the model is derived from T5-large [6]. We highlight the best score in bold.

Method	Representation	Exact ↑	MACCS ↑	RDK ↑	Morgan ↑	Valid. ↑
Results on the CheBI-20 benchmark.						
MolT5$_{\texttt{large}}$ [10]	SMILES [8]	24.0	0.704	0.663	0.562	0.86
BioT5$_{\texttt{large}}$ [11]	SELFIES [9]	28.0	0.801	0.746	0.610	**1.0**
CAMT5$_{\texttt{large}}$ (Ours)	**Context-Tree (Ours)**	**29.3**	**0.828**	**0.776**	**0.652**	**1.0**

In Table 9, we report the text-to-molecule generation results based on the models derived from the T5-large model [6]. Our CAMT5 also shows improvements in this setup, e.g., 28.0 → 29.3 in Exact and 0.746 → 0.776 in RDK, demonstrating its potential in scaling up to future large-scale models.

D Social impacts

This work will accelerate improvements in the field of molecular language models, which will affect many chemical applications such as drug discovery and material design. However, malicious or unintended usage of molecular language models (including our models) may lead to a potential threat of the generating harmful chemicals. We believe that safeguarding these models is an important future research direction, which is also widely studied in other domains (e.g., language domain [41]).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We clearly provide our contribution and scope in abstract and introduction (Section 1).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We provide the limitations and future works in Conclusion (Section 5)

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: We do not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We fully describe our method and experimental setup in Section 3, Section 4, and Appendix A.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will upload the code once the paper is accepted.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide the details in Section 4 and Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars since our experiments are computationally expensive (training language models).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We specified the computer resources in Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: This paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We provide the discussion in Appendix C.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: We will carefully examine the potential risks for misuse until we release the model.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We carefully mentioned the owners of assets in the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.