MOLECULE GENERATION THROUGH REASONING WITH LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Molecule generation is significant for its potential in scientific discovery and practical applications, e.g., accelerating drug discovery by directly generating candidate molecules. Recent attempts often frame this task as a *translation* problem from molecular caption to structural representation, such as SMILES. This paper first examines the feasibility of modeling the task as a reasoning process with large language models (LLMs), generating higher-quality molecules through structural decomposition and recombination within Chain-of-Thought (CoT). We then introduce a workflow for curating accurate CoT data, incorporating both machine and expert verification. Lastly, we demonstrate that with a limited dataset of 4,213 high-quality samples, namely **MolCoT4K**, we elicit strong reasoning capabilities for molecule generation in open-source LLMs such as Qwen2.5-7B, achieving state-of-the-art exact match accuracy over strong open-source baselines (e.g., MolT5 and LlaSMol) as well as advanced commercial LLMs like GPT-4o. Moreover, the resulting model, **MolGeneration**, attains a Pass@16 exact match accuracy of 48.46%, highlighting its strong potential for real-world experimental applications when supported by a feasible external verifier or chemistry experts. Our analysis shows that the correctness of the CoT path is crucial, while reasoning ability primarily enhances accuracy in fine-grained molecule generation. The dataset, model, and training codebase will be released to the community[1].

1 INTRODUCTION

Molecule generation plays a vital role in modern drug discovery and material design, by directly proposing a few candidate molecules rather than exhaustively searching through vast chemical spaces. A common paradigm frames this task as a translation problem, mapping natural language descriptions of molecular properties or functions into molecular representations such as SMILES (Weininger, 1988; Edwards et al., 2022). Recent advances in language models have enabled promising progress along this line (Pei et al., 2023; Liu et al., 2023; Yu et al., 2024), demonstrating the feasibility of molecule generation from textual prompts.

However, this translation-based paradigm suffers from two fundamental shortcomings. First, it relies heavily on large amounts of annotated training data, which is costly to obtain in the molecular domain due to the need for expert curation and experimental validation (Mayr et al., 2018). Second, it fails to explicitly leverage the compositional structure of molecules, leaving the process of assembling functional groups and substructures implicit within black-box end-to-end training. As a result, current systems often generate chemically plausible but structurally inaccurate molecules. Previous work has attempted to address these challenges by scaling up model size and training data (Edwards et al., 2022; Yu et al., 2024), or by introducing additional chemical knowledge, such as the BioT5 family of models (Pei et al., 2023; 2024). While these strategies yield incremental improvements, they remain fundamentally limited by their lack of explicit reasoning over molecular structure. Even strong LLM-based systems can struggle to align natural language descriptions with valid molecules.

To address these limitations, we investigate a perspective by framing molecule generation as a step-by-step reasoning task. As illustrated in Figure 1, the model is first prompted to explicitly generate intermediate structures and functional groups from the molecular description, such as the *main*

[1] https://anonymous.4open.science/r/MolGeneration-DE3F

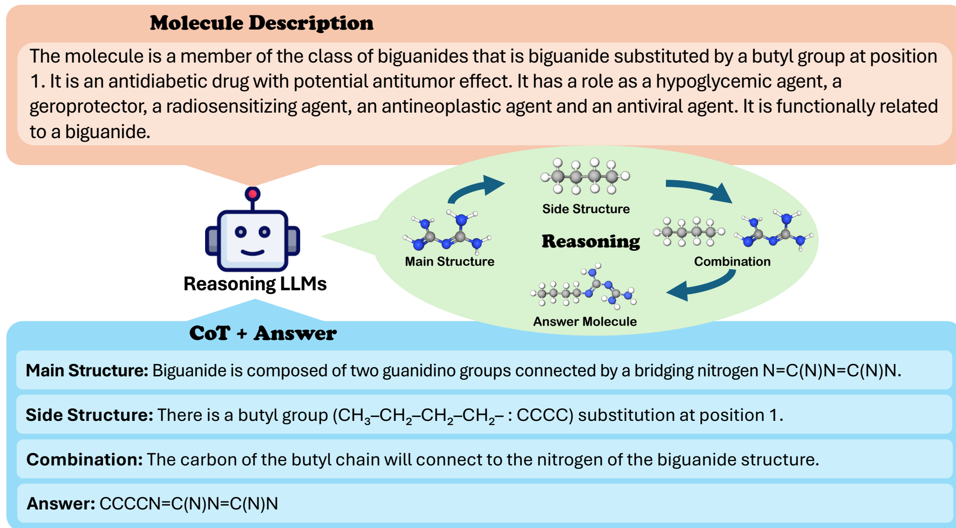


Figure 1: Illustration of molecule generation through reasoning: Given the molecule description, a reasoning LLM first decomposes the structure into a main structure N=C(N)N=C(N)N and side structure CCCC, then combines them at position 1, resulting in the final answer CCCCN=C(N)N=C(N)N. See Appendix Table 6 for more generated examples.

structure and *side structure* (substituents), and then to construct the final structural representation (SMILES) based on these components. This compositional reasoning paradigm produces interpretable traces that align with chemical knowledge and improve generation accuracy.

To facilitate this paradigm, we construct the first high-quality chain-of-thought (CoT) dataset for molecule generation, **MolCoT-4K**. In particular, we leverage the strong commercial LLM, GPT-4o-mini, to produce initial raw CoT data, and subsequently develop a dedicated curation tool that enables chemical experts to efficiently label, verify, and refine these outputs. Through this process, we obtain a total of $4{,}213$ rigorously curated high-quality samples.

Building upon MolCoT-4K, we frame a simple yet effective two-step training strategy to enable open-source LLMs with reasoning ability. Specifically, we use Qwen2.5-7B as the testbed and first perform supervised fine-tuning (SFT) on MolCoT-4K, which yields clear chemistry reasoning traces and substantial exact-match improvements over the base model. We then apply reinforcement learning (RL), where the model is encouraged to generate both a reasoning trace and a final molecule representation from the molecular description. The RL training process is guided by two types of rewards: format-based rewards, which ensure the reasoning process is included in the generation, and exact match rewards, defined as a binary exact match score.

Finally, we integrate large-scale translation pairs with our high-quality CoT samples to develop our strong molecule generation model, **MolGeneration**. On the challenging real-world evaluation dataset SMolInstruct, our MolGeneration outperforms both advanced commercial LLMs (GPT-4 and Gemini-2.5) and specialized molecular translation systems (e.g., MolT5), achieving a state-of-the-art Exact Match (EM) score of 44.97%—surpassing GPT-4 (6.40%), Gemini-2.5 (19.49%), and MolT5 (31.70%). Moreover, assuming access to an external verifier, our model reaches 48.46% EM score under 16-fold sampling, showing strong potential for real-world molecule generation when chemical experts are engaged in selecting candidates.

Our contributions are as follows:

- We demonstrate, for the first time, the effectiveness of framing molecule generation as a reasoning task: by deconstructing individual functional groups, generating intermediate results, and recombining them, one can trace a coherent reasoning path for molecule generation that leads to more accurate molecular outcomes.

- We present **MolCoT-4K**, a sample-efficient, high-quality reasoning dataset constructed with both machine and expert verification to maximize data availability, and release it to the community. To the best of our knowledge, MolCoT-4K is the first dataset that effectively elicits reasoning capabilities for molecule generation in open-source LLMs.
- Based on MolCoT-4K, we provide our specialized molecular LLM, **MolGeneration**, achieving state-of-the-art performance in molecule generation, substantially surpassing current specialized models as well as advanced commercial LLMs. Furthermore, with access to an assumed verifier, the notable improvement highlights a strong potential in a real-world experimental environment.

2 Background

Molecule representation. Molecule representation is crucial for encoding chemical structures and functions when building chemical models. Traditional hand-crafted representations include RDK (Schneider et al., 2015), Morgan (Cereto-Massagué et al., 2015), and MACCS (Durant et al., 2002) fingerprinting. RDK is more sensitive to structural changes compared to Morgan and MACCS fingerprinting (O'Boyle & Sayle, 2016). In contrast, string-based representations such as the Simplified Molecular-Input Line-Entry System (SMILES) (Weininger, 1988) serialize molecular graphs into human-readable strings. SMILES are generated from molecular graphs using a depth-first traversal. SMILES has become the standard for representing molecules in large chemical databases and machine learning pipelines, owing to its simplicity and compatibility with language models (Edwards et al., 2022; 2021; Yu et al., 2024). SELFIES (Self-Referencing Embedded Strings) (Krenn et al., 2020) was introduced as a robust molecular string representation. However, prior work (Yu et al., 2024) has shown that SELFIES offers no clear advantage for molecule generation, and the amount of annotated SELFIES data remains relatively limited. In this study, we adopt SMILES as the molecular representation and apply RDKit Landrum et al. (2025) to convert output into canonical SMILES (Weininger et al., 1989), a uniform SMILES representation, to address the potential many-to-one mapping issue for evaluation.

Language models with molecule generation. For convenient and controllable generation, Edwards et al. (2022) formulate molecule generation as a conditional language modeling task, translating natural language descriptions of molecular functions and properties directly into the corresponding SMILES. They post-train the multilingual language model, T5 (Raffel et al., 2020) with molecular datasets, resulting in MolT5 for both molecule generation and captioning. BioT5 (Pei et al., 2023) introduces a comprehensive pre-training framework that enriches cross-modal integration in biology through chemical knowledge and natural language associations. BioT5+ (Pei et al., 2024) extends the BioT5 framework and is tailored to enhance biological research and drug discovery. Mol-Instructs (Fang et al., 2024) is designed to boost LLM proficiency in the specialized domain of bio-molecular studies. LlaSMol (Yu et al., 2024) shows that post-trained LLMs can achieve strong results on a comprehensive set of chemistry tasks, including molecule generation, outperforming commercial LLMs like GPT-4 (Achiam et al., 2023) and Claude 3 Opus by a significant margin. Furthermore, Jiang et al. (2024) propose a novel framework that interweaves model fine-tuning with data augmentation to mitigate the scarcity of high-quality data. All of these models generate molecules directly in a black-box end-to-end manner, relying heavily on the availability of a large-scale pairwise dataset. The inherent chemical characteristics of molecular structures, such as the compositional information of functional groups, are not explicitly exploited. Our work addresses this gap by eliciting molecular reasoning ability from LLMs, enabling molecule generation through structural decomposition of functional groups and step-by-step reasoning.

Datasets for molecule generation. ChEBI-20 (Edwards et al., 2021) is designed for molecule retrieval. It consists of 33,010 molecule-description pairs and serves as the post-training dataset for MolT5. ChEBI-20 contains descriptions of more than 20 words. LlaSMol introduced SMolInstruct, a large-scale, high-quality molecular dataset, which covers 14 selected chemistry tasks and 3.3 million samples, providing a foundation for training and evaluating LLMs in chemistry. The molecule generation part of SMolInstruct, which consists of $56{,}498$ samples, is much larger than ChEBI-20. The L+M-24 dataset (Edwards et al., 2024) specifically targets molecule generation and captioning. This dataset focuses on functional attributes and generates natural-language descriptions of these properties via GPT-4. Additionally, existing databases such as ZINC (Irwin & Shoichet, 2005), ChEMBL (Gaulton et al., 2012), and PubChem (Kim et al., 2021) offer molecule SMILES and description paired data. However, these datasets are large but noisy. In this paper, we introduce **MolCoT-4K**, a high-quality molecular reasoning dataset constructed through a combination of machine verification and expert review. It comprises 4,213 samples, each verified via exact matching and annotated into five levels of reasoning trace availability (ranging from 1 to 5; see Section 3.2). Human experts then refine each CoT to ensure chemical soundness, yielding the final gold-standard dataset. To the best of our knowledge, this is the first dataset explicitly designed to elicit molecular reasoning ability in LLMs.

3 Data Collection, Processing, and Curation

3.1 Data Collection and Filtering

We first collect description-molecule pairs from the train splits of three widely used datasets, i.e., ChEBI-20, SMolInstruct, and L+M-24, resulting in a total number of 243,411 pairwise samples. Then, considering noisy and invalid data, we apply multiple steps of data filtering: 1) we remove samples when the corresponding SMILES cannot be processed by RDKit, including those that have invalid or incomplete molecular representations, or failure in 3D conformation generation; 2) we discard samples whose descriptions lack a clear molecular definition by requiring the text length to exceed 20 words[2], following the construction rule of ChEBI-20; and 3) we further manually remove part of the meaningless descriptions produced during step-2. Finally, we obtain a dataset D containing 55,236 samples, where each sample i is a tuple (m_i, s_i) corresponding to a molecule description and its SMILES representation.

3.2 Chain-of-Thought (CoT) Data Construction

To maximize efficiency in data construction, given a pairwise molecule dataset D, we prompt GPT-4o-mini k times for each molecule description (m), obtaining k corresponding potential CoT reasoning processes (c') prior to generating the final SMILES (s'). This procedure yields a raw dataset D' in which each sample i consists of k tuples $(m_i, c'_{ij}, s'_{ij}, s_i)$, where $j \in [1, k]$. Then, for each sample i, we retain it only if there exists an s'_{ij} that either exactly matches the gold answer s_i or can be converted into s_i using the RDKit tool. If multiple matches are found, we randomly select one s'_{ij} and the corresponding c'_{ij} as machine-generated c'_i and s'_i. This filtering process substantially reduces the dataset size from 55,236 to 4,213 samples, reflecting the difficulty current LLMs face in handling this task effectively. In the resulting dataset D'', each sample is a tuple (m_i, c'_i, s'_i).

Note that both c'_i and s'_i are machine-generated. While s'_i aligns exactly with the gold answer s_i, the availability of c'_i is not guaranteed. In practice, we observe a considerable proportion of c' that lacks sound chemical inference, e.g., main structure errors; see Appendix H for the error distribution within the dataset. To maximize data availability, we involve human experts to manually revise each c'_i for correctness and develop a CoT curation tool that supports error-level annotation, manual correction, and related I/O functions to accelerate the process; see Appendix E for details.

Finally, we construct MolCoT-4K, a high-quality molecule generation dataset of 4,213 samples, split into train/valid/test sets in an 8:1:1 ratio. Each sample is a tuple (m_i, c^*_i, c'_i, s'_i), where c^*_i is a CoT trace with chemical validity verified by experts. To simplify, we use (m_i, c_i, s_i) to denote (m_i, c^*_i, s'_i) samples in MolCoT-4K in the following text, unless specified otherwise.

4 Methods

This section outlines the training approach used to elicit chemical reasoning ability in LLMs. We adopt a two-stage post-training strategy: First, we apply SFT using (m_i, c_i, s_i) in MolCoT-4K as a warm-up stage to equip the model with basic accuracy and the ability to generate CoT responses. Note that the model is supervised by c and s given a description m. We then apply reinforcement learning to further improve the model's reasoning capability.

[2] For example, a short unclear description would be: "The molecule is cancer treatment." for $CN_1CCN(CC_2CCN(C_3C(N)=CNCC_3F)CC_2)CC_1$.

4.1 Reinforcement Learning with Molecule Reward

We apply the widely used reinforcement learning method, GRPO (Shao et al., 2024), to further improve the model's reasoning ability after SFT. GRPO discards the value function and estimates the advantage in a group-wise normalization. GRPO typically introduces a KL penalty term to regularize divergence between the online and reference policies, while we eliminate the KL penalty term to allow the policy to explore more aggressively in the chemical space.

Formally, for a pair of question-answer pair $(q, a) \in D$, GRPO method samples a group of G responses $\{o_i\}_{i=1}^{G}$ from the old behavior policy $\pi_{\theta_{\text{old}}}$. A reward model is then used to score SMILES and format correctness, yielding group rewards $\{r_i\}_{i=1}^{G}$. Outcome supervision provides the normalized reward at the end of each output o_i and sets the advantages $\hat{A}_{i,t}$ of all tokens in the output as the normalized reward:

$$\hat{A}_{i,t} = \widetilde{r}_i = \frac{r_i - \text{mean}(\{r_i\}_{i=1}^{G})}{\text{std}(\{r_i\}_{i=1}^{G})}. \tag{1}$$

The resulting GRPO loss without KL penalty is:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\min\left[\alpha_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}(\alpha_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_{i,t}\right]\right\}, \tag{2}$$

where

$$\alpha_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}. \tag{3}$$

ε is a hyper-parameter. Here, we use the molecule description and SMILES pair, (m_i, s_i) in any molecule generation dataset as q and a, respectively.

4.2 Molecular Reward Design

We follow prior RL applications to math reasoning (Shao et al., 2024; Yu et al., 2025; Liu et al., 2025) and design rewards based on correctness and output format constraints. The correctness reward ensures that the model receives a reward only when the generated SMILES exactly matches the ground-truth SMILES[3] . The format reward ensures that the model receives a reward only when the generated CoT contains both the <Think> and <Answer> tags.

However, for molecule generation, correctness and formatting alone are insufficient due to the complexity of chemical syntax and combinations. We therefore design a molecule-specific ensemble reward strategy, where we additionally introduce a SMILES validity reward that ensures the generated output corresponds to a chemically valid molecule, verified by the RDKit tool.

5 Experimental Setup

Base models. We conduct extensive experiments on advanced open-source LLMs, including the Qwen (Qwen, 2025a;b), LLaMA (Grattafiori et al., 2024), and Mistral (Jiang et al., 2023) families, covering model sizes from 0.5B to 32B.

Evaluation metrics. We employ three evaluation metrics ranging from coarse to fine: Validity, RDK-FTS (Schneider et al., 2015), and Exact Match (EM). **Validity** measures the proportion of generated SMILES strings that correspond to chemically valid molecules, regardless of correctness, processed by the RDKit tool. **RDK-FTS** encodes molecules using RDKit fingerprints and calculates the mean Tanimoto similarity (also known as the Jaccard index) between the generated and reference SMILES strings. Finally, **EM** requires the generated output to be identical to the reference SMILES, or convertible to the exact same SMILES using the RDKit tool.

[3]Our preliminary experiments show minimal gains when using similarity-based rewards, e.g., those from the RDKit tool (Landrum et al., 2025). Thus, we chose the simplest binary reward version.

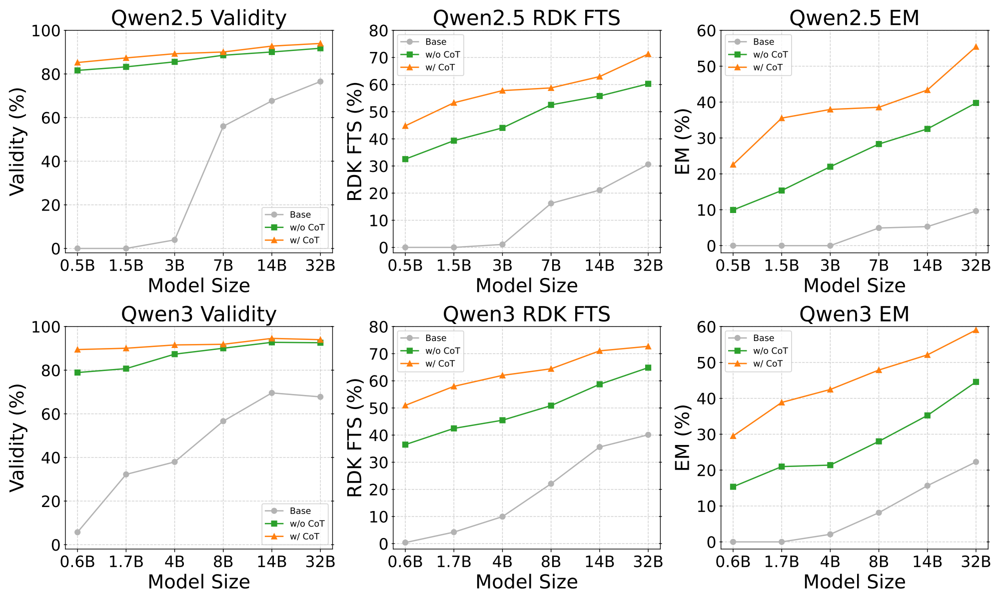


Figure 2: Performance comparison on our MolCoT-4K dataset across metrics, model scales, and training methods for Qwen2.5 and Qwen3. *Validity*, *RDK-FTS*, and *EM* represent average performance at progressively finer levels of granularity: the proportion of valid molecules, average structural similarity, and exact match accuracy, respectively. The results show that reasoning ability contributes most to improvements in fine-grained molecular generation (EM). Detailed results, along with those for other models such as LLaMA and Mistral, are provided in Appendix M.

Training setting. Our experiments were conducted with HuggingFace Transformers (Wolf et al., 2020) and TRL (von Werra et al., 2020) on NVIDIA H100 (100 GB) GPUs. In the SFT stage, we train for three epochs with a learning rate of 5×10^{-5}. We set the maximum sequence length to 2,048 tokens with AdamW, using a 0.03 warmup. In the RL stage, we apply GRPO for one epoch with a learning rate of 1×10^{-6}. We set the maximum sequence length to 2,048 tokens with AdamW, using a 0.1 cosine warmup.

Decoding Strategy. We mainly employ two decoding strategies to comprehensively evaluate the effectiveness of our method and the resulting model. (1) Beam search is applied to demonstrate performance in a constrained experimental setting; and (2) Best-of-N sampling is applied to approximate the upper bound of performance (Pass@N) when an external verifier is available. For Best-of-N settings, we set top-p as 0.9, top-k as 50, and temperature as 1.0, to ensure that samples are drawn from relatively higher-probability regions while maintaining diversity.

6 Experimental Results and Analysis

6.1 On the Role of Reasoning for Molecular Generation

In this section, we begin by evaluating the impact of incorporating decomposition and recombination information, represented as CoT traces, prior to generating the final SMILES strings. Experiments are conducted on the MolCoT-4K dataset (see Section 3.2) across a broad range of LLMs differing in size and family. We focus here on the simplest setting, supervised fine-tuning, where w/ CoT and w/o CoT denote training with (m, c, s) and (m, s) pairs, respectively, from MolCoT-4K.

Figure 2 shows the corresponding results on Qwen2.5 and Qwen3 families, measured at progressively finer levels of granularity by Validity, RDK-FTS, and EM scores (see Section 5). It is easy to see that: (1) Within the Qwen family (e.g., Qwen2.5 from 0.5B to 32B), performance improves steadily as the model size increases; (2) Compared to the base models, supervised fine-tuning substantially improves molecular generation performance. Moreover, incorporating CoT data during training (w/ CoT) consistently outperforms the traditional translation setting (w/o CoT), underscoring the necessity of explicitly decomposing and recombining structural information prior to generating the final answer; (3) It is worth noting that our reasoning model shows increasingly pronounced benefits as the evaluation becomes more fine-grained. For example, although the traditional translation approach (w/o CoT) can also generate a comparable proportion of valid molecules, when molecular similarity, and especially the degree of exact matching, is considered, the superiority of reasoning models (w/ CoT) becomes more apparent. This demonstrates that reasoning ability plays a crucial role in fine-grained molecular generation, consistent with the chemical intuition that component-wise decomposition and recombination enable more precise generation. More results on LLaMA and Mistral families can be found in Appendix M, which show consistent trends.

We further investigate the effectiveness of incorporating both SFT and RL for molecular generation, i.e., the two-stage post-training strategy, see Section 4. Experiments are also conducted on the MolCoT-4K dataset using a series of ablation strategies: (1) SFT using only pairwise data (m, s) and discarding CoT, denoted as $\textbf{SFT}_{\textbf{w/o CoT}}$, following the traditional translation paradigm (Edwards et al., 2021; Yu et al., 2024); (2) SFT with CoT data, i.e., (m, c, s), denoted as $\textbf{SFT}_{\textbf{w/ CoT}}$; (3) We additionally incorporate RL training stage after (1) and (2) to show further impacts, denoted as $\textbf{RL+SFT}_{\textbf{w/o CoT}}$ and $\textbf{RL+SFT}_{\textbf{w/ CoT}}$, respectively.

Model	**EM**	**RDK-FTS**	**Validity**
Qwen2.5-7B	4.92%	16.22%	56.00%
+ $\text{SFT}_{\text{w/o CoT}}$	28.31%	52.53%	88.57%
+ $\text{SFT}_{\text{w/ CoT}}$	38.55%	58.77%	90.06%
+ RL + $\text{SFT}_{\text{w/o CoT}}$	20.92%	37.75%	73.85%
+ RL + $\text{SFT}_{\text{w/ CoT}}$	**43.98%**	**61.63%**	**91.27%**
$\Delta_{\textbf{Improvement}}$	+39.06%	+45.41%	+35.27%

Table 1: Performance comparison under different training strategies, measured by Exact Match (EM), RDK-FTS, and validity scores, respectively.

Table 1 shows the results for these settings on the Qwen2.5-7B model, measured by EM, RDK-FTS, and Validity scores. The results clearly indicate that the base model (Qwen2.5-7B) finds this task highly challenging, achieving only 4.92% EM, 16.22% RDK-FTS, and 56.00% Validity scores. Supervised fine-tuning with pairwise data ($\text{SFT}_{\text{w/o CoT}}$), i.e., translation modeling, substantially improves performance, particularly by increasing the validity of generated SMILES strings and enhancing structural similarity. When incorporating CoT data ($\text{SFT}_{\text{w/ CoT}}$) for training, we observe clear reasoning trails during generation. It is worth noting that this strongly enhances performance by increasing exact matching results from 28.31% to 38.55%, highlighting a consistent finding with that in Figure 2 on the role in fine-grained generation. Further incorporating RL for training achieves the best results with 43.98% EM, 61.63% RDK-FTS, and 91.27% Validity scores. However, discarding CoT data ($\text{RL+SFT}_{\text{w/o CoT}}$) in this setting largely hurts performance, even worse than that of solely supervised fine-tuning ($\text{SFT}_{\text{w/o CoT}}$). The labels of the test dataset after SFT and RL are shown in Appendix I. Examples of results after RL are shown in Appendix J.

6.2 On the Role of Correctness of Reasoning Trace

Model	**EM**	**RDK-FTS**	**Validity**
Qwen2.5-7B	4.92%	16.22%	56.00%
$\text{SFT}_{\text{w/ corrected CoT}}$	**20.48%**	**43.22%**	**71.45%**
$\text{SFT}_{\text{w/ minor issue}}$	17.77%	42.96%	69.58%
$\text{SFT}_{\text{w/ element issue}}$	16.27%	41.29%	64.52%
$\text{SFT}_{\text{w/ structural issue}}$	15.36%	40.13%	64.34%
$\text{SFT}_{\text{w/ ineffective CoT}}$	7.53%	30.26%	58.73%
$\text{SFT}_{\text{w/ messed-up CoT}}$	0.60%	4.95%	20.19%

Table 2: The effectiveness of six groups of data varying noise type within CoT across EM, RDK-FTS, and Validity. Corrected CoT achieves the best performance as expected. Notably, the performance gradually decreases as the noise increases.

To more precisely examine how the correctness of the reasoning trace influences training in molecular generation, we conduct controlled experiments. From our CoT dataset, MolCoT-4K, where each sample is expressed as (m, c, s), we randomly select 1,000 samples and manually inject five types of noise into c. These noise types span mild to severe perturbations: CoT with minor issues, element issues, structural issues, ineffective CoT (i.e., removal of CoT), and a "messed-up" CoT created by shuffling the words within each reasoning path to maximize disruption (see Appendix G for details). This

Model	EM	RDK FTS	Validity
Gemini-2.5-pro	19.49%	62.72%	81.18%
Claude 3 Opus	12.30%	57.60%	92.60%
GPT-4	6.40%	42.60%	81.40%
Gemini-2.0-flash	5.29%	27.44%	64.38%
GPT-4o-mini	1.40%	12.63%	52.27%
MolT5	31.70%	**73.20%**	95.30%
LlaSMol	19.20%	61.70%	**99.70%**
MolInst	6.00%	43.60%	84.80%
ChemLLM	0.90%	14.30%	4.30%
Qwen2.5-7B	0.00%	1.08%	8.95%
MolGeneration$_{\text{w/o CoT}}$	37.02%	61.63%	89.81%
MolGeneration$_{\text{w/ CoT}}$	**44.97%**	67.69%	94.02%

Table 3: Performance comparison on the SMolInstruct dataset. Our model achieves the highest EM score of 44.97% using beam search (beam size = 5), outperforming both open-source systems (e.g., MolT5: 31.70%, LlaSMol: 19.20%) and commercial LLMs, e.g., Gemini-2.5-Pro.

process yields six datasets in total, including the clean subset from MolCoT-4K. All six datasets share the same 1,000 molecular descriptions m and corresponding SMILES s, differing only in their CoT c, thereby ensuring effective control across groups.

We further perform supervised fine-tuning on the Qwen2.5-7B model with each of these datasets. The results are presented in Table 2. Training with the dataset containing correct CoT yields the strongest performance, while increasing perturbation levels lead to a gradual decline. In particular, training with the "messed-up CoT" setting results in nearly collapsed performance. Using the ineffective CoT data (empty CoT) provides only marginal improvement over the base model, but still falls far short of the correct CoT setting, consistent with the observations in Section 6.1. Overall, the correctness of CoT is crucial to successfully elicit reasoning ability for molecular generation.

6.3 Experimental Results on SMolInstruct

In this section, we evaluate the effectiveness of incorporating reasoning ability on the widely used molecular generation benchmark, SMolInstruct. This dataset includes $56{,}498$ description–SMILES pairs in the training set, which has been largely used for translation modeling (Yang et al., 2024). The test set comprises $2{,}493$ samples. Notably, even advanced commercial LLMs struggle on this task; for instance, GPT-4 achieves only a 6.4% EM score, as reported by Yu et al. (2024), posing a challenging benchmark.

As discussed in Section 6.1, traditional translation modeling struggles to capture fine-grained structural information, yet it can still generate SMILES with relatively high coarse-grained similarity and validity. To maximize generation performance, we combine the SMolInstruct training set (56K pairwise samples) with our MolCoT-4K dataset (4K samples from the merged train and test sets). We simply upsample MolCoT-4K by a factor of 10 to balance their scales.

To highlight the role of CoT data, we follow the setting in Figure 2 and apply supervised fine-tuning on the Qwen2.5-7B model using the merged dataset with and without the CoT path, yielding our **MolGeneration** models, denoted as MolGeneration$_{\text{w/ CoT}}$ and MolGeneration$_{\text{w/o CoT}}$. We conduct an extensive comparison with both open-source chemistry-specific systems and commercial LLMs. To ensure a fair comparison, we use beam search with a beam size of 5 as the decoding strategy, which is also applied on the baseline open-source systems, e.g., MolT5 and LlaSMol.

As shown in Table 3, our models achieve state-of-the-art performance in exact match, reaching 44.97% and substantially outperforming all baseline systems. Also, the small scale of the CoT dataset contributes substantially, where training with CoT (MolGeneration$_{\text{w/ CoT}}$) clearly surpasses its translation-based version (MolGeneration$_{\text{w/o CoT}}$). Other recent strong translation-based systems, such as MolT5 and LlaSMol, perform considerably worse in EM (31.7% and 19.2%), although they attain the highest scores in structural similarity (73.2%) and validity (99.7%), respectively.

This is consistent with our findings in Section 6.1, where reasoning primarily benefits fine-grained generation, while translation modeling mainly contributes at the coarse level.

Notably, Qwen2.5-7B, the base model we trained on, fails completely on this challenging dataset, producing no exact matching SMILES. Also, it is interesting that GPT-4o-mini, the source from which our MolCoT-4K data was distilled, achieves only 1.4% in this benchmark. This underscores both the difficulty of the benchmark and the critical role of training data.

In addition, to explore the potential of MolGeneration for molecular design, we apply Best-of-N sampling to generate a set of N (N = 1,2,4,8,16) candidate molecules from MolGeneration. Assuming access to an external verifier, chemical experts can then select results from this candidate set. The Pass@N results are presented in Figure 3, where we observe a nearly linear trend as increasing sample size N.

MolGeneration exhibits steady improvements from Pass@1 to Pass@16 and achieves a Pass@16 exact match score of 48.46%. Moreover, it consistently and largely outperforms commercial LLMs such as Gemini-2.0-flash and GPT-4o-mini in this setting, while Qwen2.5-7B yields nearly 0.0% EM across all values of N. In short, MolGeneration demonstrates strong potential for future molecular design in real-world experimental settings.

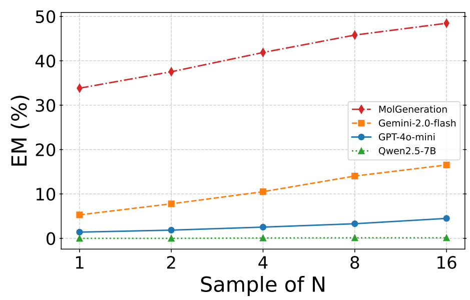


Figure 3: Performance comparison of Exact Match (EM) on CoT-augmented SMolInstruct across Pass@N (N = 1,2,4,8,16). MolGeneration exhibits steady improvements from Pass@1 to Pass@16, consistently outperforming commercial LLMs such as Gemini-2.0-flash and achieves a Pass@16 exact match score of 48.46%.

7 Conclusion

In this work, we frame molecule generation as a step-by-step reasoning process in which LLMs produce chemically valid CoT paths that decompose and recombine functional groups before generating the corresponding SMILES. To support this ability, we construct the first high-quality CoT dataset, **MolCoT-4K**, incorporating both machine and expert verification to maximize data reliability. Our extensive experiments on a wide range of open-source LLMs demonstrate that (1) despite its small scale, MolCoT-4K can effectively elicit reasoning ability for this purpose; (2) the resulting reasoning models consistently and largely outperform traditional translation-based models when trained on the same amount of data; and (3) while translation-based models can produce SMILES with relatively high structural similarity, reasoning ability proves superior in fine-grained molecular generation—reflected in substantially higher exact-match accuracy—thereby supporting the chemical intuition that component-wise decomposition and recombination enable more precise generation.

Lastly, we present our molecular reasoning model, **MolGeneration**, trained on a large amount of pairwise data and our compact CoT dataset, MolCoT-4K. MolGeneration achieves the state-of-the-art generation performance, largely surpassing both strong specific systems, e.g., MolT5, and commercial LLMs, like Gemini-2.5-pro, demonstrating strong potential for future molecular design.

REFERENCES

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.

Adrià Cereto-Massagué, María José Ojeda, Cristina Valls, Miquel Mulero, Santiago Garcia-Vallvé, and Gerard Pujadas. Molecular fingerprint similarity search in virtual screening. *Methods*, 71: 58–63, 2015.

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.

Carl Edwards, ChengXiang Zhai, and Heng Ji. Text2Mol: Cross-modal molecule retrieval with natural language queries. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 595–607, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.47. URL https://aclanthology.org/2021.emnlp-main.47/.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 375–413, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.26. URL https://aclanthology.org/2022.emnlp-main.26/.

Carl Edwards, Qingyun Wang, Lawrence Zhao, and Heng Ji. L+M-24: Building a dataset for Language+Molecules @ ACL 2024. In Carl Edwards, Qingyun Wang, Manling Li, Lawrence Zhao, Tom Hope, and Heng Ji (eds.), *Proceedings of the 1st Workshop on Language + Molecules (L+M 2024)*, pp. 1–9, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.langmol-1.1. URL https://aclanthology.org/2024.langmol-1.1/.

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. Mol-instructions: A large-scale biomolecular instruction dataset for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Tlsdsb6l9n.

Anna Gaulton, Louisa J Bellis, A Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, et al. Chembl: a large-scale bioactivity database for drug discovery. *Nucleic acids research*, 40(D1):D1100–D1107, 2012.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

John J Irwin and Brian K Shoichet. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Yinuo Jiang, Xiang Zhuang, Keyan Ding, Qiang Zhang, and Huajun Chen. Enhancing cross text-molecule learning by self-augmentation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9551–9565, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.569. URL https://aclanthology.org/2024.findings-acl.569/.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem in 2021: new data content and improved web interfaces. *Nucleic acids research*, 49(D1):D1388–D1395, 2021.

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.

Greg Landrum, Paolo Tosco, Brian Kelley, Ricardo Rodriguez, David Cosgrove, Riccardo Vianello, sriniker, Peter Gedeck, Gareth Jones, NadineSchneider, Eisuke Kawashima, Dan Nealschneider, Andrew Dalke, Matt Swain, Brian Cole, Samo Turk, Aleksandr Savelev, tadhurst cdd, Alain Vaucher, Maciej Wójcikowski, Ichiru Take, Vincent F. Scalfani, Rachel Walker, Kazuya Ujihara, Daniel Probst, Juuso Lehtivarjo, Hussein Faara, guillaume godin, Axel Pahl, and Jeremy Monat. rdkit/rdkit: 2024_09_5 (q3 2024) release, January 2025. URL `https://doi.org/10.5281/zenodo.14779836`.

Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Andreas Mayr, Günter Klambauer, Thomas Unterthiner, Marvin Steijaert, Jörg K Wegner, Hugo Ceulemans, Djork-Arné Clevert, and Sepp Hochreiter. Large-scale comparison of machine learning methods for drug target prediction on chembl. *Chemical science*, 9(24):5441–5451, 2018.

Noel M O'Boyle and Roger A Sayle. Comparing structural fingerprints using a literature-based similarity benchmark. *Journal of cheminformatics*, 8(1):36, 2016.

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. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 1102–1123, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.70. URL `https://aclanthology.org/2023.emnlp-main.70/`.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Xiaozhuan Liang, Yin Fang, Jinhua Zhu, Shufang Xie, Tao Qin, and Rui Yan. BioT5+: Towards generalized biological understanding with IUPAC integration and multi-task tuning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1216–1240, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.71. URL `https://aclanthology.org/2024.findings-acl.71/`.

Qwen. Qwen2.5 technical report, 2025a. URL `https://arxiv.org/abs/2412.15115`.

Qwen. Qwen3 technical report, 2025b. URL `https://arxiv.org/abs/2505.09388`.

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.

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.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL `https://arxiv.org/abs/2402.03300`.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl, 2020.

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.

David Weininger, Arthur Weininger, and Joseph L Weininger. Smiles. 2. algorithm for generation of unique smiles notation. *Journal of chemical information and computer sciences*, 29(2):97–101, 1989.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Botao Yu, Frazier N. Baker, Ziqi Chen, Xia Ning, and Huan Sun. LlaSMol: Advancing large language models for chemistry with a large-scale, comprehensive, high-quality instruction tuning dataset. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=lY6XTF9tPv.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

User: You are an expert in info-chemistry, especially in SMILES interpretation and translation. Provide the SMILES representation based on the following molecule description: {Molecule Description}. Analyze it step by step, and return only the SMILES within the <Answer></Answer> tags.

Figure 4: Distillation prompt. Distill Chain-of-Thought (CoT) for molecule generation based on the above prompt from GPT-4o. The placeholder *{Molecule Description}* is filled with a molecule description.

Provider	Model	Input ($/1M tok)	Output ($/1M tok)
OpenAI	GPT-4o-mini	0.15	0.60
OpenAI	GPT-4o	2.50	10.00
OpenAI	o3	10.00	40.00
Anthropic	Claude Sonnet 3.7	3.00	15.00
Google	Gemini 2.5 Pro (Thinking)	1.25 ($\leq 200k$) / 2.50 ($\geq 200k$)	10.00 ($\leq 200k$) / 15.00 ($\geq 200k$)
Google	Gemini 2.5 Flash (Hybrid reasoning)	0.30	2.50
DeepSeek	DeepSeek-R1	0.14	2.19

Table 4: Commercial model API pricing ($/million tokens). GPT-4o-mini is one of the strongest-performing and most cost-effective models.

A THE USE OF LARGE LANGUAGE MODELS

In this paper, we prompt GPT-4o-mini to generate draft CoT traces from molecule descriptions prior to human post-editing to enable relatively large-scale data curation. In addition, we also evaluate Claude, GPT, and Gemini on the MolCoT-4K and SMolInstruct test sets for comparison.

B DISTILLATION PROMPT

The prompt used to extract Chain-of-Thought (CoT) data from GPT-4o-mini is shown in Figure 4. The {Molecule Description} is replaced by actual data samples.

C COMMERCIAL REASONING MODEL API PRICING

We select GPT-4o-mini because GPT-4o-mini is one of the strongest-performing and most cost-effective models.

D EXAMPLE OF DATASET

Table 5 presents an example from the constructed dataset, which includes a molecule description of Glyphosate, its corresponding SMILES representation, and the distilled Chain-of-Thought (CoT) reasoning steps. The CoT first identifies the core structure of Glyphosate based on the description, then identifies components of this molecule, including glycine, the phosphonomethyl group, and N-alkylation, and identifies functional groups. Finally, it constructs the SMILES representation based on the above decomposition.

E CURATION TOOL

The curation is shown in Figure 5. The original CoT data is presented in the left CoT text box and can be edited. At the same time, we can label the data in the score text box and fill in the detailed issue in the reason text box. In this way, we manually correct and label the data. After editing, we can save the revision by clicking the Save changes button. The two text boxes above, which cannot be edited, are used to show the SMILES and molecule description, respectively. After saving, the curated result can be seen on the right side, including the SMILES, molecule description, corrected

Example Data: Glyphosate ($C_3H_8NO_5P$)

Molecule Description: The molecule is a phosphonic acid resulting from the formal oxidative coupling of the methyl group of methylphosphonic acid with the amino group of glycine. It is one of the most commonly used herbicides worldwide, and the only one to target the enzyme 5-enolpyruvyl-3-shikimate phosphate synthase (EPSPS). It has a role as an agrochemical, an EC 2.5.1.19 (3-phosphoshikimate 1-carboxyvinyltransferase) inhibitor and a herbicide. It is a phosphonic acid and a glycine derivative. It is a conjugate acid of a glyphosate(2-) and a glyphosate(1-).

CoT:

To generate the SMILES representation for the molecule based on the provided description, let's break down the components step by step.

1. ∗∗Identify the core structure∗∗: The molecule in question is glyphosate, which is a derivative of glycine and also contains a ∗∗phosphonic acid∗∗ group.

2. ∗∗Components∗∗:
 – ∗∗Glycine∗∗: The simplest amino acid with the structure NH_2-CH_2-COOH.
 – ∗∗Phosphonomethyl group∗∗: The substituent -CH_2-P(=O)$(OH)_2$ (a phosphonic acid attached via a methylene).
 – ∗∗N–substitution (N–alkylation)∗∗: The -CH_2-P(=O)$(OH)_2$ group is bonded to the ∗∗amino nitrogen of glycine∗∗, forming N–(phosphonomethyl)glycine.

3. ∗∗Determine functional groups∗∗:
 – Glyphosate includes a ∗∗carboxylic acid∗∗ (-COOH), a ∗∗phosphonic acid∗∗ (-P(=O)$(OH)_2$), and a ∗∗ secondary amine∗∗ (-NH-).
 – The overall structure has the amine nitrogen linked via a methylene (-CH_2-) to the phosphonic acid group.

4. ∗∗Construct the SMILES∗∗: The backbone HOOC-CH_2-NH-CH_2-P(=O)$(OH)_2$ translates to the following SMILES representation considerations:
 – A carbon chain connecting the amine and the phosphonic acid,
 – A phosphonic acid group,
 – An amino group on the glycine backbone.

5. ∗∗Final SMILES representation∗∗:
 – The SMILES string for glyphosate (∗∗N–(phosphonomethyl)glycine∗∗) is:
<Answer>O=C(O)CNCP(=O)(O)O</Answer>

SMILES: O=C(O)CNCP(=O)(O)O

Table 5: An example of the dataset is Glyphosate ($C_3H_8NO_5P$), which consists of a molecule description, reasoning steps (CoT), and the corresponding SMILES representation. The generated SMILES is enclosed within `<Answer>` tags. The CoT first identifies the core structure of Glyphosate based on the description, then identifies components of this molecule, including glycine, the phosphonomethyl group, and N-alkylation, and identifies functional groups. Finally, it constructs the SMILES representation based on the above decomposition.

CoT, label, and issue. In addition, the modification record can be exported and viewed by clicking the view export button. The curation tool will be released to the community [4].

F Labeling Criteria

To identify the different types of CoT, we label the data according to the following criteria:

- **1 Ineffective CoT:** There is an ineffective reasoning path in the CoT, including repeated descriptions, empty content in thinking tags, and simple repetition of properties from the description. For example, the CoT for reasoning CC(=O)O[Sn](C_1=CC=CC=C_1)(C_1=CC=CC=C_1)C_1=CC=CC=C_1 just repeats its description: "The molecule is an organotin compound that is the O-acetyl derivative of triphenyltin hydroxide. A fungicide used to control blights on potatoes, leaf spot diseases on

[4] urlhttps://anonymous.4open.science/r/MolGeneration-DE3F

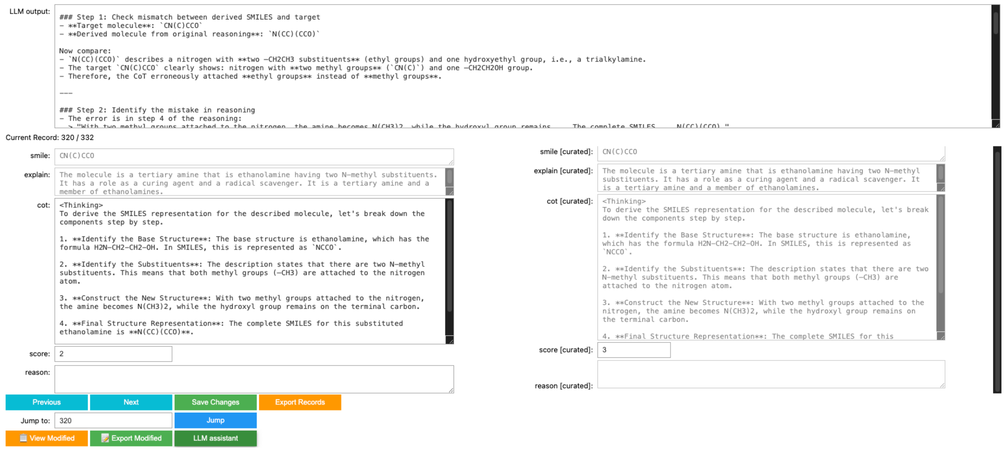


Figure 5: The user interface of the curation tool. The original CoT is presented in the left text box and can be edited. The corrected CoT is presented in the right text box. The corresponding output of assistant LLMs is presented in the top box. These buttons are used to navigate through the data, save, and export records.

sugar beet and anthracnose on beans. It has a role as an antifungal agrochemical. It is an organotin compound and an acetate ester. It derives from a fentin hydroxide.".

- **2 Main-structure issue:** The CoT contains issues with the main structure, such as mistakenly identifying a linear structure as a branched or ring structure, or vice versa. For example, O=C(/C=C/C_1=CC=CC=C_1)OCC_1=CC=CC=C_1 contains ring structures (C_1CCCCC_1), whereas the reasoning path identifies it as linear structures (CCCCCC).
- **3 Element issue:** The CoT contains issues with element counts, such as reasoning a twelve-carbon linear chain as a ten-carbon linear chain.
- **4 Minor issue:** The CoT contains issues that may not affect the main structure, including missing chiral bonds and bonds lacking between two atoms. For example, it reasons C(=O)O (the carboxyl / carboxylic acid group, –COOH) as OCO, which lacks a bond between carbon and oxygen. Another issue is, for example, reasoning a Kekulé format SMILES into an aromatic format.
- **5 Correct CoT:** The CoT does not need any revision. An example can be seen in Table 5.

G MolCoT-1K Construction

To verify the role of correctness in the reasoning trace, we construct a dataset named **MolCoT-1K**.

We randomly select 1,000 samples from the MolCoT-4K training set and inject five types of CoT issues into c^*, respectively, resulting in $c^{*'}$. The five types of CoT issues are: CoT with minor issues, CoT with element issues, CoT with structural issues, ineffective CoT, and messed-up CoT. Specifically, for ineffective CoT, we remove the reasoning content to provide the answer directly. To inject main-structure issues, we modify linear chains into rings or branches, and vice versa. The modified structures are then used to alter the corresponding content in the CoT. For example, we replace *"linear carbon chain with six carbons, CCCCCC* with *"Benzene, a ring of six carbons, C_1=CC=CC=C_1"*. To inject element issues, we randomly add or remove carbon atoms, while for minor issues, we add or remove bonds between two atoms. In addition, to verify the effectiveness of the reasoning path, we construct a messed-up CoT dataset by reshuffling the words within each sample to disrupt the reasoning path. This dataset contains the same words as the corrected CoT data but differs in the reasoning path.

After this construction process, we obtain six control-group datasets: corrected CoT, CoT with minor issues, CoT with element issues, CoT with structural issues, ineffective CoT, and messed-up CoT, each containing 1,000 samples. Since these datasets are randomly selected from the MolCoT-4K

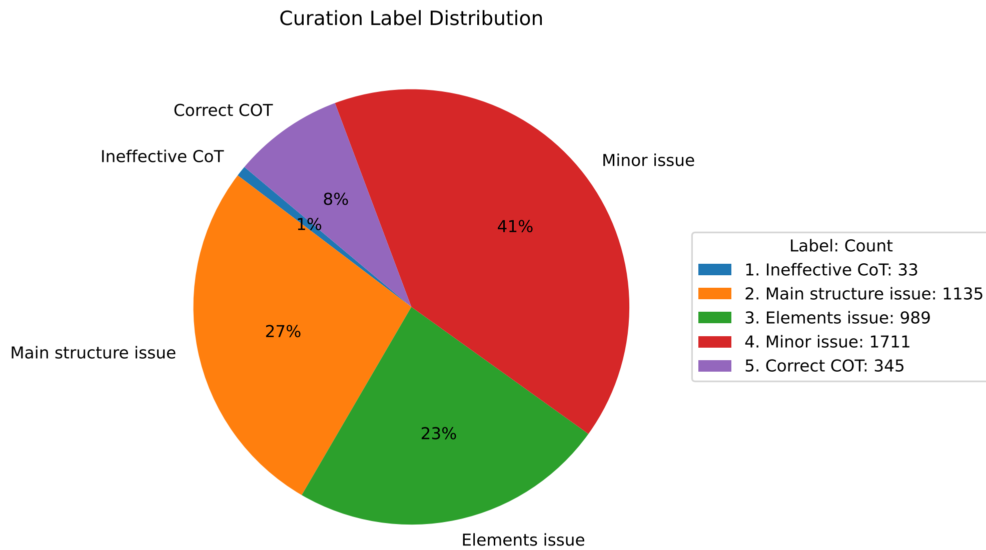


Figure 6: The label distribution of the distilled CoT data.Only 8% is correct CoT data, while 92% of the distilled CoT contains issues, with 1% not even generating valid CoT.

training set, we continue to use the MolCoT-4K test and validation sets as the MolCoT-1K test and validation sets.

H LABEL DISTRIBUTION OF COT DATA

The distribution of the CoT label can be seen in Figure 6. The results reveal that the ineffective CoT data rate is the lowest, which means that most of the distilled data is effective. The following rate is correct CoT data, which only 8%, which shows the importance of correcting distilled CoT data. Most of the data (41%) contain minor issues, such as the number of bonds between the carbon chain.

I LABEL DISTRIBUTION OF TEST DATA AFTER SUPERVISED FINE-TUNING AND REINFORCEMENT LEARNING

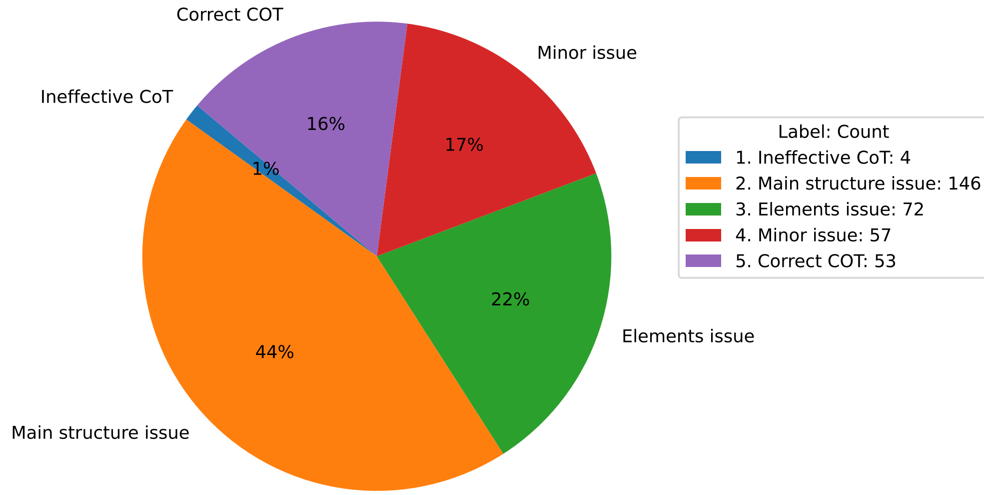
The results are presented in Figure 7. After SFT, the model can generate valid CoT with an accuracy of 99%, and the corrected CoT rate reaches 16%. The mean length of the CoT is $1,099$. After RL, the model can generate valid CoTs with nearly 100% accuracy, and the corrected CoT rate further increases to 18%. The mean length of the CoT is $1,171$. After RL, the quality of the chain-of-thought (CoT) reasoning, in terms of correctness, is also enhanced. Examples of results after RL are presented in Appendix J.

J EXAMPLES OF RESULTS AFTER REINFORCEMENT LEARNING

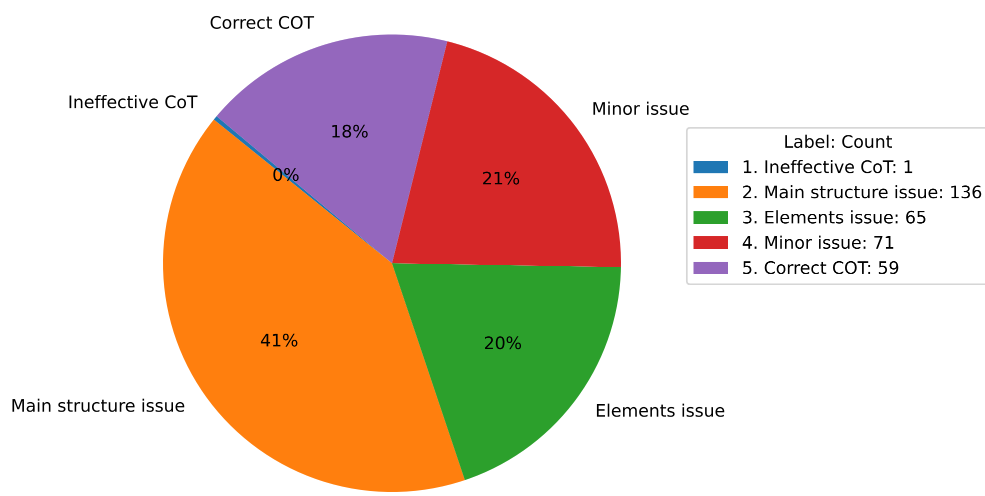
Examples are presented in Table 6. After reinforcement learning (RL), the model is able to generate reasoning steps and produce the corresponding SMILES based on the description: it identifies the core structure and substitutes, determines their positions, assembles the structure, and summarizes before providing the final answer.

K SUPERVISED FINE-TUNING PROMPT

The prompt used for Supervised Fine-Tuning is shown in Figure 8. The {Molecule Description}, {molecule generation CoT} and {ground truth SMILES} are replaced by actual data samples.



(a) Label distributions on the test set after SFT.



(b) Label distributions on the test set after SFT and RL.

Figure 7: Test set label distributions. After SFT, the model can generate valid CoT with an accuracy of 99%, with 16% correct CoT. After RL, the model can generate valid CoT with nearly 100% accuracy, with 18% correct CoT. After RL, the quality of the chain-of-thought (CoT) reasoning, in terms of correctness, is also enhanced.

L Reinforcement Learning Prompt

The prompt used for reinforcement learning is shown in Figure 9. The {Molecule Description} is replaced by actual data samples.

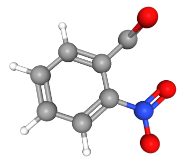
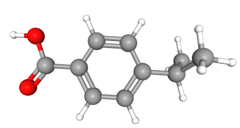
Molecule Structure&Name	Description	Chain of Thought
 2-Nitrobenzaldehyde	The molecule is benzaldehyde substituted at the ortho-position with a nitro group. It is a C-nitro compound and a member of benzaldehydes.	<Thinking> To create the SMILES representation for the described molecule, we'll follow these steps: 1. **Identify the base structure**: The base structure is benzaldehyde, which consists of a benzene ring (C6H5) with an aldehyde group (-CHO) attached. The SMILES for benzaldehyde is c1ccccc1C=O (aromatic) or its Kekulé equivalent C1=CC=CC=C1C=O. 2. **Identify the substituents**: The molecule is substituted at the ortho position with a nitro group (-NO2). The nitro group must be written correctly in SMILES as [N+](=O)[O-]. 3. **Determine the positions**: The ortho position refers to the two carbon atoms adjacent to the carbonyl carbon of the aldehyde group. In benzaldehyde, these are the ortho positions relative to the aldehyde group. 4. **Combine the components**: Attach the nitro group at the ortho position relative to the aldehyde substituent on the benzene ring. Now that we have a clear understanding of the structure, we can convert this into its SMILES notation: </Thinking> <Answer>C1=CC=C(C(=C1)C=O)[N+](=O)[O-]</Answer>
 4-Isopropylbenzoic Acid	The molecule is a cumic acid that consists of benzoic acid substituted by an isopropyl group at position 4. It has a role as a plant metabolite. It is a conjugate acid of a p-cumate.	<Thinking> To derive the SMILES representation for the described molecule, we can break down the components mentioned in the molecular description: 1. **Benzoic Acid**: This is a benzene ring (C6H5) with a carboxylic acid (COOH) group attached. The SMILES for benzoic acid is `c1ccccc1C(=O)O` or Kekulé form `C1=CC=CC=C1C(=O)O`. 2. **Substituted by an Isopropyl Group at Position 4**: An isopropyl group (-CH(CH$_3$)$_2$) is a branched alkyl group. In SMILES, the isopropyl group can be represented as CC(C). 3. **Combining the Components**: Since the isopropyl group is substituted at the -4 position of the benzoic acid, the -4 position refers to the carbon opposite the carboxylic acid on the benzene ring (para position). Thus, the isopropyl group is attached to the carbon para to the -COOH. Now that we have a clear understanding of the structure, we can convert this into its SMILES notation: </Thinking> <Answer>CC(C)C1=CC=C(C=C1)C(=O)O</Answer>

Table 6: Examples of the results. The model generates reasoning steps and finally gives the corresponding SMILES based on the description. After RL, the model identifies the core structure and substitutes, determines their positions, assembles the structure, and summarizes before providing the final answer.

M Generalization Across LLaMA and Mistral

Table 7 summarizes performance for LLaMA-3.1-8B, Mistral-7B-v0.1, and Mistral-7B-v0.3.

User: You are an expert in info-chemistry, especially in SMILES interpretation and translation. Provide the SMILES representation based on the following molecule description: {Molecule Description}. Analyze it step by step within the <Thinking></Thinking> tags and give only the SMILES within the <Answer></Answer> tags.

Assistant:
```
<Thinking>
{molecule generation CoT}
</Thinking>
<Answer>
{ground truth SMILES}
</Answer>
```

Figure 8: Supervised Fine-Tuning Prompt

User: You are an expert in info-chemistry, especially in SMILES interpretation and translation. Provide the SMILES representation based on the following molecule description: {Molecule Description}. Analyze it step by step within the <Thinking></Thinking> tags and give only the SMILES within the <Answer></Answer> tags.

Figure 9: Reinforcement Learning Prompt

Model	**EM**	**RDK FTS**	**Validity**
LlaMA-8B	0.00%	0.39%	5.72%
LlaMA-8B$_{\text{w/o CoT}}$	24.40%	45.98%	83.43%
LlaMA-8B$_{\text{w/ CoT}}$	33.74%	53.52%	87.65%
Mistral-7B-v0.1	0.00%	0.00%	0.00%
Mistral-7B-v0.1$_{\text{w/o CoT}}$	1.51%	9.43%	35.24%
Mistral-7B-v0.1$_{\text{w/ CoT}}$	4.21%	26.15%	83.43%
Mistral-7B-v0.3	1.50%	13.32%	52.71%
Mistral-7B-v0.3$_{\text{w/o CoT}}$	25.30%	44.72%	75.30%
Mistral-7B-v0.3$_{\text{w/ CoT}}$	34.34%	51.97%	87.05%

Table 7: Performance comparison across different model families (Qwen2.5 and Qwen3 in Figure 2) and scales, evaluated on the MolCoT-4K test set. Results are reported for three settings: raw base models without fine-tuning, models fine-tuned without CoT data, and models fine-tuned with CoT data. The comparison demonstrates that CoT consistently improves performance across architectures and scales.