ChemMLLM: Chemical Multimodal Large Language Model

Anonymous ACL submission

Abstract

Multimodal large language models (MLLMs) have made impressive progress in many applications in recent years. However, chemical MLLMs that can handle cross-modal understanding and generation remain underexplored. To fill this gap, in this paper, we propose ChemMLLM, a unified chemical multimodal large language model for molecule understanding and generation. Also, we design five multimodal tasks across text, molecular SMILES strings, and image, and curate the datasets. We benchmark ChemMLLM against a range of general leading MLLMs and Chemical LLMs on these tasks. Experimental results show that ChemMLLM achieves superior performance across all evaluated tasks. For example, in molecule image optimization task, ChemMLLM outperforms the best baseline (GPT-4o) by 118.9% (4.27 vs 1.95 property improvement). The code is publicly available at https://anonymous.4open.science/r/ChemMLLM-0D98/.

1 Introduction

Multimodal large language models (MLLMs) have shown strong abilities in understanding and generating content across text, images, and audio (OpenAI, 2024; Sun et al., 2024; Team, 2024; Liu et al., 2024; Zhou et al., 2024; Xie et al., 2024; Wang et al., 2024), enabling more natural human–AI interaction. Chemistry is inherently multimodal, involving textual descriptions, structured formats like SMILES (Weininger, 1988)[1], and molecular images. Recent works have demonstrated initial success in adapting MLLMs for chemical applications such as property prediction and reaction analysis (Cao et al., 2023; Zhang et al., 2024b; Luo et al., 2024; Li et al., 2025). However, these models largely focus on understanding tasks and treat images only as input. In contrast, molecular visuals are central to how chemists communicate and reason. Enabling image generation from chemical language or structure would greatly expand the expressiveness of MLLMs in chemistry (Kosenkov and Kosenkov, 2021). Yet, an integrated Chemical MLLM that supports both multimodal understanding and generation for chemistry remains lacking.

The challenges to build such a model are as follows: (1) Vision-based chemistry tasks and datasets remain underexplored. (2) Specificity of molecule image. Unlike natural images, molecule images are sparse, containing large areas of empty space, and are composed strictly of straight lines. General MLLMs result in unclear and distorted molecules (Figure 15). (3) Challenge for selecting an effective framework for chemical MLLM to seamlessly fuse diverse modalities, including discrete SMILES string/text, and continuous molecule images.

To address these issues, we propose ChemMLLM, a chemical multimodal large language model that understands and generates molecules in a unified framework. Specifically, to handle three challenges above, (1) we identify five multimodal chemistry tasks with three modalities (text, SMILES, image), which contain both generation and comprehension tasks; (2) we finetune molecule image-level Vector Quantized Generative Adversarial Network (VQGAN) to bridge the gap between molecule images and natural images; (3) we introduce the "Image Tokenizer-LLM-Image Detokenizer" architecture into multimodal chemical tasks to fuse different modalities in early stages and enable models to generate images directly. Also, we design a two-stage training strategy and prove its effectiveness empirically.

Contribution. For ease of exposition, we summarize our main contribution as:

[1] A SMILES (Simplified Molecular Input Line Entry System) string is a compact, text-based representation of a molecule's structure that encodes its atomic composition and connectivity in a linear format.

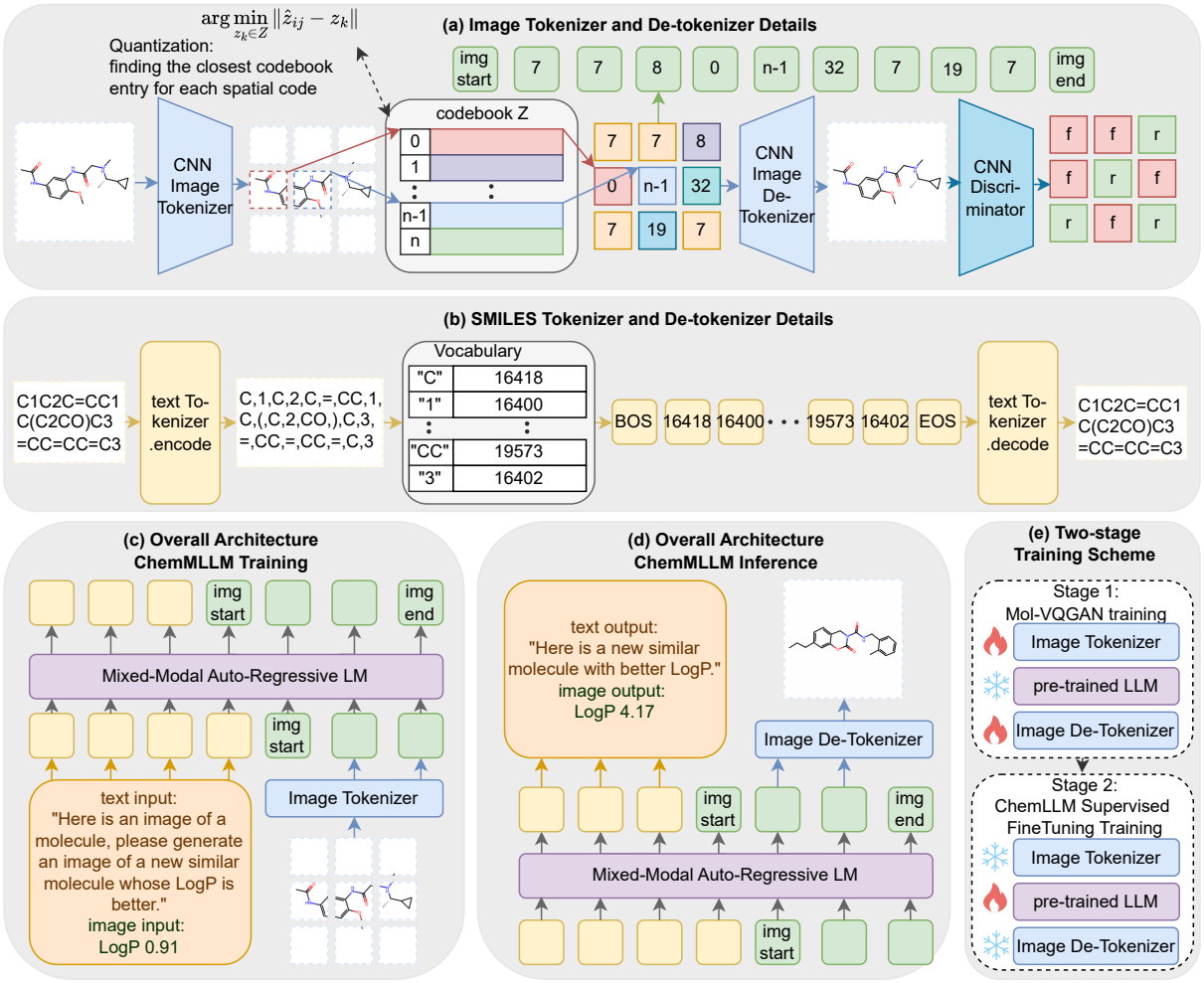


Figure 1: Overall architecture of ChemMLLM. (a) Image tokenizer and de-tokenizer. The image tokenizer employs CNN to extract spatial feature maps, where each n_z-dimensional spatial code is quantized into a discrete latent code via vector quantization (VQ). The resulting codebook indices serve as the final image tokens. Image de-tokenizer uses CNN to reconstruct image from discrete feature map. Then, a patch-based discriminator predicts whether the patch is fake (f) or real (r) (Section 3.1); (b) SMILES tokenizer and de-tokenizer. SMILES tokenization is consistent with text, and is mapped into a token sequence via text tokenizer; (c) ChemMLLM Training; (d) ChemMLLM Inference; (e) two-stage training paradigm for ChemMLLM (Section 3.3).

- **A chemical multimodal LLM understanding and generating molecule in a unified framework**. We propose and implement ChemMLLM, the first unified model to understand and generate molecules in text, SMILES, and image modality to the best of our knowledge (Section 3).
- **A multimodal chemical dataset suite.** We develop five new datasets to train and evaluate the chemical multimodal capability of MLLMs, which encompass a diverse spectrum of multimodal processing (Section 4).
- **A variety of reliable evaluation.** We benchmark the performance of different models on our proposed five tasks and ChemMLLM achieves dominating performance. For example, in molecule image optimization (image-to-image) tasks, ChemMLLM outperforms the best baseline (GPT-4o) by 118.9%, achieving a logP (optimizing property) increase of 4.27 compared to 1.95 (best baseline, GPT-4o) (Section 5).

2 Related Work

Multimodal LLM (MLLM) With the advancement of large language models and multimodal learning, numerous high-performing multimodal large language models (MLLMs) have recently emerged. Notable MLLMs include GPT-4o (OpenAI, 2024), which extends GPT-4V (OpenAI, 2023) to understand and generate across different modalities, including text, image, and audio. Emu-3 (Wang et al., 2024) unifies vision understanding and generation via discrete token model-

Model	BLEU-2 (↑)	BLEU-4 (↑)	ROUGE-1 (↑)	ROUGE-2 (↑)	ROUGE-L (↑)	METEOR (↑)
Qwen-VL-Chat	0.09 ±0.001	0.01 ±0.0009	0.32 ±0.003	0.07 ±0.001	0.22 ±0.002	0.19 ±0.001
InternVL-Chat-v1.5	0.04±0.0008	0.001 ±0.0002	0.38±0.003	0.07 ±0.001	0.26 ±0.002	0.19±0.001
LLaVA-v1.5-7B	0.08 ±0.001	0.004 ±0.0004	0.33 ±0.002	0.07 ±0.001	0.23 ±0.002	0.20 ±0.001
GPT-4o	0.16 ±0.003	0.07±0.002	0.28±0.005	0.13±0.003	0.23±0.004	0.22±0.004
ChemVLM-8B	0.15 ±0.004	0.08 ±0.003	0.28±0.006	0.13 ±0.003	0.23 ±0.005	0.23 ±0.005
ChemMLLM (ours)	**0.33** ±0.005*	**0.21**±0.005*	**0.50**±0.005*	**0.31**±0.006*	**0.43**±0.005*	**0.43**±0.005*

Table 1: Results on img2caption task (**best: bold**, second best: underlined, *: significantly better (statistically)).

ing. Chameleon (Team, 2024) aligns modalities at the token level for flexible multimodal generation, while LlamaGen (Sun et al., 2024) treats images as language-like sequences for scalable autoregressive image generation. Lumina-mGPT (Liu et al., 2024) trains a family of models to generate flexible photorealistic images from text descriptions based on Chameleon. Transfusion (Zhou et al., 2024) and Show-o (Xie et al., 2024) combine diffusion and transformer for multimodal understanding and generation. For vision understanding, well-known models include LLaVA (Liu et al., 2023), BLIP-2 (Li et al., 2023), Qwen-VL-Chat (Bai et al., 2023), InternVL-Chat (Chen et al., 2023), MiniGPT-4 (Zhu et al., 2023), Gemini (Team et al., 2023), Flamingo (Alayrac et al., 2022) and OpenFlamingo (Awadalla et al., 2023). Although current MLLMs perform well across modalities, these models struggle with chemical tasks due to a misalignment between general and domain-specific knowledge.

Chemical LLM MLLMs have also exhibited strong potential in addressing chemistry-related tasks, particularly in bridging the modality gap between textual descriptions and molecular representations. Concretely, Instruct-Mol (Cao et al., 2023) and MV-Mol (Luo et al., 2024) utilize LLaVA's architecture (Liu et al., 2023) and Q-former (Li et al., 2023) to align molecular structure and text modality, respectively. ChemLLM (Zhang et al., 2024a) uses a high-quality chemical dataset to fine-tune InternLM2 (Cai et al., 2024). ChemVLM (Li et al., 2025) extends ChemLLM (Zhang et al., 2024a) to understand images by adopting a projector-based method to align vision information and text information. UniMoT (Zhang et al., 2024b) uses a molecule tokenizer to align the graph modality molecule with text. Despite progress in chemical tasks, existing models can not generate molecular images, limiting their utility in more intuitive, visual forms of interaction. To fill this gap, We propose ChemMLLM, a unified framework that understands and generates molecules in text, SMILES, and image formats.

3 Method

Overview. Our framework uses an image tokenizer to transfer images into discrete tokens, aligning with texts and molecule's SMILES strings at the token level, known as "Image Tokenizer-LLM-Image De-tokenizer" architecture (Team, 2024). First, Section 3.1 discusses the molecule image tokenizer. Then, Section 3.2 describes how to combine images, texts and SMILES strings. Finally, the training strategy is discussed in Section 3.3. The whole pipeline is shown in Figure 1. For ease of understanding, we list key mathematical notations in Table 10 in Appendix.

3.1 Mol-VQGAN for Molecule Image Generation

Following Chameleon (Team, 2024), to enable multimodal information alignment in the early stage, images need to be discretized into token sequences similar to text. To proceed it, we train Vector Quantized Generative Adversarial Network (VQGAN) (Esser et al., 2021) on molecule images, known as Mol-VQGAN. Mol-VQGAN compresses images into a discrete latent space, which aligns well with the sequential nature of text modeling in large language models. Specifically, Mol-VQGAN uses Vector Quantized Variational Auto-Encoder (VQVAE) (Van Den Oord et al., 2017) as the generator and patch-based discriminator (Isola et al., 2017). VQVAE compresses images into discrete spaces and reconstructs images from that space. Mol-VQGAN adds a discriminator and perceptual loss to VQVAE to keep good perceptual quality and improve the performance. Formally, Mol-VQGAN takes an image $x \in \mathbb{R}^{H\times W\times 3}$ (H/W are the height/width of the input image), and transfers it into discrete representations $z_{\mathbf{q}}$ by encoding $\hat{z} \in \mathbb{R}^{h\times w\times n_z} = E(x)$ ($h/w/n_z$ are the height/width/channels of the feature map), and finding the closest codebook entry for each spatial code

Method	MW			LogP			TPSA			
	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	valid%(↑)
Qwen-VL-Chat	0.78 ±0.02	7073.3 ±858.2	58.0 ±1.8	0.14 ±0.02	4.4 ±0.46	1.4 ±0.04	0.19 ±0.02	>1e+4 ±563.6	82.8 ±1.9	99.3%
InternVL-Chat-v1.5	0.59 ±0.02	>1e+4 ±1595.5	83.6±3.6	0.04±0.05	9.2±0.62	2.3±0.08	0.29±0.03	2158.8±432.6	28.7 ±1.5	55.0%
LLaVA-v1.5-7B	0.36 ±0.08	>3e+4±5078.6	115.7±6.9	-0.003±0.05	5.1±1.0	1.6±0.08	0.01±0.04	>5e+4± $> 3e+4$	99.1±11.9	35.8%
ChemVLM-8B	0.84 ±0.02	9573.4 ±1992.9	56.9 ±4.5	0.38 ±0.05	4.9 ±0.58	1.6 ±0.08	0.26 ±0.06	5332.0 ±747.5	53.66 ±2.7	31.3%
ChemMLLM (ours)	**0.97**±0.004*	**789.7**±162.2*	**16.1**±0.72*	**0.92**±0.01*	**0.70**±0.14*	**0.52**±0.02*	**0.97**±0.005*	**152.6**±39.2*	**6.0**±0.33*	**99.6%**

Table 2: Results on img2property task: MW, LogP and TPSA (**best**, 2nd best, *: significantly better (statistically)).

Method	Hbd			Hba			Rb			QED		
	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)
Qwen-VL-Chat	-0.02 ±0.008	4.8 ±0.75	1.4±0.05	0.05±0.01	30.5±1.2	4.9±0.08	0.19±0.1	145.4±30.8	6.8±0.31	0±0.0	-	-
InternVL-Chat-v1.5	0.03±0.05	9.9±0.89	2.2±0.09	0.22±0.04	10.2±0.85	2.4±0.08	0.04±0.04	43.6±4.3	4.8±0.19	0.003±0.03	4.0±3.2	0.4±0.08
LLaVA-v1.5-7B	0.004 ±0.05	53.3±27.3	3.7±0.33	0.04±0.05	23.7±5.4	2.9±0.19	0.03 ±0.04	39.9±10.4	3.8±0.26	-0.11±0.08	>1e+5± $> 1e+5$	24.5±18.6
ChemVLM-8B	0.49 ±0.09	4.2 ±0.85	1.3 ±0.08	0.32 ±0.07	27.2 ±1.95	4.5 ±0.14	0.10 ±0.06	45.8 ±5.02	5.5 ±0.22	-0.003 ±0.06	0.24 ±0.02	0.37 ±0.01
ChemMLLM (ours)	**0.96**±0.004*	**0.18**±0.02*	**0.13**±0.01*	**0.94**±0.007*	**0.79**±0.12*	**0.44**±0.02*	**0.94**±0.01*	**1.6**±0.33*	**0.59**±0.03*	**0.91**±0.006*	**0.008**±0.0004*	**0.06**±0.002*

Table 3: Results on img2property task: Hbd, Hba, Rb, and QED (**best**, 2nd best, *: significantly better (statistically)).

$\hat{z}_{ij} \in \mathbb{R}^{n_z}$ (also known as vector quantization (VQ) process, denoted $\mathbf{q}(\cdot)$):

$$z_q = \mathbf{q}(\hat{z}) = \left(\arg\min_{z_k \in Z} \|\hat{z}_{ij} - z_k\| \right) \in \mathbb{R}^{h \times w \times n_z}, \tag{1}$$

where Z is codebook, a set of learnable vectors, and $z_k \in Z = \{z_i\}_{i=1}^{n} \subset \mathbb{R}^{n_z}$. Then, Mol-VQGAN reconstructs image $\hat{x} \in \mathbb{R}^{H \times W \times 3}$ from $z_{\mathbf{q}}$:

$$\hat{x} = G(z_q) = G(\mathbf{q}(E(x))). \tag{2}$$

The discretization and reconstruction process is illustrated in Figure 1(a).

The VQGAN's objective function is:

$$\begin{aligned} \min_{E,G,Z} \max_{D} \big[& \mathcal{L}_{vqvae}(E, G, Z) \\ & + \lambda_1 \mathcal{L}_{perceptual}(E, G, Z) \\ & + \lambda_2 \mathcal{L}_{GAN}(\{E, G, Z\}, D) \big], \end{aligned} \tag{3}$$

where (1) the first term $\mathcal{L}_{vqvae}(E, G, Z) = ||x - G(\hat{z} - sg(\hat{z} - z_q))||_2^2 + ||sg[\hat{z}] - z_q||_2^2 + ||sg[z_q] - \hat{z}||_2^2$ is VQVAE loss, where $\mathcal{L}_{rec} = ||x - G(\hat{z} - sg(\hat{z} - z_q))||_2^2$ is reconstruction loss. Since the quantization process is non-differentiable, a stop-gradient operation $sg[\cdot]$ is used so that the forward pass operates on quantized vectors, whereas the backward pass leverages continuous vectors for gradient computation (Van Den Oord et al., 2017); (2) the second term $\mathcal{L}_{perceptual}(E, G, Z) = ||P(x) - P(G(\hat{z} - sg(\hat{z} - z_q)))||_2^2$ is perceptual loss, P denotes a perceptual model like Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018), which is used to extract the high-level semantic features; (3) the third term $\mathcal{L}_{GAN}(\{E, G, Z\}, D) = \log D(x) + \log(1 - D(\hat{z} - sg(\hat{z} - z_q)))$ is GAN loss, D is a patch-based discriminator (Isola et al., 2017) aiming to differentiate original and reconstructed images. λ_1 is a hyperparameter and λ_2 is an adaptive weight computed dynamically to balance the weight of $\mathcal{L}_{GAN}$ and stabilize training, following VQGAN (Esser et al., 2021).

3.2 ChemMLLM

Chemical molecules are typically encoded in the format of Simplified Molecular Input Line Entry System (SMILES) (Weininger, 1988), a compact ASCII string, serving as a distinct modality in computational chemistry (Anderson et al., 1987). In ChemMLLM, SMILES tokenization is the same as the text. Specifically, SMILES is mapped into a token sequence via Chaemelon (Team, 2024) text tokenizer trained based on Byte Pair Encoding (BPE) algorithm (Sennrich et al., 2015). The process is shown in Figure 1(b).

After training the Mol-VQGAN, ChemMLLM uses it as image tokenizer to align image and text at token level, unifying the training and inference for image and text by maximizing the standard next-token prediction cross-entropy loss:

$$\begin{aligned} \mathcal{L}_{LLM} = & \sum_{i=1}^{L} \log p_\theta(s_i | s_1, ..., s_{i-1}) \\ & + \lambda \sum_{k} (\log \sum_{j=1}^{V} \exp(z_{k,j}))^2, \end{aligned} \tag{4}$$

where λ is a hyper-parameter; in the first term, L is the length of total sequence tokens; $s_i \in S = \{S_I, S_T\}$. S_I is the image tokens sequence tokenized by the image tokenizer, and S_T is the text/SMILES token sequence tokenized by the text tokenizer. The second term is z-loss, a regularization term to mitigate the problem of logit shift in the final softmax and stabilize training (Chowdhery

Method	MW			LogP			TPSA			valid%(↑)
	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	
InternVL-Chat-v1.5	0.04±0.05	$>1e+5\pm>1e+5$	210.5±29.4	-0.02±0.11	23.4±9.3	2.96±0.31	0.04±0.11	$>1e+4\pm6205.6$	58.9±8.1	75.0%
LLaVA-v1.5-7B	0.07±0.27	$>2e+6\pm>9e+5$	894.9±285.0	-0.52±0.16	1104.4±522.3	18.0±6.7	0.48 ±0.27	$>1e+5\pm>1e+5$	146.3±88.5	8.5%
GPT-4o	**0.77**±0.04	**7633.2**±1954.0	**55.7**±4.7*	**0.48**±0.07	**5.7**±0.87	**1.7**±0.11	0.69±0.04	1209.2±211.1	**24.3**±1.7	99.0%
ChemLLM-7B-Chat	-0.25±0.08	$>7e+4\pm>1e+4$	244.8±16.2	0.0005±0.05	10.0±1.2	2.7±0.19	-0.35±0.09	7666.0±937.2	80.6±4.3	35.0%
ChemVLM-8B	0.14±0.15	$>2e+5\pm>1e+5$	197.39±30.9	0.08±0.08	9.0±1.5	2.2±0.14	0.22±0.22	$>2e+4\pm>2e+4$	62.5±11.0	**100.0%**
ChemMLLM (ours)	0.71±0.05	$>3e+4\pm>1e+4$	119.5±12.1	0.42±0.05	13.2±9.0	1.8±0.26	**0.71**±0.05	**1191.6**±183.6	26.5±1.91	65.5%

Table 4: Results on property2img task: MW, LogP and TPSA (**best**, 2nd best, *: significantly better (statistically)).

Method	Hbd			Hba			Rb			QED		
	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)	Pearson (↑)	MSE (↓)	MAE (↓)
InternVL-Chat-v1.5	-0.1±0.08	7.3±3.8	1.5±0.18	0.09±0.08	55.3±31.1	3.4±0.52	0.06±0.08	44.9±15.5	4.3±0.41	0.07±0.08	0.09±0.009	0.24±0.01
LLaVA-v1.5-7B	0.0±0.0	-	-	0.44±0.27	529.7±495.4	9.3±5.0	-0.24±0.26	5789.7±3515.2	37.5±15.7	0.11±0.24	0.10±0.03	0.25±0.04
GPT-4o	**0.58**±0.07	1.8±0.35	0.86±0.07	0.57±0.07	6.2±1.0	**1.7**±0.12	0.45±0.07	**11.9**±2.1	**2.4**±0.17	**0.59**±0.04*	**0.03**±0.003	**0.15**±0.008
ChemLLM-7B-Chat	-0.23±0.07	3.4±0.82	1.5±0.14	-0.22±0.06	30.3±4.3	4.9±0.31	-0.22±0.07	46.9±6.5	5.9±0.43	-0.07±0.03	0.07±0.009	0.24±0.01
ChemVLM-8B	0.47±0.30	6.0±0.99	1.7±0.12	0.02±0.13	75.6±58.4	3.49±0.56	0.44±0.29	25.8±4.3	3.7±0.24	0.10±0.06	0.08±0.007	0.24±0.01
ChemMLLM (ours)	0.45±0.08	**1.5**±0.32	**0.83**±0.08	**0.66**±0.06	**5.8**±0.84	1.8±0.13	**0.62**±0.05*	17.3±8.1	2.4±0.29	0.34±0.08	0.08±0.009	0.24±0.01

Table 5: Results on property2img Task: Hbd, Hba, Rb, and QED (**best**, 2nd best, *: significantly better).

et al., 2023), where $z_{k,j}$ denotes the logit in the last layer, V denotes the size of the vocabulary.

Specifically, ChemMLLM adopts Chameleon VQGAN as the image tokenizer/de-tokenizer and Chameleon-7B as the language model. The image tokenizer takes images of 256×256 resolution as input. Simultaneously, text and SMILES information passes through the text tokenizer to be converted to text tokens. The text, SMILES and image tokens are then concatenated to form a unified token sequence to feed into the LLM during training and inference.

3.3 Training

ChemMLLM's training can be divided into two stages: (i) Mol-VQGAN training and (ii) ChemMLLM supervised fine-tuning (SFT) training, as shown in Figure 1(e).

(i) Mol-VQGAN training. The original Chameleon VQGAN is only trained on the natural image dataset and can not discrete and reconstruct molecule images well. So, the first stage focuses on improving VQGAN's performance in encoding and decoding molecule images. Concretely, we use the well-trained VQGAN (trained on natural images) as the initialization and then fine-tune it on molecule image datasets.

(ii) ChemMLLM Supervised Fine-Tuning Training. In the second stage, we freeze the Mol-VQGAN and only finetune the language model on 5 downstream tasks. We utilize Lumina-mGPT (Liu et al., 2024) as training framework to train our ChemMLLM and uses Chameleon-7B as the base model. The weight related to the image tokens in the last layer will first be initialized as zero during finetuning. LLM uses the output of Mol-VQGAN as finetuning data, *i.e.*, the data is first pre-tokenized by Mol-VQGAN and text tokenizer into token sequences and then fed into LLM.

Task	Input	Output	Source	# train/test
molecule image captioning (img2caption)	image +text	text	chebi-20 (Edwards et al., 2022) mol-instruct (Fang et al., 2023)	70K/3K
molecule image property prediction (img2property)	image +text	text	PubChem (Kim et al., 2021)	95K/5K
image-to-SMILES conversion (img2smiles)	image +text	SMILES	PubChem (Kim et al., 2021)	95K/5K
controllable multi-objective molecule image design (property2img)	text	image	PubChem (Kim et al., 2021)	95K/5K
molecule image optimization (img2img)	image +text	image	TDC (Huang et al., 2021)	157K/17K

Table 6: Tasks and datasets.

4 Tasks and Data Curation

In this paper, we design five vision-based chemistry research tasks, defined as follows.

(1) **Molecule image captioning (img2caption)** is an image-to-text task, where the models are expected to generate a caption concerning the source, functionality, structure feature and usage for each molecule image. This image-to-caption task requires models to translate molecule images into natural language descriptions, which is a process mirroring how chemists annotate experimental data. Examples for this task are shown in Table 11.

(2) **Molecule image property prediction (img2property)** is an image-to-text task, where models are expected to generate the value of seven different important properties for each molecule image, including molecule weight (MW), Partition Coefficient (P) of a solute between octanol and water (LogP), Topological Polar Surface Area (TPSA), Hydrogen Bond Donor (Hbd), Hydrogen Bond Acceptor (Hba), Rotatable Bond (Rb), and Quantitative Estimate of Drug-likeness (QED). More details for the properties can be found in

Model	domain	architecture	txt2txt	img2txt	txt2img	img2img
Qwen-VL-Chat	general	text tokenizer, vision encoder	✓	✓	×	×
InternVL-Chat-v1.5	general	text tokenizer, vision encoder	✓	✓	×	×
LLaVA-v1.5-7B	general	text tokenizer, vision encoder	✓	✓	×	×
GPT-4o	general	close-sourced	✓	✓	✓	✓
ChemLLM-7B-Chat	chemistry	text tokenizer	✓	×	×	×
ChemVLM-8B	chemistry	text tokenizer, vision encoder	✓	✓	×	×
ChemMLLM (ours)	chemistry	text tokenizer, vision tokenizer/de-tokenizer	✓	✓	✓	✓

Table 7: Architectures and capabilities of MLLMs and Chemical LLMs approaches.

Appendix G. This image-to-property prediction task evaluates a model's ability to infer key chemical properties directly from 2D molecular images, enabling researchers to extract actionable insights from molecular images without specialized software, which could accelerate high-throughput screening in drug/material design (Lu et al., 2021). Table 12 shows some examples.

(3) **Image-to-SMILES conversion (img2smiles)** is a fundamental chemistry task, where models are expected to recognize the SMILES in each molecular image. The image-to-SMILES translation task challenges models to convert 2D molecular images into SMILES strings, requiring precise recognition of atoms, bonds, rings, and stereochemistry. Examples for this task are shown in Table 13.

(4) **Controllable multi-objective molecule image design (property2img)** is the inverse problem of molecule image property prediction and is a text-to-image task, where models are expected to generate the image of a molecule conditioned on target properties. It is the core of molecule design (Du et al., 2022). The challenge lies in simultaneously optimizing multiple property constraints while maintaining chemical validity. Examples for this task are shown in Table 14.

(5) **Molecule image optimization (img2img)** is an image-to-image task, where models take a molecular structure with less desirable molecular properties (*e.g.*, LogP) as input and generate a similar molecular structure with more desirable properties while preserving desired chemical properties. It imitates the process of lead optimization, a fundamental problem in drug discovery (Huang et al., 2021; Fu et al., 2020). Examples for this task are shown in Table 15.

4.1 Data Curation

We employ RDKit (Landrum et al., 2006) to convert the original SMILES strings into molecular images across all five tasks. We primarily follow the methodology of SketchMol (Wang et al., 2025) and ChemVLM (Li et al., 2025) for data curation and diversity natural language templates synthesis. The input/output modalities, raw data sources, and sizes of training/test sets for all tasks are shown in Table 6. Further details on data curation are provided in Appendix A.

5 Experiment

5.1 Experimental Setup

Baseline Methods cover both general multimodal LLM and chemical LLM. For general-domain multimodal LLM, we chose Qwen-VL-Chat (Bai et al., 2023), InternVL-Chat-v1.5 (Chen et al., 2023), LLaVA-v1.5-7B (Liu et al., 2023) and GPT-4o (OpenAI, 2024). For chemical LLM, we chose ChemLLM-7B-Chat (Zhang et al., 2024a) and ChemVLM-8B (Li et al., 2025). We compare their capabilities in Table 7. Please refer to Appendix D for more descriptions.

Evaluation metrics and **implementation details** are elaborated in Appendix F and H, respectively. The code is publicly available at `https://anonymous.4open.science/r/ChemMLLM-0D98/`.

5.2 Result

Molecule image captioning (img2caption). We compare our model with various multimodal LLMs (MLLMs) including Qwen-VL-Chat (Bai et al., 2023), InternVL-Chat-v1.5 (Chen et al., 2023), LLaVA-v1.5-7B (Liu et al., 2023), GPT-4o (OpenAI, 2024), ChemVLM-8B (Li et al., 2025). The evaluation results are shown in Table 1. Our model exhibits strong performance on this task, outperforming all competing MLLM models on all six metrics. An example is shown in Figure 2. Our model generates captions that closely match the ground truth, while Qwen-VL-Chat includes fewer semantically informative details.

Molecule image property prediction (img2property). GPT-4o can not predict

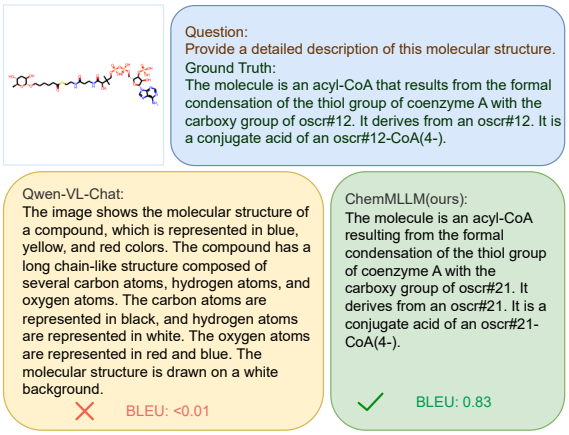


Figure 2: An example on img2caption task, comparison between Qwen-VL-Chat and our ChemMLLM.

properties from image directly, so in this task we do not compare with GPT-4o. As shown in Table 2 and 3, our model consistently outperforms competing methods across all seven molecular properties, yielding the highest Pearson correlation coefficients alongside the lowest MSE and MAE values. An example is shown in Figure 3. Among the properties predicted, our model has 5 accurate values and 2 close values while Qwen-VL-Chat has 2 close values and 5 inaccurate values.

Model	Avg Sim (↑)	Tani@1.0 (↑)	valid%(↑)
Qwen-VL-Chat	0.08 ±0.006	0.0 ±0.0	8.2%
InternVL-Chat-v1.5	0.09 ±0.003	0.0 ±0.0	20.7%
LLaVA-v1.5-7B	0.05 ±0.004	0.0 ±0.0	11.1%
GPT-4o	0.29±0.005	0.01 ±0.004	74.5%
ChemVLM-8B	0.55 ±0.009	0.15 ±0.01	85.2%
ChemMLLM (ours)	**0.75**±0.009*	**0.49**±0.01*	**97.1%**

Table 8: Results on img2smiles Task. Tanimoto similarities are written as Avg Sim, and Tanimoto@1.0 written as Tani@1.0 (**best**, 2nd best, *: significantly better).

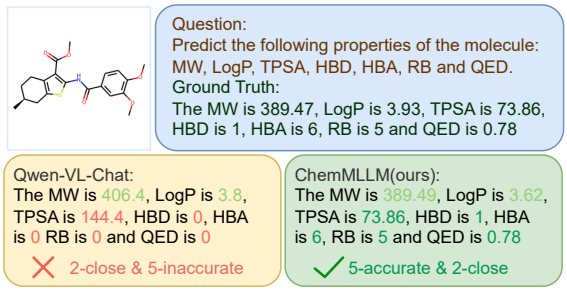


Figure 3: A comparison of answers on img2property task on Qwen-VL-Chat and our ChemMLLM. Accurate answers are highlighted in bottle-green, close answers are highlighted in light-green and inaccurate answers are highlighted in red.

Image-to-SMILES conversion (img2smiles). The evaluation results are shown in Table 8. ChemMLLM performances best in both Tanimoto similarity and Tanimoto@1 metrics. For Tanimoto similarity, ChemMLLM (0.75) surpasses domain-specific model ChemVLM (0.55) by 36.4%. An example is shown in Figure 4. Our model recognizes SMILES from images successfully, while GPT-4o predicts wrong SMILES with low Tanimoto similarity.

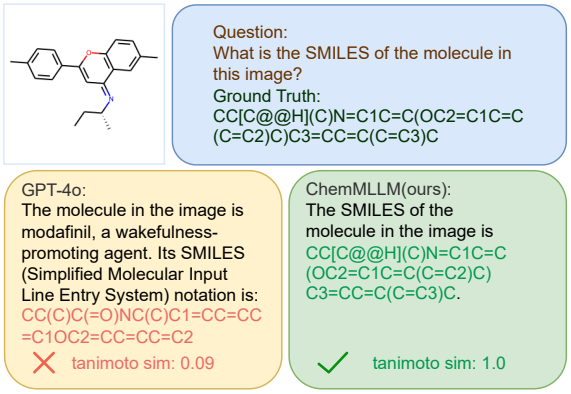


Figure 4: A comparison of GPT-4o's and ChemMLLM's answers on img2smiles task.

Controllable multi-objective molecule image design (property2img). Since the GPT-4o API does not perform as well on this task as its web interface, and other MLLMs lack the capability to generate images, we treat this task as a purely text-based problem when evaluating other MLLMs. Specifically, our model is used to generate molecular images, while other MLLMs are tasked with directly generating the corresponding SMILES strings in text form. Given this setting, we also include ChemLLM-Chat-7B (Zhang et al., 2024a), a domain-specific chemical language model, in our evaluation. Furthermore, we exclude Qwen-VL-Chat (Bai et al., 2023) from comparison on this task, as it fails to generate valid SMILES strings on all test samples. The evaluation results are shown in Table 4 and 5. Our model achieves top-2 or better performance on 90% of the evaluation metrics. Several examples are shown in Figure 5. Our model generates the image molecules with desired properties directly. For more result examples for this task, please refer to Figure 12.

Molecule image optimization (img2img). Since the GPT-4o API does not achieve the same level of performance on this task as its web-based interface and other MLLMs lack the ability to generate

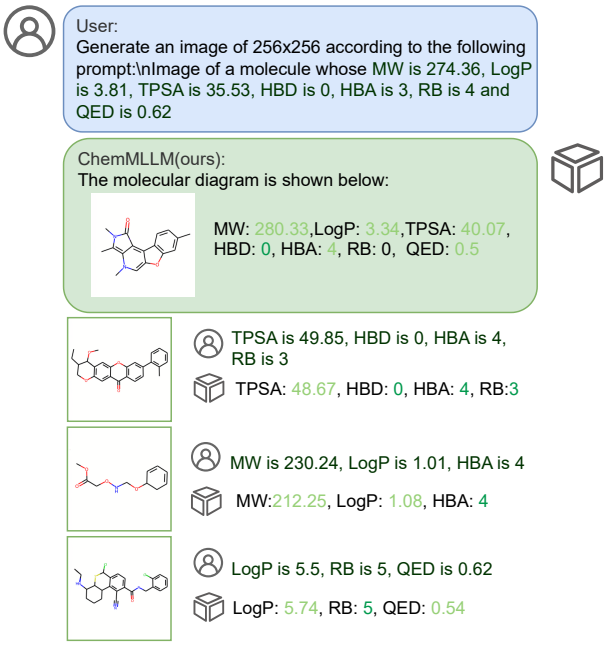


Figure 5: Examples on property2img task on our ChemMLLM. Accurate answers are highlighted in bottle-green, close answers are highlighted in light-green.

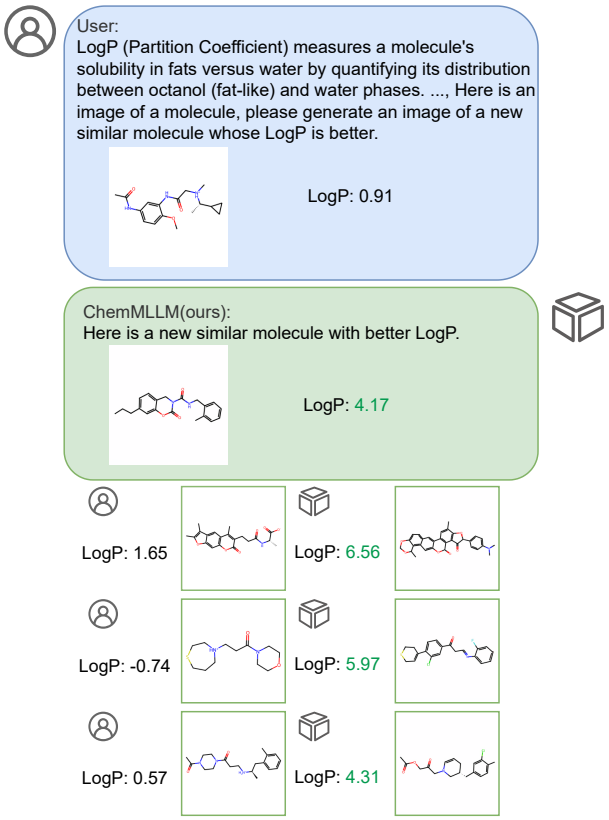


Figure 6: Examples of ChemMLLM on img2img task.

images, we formulate the evaluation as an image-to-text task for these models. Specifically, our model generates molecular images directly given optimized molecule images, while other MLLMs are required to generate SMILES strings. As GPT-4o cannot directly generate SMILES with optimized LogP from molecular images, we instead treat its evaluation as a text-to-text task, by providing the input SMILES in textual form rather than as images. As shown in Table 9, ChemMLLM achieves the highest increase in LogP, outperforms GPT-4o by 118.9%. Several examples are shown in Figure 6. Our model generates molecule images with higher LogP directly. For more result examples for this task, please refer to Figure 13.

Model	Increased LogP (↑)	Diversity (↑)	Novelty (↑)	valid% (↑)
Qwen-VL-Chat	-2.0 ±0.11	0.95±0.01	1.0±0.0	4.0%
InternVL-Chat-v1.5	-0.77±0.17	0.90±0.004	1.0±0.0	48.0%
LLaVA-v1.5-7B	-0.86±0.59	**0.96**±0.005	1.0±0.0	37.5%
GPT-4o	1.95±0.08	0.86±0.002	1.0±0.0	**99.0%**
ChemVLM-8B	0.45 ±0.14	0.87±0.002	0.97±0.01	92.5%
ChemMLLM (ours)	**4.2**±0.44	0.88±0.001	**1.0**±0.0	91.0%

Table 9: Results on img2img task (**best**, 2nd best).

5.3 Ablation Study

We conduct an ablation study on property2img task (see Appendix I) to assess the effects of Mol-VQGAN training and data augmentation (especially image rotation). Results show that both components significantly improve the correlation between generated images and molecular properties, with their combination yielding the best overall performance. This highlights the importance of high-quality visual representations in enhancing multimodal chemical tasks.

6 Conclusion

This paper has proposed ChemMLLM, a chemical multimodal large language model that handles molecule comprehension and generation across three modalities (text, SMILES string, molecule image). By jointly modeling text, SMILES strings, and molecule images, ChemMLLM enables seamless cross-modal comprehension and generation, outperforming state-of-the-art MLLMs and specialized chemical LLMs across a range of tasks. Also, we design five cross-modal chemistry tasks and curate datasets, providing a valuable resource for multimodal AI in chemistry. The experimental results demonstrate ChemMLLM's strong performance, highlighting its potential for real-world drug and material discovery.

Limitations

Despite its promising capabilities, our work has several limitations that point to important directions for future research: Currently, ChemMLLM incorporates only three modalities — text, SMILES strings, and 2D molecule images. Real-world chemical data often includes richer modalities such as 3D molecular structures, quantum mechanical properties, or spectroscopic data. Incorporating these would significantly enhance the model's ability to capture complex molecular behaviors and interactions. Also, our evaluation is primarily focused on proof-of-concept chemistry tasks. Further studies are needed to validate the model's performance in real-world applications such as drug discovery, materials design, or chemical synthesis planning.

Ethics Statement

The development and application of chemical AI models, such as ChemMLLM, raise important ethical considerations. We ensure that all datasets used in this work are sourced from publicly available, non-sensitive chemical data and do not involve personal or private information. The model is designed for scientific research purposes, including drug discovery and materials science, with the aim of advancing chemical understanding and innovation. We advocate for responsible use of AI in chemistry and encourage transparency, reproducibility, and fairness in future deployments of such technologies.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.

Eric Anderson, Gilman D Veith, and David Weininger. 1987. *SMILES, a line notation and computerized interpreter for chemical structures*. US Environmental Protection Agency, Environmental Research Laboratory.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Jinze Bai, Shuai Bai, Shusheng Yang, Sinan Wang, Zhitang Tan, Penghui Wang, Junyang Chen, Jun Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

He Cao, Zijing Liu, Xingyu Lu, Yuan Yao, and Yu Li. 2023. Instructmol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery. *arXiv preprint arXiv:2311.16208*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2023. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Yuanqi Du, Tianfan Fu, Jimeng Sun, and Shengchao Liu. 2022. Molgensurvey: A systematic survey in machine learning models for molecule design. *arXiv preprint arXiv:2203.14500*.

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

Patrick Esser, Robin Rombach, and Bjorn Ommer. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883.

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. 2023. Mol-instructions: A large-scale biomolecular instruction dataset for large language models. *arXiv preprint arXiv:2306.08018*.

Tianfan Fu, Cao Xiao, and Jimeng Sun. 2020. CORE: Automatic molecule optimization using copy and refine strategy. *AAAI*.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. 2021. Therapeutics data commons: machine learning datasets and tasks for therapeutics. *NeurIPS Track Datasets and Benchmarks*.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Dhiraj Kalamkar, Dheevatsa Mudigere, Naveen Mellempudi, Dipankar Das, Kunal Banerjee, Sasikanth Avancha, Dharma Teja Vooturi, Nataraj Jammalamadaka, Jianyu Huang, Hector Yuen, et al. 2019. A study of bfloat16 for deep learning training. *arXiv preprint arXiv:1905.12322*.

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

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Yana Kosenkov and Dmitri Kosenkov. 2021. Computer vision in chemistry: Automatic titration.

Greg Landrum et al. 2006. Rdkit: Open-source cheminformatics.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Junxian Li, Di Zhang, Xunzhi Wang, Zeying Hao, Jingdi Lei, Qian Tan, Cai Zhou, Wei Liu, Yaotian Yang, Xinrui Xiong, et al. 2025. Chemvlm: Exploring the power of multimodal large language models in chemistry area. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 415–423.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Dongyang Liu, Shitian Zhao, Le Zhuo, Weifeng Lin, Yu Qiao, Hongsheng Li, and Peng Gao. 2024. Lumina-mgpt: Illuminate flexible photorealistic text-to-image generation with multimodal generative pretraining. *arXiv preprint arXiv:2408.02657*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Yingzhou Lu, Chiung-Ting Wu, Sarah J Parker, Lulu Chen, Georgia Saylor, Jennifer E Van Eyk, David M Herrington, and Yue Wang. 2021. COT: an efficient python tool for detecting marker genes among many subtypes. *bioRxiv*, pages 2021–01.

Yizhen Luo, Kai Yang, Massimo Hong, Xing Yi Liu, Zikun Nie, Hao Zhou, and Zaiqing Nie. 2024. Learning multi-view molecular representations with structured and unstructured knowledge. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2082–2093.

OpenAI. 2023. Gpt-4v(ision) system card. Accessed: 2024-07-20.

OpenAI. 2024. Gpt-4o: Our most advanced ai model. Accessed: 2024-07-20.

Yujie Qian, Jiang Guo, Zhengkai Tu, Zhening Li, Connor W. Coley, and Regina Barzilay. 2023. MolScribe: Robust molecular structure recognition with image-to-graph generation. *Journal of Chemical Information and Modeling*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

OpenEye Scientific Software. 2023. *OpenEye Toolkits Documentation*.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. 2024. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*.

Chameleon Team. 2024. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Aaron Van Den Oord, Oriol Vinyals, et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. 2024. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*.

Zixu Wang, Yangyang Chen, Pengsen Ma, Zhou Yu, Jianmin Wang, Yuansheng Liu, Xiucai Ye, Tetsuya Sakurai, and Xiangxiang Zeng. 2025. Image-based generation for molecule design with sketchmol. *Nature Machine Intelligence*, pages 1–12.

David Weininger. 1988. 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.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. 2024. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, et al. 2024a. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*.

Juzheng Zhang, Yatao Bian, Yongqiang Chen, and Quanming Yao. 2024b. Unimot: Unified molecule-text language model with discrete token representation. *arXiv preprint arXiv:2408.00863*.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. 2023. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. 2024. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Data Curation Details

(1) **img2caption**: The dataset used for this task is sourced from chebi-20 (Edwards et al., 2022)and mol-instruct (Fang et al., 2023). The original datasets contain SMILES-caption pairs. We utilize RDKit (Landrum et al., 2006) to transfer the SMILES into 256×256 image to form image-caption pairs. For mol-instruct, we filter captions shorter than 150 words to screen clearer descriptions. We only use the test set of chebi-20 as the test set and the partitioning of the dataset is the same as chebi-20 (Edwards et al., 2022).

(2) **img2property**: The dataset used for this task is sourced from PubChem (Kim et al., 2021). For one compound, there are three fileds related to it's SMILES, *i.e.*, "PUBCHEM_SMILES" which means The Simplified Molecular Linear Input Specification (SMILES) for compounds, it is a string used to represent the chemical structure, "PUBCHEM_OPENEYE_CAN_SMILES" which means Canonical SMILES generated using the OpenEye tool (Software, 2023) and "PUBCHEM_OPENEYE_ISO_SMILES" which means The isomer SMILES generated using the OpenEye tool. The processing steps are as follows:

1. Extract all the three fields from PubChem and remove the duplicate SMILES.
2. Use the Draw.MolToImage() function in RDKit (Landrum et al., 2006) to transfer SMILES into images.
3. Sample 100,000 SMILES for this work.
4. Choose 7 important properties as the prediction objective, *i.e.*, MW, LogP, TPSA, HBD, HBA, RB and QED. Then use RDKit to calculate the 7 properties for each sampled SMILES.
5. Use natural language templates to integrate properties into natural language to form the final image-property answer pairs.
6. Divide the dataset into training and test set by the ratio of 95:5.

(3) **img2smiles**: The dataset and the construction steps are the same as the img2property task, the difference is that this task only apply templates for SMILES to construct image-SMILES answer pairs.

(4) **property2img**: This task is a reverse task of img2property. By swapping the question and answer of the img2property dataset, we construct property prompt-image pairs for property2img task.

(5) **img2img**: The dataset for this task is sourced from TDC (Huang et al., 2021). The original dataset contains SMILES-SMILES pairs. The previous molecule's LogP is lower while the latter molecule's LogP is higher. We also use Draw.MolToImage() function in RDKit (Landrum et al., 2006) to transfer them into image-image pairs. For data partitioning, we divide the training set and test set by the ratio of 9:1.

All the data used in this paper is publicly available.

B Mathematical Notations

For ease of understanding, we list key mathematical notations in Table 10 in Appendix.

C Data Examples

(1) **Molecule image captioning (img2caption).** As shown in Table 11, the input/output for img2caption task is text/image-text pair. The input is a question asking models to give captions for molecule images and the output is a caption concerning the source, functionality, structure feature and usage for each molecule image.

(2) **Molecule image property prediction (img2property).** As shown in Table 12, the input/output for img2property task is text/image-text pair. The input is a question asking models to predict seven properties for given molecule images and the output is a natural language answer describing the seven properties.

(3) **Image-to-SMILES conversion (img2smiles).** As shown in Table 13, the input/output for img2smiles task is text/image-text/SMILES pair. The input is a question asking models to recognize the SMILES in the given molecule images and the output is a natural language answer describing the SMILES in the image.

(4) **Controllable multi-objective molecule image design (property2img).** As shown in Table 14, the input/output for property2img task is text-text/image pair. The input is a question asking models to generate images according to the given values of the seven properties, and the output is the generated molecule image with the given values of the seven properties.

(5) **Molecule image optimization (img2img).** As shown in Table 15, the input/output for img2img task is text/image-text/image pair. The input is a question describing the meaning of LogP and then asking models to optimize the LogP property for given molecule images, and the output is the optimized molecule image with better LogP.

D Baseline Methods

- **Qwen-VL-Chat** (Bai et al., 2023) is an open-source multimodal conversational model developed by Alibaba Cloud, extending the Qwen-VL architecture to support complex visual-language interaction through instruction tuning. It integrates a frozen CLIP-ViT-G/14 vision encoder with the Qwen-7B language model via a trainable vision-language connector. Images are encoded into 1024-dimensional patch embeddings by the vision encoder, which are then linearly projected into the token embedding space of the language model (hidden size 4096). The language backbone consists of 32 transformer decoder layers, each employing multi-head masked self-attention with rotary position embeddings (RoPE), followed by a SwiGLU-activated feed-forward network with an intermediate dimension of 11008. Qwen-VL-Chat uses a 2048-token context window and supports dynamic multimodal prompts comprising text, images, and region boxes. It is instruction-tuned on a large-scale, GPT-generated multimodal dataset containing both single- and multi-turn visual conversations, enabling capabilities in visual question answering, dense captioning, document OCR, and multi-image reasoning. The model achieves high performance on benchmarks such as MME, SEED-Bench, and MMBench, demonstrating strong alignment between visual and linguistic modalities.

- **InternVL-Chat-V1.5** (Chen et al., 2023) is an open-source vision-language instruction-following model developed by OpenGVLab, designed to support high-resolution, multilingual, and multi-turn visual conversations. The model integrates a powerful ViT-based vision encoder, InternViT-6B, with the InternLM2-Chat-20B language model via a trainable multi-layer perceptron (MLP) connector. InternViT encodes images into patch embeddings with dynamic resolution support, allowing the model to process up to 40 image tiles of size 448×448, effectively supporting 4K-level inputs. These embeddings are projected into the language model's token space to enable seamless multimodal interaction. The language backbone consists of 64 transformer decoder layers with rotary posi-

Table 10: Mathematical notations.

Notations	Descriptions
$x/\hat{x} \in \mathbb{R}^{H\times W\times 3}$	the input/reconstructed molecule image
H/W	the height/width of the input image
h/w	the height/width of the feature map
n_z	the channels of the feature map, same as the dimension of the codebook vector
E	the encoder of VQGAN, a convolutional neural network (CNN) that extracts features from original image
$\hat{z} \in \mathbb{R}^{h\times w\times n_z}$	the continuous feature map encoded by $E(x)$
$\hat{z}_{ij} \in \mathbb{R}^{n_z}$	the spatial code $\in \mathbb{R}^{n_z}$ at position i, j in the feature map; $(i, j) \in \{0, 1, \ldots, h\} \times \{0, 1, \ldots, w\}$
$\mathbf{q}$	vector quantization process in VQGAN, which transfers continuous feature into discrete feature
$Z = \{z_i\}_{i=1}^{n} \subset \mathbb{R}^{n_z}$	the codebook in VQGAN, a dictionary that represents the latent discrete space
$z_k \in \mathbb{R}^{n_z}$	entry in the codebook
$z_q \in \mathbb{R}^{h\times w\times n_z}$	the quantized feature map quantified by $\mathbf{q}(\hat{z})$
G	the decoder of VQGAN, a CNN that reconstructs image from latent discrete space
$sg[\cdot]$	the stop-gradient operation
$\mathcal{L}_{vqvae}$	origin VQVAE training loss
$\mathcal{L}_{rec}$	the reconstruction loss
$\mathcal{L}_{perceptual}$	the perceptual loss
$\mathcal{L}_{GAN}$	GAN loss
D	the discriminator to identify x and $\hat{x}$, a patch-based discriminator (Isola et al., 2017)
$\mathcal{L}_{LLM}$	next-token prediction cross-entropy loss with z-loss for training large language model
$p_\theta(s_i\|s_1, \ldots, s_{i-1})$	the probability of s_i given $s_1, \ldots, s_{i-1}$
S_I	image token sequence tokenized by image tokenizer
S_T	text/SMILES token sequence tokenized by text tokenizer
S	the concatenated sequence of image token sequence and text/SMILES token sequence
L	the size of total sequence tokens
V	the size of vocabulary
$z_{k,j} \in \mathbb{R}$	the logit at last layer.
$\lambda, \lambda_1, \lambda_2$	hyper-parameters or adaptively calculated parameters to adjust the weight of different loss functions

tional embeddings (RoPE), multi-head masked self-attention, and SwiGLU-activated feedforward layers. The model uses a maximum context length of 4096 tokens and is instruction-tuned on a high-quality bilingual dataset containing document images, natural images, and complex multimodal dialogues. InternVL-Chat-V1.5 achieves strong performance on benchmarks including MME, MMBench, and AI2D, demonstrating robust capabilities in visual question answering, document OCR, visual reasoning, and bilingual understanding. The total parameter count is approximately 25.5 billion, with both the vision encoder and language model jointly fine-tuned during the instruction-following phase.

- **LLaVA-v1.5-7B** (Large Language and Vision Assistant) (Liu et al., 2023) is an open-source vision-language instruction-tuned model that integrates a pre-trained CLIP-ViT-L/14 vision encoder with the LLaMA-7B language model through a projection network. LLaVA-7B processes visual inputs by encoding images into 1024-dimensional patch embeddings via the vision encoder, which are then projected into the language model's token space through a trainable linear layer (hidden size 4096). The language model comprises 32 transformer decoder blocks, each with masked self-attention (key/-query size 4096, 32 attention heads), followed by a feed-forward network with SwiGLU activation and an intermediate dimension of 11008. The self-attention layers use RoPE (rotary positional

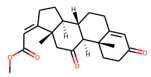
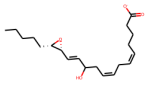
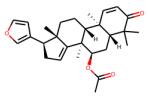
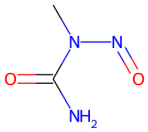
input	output
Give an overview of this molecule's properties. 	The molecule is a steroid ester that is methyl (17E)-pregna-4,17-dien-21-oate substituted by oxo groups at positions 3 and 11. It is a 3-oxo-Delta(4) steroid, an 11-oxo steroid, a steroid ester and a methyl ester. It derives from a hydride of a pregnane.
Explain the key traits of this molecule. 	The molecule is an epoxy(hydroxy)icosatrienoate that is the conjugate base of 11 hydroxy-(14R,15S)-epoxy-(5Z,8Z,12E)-icosatrienoic acid, obtained by deprotonation of the carboxy group; major species at pH 7.3. It is a conjugate base of an 11 hydroxy-(14R,15S)-epoxy-(5Z,8Z,12E)-icosatrienoic acid.
Provide a detailed description of this molecular structure. 	The molecule is a tetracyclic triterpenoid that is 4,4,8-trimethylandrosta-1,14-diene substituted by an oxo group at position 3, an acetoxy group at position 7 and a furan-3-yl group at position 17. Isolated from Azadirachta indica, it exhibits antiplasmodial and antineoplastic activities. It has a role as an antineoplastic agent, an antiplasmodial drug and a plant metabolite. It is an acetate ester, a cyclic terpene ketone, a member of furans, a limonoid and a tetracyclic triterpenoid.
List the notable attributes of this molecule. 	The molecule is a member of the class of N-nitrosoureas that is urea in which one of the nitrogens is substituted by methyl and nitroso groups. It has a role as a carcinogenic agent, a mutagen, a teratogenic agent and an alkylating agent.

Table 11: Example for img2caption task.

embeddings) and support a context window of 2048 tokens. LLaVA-7B leverages instruction-tuning on 558K GPT-4 generated multimodal instruction-following samples, aligning visual and textual representations for tasks such as visual QA and image captioning. The total number of trainable parameters is approximately 7 billion, with the vision encoder frozen during fine-tuning.

- **GPT-4o** (OpenAI, 2024) (Generative Pre-trained Transformer 4 Omni) is a state-of-the-art multimodal foundation model developed by OpenAI, designed to natively process and reason across text, images, and audio modalities. Unlike previous GPT-4 variants that rely on separate vision encoders, GPT-4o employs a unified transformer architecture that jointly encodes multimodal inputs, enabling low-latency and high-fidelity interactions. The model supports up to 128k tokens of context and exhibits strong performance across a wide range of tasks, including natural language understanding, image captioning, document analysis, and spoken language comprehension. GPT-4o achieves significant improvements in visual reasoning (*e.g.*, charts, diagrams, OCR), math problem solving, and multilingual capability, surpassing the capabilities of GPT-4-turbo while operating with lower inference latency. It is instruction-tuned on a diverse and extensive corpus of text and multimodal data, and optimized for both conversational fluency and factual grounding. We also utilized GPT-4o to assist in the writing of this paper.

- **ChemLLM-7B-Chat** (Zhang et al., 2024a) is an open-source domain-specific large language model framework tailored for chemical sciences, designed to address the limitations of general-purpose LLMs in structured scientific domains. The model is instruction-tuned using ChemData, a templated dataset that transforms structured chemical knowledge (*e.g.*, molecular properties, SMILES strings, compound databases) into natural language instructions across a variety of chem-

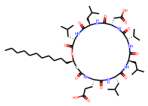
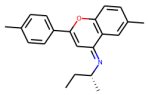
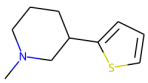
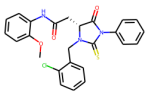
input	output
Predict the following properties of the molecule: MW, LogP, TPSA, HBD, HBA, RB and QED. 	The MW is 1022.34, LogP is 4.8, TPSA is 304.6, HBD is 9, HBA is 11, RB is 24 and QED is 0.05
Predict the following properties of the molecule: MW, LogP, TPSA, HBD, HBA, RB and QED. 	The MW is 305.42, LogP is 5.42, TPSA is 25.5, HBD is 0, HBA is 2, RB is 3 and QED is 0.63
Predict the following properties of the molecule: MW, LogP, TPSA, HBD, HBA, RB and QED. 	The MW is 181.3, LogP is 2.56, TPSA is 3.24, HBD is 0, HBA is 2, RB is 1 and QED is 0.64
Predict the following properties of the molecule: MW, LogP, TPSA, HBD, HBA, RB and QED. 	The MW is 479.99, LogP is 4.88, TPSA is 61.88, HBD is 1, HBA is 4, RB is 7 and QED is 0.49

Table 12: Example for img2property task.

ical tasks. ChemLLM adopts a decoder-only transformer architecture based on InternLM2-Base-7B (Cai et al., 2024), and is fine-tuned to handle both single- and multi-turn dialogues. The model supports tasks such as molecular property prediction, compound generation, synthetic route planning, and reaction condition recommendation.

- **ChemVLM-8B** is an open-source multimodal domain-specific large language model specifically designed for chemistry-related tasks, aiming to bridge the gap between vision and language understanding in the chemical domain. The model adopts a ViT-MLP-LLM architecture, integrating a vision encoder based on Vision Transformer (ViT), a multi-layer perceptron (MLP) as a projection layer, and a 20-billion-parameter decoder-only language model ChemLLM-20B (Zhang et al., 2024a) as the backbone. ChemVLM processes visual inputs such as molecular structures, chemical reaction schemes, and spectra by encoding images into patch embeddings through ViT, which are then linearly projected to the token space of the language model. It is instruction-tuned on a constructed dataset of 1.2M multimodal samples covering tasks like molecule captioning, reaction classification, and chemical structure understanding.

E More Visual Result

To better visualize results for different run on five tasks, we draw metric bar for each tasks. The metric bar for img2caption, img2property, img2smiles, property2img, img2img task is shown in Figure 7, Figure 8, Figure 10, Figure 11, Figure 9, respectively.

For the two image generation task, we provide more examples to better display the ability of ChemMLLM. More examples for property2img task are shown in Figure 12. More examples for img2imgtask are shown in Figure 13.

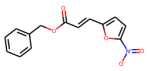
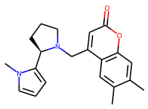
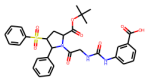
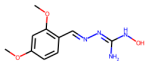
input	output
What is the SMILES of the molecule in this image? 	The SMILES of the molecule in the image is C1=CC=C(C=C1)COC(=O)/C=C/C2=CC=C(O2)[N+](=O)[O-].
What is the SMILES of the molecule in this image? 	The SMILES of the molecule in the image is CC1=CC2=C(C=C1C)OC(=O)C=C2CN3CCC[C@@H]3C4=CC=CN4C.
What is the SMILES of the molecule in this image? 	The SMILES of the molecule in the image is CC(C)(C)OC(=O)C1CC(C(N1C(=O)CNC(=O)NC2=CC=CC(=C2)C(=O)O)C3=CC=CC=C3)S(=O)(=O)C4=CC=CC=C4.
What is the SMILES of the molecule in this image? 	The SMILES of the molecule in the image is COC1=CC(=C(C=C1)C=N/N=C(\\N)/NO)OC.

Table 13: Example for img2smiles task.

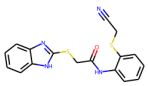
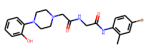
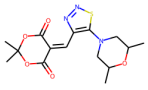
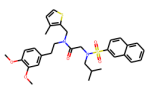
input	output
Generate an image of 256x256 according to the following prompt:\n Image of a molecule whose MW is 354.46, LogP is 3.91, TPSA is 81.57, HBD is 2, HBA is 5, RB is 6 and QED is 0.66	The molecular diagram is shown below: 
Generate an image of 256x256 according to the following prompt:\n Image of a molecule whose MW is 461.36, LogP is 2.34, TPSA is 84.91, HBD is 3, HBA is 5, RB is 6 and QED is 0.62	See the molecular depiction: 
Generate an image of 256x256 according to the following prompt:\n Image of a molecule whose MW is 353.4, LogP is 1.37, TPSA is 90.85, HBD is 0, HBA is 9, RB is 2 and QED is 0.45	The molecular diagram is shown below: 
Generate an image of 256x256 according to the following prompt:\n Image of a molecule whose MW is 594.8, LogP is 6.15, TPSA is 76.15, HBD is 0, HBA is 6, RB is 13 and QED is 0.19	See the molecular depiction: 

Table 14: Example for property2img task.

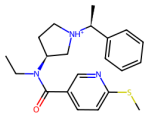
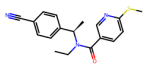
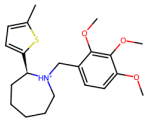
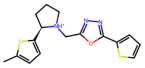
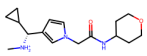
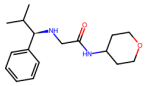
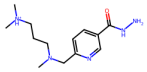
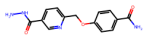
input	output
LogP (Partition Coefficient) measures a molecule's solubility in fats versus water by quantifying its distribution between octanol (fat-like) and water phases. Calculated as the logarithm of the concentration ratio (LogP = log[octanol]/[water]), it predicts drug absorption and permeability—higher values (>0) indicate greater fat solubility, while lower values (<0) suggest water solubility. Ideal drug candidates typically have LogP between 0-3 for optimal bioavailability. Here is an image of a molecule, please generate an image of a new similar molecule whose LogP is better. 	Here is a new similar molecule with better LogP. 
LogP (Partition Coefficient) measures a molecule's solubility in fats versus water by quantifying its distribution between octanol (fat-like) and water phases.... Here is an image of a molecule, please generate an image of a new similar molecule whose LogP is better. 	Here is a new similar molecule with better LogP. 
LogP (Partition Coefficient) measures a molecule's solubility in fats versus water by quantifying its distribution between octanol (fat-like) and water phases....Here is an image of a molecule, please generate an image of a new similar molecule whose LogP is better. 	Here is a new similar molecule with better LogP. 
LogP (Partition Coefficient) measures a molecule's solubility in fats versus water by quantifying its distribution between octanol (fat-like) and water phases....Here is an image of a molecule, please generate an image of a new similar molecule whose LogP is better. 	Here is a new similar molecule with better LogP. 

Table 15: Example for img2img task.

F Evaluation Metrics

(1) **img2caption**: We use BLEU-2/4, ROUGE-1/2/L, and METEOR to evaluate the quality of generated captions against reference texts; (2) **img2property**: We extract seven molecular properties from LLM-generated outputs and evaluate the accuracy using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Pearson correlation; (3) **img2smiles**: We extract SMILES strings from the LLM outputs and adopt Tanimoto similarity and Tanimoto hit 1.0 (tanimoto@1.0) that measures the percentage of exact matches (similarity = 1.0); (4) **property2img**: Generated molecular images are converted to SMILES via MolScribe (Qian et al., 2023), from which properties are computed and evaluated using MSE, MAE, and Pearson correlation. Each model is run five times, and the best result is reported; (5) **img2img**: We use Increased LogP, as well as molecular diversity and novelty to measure the optimized molecule. Other settings

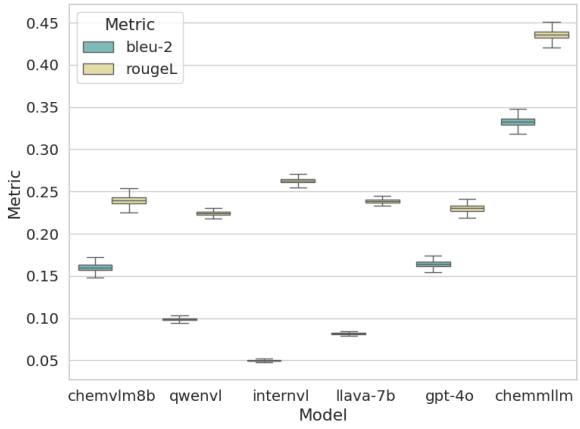


Figure 7: Metric bar for different runs of img2caption task.

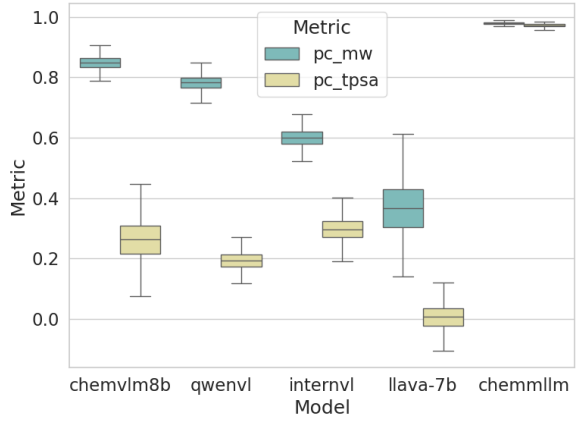


Figure 8: Metric bar for different runs of img2property task. pc_mw means the Pearson correlation between the predicted molecule weight and groundtruth and pc_tpsa means the Pearson correlation between the predicted topological polar surface area and groundtruth.

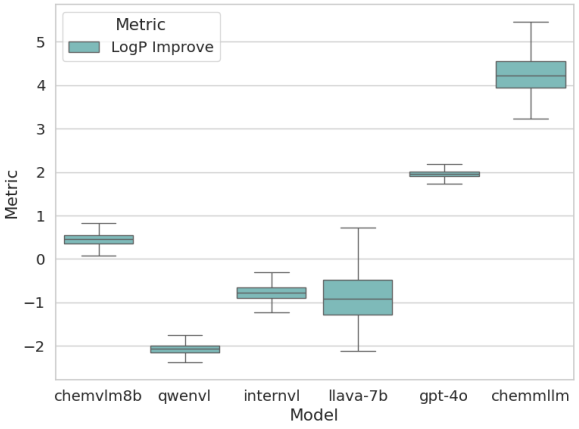


Figure 9: Metric bar for different runs of img2imgtask. LogP Improve means Increased LogP, which is the increase in LogP of the optimized molecule relative to the original molecule.

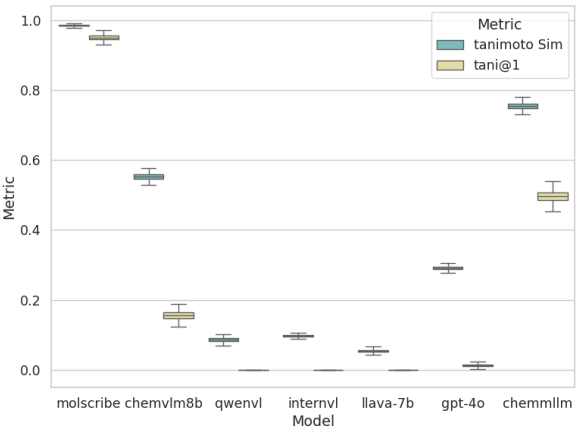


Figure 10: Metric bar for different runs of img2smiles task.

are the same as property2img.

The detailed explanations of each metric are as follows:

- **BLEU-N** (Bilingual Evaluation Understudy) is an automatic evaluation metric for machine-generated text that assesses how closely a candidate sentence matches one or more reference sentences. It uses the modified precision of n-grams up to length N. It is defined as

$$\text{BLEU-N} = \text{BP} \cdot \exp\left(\frac{1}{N}\sum_{n=1}^{N}\log p_n\right), \quad (5)$$

  where p_n denotes the modified n-gram precision for n-grams of size n, and BP is the brevity penalty, which penalizes short candidate sentences to prevent artificially high scores. The BLEU-N score ranges from 0 to 1, where a higher score indicates better overlap with the reference text in terms of n-gram content. A higher BLEU-N value generally reflects better fluency and adequacy in the generated text. In our experimental evaluation, we use the word_tokenize() function from the NLTK library (Bird et al., 2009) to do tokenization and employ the sentence_bleu() metric with uniform weights for all n-gram precision calculations (*i.e.*, equal weights for 1- to 4-gram contributions).

- **ROUGE-N** (Recall-Oriented Understudy for Gisting Evaluation) is a recall-based metric that measures the overlap of n-grams between a candidate text and one or more reference

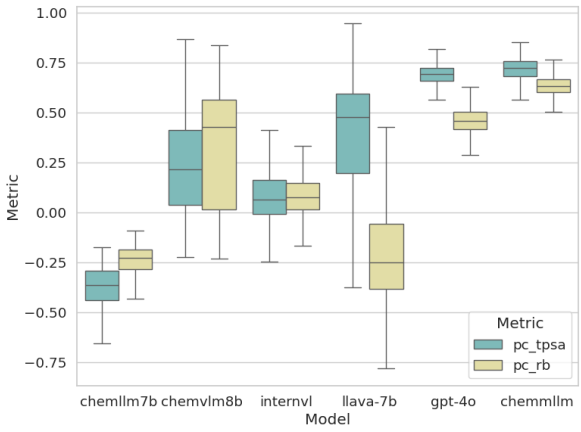


Figure 11: Metric bar for different runs of property2img task. pc_tpsa means the Pearson correlation between the predicted topological polar surface area and ground truth, and pc_rb means the Pearson correlation between the predicted rotatable bond and ground truth.

texts. It is defined as

$$\text{ROUGE-N} = \frac{\sum_{S \in \{\text{Ref}\}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \{\text{Ref}\}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_n)}, \quad (6)$$

where n denotes the length of the n-grams (*e.g.*, ROUGE-1 for unigrams, ROUGE-2 for bigrams), and $\text{Count}_{\text{match}}(\text{gram}_n)$ is the number of n-grams in the reference that also appear in the candidate text. ROUGE-N values range from 0 to 1 and a higher ROUGE-N value indicates better performance. In our experimental evaluation, we employ the rouge_scorer() metric from the rouge_score (Lin, 2004) library.

- **ROUGE-L** evaluates the quality of generated text by measuring the longest common subsequence (LCS) between the candidate and reference texts. It is defined as

$$\text{ROUGE-L} = \frac{(1+\beta^2) \cdot \text{Precision} \cdot \text{Recall}}{\text{Recall} + \beta^2 \cdot \text{Precision}}, \quad (7)$$

where $\text{Precision} = \frac{\text{LCS}(X,Y)}{|X|}$ and $\text{Recall} = \frac{\text{LCS}(X,Y)}{|Y|}$, with X and Y denoting the candidate and reference sequences, respectively. β is typically set to favor recall ($\beta = 1.2$ in common settings). ROUGE-L scores range from 0 to 1, with higher values indicating better preservation of the reference's sequence and structure.

- **METEOR** (Metric for Evaluation of Translation with Explicit ORdering) is a metric designed to evaluate the quality of machine-generated text by aligning it to one or more reference texts. It is defined as

$$\text{METEOR} = F_{\text{mean}} \cdot (1 - \text{Penalty}), \quad (8)$$

where $F_{\text{mean}} = \frac{10 \cdot P \cdot R}{R + 9P}$, with P and R denoting unigram precision and recall, respectively. The penalty is a function of the number of chunks in the alignment, designed to penalize disordered matches. METEOR scores range from 0 to 1, with higher scores indicating better alignment with the reference text. In our experimental evaluation, we perform tokenization using the word_tokenize() function and employ the METEOR metric (meteor_score()), both implemented in the NLTK library (Bird et al., 2009).

- **Mean Squared Error (MSE)** measures the average of the squares of the difference between the forecasted value and the actual value. It is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \widehat{y}_i)^2, \quad (9)$$

where N is the size of the test set; y_i and $\widehat{y}_i$ denote the ground truth and predicted score of the i-th data sample in the test set, respectively. MSE value ranges from 0 to positive infinity. A lower MSE value indicates better performance.

- **Mean Absolute Error (MAE)** measures the absolute value of the difference between the predicted value and the actual value. It is defined as

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \widehat{y}_i|, \quad (10)$$

where N is the size of the test set; y_i and $\widehat{y}_i$ denote the ground truth and predicted score of the i-th data sample in the test set, respectively. MAE value ranges from 0 to positive infinity. It emphasizes the ranking order of the prediction instead of the absolute value. A lower MAE value indicates better performance.

- **Pearson Correlation** (PC) is defined as the covariance of the prediction and the ground

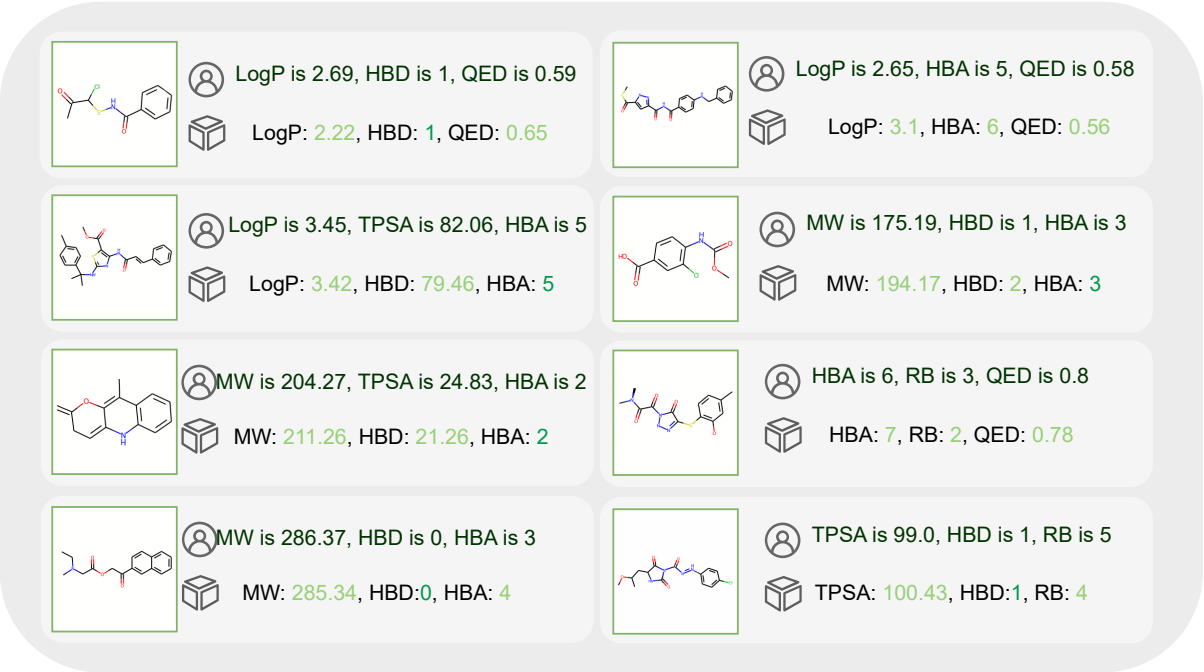


Figure 12: More examples for property2img task.

truth divided by the product of their standard deviations. For two random variables x and y, Pearson Correlation is formally defined as

$$\text{PC} = \frac{\mathbb{E}[(x-\mu_x)(y-\mu_y)]}{\sigma_x \sigma_y}, \quad (11)$$

In the regression task, suppose there are N data points in the test set, y_i is the ground truth of the i-th data sample, $\widehat{y}_i$ is the prediction for i-th data, Pearson Correlation becomes

$$\text{PC} = \frac{\sum_{i=1}^{N}\left((y_i-\mu_y)(\widehat{y}_i-\mu_{\widehat{y}})\right)}{\sigma_y \sigma_{\widehat{y}}}, \quad (12)$$

where $\mu_y = \frac{1}{N}\sum_{j=1}^{N} y_j$ and $\mu_{\widehat{y}} = \frac{1}{N}\sum_{j=1}^{N} \widehat{y}_j$ are mean of ground truth and prediction, respectively. $\sigma_y = \sum_{i=1}^{N}(y_i - \frac{1}{N}\sum_{j=1}^{N} y_j)^2$ and $\sigma_{\widehat{y}} = \sum_{i=1}^{N}(\widehat{y}_i - \frac{1}{N}\sum_{j=1}^{N} \widehat{y}_j)^2$ are the standard deviations of ground truth and prediction, respectively. The value ranges from -1 to 1. A higher Pearson correlation value indicates better performance.

- **Tanimoto similarity** is to measure the similarity between two molecules. Tanimoto similarity is also known as the Jaccard coefficient, *i.e.*, the ratio of their intersection to their union over two chemical fingerprint vectors.

$$\text{sim}(X,Y) = \frac{|\mathbf{b}_X \cap \mathbf{b}_Y|}{|\mathbf{b}_X \cup \mathbf{b}_Y|}, \quad (13)$$

where $\mathbf{b}_X$ is the binary fingerprint vector for the molecule X. Tanimoto distance between two molecules is defined as one minus Tanimoto similarity.

$$\text{Tanimoto-distance}(X,Y) = 1 - \text{sim}(X,Y), \quad (14)$$

Also, given a set of chemical compounds, we are typically interested in their diversity, which is defined based on Tanimoto distance. Specifically, diversity is defined as the average pairwise Tanimoto distance between the molecular fingerprints,

$$\text{diversity}(\mathcal{Z}) = 1 - \frac{1}{|\mathcal{Z}|(|\mathcal{Z}|-1)} \sum_{X,Y\in\mathcal{Z}, X\neq Y} \text{sim}(X,Y), \quad (15)$$

where $\mathcal{Z}$ is the set of generated molecules to evaluate.

- **Tanimoto@1** is to measures the proportion of generated molecules that exactly match the ground-truth molecule in terms of Tanimoto similarity. Specifically, it computes the ratio of generated molecules whose Tanimoto similarity with the corresponding ground-truth

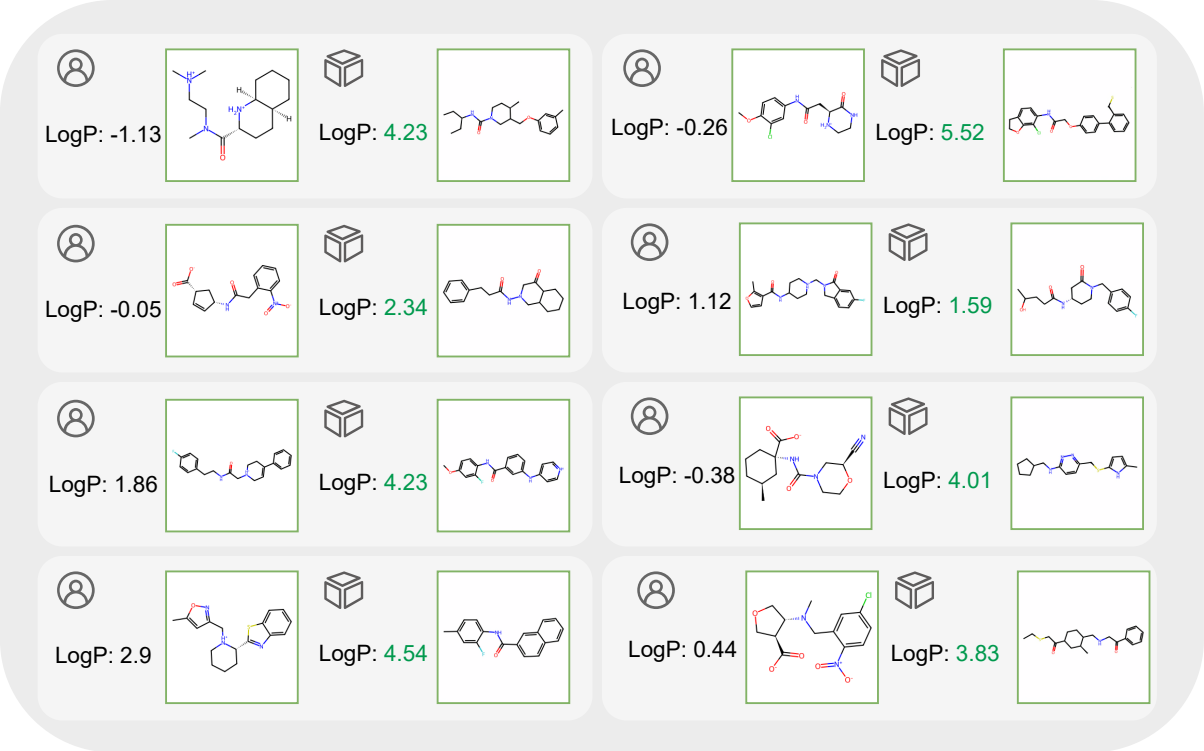


Figure 13: More examples for img2img task.

molecule is equal to 1. It is defined as

$$\text{Tanimoto@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left[\text{Tanimoto}(f_i^{\text{gen}}, f_i^{\text{true}}) = 1\right], \tag{16}$$

where N is the number of generated molecules, f_i^{gen} and f_i^{true} are the Morgan fingerprints of the i-th generated and ground-truth molecules, respectively, and $\mathbb{I}[\cdot]$ is the indicator function. The score ranges from 0 to 1, where a higher Tanimoto@1 indicates better exact matching performance between generated and reference molecules.

- **Increased LogP** is to evaluate molecular optimization performance. It measures the average increase in the LogP of molecules after optimization. For each molecule, the improvement is computed as the difference between the LogP value of the optimized molecule and that of the original molecule. The final score is the mean improvement across all molecule pairs. It is defined as

$$\text{Increased LogP} = \frac{1}{N}\sum_{i=1}^{N} \left(\text{LogP}(m_i^{\text{opt}}) - \text{LogP}(m_i^{\text{orig}})\right), \tag{17}$$

where N is the number of molecule pairs, m_i^{orig} and m_i^{opt} denote the i-th original and optimized molecules, respectively. A higher LogP Improvement value indicates a greater enhancement of the LogP property through the optimization process.

- **Diversity** is a metric used to quantify the structural variety within a set of molecules. It is defined as the average pairwise Tanimoto distance between the Morgan fingerprints of the molecules:

$$\text{Diversity} = \frac{2}{N(N-1)}\sum_{i=1}^{N}\sum_{j=i+1}^{N} \left(1 - \text{Tanimoto}(f_i, f_j)\right), \tag{18}$$

where N is the number of molecules in the set, and f_i and f_j are the Morgan fingerprints of the i-th and j-th molecules, respectively. The Tanimoto similarity measures the overlap between two binary fingerprints, and the distance is computed as $1 -$ Tanimoto. The diversity values range from 0 to 1, with higher values indicating greater chemical diversity.

- **Novelty** evaluates the proportion of generated molecules that are not present in the training

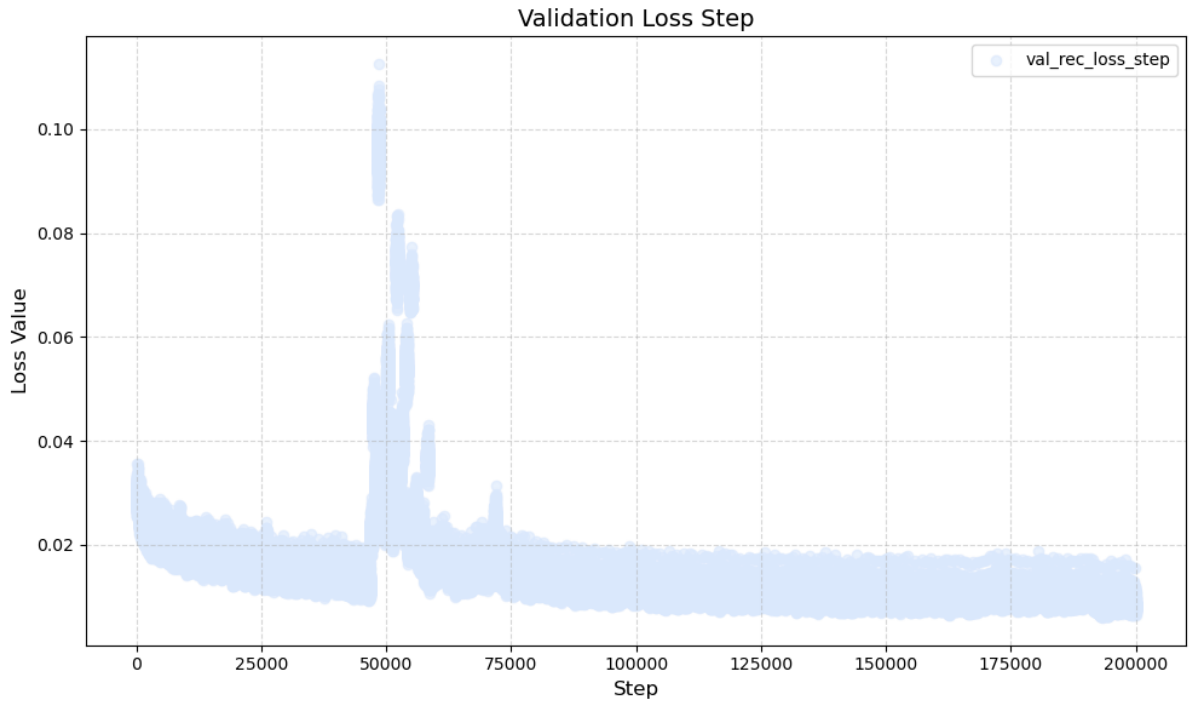


Figure 14: Training curve from start to the best check point, we apply GAN loss after training in E,G, and Z for a period of steps for stability. As shown in the validation loss curve, GAN loss is introduced at 45000 steps and cause the oscillation of validation loss and finally converge to stable result.

set. It reflects the ability of a generative model to produce novel chemical structures, rather than simply memorizing and replicating the training data. It is defined as

$$\text{Novelty} = \frac{|\mathcal{G} \setminus \mathcal{T}|}{|\mathcal{G}|}, \tag{19}$$

where $\mathcal{G}$ denotes the set of generated molecules, and $\mathcal{T}$ denotes the set of molecules in the training set. The numerator counts the number of molecules in $\mathcal{G}$ that are not in $\mathcal{T}$. The score ranges from 0 to 1, with higher values indicating greater novelty.

- **Valid%** is to evaluate the structural and syntactic validity of model outputs in instruction-following tasks involving molecule generation. It measures the proportion of outputs that are both (1) successfully parsed according to a predefined structured format (*i.e.*, instruction-following), and (2) contain syntactically valid SMILES strings, if any are present. It is defined as

$$\text{Valid Rate} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[\text{structured}(o_i) \wedge \text{valid}(o_i)\right], \tag{20}$$

where N is the total number of model outputs, o_i is the i-th output, structured$(\cdot)$ checks whether the output follows the expected structured format, and valid$(\cdot)$ verifies the syntactic validity of any SMILES strings present in the output. The score ranges from 0 to 1, with a higher Valid% indicates better adherence to the required output format and chemical validity.

Also, we conduct statistical testing to check if the improvement is statistically significant.

G Molecular Properties

- **MW**: Molecular Weight (MW) is the sum of atomic masses of all atoms in a molecule (units: g/mol or Da). It influences physicochemical properties such as solubility, diffusion rate, and bioavailability. MW should be below 500 Da for optimal oral bioavailability.
- **LogP**: Octanol-water Partition Coefficient (LogP) assesses the solubility and synthetic accessibility of a chemical compound. The LogP score of a molecule ranges from $-\infty$ to $+\infty$.
- **TPSA**: Topological Polar Surface Area

(TPSA) quantifies the surface area contributed by polar atoms, typically oxygen and nitrogen, including their attached hydrogens. The theoretical TPSA ranges from 0 to several hundreds or even thousands of $Å^2$ for highly polar or large biomolecules.

- **HBD**: Hydrogen Bond Donor (HBD) counts the number of polar functional groups (*e.g.*, -OH, -NH) in a molecule that can donate hydrogen atoms to form hydrogen bonds.

- **HBA**: Hydrogen Bond Acceptor (HBA) counts the number of atoms (*e.g.*, O, N, S, F) in a molecule capable of accepting hydrogen bonds via lone electron pairs. Typical small-molecule drugs containing 2–10 HBA sites.

- **RB**: Rotatable Bond (RB) counts the number of non-ring single bonds (*e.g.*, C-C, C-O, C-N) in a molecule that allow free rotation at room temperature. Optimal drug-like compounds typically contain $\leq$10 rotatable bonds (RB).

- **QED**: Quantitative Estimate of Drug-likeness (QED) is an integrative score to evaluate compounds' favorability to become a drug. The QED value ranges from 0 to 1. A higher value is more desirable.

H Implementation Details

For the first training stage, Mol-VQGAN is trained on 8 NVIDIA A800×80G GPUs for two epochs. The batch size is set to 16 and the base learning rate is set to 4.5e-06. We use Adam (Kingma and Ba, 2014) as optimizer and $\mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{perceptual}$ as monitor to validate the model and save the best checkpoint during 2-epoch training. For the second training stage, ChemMLLM is trained on 8 NVIDIA A800×80G GPUs for three epochs. The batch size is set to be 16, the base learning rate is set to be 2e-5 and z-loss weight is set to be 1e-5. We use AdamW (Loshchilov and Hutter, 2017) as optimizer and employ mixed-precision training, utilizing brain floating point 16 precision (bf16) (Kalamkar et al., 2019) for forward propagation and 32-bit floating point precision (fp32) for backward propagation to balance the training efficiency and stability. To handle distributed training, we apply PyTorch Fully Shared Data Parallel (FSDP) (Zhao et al., 2023) strategy. We train three variants of ChemMLLM for different tasks.

Mol-VQGAN Training For Mol-VQGAN training code, we use the official implementation code of original VQGAN (Esser et al., 2021). Specifically, we utilize the synthesized image datasets to let the original Chameleon VQGAN learn how to understand and generate molecule images. We sample 1,000,000 molecule images from PubChem synthesized by RDKit (Landrum et al., 2006) and combine them with all images synthesized in 5 downstream tasks as training dataset for Mol-VQGAN. All parameters of Encoder, Decoder and codebook of VQGAN are trained. We first train VQGAN on this 1 million-level molecule image dataset for two epochs and save the best check point according to $\mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{perceptual}$. After the initial training, we find that it can not reconstruct image well on the dataset with less data size like img2caption dataset, so we do continuous training based on the best checkpoint using small-size datasets. Specifically, we continue training the 1-million best check point on img2caption images for five epochs and save the best checkpoint. Finally, we get the well-trained Mol-VQGAN to tokenize molecule images.The original VQGAN will result in unclear and distorted images when encoding and decoding molecule images. After training, Mol-VQGAN can encode and decode molecule image almost the same with original image. Several examples are shown in Figure 15. The validation curve from start to the best checkpoint is shown in Figure 14.

ChemMLLM Supervised FineTuning Training For ChemMLLM training code, we utilize LuminamGPT framework (Liu et al., 2024). We train three variants ChemMLLM ChemMLLM-pro2img and ChemMLLM-img2img.

For ChemMLLM, we train it on img2caption, img2property, img2smiles and property2img datasets. Apart from the four datasets, we also train it on SMILES molecule image generation (smiles2img) and text based drug design dataset (caption2img), which is the inverse task of img2smiles and img2caption. smiles2img requires model to receive a SMILES and output the corresponding image and caption2img ask model to output molecule image according to molecule caption. We train ChemMLLM on the six datasets for three epochs; For property2img task, we train and evaluate ChemMLLM-pro2img. We find that directly training the model on raw data can not achieve good performance, so we do data augmentation by rotating images by 90°,180°, and 270° so as to generate augmented data that is four times the size of the

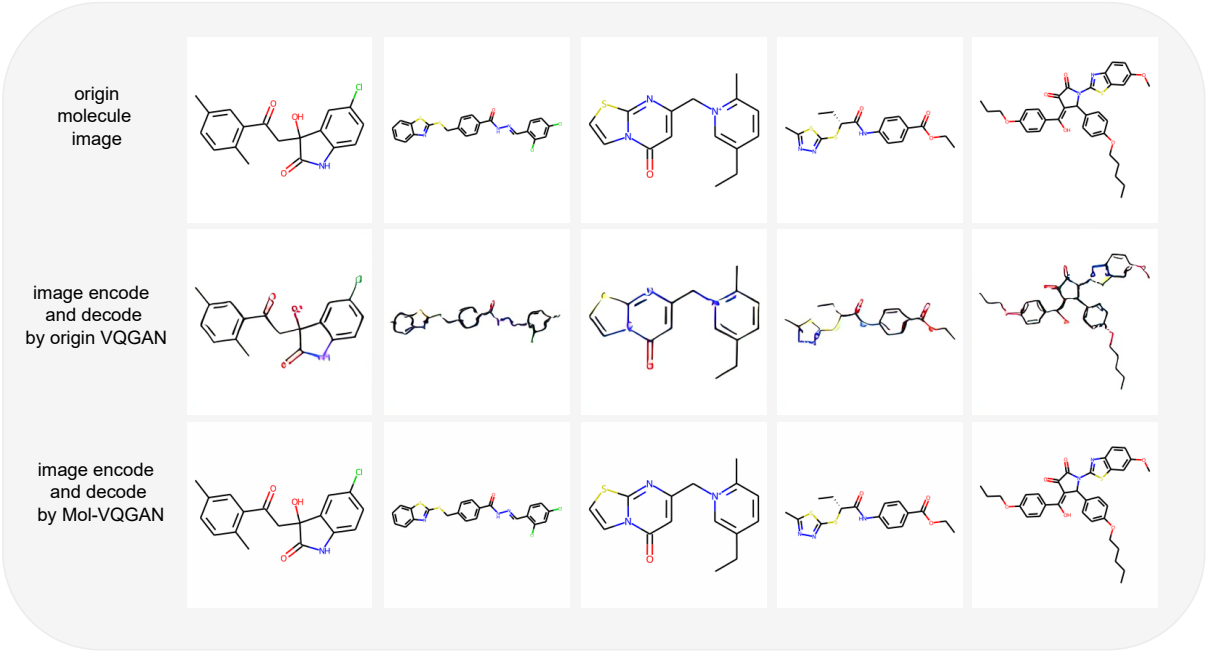


Figure 15: Examples of origin VQGAN and Mol-VQGAN. The origin images is shown in the first row. As shown in the second row, the origin VQGAN can not encode and decode molecule image clearly and accurately, resulting in distorted atoms and bonds which are hard to distinguish. As shown in the third row, our well-trained Mol-VQGAN can encode and decode molecule images with clear atoms and bonds, which is almost the same as the origin molecule images.

original dataset. We train ChemMLLM-pro2img on the augmented dataset for two epochs; For img2img task, we train and evaluate ChemMLLM-img2img. we train it on img2img dataset for two epochs. The training hyper-parameters primarily follow Lumina-mGPT (Liu et al., 2024).

The details of data information for training different ChemMLLM variants are shown in Table 16. The details of training settings for training different ChemMLLM variants are shown in Table 17.

Model	task	# Training	# Test
ChemMLLM	img2caption	69,799	3,300
	img2property	95,000	5,000
	img2smiles	95,000	5,000
	property2img	95,000	5,000
	smiles2img	95,000	5,000
	caption2img	72,143	3301
ChemMLLM-pro2img	property2img	380,000	20,000
ChemMLLM-img2img	img2img	157,673	17,520

Table 16: Detailed dataset information for training different ChemMLLM variants.

Settings	ChemMLLM	ChemMLLM-pro2img	ChemMLLM-img2img
z-loss weight	1e-05	1e-05	1e-05
warmup epochs	0.01	0.01	0.01
learning rate	2e-05	2e-05	2e-05
weight decay	0.1	0.1	0.1
drop rate	0.05	0.05	0.05
total bacth size	$16 \times 8 \times 1$	$16 \times 8 \times 1$	$8 \times 4 \times 1$
GPUs for training	8×A800 (80G)	8× A800 (80G)	4× A800 (80G)
GPUs hours(h)	65	30.6	25.4

Table 17: Detailed training settings for training different ChemMLLM variants.

I Ablation Study

We conduct ablation study on property2img task with a focus on evaluating the impact of Mol-VQGAN training and data augmentation. The effectiveness of the generated molecular images is assessed through the Pearson correlation coefficients between the predicted and ground truth values of several molecular properties, including MW, LogP, TPSA, Hbd, Hba, Rb, and QED. we use a subset of 200 samples of he property2img task and do 5-shot evaluation.

The result is shown in Table 18. When both fine-tuned VQGAN and data augmentation are employed, the model achieves the highest correlation scores across all evaluated properties. Notably, the correlations for MW (0.71), TPSA (0.71), Hba (0.66), and Rb (0.62) indicate that the generated images capture molecular structure and features that align well with the original textual descriptions. This demonstrates the efficacy of our approach in enhancing the semantic fidelity and chemical rele-

data augmentation	VQGAN	MW Pearson (↑)	LogP Pearson (↑)	TPSA Pearson (↑)	Hbd Pearson (↑)	Hba Pearson (↑)	Rb Pearson (↑)	QED Pearson (↑)
✓	✓	**0.71**	**0.42**	**0.71**	**0.45**	**0.66**	**0.62**	0.34
×	✓	0.46	0.04	0.40	0.08	0.53	0.17	0.26
×	×	0.55	0.06	-0.05	0.06	-0.06	0.31	**0.35**

Table 18: Ablation study.

vance of the generated visual representations. Removing data augmentation while retaining the fine-tuned VQGAN leads to a significant drop in performance, especially for properties such as LogP (reduced to 0.04) and TPSA (0.40), highlighting the importance of data augmentation in improving the model's generalization and robustness during training. The performance further deteriorates when both fine-tuning and data augmentation are removed. In this setting, the model yields the lowest correlations, with some properties (*e.g.*, TPSA at -0.05 and Hba at -0.06) exhibiting negative correlation, suggesting that the general VQGAN trained on natural images fails to preserve critical molecular features necessary for reliable property prediction.

In summary, these results clearly demonstrate that both fine-tuning Mol-VQGAN and applying data augmentation play complementary and crucial roles in enhancing the quality and accuracy of chemical multimodal tasks.