Editing Multimodal Molecule Language Models

Anonymous ACL submission

Abstract

Understanding multimodal molecular knowledge is crucial for advancing biomedicine, chemistry, and materials science. Molecule language models (MoLMs) have become powerful tools in these domains, integrating structural representations (e.g., SMILES strings, 2D graphs) with contextual descriptions (e.g., physicochemical properties, biomedical applications). However, MoLMs can encode and propagate inaccuracies due to low-quality training data or malicious manipulation. While model editing has been explored for general-domain AI, its application to MoLMs remains uncharted, presenting unique challenges due to the multifaceted and interdependent nature of molecular knowledge. In this paper, we take the first step toward MoLM editing for two critical tasks: molecule-to-caption generation and caption-to-molecule generation. To address molecule-specific challenges, we propose MolEdit, a novel framework that enables targeted modifications while preserving unrelated molecular knowledge. To systematically evaluate editing performance, we introduce MEBench, a comprehensive benchmark assessing multiple dimensions, including reliability, locality, and generality. Extensive experiments on MEBench highlight the distinct challenges of MoLM editing and demonstrate MolEdit's superiority over existing methods.

1 Introduction

Understanding molecular knowledge is crucial across various scientific fields, such as biomedicine (Zhang et al., 2024b; Pei et al., 2024a), chemistry (Liao et al., 2024; Xiao et al., 2024), and materials science (Lei et al., 2024). Pre-trained and fine-tuned on diverse multimodal data, molecule language models (MoLMs) encode multimodal molecular knowledge, encompassing structural representations (e.g., SMILES strings, 2D graphs) and contextual descriptions (e.g., physicochemical properties, biomedical applications) (Su et al., 2022; Pei et al., 2024b; Cao et al., 2023). With rich knowledge integrated, MoLM encoders process input representations, while decoders generate outputs across modalities, enabling key applications such as molecule-to-caption generation and caption-to-molecule design (Luo et al., 2023). However, during knowledge integration, inaccurate or misleading information can be introduced, either from low-quality training data (Deng et al., 2024b) or because of malicious knowledge manipulation (Chen et al., 2024), posing risks in downstream applications. For instance, a compromised MoLM might incorrectly describe *Naphthalene*—a highly carcinogenic organic molecule—as "a benign human metabolite", potentially leading to critical errors in drug discovery pipelines. Hence, there is a pressing need to refine MoLMs to correct inaccurate or misleading knowledge, as shown in Figure 1. Recently, model editing has emerged as an efficient approach to modifying specific knowledge while preserving other information (Wang et al., 2024b; Zhang et al., 2024a; Mazzia et al., 2024). It

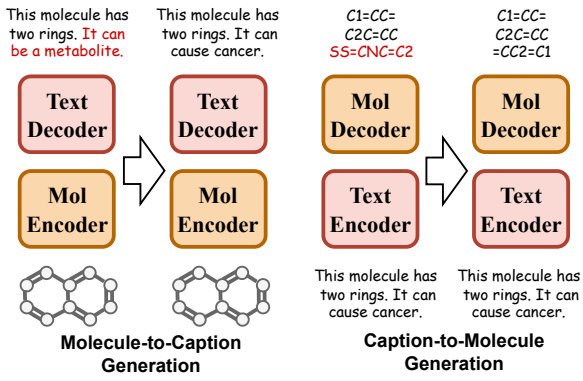

Figure 1: An illustration of MoLM editing for two tasks: correcting inaccurate captions in *molecule-to-caption generation* and fixing mismatched or invalid molecules in *caption-to-molecule generation*.

has been widely applied to general-domain large language models (De Cao et al., 2021; Meng et al., 2022b) and multimodal language models (Huang et al., 2024b; Cheng et al., 2023).

However, no existing work has explored how model editing can be adapted to MoLMs, which presents two inherent molecule-specific challenges. First, molecular knowledge is inherently *multi-faceted* (Cao et al., 2023)—a molecule consists of multiple functional groups, while its caption comprises distinct descriptive components. Each of these elements represents a specific aspect of molecular properties and may exhibit varying sensitivities to editing. Consequently, modifying multifaceted molecular knowledge poses the risk of over-editing certain aspects while under-editing others (Javadi, 2024; Zheng et al., 2023). Second, shared functional groups and contextual descriptions create *interdependencies* among molecules, so editing one molecule's knowledge can unintentionally affect others with similar features. This makes it difficult to ensure edits remain localized, violating the principle of locality—where modifications should only impact the intended target.

To address these challenges, we propose MolEdit, the first framework for editing multimodal MoLMs in caption generation and molecule design tasks. Our approach enables precise, targeted updates to compositional molecular knowledge while preserving unrelated information. To address the first challenge, we design a *Multi-Expert Knowledge Adapter* (MEKA) that directs different facets of molecular knowledge to specialized editing experts, enabling fine-grained control over multifaceted updates. For the second challenge, we introduce an *Expertise-Aware Editing Switcher* (EAES), which maintains a memory bank of edited molecular knowledge. The knowledge adapter is activated only when a new input exhibits high expertise overlap with stored edits, minimizing unintended interference with unrelated knowledge. Furthermore, since incorrect outputs can arise from interactions between different modalities, our approach edits both MoLM encoders and decoders to ensure comprehensive refinement.

To enable systematic evaluation, we introduce MEBench, the first benchmark for editing MoLMs, which rigorously assesses reliability (editing accuracy), locality (preservation of unrelated knowledge), and generality (consistency across varied textual descriptions of the same concept). Comprehensive experiments on MEBench demonstrate that MolEdit outperforms existing knowledge editing methods. Overall, our key contributions are as follows:

- **Problem Formulation**: We conduct the first systematic study on model editing for MoLMs, identifying and formalizing molecule-specific challenges.
- **Benchmark Construction**: We introduce MEBench, a comprehensive evaluation benchmark that rigorously assesses three key dimensions: reliability, locality, and generality.
- **Framework Design**: We propose MolEdit, a novel editing framework incorporating a *Multi-Expert Knowledge Adapter* and an *Expertise-Aware Editing Switcher* to address molecule-specific challenges.
- **Experimental Evaluation**: Extensive experiments on MEBench demonstrate that MolEdit outperforms existing knowledge editing methods across all three evaluation dimensions.

2 Related Work

Molecule Language Model. Inspired by general-domain language models, molecule language models (MoLMs) learn rich molecular representations for various tasks (Liu et al., 2024b; Pei et al., 2024a). Given the multimodal nature of molecular knowledge, existing MoLMs align heterogeneous inputs during pretraining (Liu et al., 2023a; Pei et al., 2023). While some frameworks adopt a unified generative approach (Fang et al., 2023; Zeng et al., 2022; Christofidellis et al., 2023; Zhao et al., 2023), we focus on contrastive methods (Su et al., 2022; Luo et al., 2023; Liu et al., 2023b; Li et al., 2024b; Liu et al., 2024a; Luo et al., 2024), which employ separate encoders for each modality, enabling greater flexibility in handling diverse data sources. These models are then fine-tuned for specialized tasks such as molecule-to-caption and caption-to-molecule generation (Su et al., 2022; Li et al., 2024a; Gong et al., 2024; Liu et al., 2024c). However, both pretraining and fine-tuning can introduce inaccurate or misleading knowledge (Dong et al., 2022; Huang et al., 2024a), highlighting the need for model editing.

Model Editing. Model editing seeks to efficiently and precisely modify specific factual knowledge within AI systems (Mazzia et al., 2024). Gradient-based approaches (Sinitsin et al., 2020;

Ni et al., 2023), including meta-learning (Cheng et al., 2024; Mitchell et al., 2021) and locate-then-edit methods (Meng et al., 2022a,b), directly update model parameters but risk unintended alterations to unrelated knowledge. In contrast, external memorization-based techniques mitigate this issue by isolating new and existing knowledge, employing methods such as counterfactual models (Mitchell et al., 2022), adapters (Hartvigsen et al., 2024; Huang et al., 2023; Wang and Li, 2024), and textual context-based edits (Zheng et al., 2023; Madaan et al., 2022). However, these approaches struggle to handle the multifaceted and interdependent nature of molecular knowledge. To address this, MolEdit introduces a multi-expert knowledge adapter to capture diverse molecular expertise and an expertise-aware editing switcher to ensure edits apply only to highly relevant inputs.

3 Preliminary

Notations. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represent a molecular graph, where $\mathcal{V}$ denotes the set of atoms (nodes) and $\mathcal{E}$ represents covalent bonds (edges). Each molecular graph consists of N_g subgraphs $g_{i}{}_{i=1}^{N_g}$, each corresponding to a functional group. The molecular structure can also be expressed as a SMILES string, denoted as $\mathcal{S}$. Additionally, each molecule is associated with a caption $\mathcal{T} = t^1, ..., t^{N_t}$ containing N_t textual descriptions.

Molecule Language Model. Given molecular representations $\mathcal{G}$, $\mathcal{S}$, and captions $\mathcal{T}$, molecule language models (MoLMs) learn aligned cross-modal representations by utilizing a structure and a text encoder during pretraining. A task-specific decoder is then appended for downstream generation tasks.

Caption Generation. In the caption generation task, a pretrained text decoder is appended to the structure encoder and fine-tuned to generate captions $\mathcal{T}$ given a molecular representation $\mathcal{G}$, $\mathcal{S}$. We denote the fine-tuned MoLMs for this task as f_{cap}.

Molecule Generation. Similarly, in the molecule generation task, a pretrained molecule generation decoder is appended to the text encoder and fine-tuned to generate SMILES representations $\mathcal{S}$ given a caption $\mathcal{T}$ [1]. We denote the fine-tuned MoLMs for this task as f_{gen}.

[1]We follow the definition of existing MoLMs (Luo et al., 2023, 2024; Edwards et al., 2022)

4 Editing Molecule Language Model

In this section, we first introduce the task of editing MoLMs for molecule and caption generation (§4.1). We then present MEBench, the first benchmark for evaluating MoLM editing (§4.2), followed by MolEdit, a novel framework designed to address molecule-specific challenges (§4.3).

4.1 Task Definition

Editing Caption Generation. MoLMs may produce inaccurate captions that require correction. To assess editing effectiveness, we evaluate caption generation along two dimensions. First, Reliability measures how well the edited captions align with expert-curated ground truth. Given a dataset $\mathcal{D}^{\mathcal{T}}$edit containing molecules with initially incorrect captions, we compute the semantic similarity between the captions generated by the edited MoLMs, $\tilde{f}_{cap}$, and the target captions $\mathcal{T}$:

$$\mathcal{M}^{\mathcal{T}}_{rel} = \mathbb{E}_{(\mathcal{G},\mathcal{S},\mathcal{T})\sim\mathcal{D}^{\mathcal{T}}_{\text{edit}}}(\mathbf{SIM}_T(\tilde{f}_{cap}(\mathcal{G},\mathcal{S}),\mathcal{T})), \tag{1}$$

where $\mathbf{SIM}_T$ is the metric to measure text similarity. Secondly, **Locality** preserves existing knowledge by minimizing deviations in unedited captions. For a dataset $\mathcal{D}^{\mathcal{T}}_{\text{loc}}$ with knowledge unrelated to $\mathcal{D}^{\mathcal{T}}_{\text{edit}}$, we compare outputs before and after editing:

$$\mathcal{M}^{\mathcal{T}}_{loc} = \mathbb{E}_{(\mathcal{G},\mathcal{S})\sim\mathcal{D}^{\mathcal{T}}_{\text{loc}}}(\mathbf{SIM}_T(\tilde{f}_{cap}(\mathcal{G},\mathcal{S}), f_{cap}(\mathcal{G},\mathcal{S}))). \tag{2}$$

Editing Molecule Generation. MoLMs can also generate invalid molecule SMILES that necessitate correction. Similar to editing caption generation, we evaluate the **Reliability** and **Locality** with edited MoLMs for molecule generation $\tilde{f}_{gen}$:

$$\mathcal{M}^{\mathcal{S}}_{rel} = \mathbb{E}_{(\mathcal{S},\mathcal{T})\sim\mathcal{D}^{\mathcal{S}}_{\text{edit}}}(\mathbf{SIM}_G(\tilde{f}_{gen}(\mathcal{T}),\mathcal{S})), \tag{3}$$

$$\mathcal{M}^{\mathcal{S}}_{loc} = \mathbb{E}_{(\mathcal{T})\sim\mathcal{D}^{\mathcal{S}}_{\text{loc}}}(\mathbf{SIM}_G(\tilde{f}_{gen}(\mathcal{T}), f_{gen}(\mathcal{T}))), \tag{4}$$

where $\mathcal{D}^{\mathcal{S}}_{\text{edit}}$ contain molecules requiring knowledge editing and $\mathcal{D}^{\mathcal{S}}_{\text{loc}}$ ought to remain unchanged during editing. $\mathbf{SIM}_G$ is the metric to measure molecule similarity. Additionally, descriptions with the same semantic meaning, despite differences in phrasing, should generate the same molecule. To evaluate this, we assess the model's output consistency for equivalent inputs (e.g., rephrased descriptions) using a **generality** dataset, as shown below:

$$\mathcal{M}^{\mathcal{S}}_{gen} = \mathbb{E}_{\substack{(\mathcal{T}_r)\sim\mathcal{N}(\mathcal{T})\\(\mathcal{S},\mathcal{T})\sim\mathcal{D}^{\mathcal{S}}_{\text{edit}}}}(\mathbf{SIM}_G(\tilde{f}_{gen}(\mathcal{T}_r),\mathcal{S})), \tag{5}$$

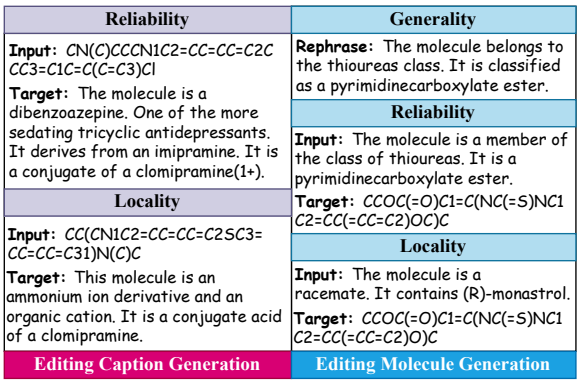

Figure 2: A sample illustration of MEBench. It includes three evaluation dimensions for two tasks: Reliability (molecules requiring editing), Locality (similar but untargeted molecules), and Generality (rephrased captions of the Reliability inputs).

where the $\mathcal{N}(\cdot)$ denotes the generalization set of each description.

4.2 Benchmark Construction

This subsection provides a brief overview of the MEBench construction process. An illustration of MEBench samples is also provided in Figure 2.

4.2.1 Editing Caption Generation

Reliability. To assess the effectiveness of knowledge editing, we construct a caption reliability dataset, $\mathcal{D}^{\mathcal{T}}_{\text{edit}}$. We first identify suboptimal entries shared across multiple MoLMs within the widely used CheBI-20 dataset for caption generation, using its ground truth captions as the editing targets. Additionally, to enable a more fine-grained evaluation of editing capabilities (as discussed in Section 5.4), the targets are decomposed into distinct descriptions, each capturing a specific aspect of molecular knowledge.

Locality. To assess the ability of MoLMs to preserve existing knowledge, we construct a caption locality dataset, $\mathcal{D}^{\mathcal{T}}_{\text{loc}}$. Specifically, we extract high-accuracy entries from the CheBI-20 training set to serve as the basis for this dataset. Since model editing is more likely to affect knowledge that is semantically similar to the editing target, we select locality samples with high similarity to those in the reliability dataset, making the editing task more challenging. Similar to UnKEBench (Deng et al., 2024a), both the reliability and locality datasets contain unstructured knowledge, as molecule captions are complex and involve multiple entities.

4.2.2 Editing Molecule Generation

Reliability. To assess the effectiveness of knowledge editing in improving molecule SMILES generation, we construct the molecule reliability dataset, $\mathcal{D}^{\mathcal{S}}_{\text{edit}}$. Following a similar approach to caption editing dataset construction, we identify suboptimal entries shared across multiple MoLMs within the CheBI-20 dataset. The ground truth SMILES representations in the dataset serve as the editing targets.

Locality. To construct the generation locality dataset, $\mathcal{D}^{\mathcal{S}}_{\text{loc}}$, we select high-accuracy entries from multiple MoLMs within the CheBI-20 training set. To ensure a rigorous evaluation of the model's ability to preserve existing knowledge, we choose entries with the highest semantic similarity to those in the reliability dataset.

Generality. To assess a model's ability to generalize across different phrasings, we construct the generality dataset, $\mathcal{N}(\mathcal{T})$. This dataset consists of rephrased captions from the molecule reliability dataset, evaluating whether semantically equivalent descriptions consistently generate the same molecular structures.

4.3 Methodology

To tackle the challenges of editing MoLMs, we propose MolEdit, a novel framework designed to enable localized updates to multifaceted molecular knowledge. As shown in Figure 3, MolEdit comprises two key components: (1) *Multi-Expert Knowledge Adapter* - This module disentangles and customizes the editing of diverse molecular knowledge (e.g., functional groups in molecules, descriptive elements in captions) by dynamically routing them to specialized editing experts via a Mixture-of-Experts (MoE) architecture. (2)*Expertise-Aware Editing Switcher* - This component ensures edits are applied only to highly relevant inputs by leveraging a memory bank of expertise embeddings, activating modifications only when the input exhibits substantial overlap with stored edits.

4.3.1 Multi-Expert Knowledge Adapter

Since errors can originate from both input and output modalities (Cheng et al., 2023), MolEdit enables knowledge editing by wrapping selected layers in both the encoder and decoder with adapters, allowing localized parameter updates. Taking the molecule generation editing task as an example, when editing the encoder at layer l, we employ P distinct experts to perform customized edits for

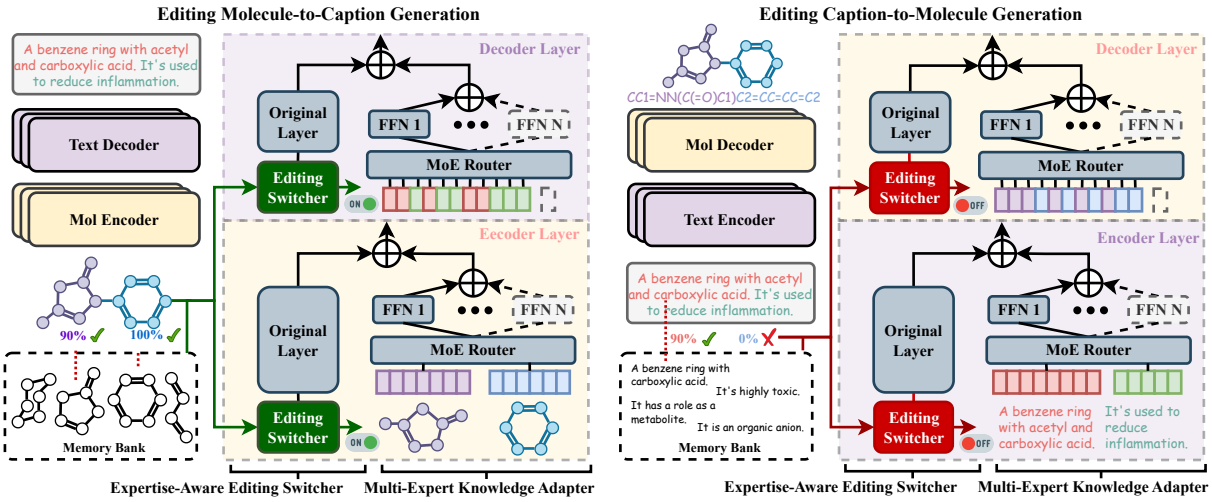

Figure 3: An overview of MolEdit to edit MoLMs for molecule/caption generation by modifying a chosen layer in either the encoder and decoder. It is composed of two components: (1) Multi-Expert Knowledge Adapter (MEKA) and (2) Expertise-Aware Editing Switcher (EAES). Specifically, MEKA utilizes expertise-wise MoE for encoder and token-wise MoE for decoder to route expertise to different editing experts (instantiated as FFN). EAES stores edited knowledge expertise (functional groups/descriptions) and activates MEKA only when all input expertise finds a similar match in its memory bank during inference.

different descriptions in the input. Each expert is instantiated as a feed-forward network (FFN) layer. To achieve this, a gating function dynamically routes the $(l-1)$-th layer embeddings of each description to the appropriate expert, illustrated as,

$$\mathcal{G}^n = \text{top}_k\left(\text{softmax}\left(\mathbf{W}_g \cdot \Sigma_{i\in t_n}(z_i^{l-1})/|t^n| + \epsilon\right)\right), \tag{6}$$

where t^n is the n-th description and z_i^{l-1} is the embedding of i-th token in t^n at $(l-1)$-th layer. $\mathbf{W}_g$ is the trainable weights in gate decision, while ϵ denotes the noise term. The $\text{top}_k(\cdot)$ operator zeros out all but the top-k values. After getting the gate decision vector $\mathcal{G}^n$ of the n-th description, the corresponding output is generated through a weighted aggregation of each expert's computation on z_i^l,

$$z_i^l = f^l(z_i^{l-1}) + \sum_{p=1}^{P} \mathcal{G}_p^n \cdot \mathbf{W}_p \cdot z_i^{l-1}, \quad i \in t^n, \tag{7}$$

where $\mathbf{W}_p$ is the trainable weights. When editing the decoder at layer l', the ground truth expertise segmentation is unavailable. As a result, the MoE adapter is applied to each token, which is expected to route tokens associated with different expertise to the appropriate experts, demonstrated as,

$$\begin{aligned} \mathcal{G}^i &= \text{top}_k\left(\text{softmax}\left(\mathbf{W}_g^{'} \cdot z_i^{l'-1} + \epsilon\right)\right), \\ z_i^{l'} &= f^{l'}(z_i^{l'-1}) + \lambda \sum_{p=1}^{P} \mathcal{G}_p^i \cdot \mathbf{W}_p^{'} \cdot z_i^{l'-1}. \end{aligned} \tag{8}$$

Similarly, for editing caption generation, the input molecule is composed of multiple functional groups, each representing a distinct molecule expertise. We route by functional groups during encoder editing and by token during decoder editing due to the lack of predefined expertise in decoder.

4.3.2 Expertise-Aware Editing Switcher

In order to minimize unintended interference with untargeted molecules that share a few functional groups or descriptions, we design a expertise-aware switching mechanism that only allows activation of the knowledge adapters for molecules with high expertise overlap with edited molecules. Specifically, we keep an expertise-based memory bank that stores the expertise-wise knowledge through the form of encoder embeddings of the edit queries:

$$\mathcal{Z} = \{\bar{z}_{n_j}\}, \quad \bar{z}_{n_j} = \Sigma_i^{i\in t_{n_j}}(z_i^{enc})/|t_{n_j}|, \tag{9}$$

where z_i^{enc} represents the encoder embedding of the i-th token (node) within the n_j-th description (functional group) of the j-th sample. During inference, the switcher compares each expertise in the input to the expertise of stored edits in the memory bank $\mathcal{Z}$. The adapter is activated only if all expertise distances are below a threshold ϵ:

$$z_i^l = \begin{cases} \text{MolEdit}\left(z_i^{l-1}\right) & \text{if } \max_n\left(d\left(\bar{z}_n, \mathcal{Z}\right)\right) < \epsilon, \\ f^l\left(z_i^{l-1}\right) & \text{otherwise,} \end{cases} \tag{10}$$

where $d(\cdot)$ is a distance function.

		MoMu			MolFM		
		BLEU-4↑	LEV↓	MACCS↑	BLEU-4↑	LEV↓	MACCS↑
Reliability	FT (Encoder)	0.758 (±0.022)	22.974 (±3.224)	0.987 (±0.018)	**0.983** (±0.000)	**2.015** (±0.018)	0.998 (±0.001)
	FT (Decoder)	0.711 (±0.002)	29.936 (±0.112)	0.938 (±0.002)	0.894 (±0.035)	11.560 (±4.244)	0.977 (±0.013)
	FT (All)	0.781 (±0.000)	19.940 (±0.035)	**1.000** (±0.000)	0.977 (±0.011)	2.554 (±1.091)	0.995 (±0.001)
	MEND	0.802 (±0.019)	21.079 (±1.360)	0.834 (±0.011)	0.789 (±0.003)	23.547 (±0.125)	0.869 (±0.003)
	GRACE	0.718 (±0.000)	28.331 (±0.025)	0.938 (±0.000)	0.770 (±0.001)	11.464 (±15.122)	0.987 (±0.004)
	MolEdit	**0.953** (±0.025)	**4.667** (±2.154)	0.989 (±0.008)	0.975 (±0.003)	2.862 (±0.479)	**1.000** (±0.000)
Locality	FT (Encoder)	0.829 (±0.001)	18.786 (±0.089)	0.881 (±0.008)	0.829 (±0.001)	19.622 (±0.154)	0.943 (±0.005)
	FT (Decoder)	0.881 (±0.001)	12.562 (±0.094)	0.936 (±0.003)	0.725 (±0.001)	31.195 (±0.203)	0.826 (±0.002)
	FT (All)	0.891 (±0.004)	11.756 (±0.413)	0.949 (±0.005)	0.803 (±0.009)	21.985 (±1.189)	0.909 (±0.006)
	MEND	0.912 (±0.028)	9.512 (±3.291)	0.940 (±0.030)	0.833 (±0.020)	17.581 (±1.750)	0.937 (±0.014)
	GRACE	0.859 (±0.000)	15.732 (±0.005)	0.937 (±0.000)	0.757 (±0.001)	30.112 (±0.121)	0.894 (±0.015)
	MolEdit	**0.980** (±0.007)	**1.853** (±0.727)	**0.997** (±0.000)	**0.918** (±0.034)	**9.167** (±3.963)	**0.991** (±0.007)
Generality	FT (Encoder)	0.526 (±0.013)	55.008 (±0.359)	0.629 (±0.012)	0.727 (±0.021)	30.209 (±1.162)	0.697 (±0.051)
	FT (Decoder)	0.590 (±0.011)	48.795 (±1.465)	0.753 (±0.021)	0.665 (±0.002)	38.035 (±0.172)	0.649 (±0.023)
	FT (All)	0.586 (±0.006)	47.872 (±1.192)	0.745 (±0.013)	0.713 (±0.002)	30.755 (±2.094)	0.691 (±0.017)
	MEND	0.707 (±0.025)	30.551 (±2.379)	0.721 (±0.009)	0.731 (±0.000)	30.945 (±0.513)	0.678 (±0.052)
	GRACE	0.703 (±0.000)	32.574 (±0.000)	0.913 (±0.000)	0.645 (±0.000)	44.521 (±0.732)	0.892 (±0.029)
	MolEdit	**0.842** (±0.028)	**15.609** (±2.304)	**0.917** (±0.020)	**0.796** (±0.020)	**23.314** (±1.576)	**0.895** (±0.044)

Table 1: Main results on MEBench for editing MoMu and MolFM in molecule generation, evaluated across three dimensions: Reliability, Locality, and Generality. Each dimension uses three metrics: BLEU-4, LEV, and MACSS. The best and second-best results are shown in **bold** and underlined, respectively. FT denotes fine-tuning.

		MoMu			MolFM		
		BLEU-2↑	METEOR↑	ROUGE-1↑	BLEU-2↑	METEOR↑	ROUGE-1↑
Reliability	FT (Encoder)	0.334 (±0.005)	0.365 (±0.006)	0.472 (±0.007)	0.321 (±0.014)	0.346 (±0.019)	0.448 (±0.023)
	FT (Decoder)	0.784 (±0.017)	0.813 (±0.014)	0.854 (±0.013)	0.916 (±0.000)	0.937 (±0.000)	0.958 (±0.000)
	FT (All)	0.886 (±0.034)	0.907 (±0.030)	0.925 (±0.024)	0.921 (±0.001)	0.941 (±0.001)	0.960 (±0.001)
	MEND	0.557 (±0.034)	0.569 (±0.024)	0.631 (±0.022)	0.627 (±0.025)	0.652 (±0.014)	0.715 (±0.007)
	GRACE	0.928 (±0.000)	0.947 (±0.000)	0.965 (±0.000)	0.928 (±0.000)	0.947 (±0.000)	0.965 (±0.000)
	MolEdit	**0.978** (±0.006)	**0.978** (±0.007)	**0.982** (±0.005)	**0.977** (±0.006)	**0.979** (±0.004)	**0.983** (±0.004)
Locality	FT (Encoder)	0.511 (±0.006)	0.536 (±0.010)	0.604 (±0.006)	0.491 (±0.054)	0.508 (±0.065)	0.587 (±0.052)
	FT (Decoder)	0.637 (±0.023)	0.653 (±0.030)	0.699 (±0.023)	0.669 (±0.000)	0.688 (±0.000)	0.732 (±0.000)
	FT (All)	0.625 (±0.066)	0.637 (±0.063)	0.688 (±0.056)	0.656 (±0.004)	0.671 (±0.006)	0.721 (±0.007)
	MEND	0.615 (±0.010)	0.633 (±0.011)	0.676 (±0.016)	0.844 (±0.005)	0.856 (±0.006)	0.873 (±0.005)
	GRACE	0.875 (±0.013)	0.883 (±0.006)	0.904 (±0.005)	0.893 (±0.002)	0.899 (±0.001)	0.917 (±0.002)
	MolEdit	**0.978** (±0.010)	**0.981** (±0.009)	**0.983** (±0.008)	**0.991** (±0.001)	**0.991** (±0.000)	**0.993** (±0.000)

Table 2: Main results on MEBench for editing MoMu and MolFM in molecule generation, evaluated across Reliability and Locality dimensions. Each dimension uses three metrics: BLEU-2, METEOR, and ROUGE-1. The best and second-best results are shown in **bold** and underlined, respectively. FT denotes fine-tuning.

5 Experiments

We aim to answer three key research questions: **RQ1**: How does MolEdit perform compared to baselines on MEBench? **RQ2**: How does each component contribute to MolEdit's performance? **RQ3**: How can we explain the effectiveness of each modules? The analysis is presented below.

5.1 Experiment Settings

Below we briefly introduce to the experiment settings, with details explained in Appendix A.

5.1.1 MoLM Backbone

In this paper, we use two widely used multimodal MoLMs as the editing backbones: MoMu (Su et al., 2022) and MolFM (Luo et al., 2023).

MoMu. MoMu is a multimodal MoLM aligning molecule graphs and text through contrastive learning (Li et al., 2021). MoMu utilizes GIN (Xu et al., 2018) and Bert (Devlin, 2018) as the graph and text encoders, and MolT5 (Edwards et al., 2022) as the decoder for molecule and caption generation.

MolFM. MolFM is a multimodal MoLM integrating molecular structures, biomedical texts,

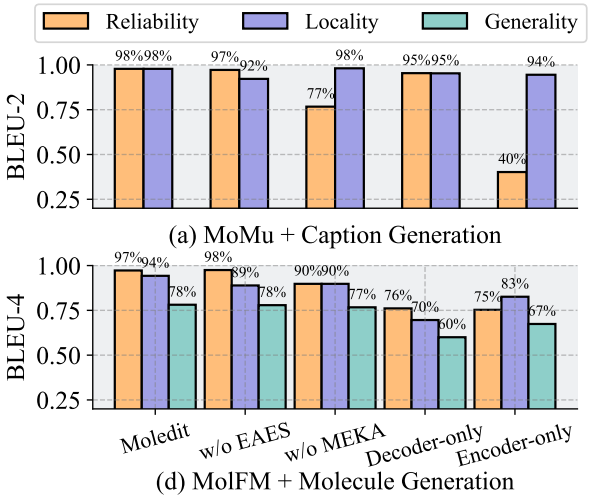

Figure 4: Ablation study on MoMu in caption generation and MolFM in molecule generation under three evaluation dimensions. We evaluate BLEU-2 for caption generation and BLEU-4 for molecule generation. EAES denotes Expertise-Aware Editing Switcher and MEKA denotes Multi-Expert Knowledge Adapter.

and knowledge graphs via cross-modal attention. MolFM utilizes GraphMVP (Liu et al., 2021) and BERT as the graph and text encoders, and MolT5 as the decoder for molecule and caption generation.

5.1.2 Baselines

Fine-tune. Fine-tune adapts pre-trained language models to specific tasks and is a common baseline for knowledge editing (Li et al., 2023; Zhong et al., 2023). We experiment with three fine-tuning strategies: editing the encoder, decoder, and both.

MEND. MEND (Mitchell et al., 2021) enables efficient local edits using a single input-output pair. It employs auxiliary editing networks that transform model gradients via low-rank decomposition.

GRACE. GRACE (Hartvigsen et al., 2024) employs a deferral mechanism to decide whether to activate an adapter by comparing a given input against a codebook of stored edits. We extend both GRACE and MEND to a multi-modal setting.

5.1.3 Metrics

To evaluate each method on MEBench, we use standard metrics commonly applied to MoLMs. For editing molecule generation, we employ BLEU-4 (Papineni et al., 2002), and Levenshtein (Yujian and Bo, 2007), MACCS FTS (Zhang et al., 2024c). For editing caption generation, we use BLEU-2, METEOR (Banerjee and Lavie, 2005), and ROUGE-1 (Lin, 2004).

5.2 Main Experiments

To answer **RQ1**, we first evaluate MolEdit's performance against all baseline methods on MEBench for both molecule and caption generation tasks. We make the following observations in Table 1 and Table 2: (1) MolEdit consistently outperforms baselines across most metrics in three dimensions for both generation tasks. Specifically, MolEdit outperforms the second-best method by 10% in Reliability, by 10% in Locality, and by 10% in Generality at most, which demonstrates MolEdit's ability to update molecule-specific knowledge precisely. (2) Different methods excel in different evaluation dimensions on MEBench. Fine-tuning, while effective at reliable knowledge updates, struggles to preserve untargeted knowledge, particularly in molecule generation. MEND performs better at knowledge preservation but underperforms in Reliability, potentially due to the complexity of unstructured molecular knowledge, which poses a challenge for meta-learning approaches. GRACE excels at editing caption generation, but performs poorly in editing molecule generation, underscoring the necessity for molecule-specific solutions. (3) The choice of the edited module significantly influences performance. Fine-tuning the encoder generally enhances performance in editing molecule generation (particularly for MolFM) but leads to weaker performance in editing caption generation, suggesting that knowledge is stored in different locations depending on the task and MoLM. Furthermore, the superior performance achieved by editing both modules highlights the presence of synergistic knowledge storage across them.

5.3 Ablation Study

To answer **RQ2**, we assess the contribution of each module in MolEdit to the performance. We use *w/o MEKA* to denote the use of a plain LoRA adapter without MoE and *w/o EAES* to denote calculating similarity at the sample level rather than the expertise level. Additionally, we evaluate the impact of editing only the encoder or decoder in MolEdit. The results are presented in Figure 4. We make the following observations: (1) Both MEKA and EAES contribute to overall performance, validating their effectiveness in addressing molecule-specific challenges in MoLM editing. (2) EAES enhances Locality performance, indicating that it successfully defers untargeted inputs to unedited layers. MEKA improves Reliability performance, suggesting that multiple experts are better equipped to han-

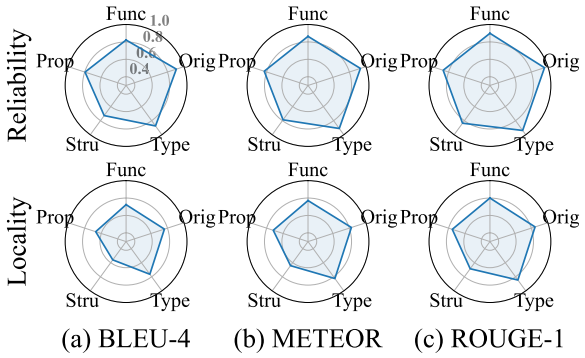

Figure 5: Performance of fine-tuning MoMu on a variation of MEBench. Each subset in this variation targets caption editing where only a single type of expertise requires modification. The expertise is labeled by domain experts and describes different molecular aspects, including (1) *Function* (Func), (2) *Origin* (Orig), (3) *Structure* (Stru), (4) *Type*, and (5) *Property* (Prop).

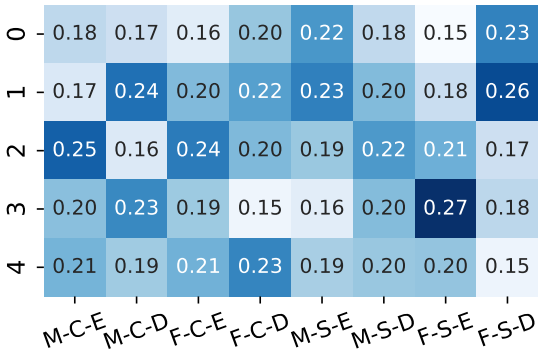

Figure 6: The activation distribution of the experts under different settings. For the label in *x*-axis, "M" denotes MoMu, "F" denotes MolFM; "C" denotes editing caption generation, "S" denotes editing molecule generation; "E" denotes editing encoder, "D" denotes decoder.

dle multifaceted molecular knowledge. (3) Editing both the encoder and decoder outperforms editing either component individually, highlighting synergistic knowledge storage across these components.

5.4 Rationale Validation

To answer **RQ3**, we aim to provide a deeper analysis of each module's rationale. We first validate two interconnected principles behind the multi-expert knowledge adapter: **(1) The necessity of expertise-wise editing.** We hypothesize that different expertise exhibits varying sensitivities to editing, posing risks of both over-editing and under-editing, thereby necessitating expertise-specific adjustments. To validate this, we curate a specialized dataset from MEBench for molecular caption generation, where each subset focuses on modifying a single targeted expertise. By evaluating fine-tuning performance on this dataset (Figure 5), we observe that editing sensitivities vary across expertise domains, with structure edits being the most challenging. This divergence underscores the importance of adopting expertise-wise editing strategies. **(2) The effectiveness of expertise-wise editing.** We validate this by analyzing the expert activation distribution of MoE under different experimental settings, as shown in Figure 6. The results indicate that all experts in both the encoder and decoder consistently exhibit high activation rates, demonstrating that the MoE effectively isolates and routes different expertise during editing.

Next, we validate **the effectiveness of expertise-aware editing switcher (EAES)** in selectively activating relevant knowledge while suppressing untargeted information. As shown in Table 3, EAES achieves significantly higher switching accuracy than non-expertise-aware alternatives, with improvements of up to 90.7%. By validating these principles, we demonstrate that MolEdit's modular design effectively addresses molecule-specific editing challenges, resulting in enhanced performance.

	M-Cap	F-Cap	M-Mol	F-Mol
Accuracy	0.827	0.772	0.635	0.529
Δ ↑	65.5%	54.4%	90.7%	58.9%

Table 3: Accuracy of Expertise-Aware Editing Switcher under different settings. **Δ** ↑ represents the improvement in accuracy compared to switchers that do not consider expertise. "M" denotes MoMu, "F" denotes MolFM; "Cap" denotes editing caption generation, "Mol" denotes editing molecule generation.

6 Conclusion

In this paper, we make the first attempt to edit MoLMs for both molecule and caption generation. To address the unique challenges associated with molecular editing, we propose MolEdit, a novel framework which consists of two key components: (1) MEKA, which directs molecular knowledge to specialized editing experts, enabling fine-grained control over multi-faceted updates; and (2) EAES, which maintains a memory bank of edited molecular expertise to activate MEKA only for highly relevant inputs. To enable systematic evaluation, we introduce MEBench, a comprehensive benchmark that assesses three critical dimensions: Reliability, Locality, and Generality. Extensive experiments demonstrate the effectiveness of MolEdit.

Limitation

Task Scope. While our work lays the foundation for editing Molecular Language Models (MoLMs), its scope is currently limited to molecule-to-caption and caption-to-molecule generation. Broader applications of MoLMs, such as molecular property prediction, cross-modal retrieval, and IUPAC name prediction, remain unexplored in the context of model editing. These tasks also rely on accurate molecular knowledge and may require specialized editing strategies to address domain-specific errors. Extending our approach to a wider range of applications presents an exciting direction for future research.

MoLM Generality. Our experiments focus on two representative Molecular Language Models (MoLMs), MoMu and MolFM, but the increasing diversity of molecular foundation models—including decoder-only and unified generative architectures—introduces new challenges. Expanding our methodology to accommodate a broader range of model architectures, scales, and training paradigms will further enhance its practical utility and robustness in real-world deployment scenarios.

Benchmark Limitation. While MEBench provides a comprehensive evaluation of MoLM editing capabilities, it is limited in both scale and knowledge structure. Specifically, MEBench is constructed from a small set of suboptimal entries generated by MoLMs, constraining its overall scale. Moreover, its knowledge remains unstructured, whereas a more structured MoLM editing dataset—organized in a source-relation-target format—could introduce new challenges. Developing a large-scale MoLM editing benchmark that aligns structurally with molecular knowledge graphs would be a promising direction for future research.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

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*.

Canyu Chen, Baixiang Huang, Zekun Li, Zhaorun Chen, Shiyang Lai, Xiongxiao Xu, Jia-Chen Gu, Jindong Gu, Huaxiu Yao, Chaowei Xiao, et al. 2024. Can editing llms inject harm? *arXiv preprint arXiv:2407.20224*.

Siyuan Cheng, Bozhong Tian, Qingbin Liu, Xi Chen, Yongheng Wang, Huajun Chen, and Ningyu Zhang. 2023. Can we edit multimodal large language models? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13877–13888, Singapore. Association for Computational Linguistics.

Siyuan Cheng, Ningyu Zhang, Bozhong Tian, Xi Chen, Qingbin Liu, and Huajun Chen. 2024. Editing language model-based knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17835–17843.

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

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. *arXiv preprint arXiv:2104.08164*.

Jingcheng Deng, Zihao Wei, Liang Pang, Hanxing Ding, Huawei Shen, and Xueqi Cheng. 2024a. Unke: Unstructured knowledge editing in large language models. *arXiv preprint arXiv:2405.15349*.

Yifan Deng, Spencer S Ericksen, and Anthony Gitter. 2024b. Chemical language model linker: blending text and molecules with modular adapters. *arXiv preprint arXiv:2410.20182*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. *arXiv preprint arXiv:2210.03329*.

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*.

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*.

Haisong Gong, Qiang Liu, Shu Wu, and Liang Wang. 2024. Text-guided molecule generation with diffusion language model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 109–117.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2024. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Advances in Neural Information Processing Systems*, 36.

Baixiang Huang, Canyu Chen, Xiongxiao Xu, Ali Payani, and Kai Shu. 2024a. Can knowledge editing really correct hallucinations? *arXiv preprint arXiv:2410.16251*.

Han Huang, Haitian Zhong, Tao Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2024b. Vlkeb: A large vision-language model knowledge editing benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. 2023. Transformer-patcher: One mistake worth one neuron. *arXiv preprint arXiv:2301.09785*.

Saeedeh Javadi. 2024. *Knowledge Editing in Large Language Model*. Ph.D. thesis, Politecnico di Torino.

Ge Lei, Ronan Docherty, and Samuel J Cooper. 2024. Materials science in the era of large language models: a perspective. *Digital Discovery*.

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

Sihang Li, Zhiyuan Liu, Yanchen Luo, Xiang Wang, Xiangnan He, Kenji Kawaguchi, Tat-Seng Chua, and Qi Tian. 2024b. Towards 3d molecule-text interpretation in language models. *arXiv preprint arXiv:2401.13923*.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. 2021. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*.

Zhoubo Li, Ningyu Zhang, Yunzhi Yao, Mengru Wang, Xi Chen, and Huajun Chen. 2023. Unveiling the pitfalls of knowledge editing for large language models. *arXiv preprint arXiv:2310.02129*.

Chang Liao, Yemin Yu, Yu Mei, and Ying Wei. 2024. From words to molecules: A survey of large language models in chemistry. *arXiv preprint arXiv:2402.01439*.

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

Pengfei Liu, Yiming Ren, Jun Tao, and Zhixiang Ren. 2024a. Git-mol: A multi-modal large language model for molecular science with graph, image, and text. *Computers in biology and medicine*, 171:108073.

Pengfei Liu, Jun Tao, and Zhixiang Ren. 2024b. Scientific language modeling: A quantitative review of large language models in molecular science. *arXiv preprint arXiv:2402.04119*.

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

Shengchao Liu, Hanchen Wang, Weiyang Liu, Joan Lasenby, Hongyu Guo, and Jian Tang. 2021. Pre-training molecular graph representation with 3d geometry. *arXiv preprint arXiv:2110.07728*.

Xianggen Liu, Yan Guo, Haoran Li, Jin Liu, Shudong Huang, Bowen Ke, and Jiancheng Lv. 2024c. Drugllm: Open large language model for few-shot molecule generation. *arXiv preprint arXiv:2405.06690*.

Zhiyuan Liu, Sihang Li, Yanchen Luo, Hao Fei, Yixin Cao, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. 2023b. Molca: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. *arXiv preprint arXiv:2310.12798*.

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.

Yizhen Luo, Kai Yang, Massimo Hong, Xingyi Liu, and Zaiqing Nie. 2023. Molfm: A multimodal molecular foundation model. *arXiv preprint arXiv:2307.09484*.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022. Memory-assisted prompt editing to improve gpt-3 after deployment. *arXiv preprint arXiv:2201.06009*.

Vittorio Mazzia, Alessandro Pedrani, Andrea Caciolai, Kay Rottmann, and Davide Bernardi. 2024. A survey on knowledge editing of neural networks. *IEEE Transactions on Neural Networks and Learning Systems*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2021. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022. Memory-based model editing at scale. In *International Conference on Machine Learning*, pages 15817–15831. PMLR.

Shiwen Ni, Dingwei Chen, Chengming Li, Xiping Hu, Ruifeng Xu, and Min Yang. 2023. Forgetting before learning: Utilizing parametric arithmetic for knowledge updating in large language models. *arXiv preprint arXiv:2311.08011*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Jinhua Zhu, Yue Wang, Zun Wang, Tao Qin, and Rui Yan. 2024a. Leveraging biomolecule and natural language through multi-modal learning: A survey. *arXiv preprint arXiv:2403.01528*.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Jinhua Zhu, and Rui Yan. 2024b. 3d-molt5: Towards unified 3d molecule-text modeling with 3d molecular tokenization. *arXiv preprint arXiv:2406.05797*.

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

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

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

Anton Sinitsin, Vsevolod Plokhotnyuk, Dmitriy Pyrkin, Sergei Popov, and Artem Babenko. 2020. Editable neural networks. *arXiv preprint arXiv:2004.00345*.

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

Taffee T Tanimoto. 1958. Elementary mathematical theory of classification and prediction.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024a. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *arXiv preprint arXiv:2405.14768*.

Renzhi Wang and Piji Li. 2024. Lemoe: Advanced mixture of experts adaptor for lifelong model editing of large language models. *arXiv preprint arXiv:2406.20030*.

Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. 2024b. Knowledge editing for large language models: A survey. *ACM Computing Surveys*, 57(3):1–37.

Yi Xiao, Xiangxin Zhou, Qiang Liu, and Liang Wang. 2024. Bridging text and molecule: A survey on multimodal frameworks for molecule. *arXiv preprint arXiv:2403.13830*.

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

Li Yujian and Liu Bo. 2007. A normalized levenshtein distance metric. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1091–1095.

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

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, et al. 2024a. A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.

Qiang Zhang, Keyan Ding, Tianwen Lv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, et al. 2024b. Scientific large language models: A survey on biological & chemical domains. *ACM Computing Surveys*.

Yikun Zhang, Geyan Ye, Chaohao Yuan, Bo Han, Long-Kai Huang, Jianhua Yao, Wei Liu, and Yu Rong. 2024c. Atomas: Hierarchical alignment on molecule-text for unified molecule understanding and generation. *arXiv preprint arXiv:2404.16880*.

Haiteng Zhao, Shengchao Liu, Ma Chang, Hannan Xu, Jie Fu, Zhihong Deng, Lingpeng Kong, and Qi Liu. 2023. Gimlet: A unified graph-text model for instruction-based molecule zero-shot learning. *Advances in Neural Information Processing Systems*, 36:5850–5887.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? *arXiv preprint arXiv:2305.12740*.

Zexuan Zhong, Zhengxuan Wu, Christopher D Manning, Christopher Potts, and Danqi Chen. 2023. Mquake: Assessing knowledge editing in language models via multi-hop questions. *arXiv preprint arXiv:2305.14795*.

A Implemention Details

In this section, we will introduce the implementation details of MolEdit. The Multi-Expert Knowledge Adapter's similarity threshold ϵ is 0.98 for MolFM molecule generation, 0.8 for MoMu molecule generation, and 0.9 for caption generation. The number of experts P is set to 5, with top-k fixed at 1. The distance function $d(\cdot)$ is instantiated as cosine similarity. Following prior work (Cheng et al., 2023; Wang et al., 2024a), we edit mid-to-late layers (specifically, layer 4 of the encoder and layer 10 of the decoder in our work) for all tasks and backbones. We edit one piece of knowledge at a time for molecule generation and two pieces for caption generation, due to batch normalization in the MoMu and MolFM molecule encoders. In addition, learning rates are 1e-4 for caption generation, 2e-5 for MoMu molecule generation, and 1e-5 for MolFM molecule generation. All experiments were conducted on an NVIDIA A100 server with four GPUs.

B Dataset Statistics

As introduced previously, MEBench consists of two core components: (1) molecule generation editing, with **390** total samples, where each editing sample includes a Reliability (verifying direct edits), a Locality (preserving unrelated knowledge), and a Generality sample (maintaining consistency across rephrasing); and (2) caption generation editing with **572** total samples, where each editing sample contains a Reliability and a Locality sample.

We further propose a variant of MEBench comprising five specialized subsets, each targeting a single molecular expertise type: the **Function** set (192 samples) modifies molecule functions (e.g., "It has a role as a progestin."), the **Origin** set (114 samples) edits molecular derivations (e.g., "It derives from a D-mannitol."), the **Property** set (102 samples) adjusts chemical properties (e.g., "It is a conjugate acid of a 1-deoxy-D-gluconate."), the **Structure** set (570 samples) revises structural descriptors (e.g., "The molecule is an artemoin in which the two hydroxy groups on the C-30 side-chain are located at positions 19 and 20."), and the **Type** set (358 samples) updates categorical classifications (e.g., "It is a scalastatin, a 3beta-D-glucoside, a scaloside and a 3beta-hydroxy steroid.").

C Supplementary Experiments

In this section, we present supplementary experiments on MEBench, expanding on overall performance and ablation studies.

To answer **R1**, we compare MolEdit against all baselines using additional metrics for a more comprehensive evaluation. For molecule generation editing, we include RDK FTS (Schneider et al., 2015), MORGAN FTS (Rogers and Hahn, 2010), Exact (Edwards et al., 2022), and Bad (Edwards et al., 2022). FTS stands for fingerprint Tanimoto similarity (Tanimoto, 1958), Exact denotes exact match rates with the ground truth molecule SMILES strings, and Bad denotes invalid SMILES string rates. For caption generation editing, we add BLEU-4, ROUGE-2, and ROUGE-L. The results are shown in Table 4 and Table 5. We make similar observations as in main text that MolEdit consistently outperforms all baselines across most metrics in three dimensions for both generation tasks, further demonstrating its precise molecule-specific knowledge updating capabilities..

To answer **R2**, we provide a comprehensive investigation of the contribution of different modules in MolEdit. We make similar observations from Figure 7 and Figure 8 as in the main paper: (1) MEKA and EAES both contribute to the overall performance, validating their effectiveness for molecule-specific MoLM editing challenges. (2) EAES improves Locality by deferring untargeted inputs. MEKA improves Reliability, showing that multiple experts can better handle multifaceted molecular knowledge. (3) Editing both the encoder and decoder outperforms editing either component alone, indicating a synergistic distribution of knowledge across these modules.

D Packages Required

Below we list the key packages and their associated versions in our implementation.

- torch == 2.5.1
- torch-geometric == 2.6.1
- rdkit == 2024.3.6
- transformers == 4.46.3
- local-attention == 1.9.15
- SentencePiece == 0.2.0
- pandas == 2.2.3
- numpy == 2.2.2
- einops == 0.8.0
- scanpy == 1.10.4

		MoMu				MolFM			
		RDK↑	MORGAN↑	Exact↑	Bad↓	RDK↑	MORGAN↑	Exact↑	Bad↓
Reliability	FT (Encoder)	0.979 (±0.029)	0.976 (±0.030)	0.747 (±0.092)	0.221 (±0.058)	0.996 (±0.001)	0.995 (±0.001)	0.740 (±0.009)	0.240 (±0.009)
	FT (Decoder)	0.910 (±0.003)	0.901 (±0.002)	0.606 (±0.002)	0.297 (±0.004)	0.971 (±0.015)	0.961 (±0.021)	0.505 (±0.073)	0.429 (±0.063)
	FT (All)	**1.000** (±0.000)	**0.999** (±0.000)	**0.822** (±0.001)	**0.174** (±0.000)	0.990 (±0.003)	0.987 (±0.004)	0.695 (±0.029)	0.269 (±0.018)
	MEND	0.734 (±0.024)	0.705 (±0.022)	0.250 (±0.016)	0.445 (±0.002)	0.780 (±0.005)	0.755 (±0.007)	0.088 (±0.002)	0.801 (±0.002)
	GRACE	0.918 (±0.000)	0.909 (±0.000)	0.636 (±0.000)	0.267 (±0.000)	0.985 (±0.005)	0.980 (±0.006)	0.699 (±0.013)	0.250 (±0.020)
	MolEdit	0.986 (±0.009)	0.975 (±0.016)	0.742 (±0.024)	0.201 (±0.009)	**0.999** (±0.000)	**0.998** (±0.000)	**0.764** (±0.022)	**0.223** (±0.018)
Locality	FT (Encoder)	0.808 (±0.006)	0.787 (±0.001)	0.301 (±0.005)	0.513 (±0.007)	0.879 (±0.007)	0.881 (±0.006)	0.203 (±0.022)	0.713 (±0.022)
	FT (Decoder)	0.899 (±0.007)	0.876 (±0.005)	0.415 (±0.000)	0.408 (±0.004)	0.696 (±0.005)	0.682 (±0.014)	0.041 (±0.004)	0.865 (±0.009)
	FT (All)	0.920 (±0.006)	0.903 (±0.010)	0.472 (±0.000)	0.382 (±0.017)	0.819 (±0.020)	0.819 (±0.014)	0.129 (±0.024)	0.765 (±0.027)
	MEND	0.903 (±0.041)	0.885 (±0.048)	0.494 (±0.082)	0.336 (±0.033)	0.889 (±0.024)	0.886 (±0.023)	0.186 (±0.020)	0.731 (±0.018)
	GRACE	0.899 (±0.000)	0.887 (±0.000)	0.372 (±0.000)	0.500 (±0.000)	0.809 (±0.000)	0.848 (±0.001)	0.032 (±0.002)	0.949 (±0.007)
	MolEdit	**0.996** (±0.001)	**0.992** (±0.002)	**0.769** (±0.015)	**0.214** (±0.009)	**0.982** (±0.014)	**0.979** (±0.012)	**0.404** (±0.024)	**0.569** (±0.018)
Generality	FT (Encoder)	0.422 (±0.002)	0.364 (±0.004)	0.022 (±0.002)	0.623 (±0.018)	0.522 (±0.074)	0.484 (±0.060)	0.022 (±0.013)	0.855 (±0.013)
	FT (Decoder)	0.646 (±0.030)	0.592 (±0.035)	0.160 (±0.016)	0.550 (±0.005)	0.503 (±0.039)	0.450 (±0.028)	0.012 (±0.002)	0.928 (±0.004)
	FT (All)	0.617 (±0.018)	0.571 (±0.024)	0.150 (±0.023)	0.541 (±0.008)	0.533 (±0.065)	0.498 (±0.064)	0.018 (±0.004)	0.869 (±0.025)
	MEND	0.568 (±0.012)	0.524 (±0.009)	0.099 (±0.005)	0.535 (±0.002)	0.471 (±0.045)	0.476 (±0.045)	0.004 (±0.002)	0.865 (±0.005)
	GRACE	0.872 (±0.000)	**0.859** (±0.000)	**0.523** (±0.000)	**0.346** (±0.000)	**0.811** (±0.044)	0.789 (±0.032)	0.091 (±0.009)	0.829 (±0.005)
	MolEdit	**0.887** (±0.033)	0.856 (±0.022)	0.465 (±0.049)	**0.346** (±0.062)	0.808 (±0.090)	**0.790** (±0.071)	**0.119** (±0.024)	**0.776** (±0.002)

Table 4: Additional results on MEBench for editing MoMu and MolFM in molecule generation, evaluated across three dimensions: Reliability, Locality, and Generality. Each dimension uses four metrics: RDK, MORGAN, Exact, and Bad. The experiments are run with multiple random seeds, where the average and standard deviations are recorded. The best and second-best results are shown in **bold** and underlined, respectively. FT denotes fine-tuning.

		MoMu			MolFM		
		BLEU-4↑	ROUGE-2↑	ROUGE-L↑	BLEU-4↑	ROUGE-2↑	ROUGE-L↑
Reliability	FT (Encoder)	0.229 (±0.006)	0.283 (±0.009)	0.405 (±0.007)	0.216 (±0.014)	0.258 (±0.027)	0.375 (±0.026)
	FT (Decoder)	0.760 (±0.018)	0.813 (±0.016)	0.839 (±0.013)	0.914 (±0.000)	0.954 (±0.000)	0.956 (±0.000)
	FT (All)	0.880 (±0.040)	0.920 (±0.034)	0.928 (±0.029)	0.919 (±0.001)	0.958 (±0.001)	0.959 (±0.001)
	MEND	0.494 (±0.033)	0.519 (±0.021)	0.591 (±0.022)	0.567 (±0.024)	0.612 (±0.008)	0.675 (±0.007)
	GRACE	0.928 (±0.000)	0.964 (±0.000)	0.965 (±0.000)	0.928 (±0.000)	0.965 (±0.000)	0.965 (±0.000)
	MolEdit	**0.975** (±0.006)	**0.975** (±0.008)	**0.979** (±0.006)	**0.974** (±0.006)	**0.977** (±0.005)	**0.981** (±0.004)
Locality	FT (Encoder)	0.438 (±0.008)	0.472 (±0.007)	0.560 (±0.006)	0.417 (±0.060)	0.451 (±0.071)	0.538 (±0.060)
	FT (Decoder)	0.581 (±0.027)	0.599 (±0.029)	0.664 (±0.025)	0.615 (±0.000)	0.637 (±0.000)	0.699 (±0.000)
	FT (All)	0.567 (±0.076)	0.584 (±0.077)	0.651 (±0.064)	0.601 (±0.005)	0.624 (±0.009)	0.687 (±0.007)
	MEND	0.561 (±0.011)	0.582 (±0.013)	0.646 (±0.015)	0.819 (±0.006)	0.827 (±0.006)	0.859 (±0.005)
	GRACE	0.860 (±0.011)	0.877 (±0.006)	0.896 (±0.003)	0.878 (±0.001)	0.894 (±0.001)	0.910 (±0.001)
	MolEdit	**0.976** (±0.011)	**0.978** (±0.011)	**0.982** (±0.009)	**0.990** (±0.000)	**0.990** (±0.000)	**0.992** (±0.000)

Table 5: Additional results on MEBench for editing MoMu and MolFM in molecule generation, evaluated across Reliability and Locality dimensions. Each dimension uses three metrics: BLEU-4, ROUGE-2, ROUGE-L. The experiments are run with multiple random seeds, where the average and standard deviations are recorded. The best and second-best results are shown in **bold** and underlined, respectively. FT denotes fine-tuning.

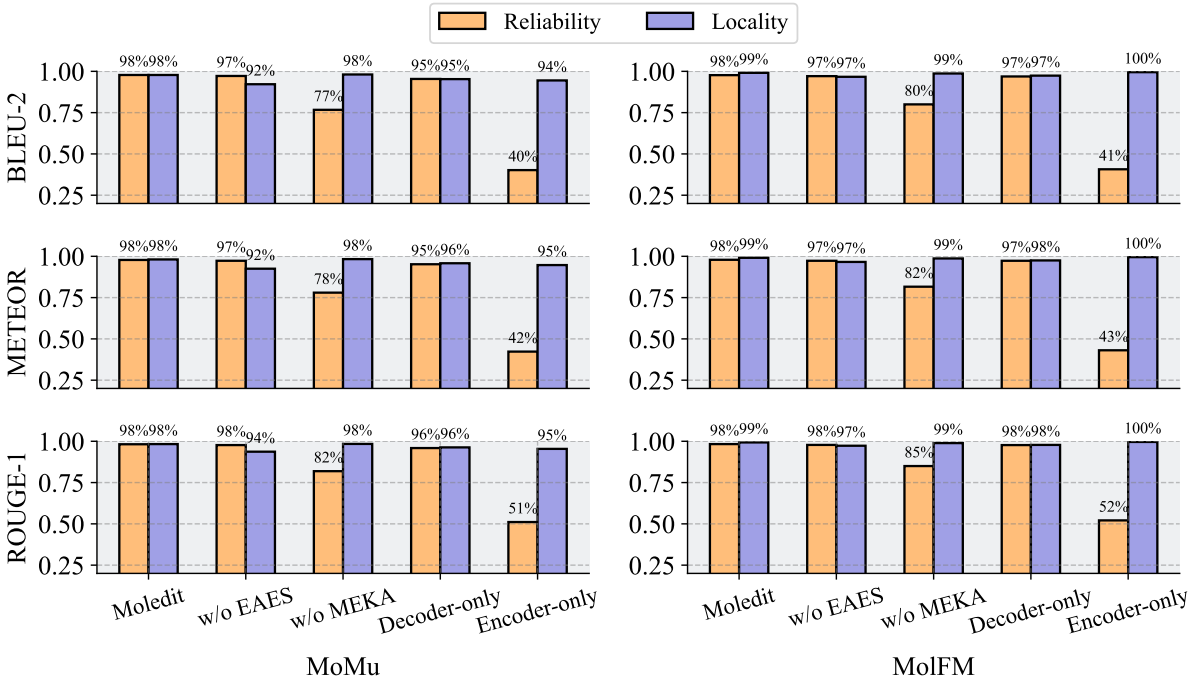

Figure 7: Ablation study for editing caption generation under the Reliability and Locality evaluation dimensions. For each dimension, we perform the evaluation by using three metrics: BLEU-2, METEOR, and ROUGE-1. EAES denotes Expertise-Aware Editing Switcher while MEKA denotes Multi-Expert Knowledge Adapter.

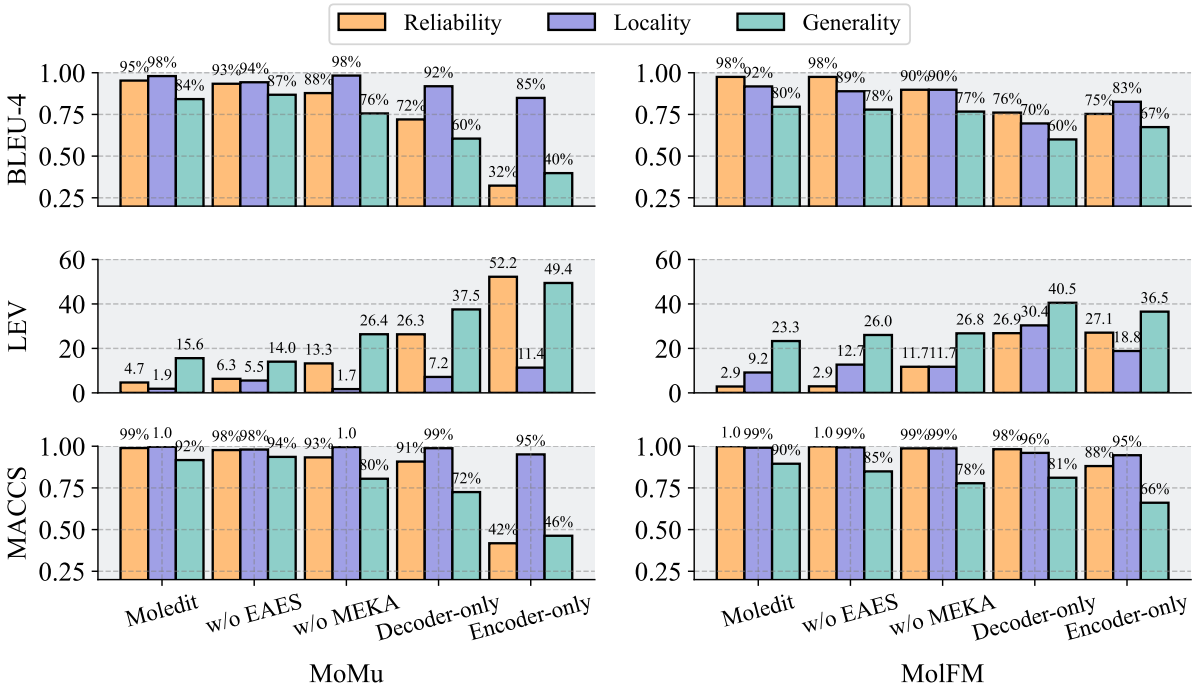

Figure 8: Ablation study for editing molecule generation under the Reliability, Locality, and Generality dimensions. For each dimension, we perform the evaluation by using three metrics: BLEU-4, LEV, and MACCS. EAES denotes Expertise-Aware Editing Switcher while MEKA denotes Multi-Expert Knowledge Adapter.