MolSC: Leveraging Substituent Contributions to Enhance Fine-grained Molecular Understanding in LLMs

Anonymous ACL submission

Abstract

Recent advances in natural language processing have led to molecular Large Language Models (LLMs) with strong performance across diverse chemistry tasks. However, they still struggle to capture fine-grained structure-property relationships, particularly how small, localized modifications alter a molecule's behavior. To address this limitation, we introduce `MolSC`, a dataset of substituent contributions, defined as property changes induced by attaching specific substituents to molecular scaffolds. Curated from manually annotated bioactivity records, `MolSC` spans structural-alert liability, target-specific bioactivity, and physicochemical descriptors, and contains 181K instruction-tuning examples for training. We further propose `MolSC-Bench`, a held-out evaluation benchmark of 1,541 examples disjoint from `MolSC` at the scaffold, substituent, and molecule levels. Our experiments show that existing molecular LLMs and strong proprietary models such as GPT-5.2 and Gemini-3-Flash show limited reliability in substituent contribution prediction. In contrast, training on `MolSC` substantially improves this ability and yields consistent gains on molecular reaction-related tasks and property prediction tasks. These results highlight substituent contribution learning as a key component of fine-grained molecular understanding.[1]

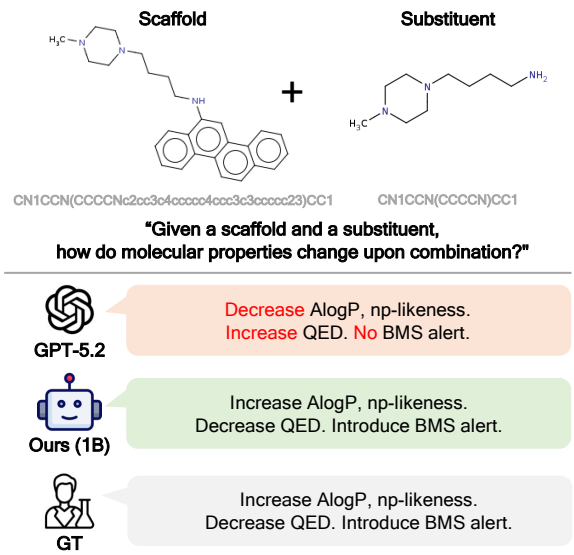


Figure 1: A `MolSC-Bench` example where our `MolSC`-trained model (Ours) correctly predicts property changes induced by substituent attachment, while GPT-5.2 predicts the opposite direction across all three property axes and misses the newly introduced BMS alert. Ground-truth (GT) changes are derived from ChEMBL.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of natural language tasks. Building on this success, recent work has extended LLMs to the chemical domain by training them on textual molecular representations such as SMILES (Weininger, 1988) and SELFIES (Krenn et al., 2020) through molecular instruction tuning (Fang et al., 2023). The resulting molecular LLMs (M. Bran et al., 2024; Cao et al., 2025) have achieved strong performance on diverse chemistry tasks, including reaction prediction (Wei et al., 2010), molecular property prediction (Wu et al., 2018), and molecule-text translation (Edwards et al., 2022).

Despite recent advances, current molecular LLMs still struggle to capture fine-grained structure-property relationships, i.e., how local structural changes affect molecular properties and activities, a key ability for lead optimization in drug discovery (Free and Wilson, 1964). Most existing molecular datasets pair each molecule with its overall property as a single unit (Wu et al., 2018), without supervising how individual substituents contribute to specific properties. Such molecule-level abstraction leads to representations that conflate functionally distinct compounds and restricts the model's ability to reason about local structural vari-

[1] Our data and code will be publicly available.

ations. Consequently, recent molecular LLMs struggle with functional group-level structure-property reasoning (Liu et al., 2026) and frequently misidentify substituents or misinterpret how they shape molecular properties (Yang et al., 2025).

While a few studies have attempted to enhance such fine-grained molecular understanding by leveraging molecular fragments or functional groups (Yang et al., 2025; Park et al., 2025), these approaches focus on aligning fragments with their textual descriptions or grouping them as structural tokens, without modeling the quantitative effect of each substituent on specific properties. As a result, how individual substituents shape specific molecular properties has remained largely implicit in existing datasets, motivating us to explicitly model substituent-level property contributions.

In this paper, we introduce **MolSC**, a dataset that captures how individual substituents contribute to molecular properties. We construct it by decomposing molecules from ChEMBL (Mendez et al., 2019) into scaffolds and substituents, defining each substituent's contribution as the property difference between the molecule and its scaffold. These contributions span three axes: structural-alert liability, target-specific bioactivity, and physicochemical descriptors. In total, MolSC provides 181K substituent contributions as training signals, directly supervising how local structural changes shape molecular properties. We further propose MolSC-Bench, a held-out benchmark of 1,541 examples disjoint from the training set at the scaffold, substituent, and molecule levels, to evaluate how well molecular LLMs understand substituent contributions.

Using MolSC-Bench, we show that existing molecular LLMs and even powerful proprietary models such as GPT-5.2 and Gemini-3-Flash struggle to reliably predict substituent contributions. In contrast, our model trained on MolSC with improved molecular representations not only achieves strong performance on MolSC-Bench but also is associated with consistent gains on downstream tasks, including molecular reaction-related tasks and molecular property prediction. Notably, our 3B model surpasses prior approaches that rely on larger backbones and auxiliary 2D molecular graphs, using only 1D molecular representations.

Our main contributions are as follows:

- We introduce MolSC and MolSC-Bench, a training dataset and benchmark of measured substituent contributions.
- We reveal that existing molecular LLMs and powerful proprietary models such as GPT-5.2 often fail to predict substituent contributions.
- We show that training on MolSC improves downstream molecular tasks, highlighting the importance of substituent contributions.

2 Related Work

2.1 Molecule Language Modeling

Driven by the success of deep learning and natural language processing, researchers have extensively applied these techniques to the chemical domain, interpreting molecules as either text sequences or atom-level graphs (Chithrananda et al., 2020; Ross et al., 2022). These approaches have gained attention for enabling more efficient drug candidate screening than traditional wet-lab pipelines. More recently, molecule instruction-tuning methods (Fang et al., 2023; Yu et al., 2024a) have sought to leverage the adaptability of LLMs for molecular tasks. However, LLMs still exhibit limited structural understanding of molecules, motivating continued efforts to address this challenge (Park et al., 2024a; Hu et al., 2025). For example, InstructMol integrates a 2D graph encoder with an LLM (Cao et al., 2025), while subsequent work (Pei et al., 2025) further incorporates 3D molecular topology to improve generation reliability. Nevertheless, existing approaches continue to struggle with fine-grained molecular understanding, often underperforming small-sized specialist models (<0.3B parameters) (Park et al., 2025). In this work, we aim to bridge this gap by explicitly modeling substituent contributions to molecular properties, enabling more precise molecular reasoning.

2.2 Fine-grained Molecular Understanding

Efforts to improve molecular understanding in language models have explored learning from fine-grained molecular decompositions. For property prediction, prior work has proposed fine-grained alignment across multiple molecular modalities (Feng et al., 2023; Yu et al., 2024b; Park et al., 2024b), while molecule-caption alignment methods associate molecular fragments with textual phrases to capture structure-language correspondence (Zhang et al., 2025; Park et al., 2025). To further enhance structural perception, hierarchical tokenization schemes encode atom-, motif-, and molecule-level information (Chen et al., 2025), and

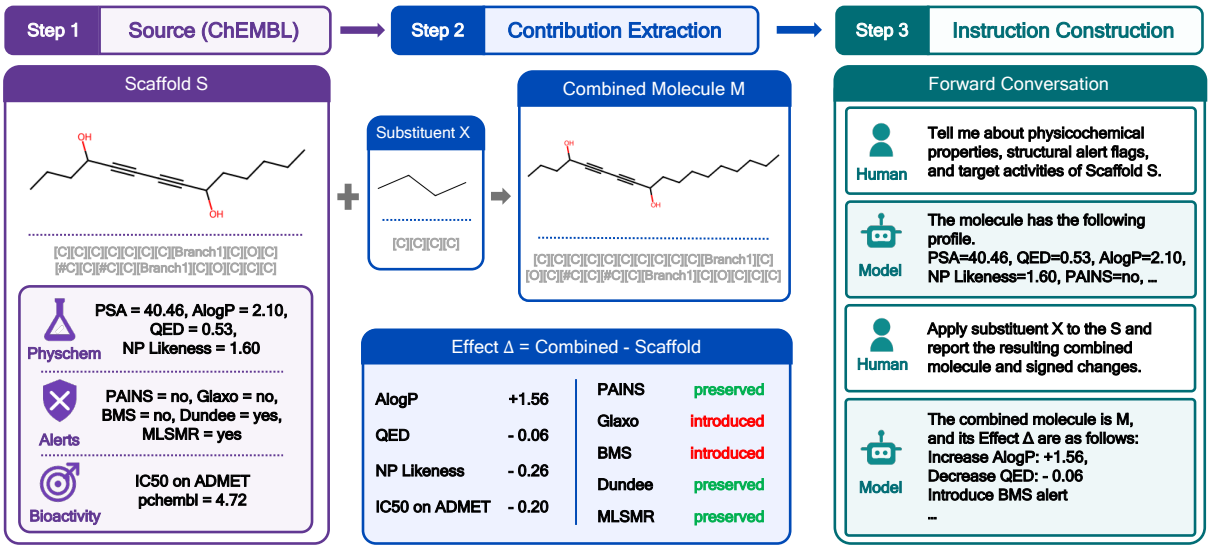


Figure 2: Data generation process of MolSC. We divide each molecule into a scaffold and a substituent, and define the substituent contribution as the property difference between the molecule and the scaffold. We then build an instruction-tuning dataset with predefined templates. Templates and examples are provided in Appendix C.

	MolSC	MolSC-Bench
Contributions	181,098	1,541
Unique scaffolds	100,076	1,541
Unique substituents	20,541	1,541
Unique molecules	164,650	1,518
Instructions	149,251	1,541

Table 1: Statistics of MolSC and MolSC-Bench.

datasets with multi-level annotations have been introduced to better link molecular structures to textual descriptions (Yang et al., 2025). In parallel, large-scale synthesis and decomposition pretraining has been shown to benefit reaction prediction tasks (Lin et al., 2025). However, these approaches primarily rely on surface-level signals, such as structural presence or alignment, without explicitly modeling how substructures influence task outcomes. In contrast, our work directly models how substituents change molecular properties across diverse chemical tasks.

3 Method

In this section, we introduce our proposed dataset and training strategy. First, we describe the construction of MolSC, which provides explicit fragment contribution information that enables us to train and evaluate models on fine-grained structure-property signals (§3.1). Second, we describe how we leverage MolSC to train a molecular LLM that achieves deep molecular understanding (§3.3). Figure 2 illustrates the data construction process.

3.1 MolSC

The contribution of individual substituents to molecular properties has long been foundational in chemical modeling, where biological activity is often decomposed into additive substituent effects (Free and Wilson, 1964). This principle continues to underpin modern structure-activity and structure-property modeling (Cherkasov et al., 2014; Muratov et al., 2020). However, such substituent-level information is rarely available to molecular LLMs as a direct learning signal, as existing datasets pair each molecule with its overall property and leave the effect of any individual substituent buried in molecule-level labels.

We construct MolSC, a dataset that explicitly captures substituent contributions to molecular properties. For each molecule formed by attaching a substituent to a scaffold, we record the signed property change induced by the substituent across three axes: structural-alert liability, target-specific bioactivity, and physicochemical descriptors.

3.1.1 Substituent Contribution Extraction

To obtain substituent-level supervision at scale, we ground MolSC in ChEMBL (Mendez et al., 2019), which provides annotations for all three axes central to drug discovery: structural-alert SMARTS patterns that flag liability concerns, assay-level bioactivity records that report target engagement, and physicochemical descriptors that summarize developability.

For each original molecule M extracted from

ChEMBL, we apply the Hussain-Rea fragmentation algorithm (Hussain and Rea, 2010) to obtain scaffolds and substituents. This process severs a single acyclic bond between two heavy atoms, yielding an ordered pair (S, X) where S denotes the scaffold and X the substituent. We retain only those scaffolds that themselves carry annotations along all three axes, ensuring that each scaffold provides a complete baseline against which the substituent's effect can be defined.

We then annotate substituent-induced changes along three property axes:

- **Structural-alert liability.** Whether the substituent *introduces*, *removes*, or *preserves* any alert from five expert-curated structural-alert sets: PAINS, Glaxo, BMS, Dundee, and MLSMR.
- **Target-specific bioactivity.** The signed activity change $\Delta_b = b_M - b_S$ for each *(target, activity type)* pair measured for both the scaffold and the original molecule in the same assay, reducing inter-laboratory variability (Kalliokoski et al., 2013). We use eight standard ChEMBL activity types: IC50, Ki, Kd, EC50, AC50, XC50, Potency, and ED50.
- **Physicochemical descriptors.** The signed descriptor change $\Delta_P = P(M) - P(S)$ for five descriptors that can either increase or decrease after substituent attachment: PSA, ALogP, HBD, QED, and NP-likeness.[2]

Together, these records yield approximately 1.2M substituent contributions, providing a substituent-resolved view of how local structural changes affect molecular properties.

3.1.2 Data Refinement

To improve the quality of the learning signal, we refine the extracted contributions by applying two filters. First, to ensure balanced learning across scaffolds, we drop over-represented scaffolds (those carrying more than ten distinct substituents, such as dinitrobenzene). Although they make up only 4.2% of all scaffolds, they account for 40.8% of the extracted contributions, and would otherwise dominate training at the expense of less frequent scaffolds. Second, we drop overly simple substituents with fewer than three heavy atoms, such as methyl and halogen groups, which act as atom-level decorations rather than chemically meaningful substituents. Single-atom substituents alone account for 42.5% of all bioactivity records, and retaining them risks degenerate representations. These two filters remove approximately 78% of the initial pool, yielding the final `MolSC` dataset comprising 181,098 substituent contributions.[3]

3.1.3 Instruction Construction

To effectively leverage the instruction-following ability of LLMs for fine-grained structure-property reasoning, we verbalize each substituent contribution into a multi-turn instruction conversation. Each conversation is designed to supervise three abilities: profiling all molecular properties from ChEMBL, performing structural synthesis or decomposition, and predicting property changes when a specific substituent is attached.

We construct conversations in both *forward* and *backward* directions, corresponding to substituent attachment and detachment, inspired by Lin et al. (2025). These directions also reflect lead optimization and substructure analysis in drug discovery. In the forward direction, the model first receives S and is asked to predict its full profile across the three axes; it is then given X and asked to produce the resulting molecule M together with the property changes induced by attaching X. In the backward direction, the model first receives M and is asked to predict its full profile across the three axes; it is then asked to decompose M into S and X together with the property contributions attributable to X. Contributions sharing the same anchor, either S in the forward direction or M in the backward direction, are merged into a single conversation so that the model can learn the substituent landscape of each anchor jointly. Examples and templates are provided in Appendix C.

3.2 `MolSC-Bench`

For a clean evaluation of substituent-level reasoning, we construct `MolSC-Bench` from 1,541 contributions whose scaffold, substituent, and original molecule are all unseen in `MolSC`. From these contributions, we define two complementary evaluation tasks that probe molecular understanding at different levels of granularity. **Task 1 (Property Profiling)** provides a molecule and asks the model

[2] We exclude descriptors that change monotonically with attachment, such as molecular weight, heavy-atom count, and Lipinski violations.

[3] We also exclude contributions whose S, X, or M overlaps with any molecule in the downstream evaluation sets.

Method	Task 1: Property Profiling				Task 2: Substituent Contribution Prediction				
	SR (%)	propMAE ↓	alertAcc ↑	bioMAE ↓	SR (%)	ΔpropMAE ↓	ΔalertAcc ↑	ΔbioMAE ↓	dirAcc ↑
Mol-Instruction-8B (2023)	40.6	24.019	0.667	–	65.1	9.263	0.075	**0.735**	0.454
ChemLLM-20B (2024)	99.0	15.854	0.700	6.326	100.0	8.300	0.049	0.960	0.146
ChemDFM-v2-14B (2025)	98.8	5.627	0.627	7.312	100.0	7.928	0.068	0.804	0.199
Gemini-3-Flash	100.0	9.013	0.667	1.473	100.0	2.535	0.100	0.799	0.722
GPT-5.2	100.0	10.531	0.517	**0.665**	100.0	3.519	0.219	2.469	0.706
Ours-1B w/o SC, dual-view	100.0	0.616	0.947	0.817	100.0	9.305	0.422	0.776	0.525
Ours-1B w/o dual-view	100.0	0.465	0.959	0.764	100.0	0.727	0.674	0.797	0.908
Ours-1B	100.0	<u>0.386</u>	<u>0.973</u>	<u>0.728</u>	100.0	<u>0.639</u>	<u>0.714</u>	<u>0.754</u>	<u>0.917</u>
Ours-3B	100.0	**0.307**	**0.978**	0.731	100.0	**0.561**	**0.737**	**0.735**	**0.923**

Table 2: Results on `MolSC-Bench`. SR denotes Success Rate, the proportion of outputs that follow the required format. SC denotes Substituent Contribution; w/o SC refers to the model trained only on property prediction from `MolSC`. dual-view refers to our training strategy mentioned in §3.3. **Bold** and <u>underline</u> indicate the best and second-best results, respectively.

to predict its full profile across all three axes, evaluating whether the model can infer molecule-level properties. **Task 2 (Substituent Contribution Prediction)** provides a scaffold, a substituent to attach, and the scaffold's baseline values on the affected axes, and asks the model to predict the signed property changes induced by the attachment. Since the resulting molecule is never given, the model must infer the attachment outcome rather than read it off a known structure.

3.3 Training Strategy

We train a general-purpose LLM on `MolSC` using supervised fine-tuning. For a multi-turn conversation sequence $x = (x_1, \ldots, x_T)$ constructed in §3.1, we optimize the standard causal language modeling objective:

$$\mathcal{L} = -\sum_{t=1}^{T} m_t \log p_\theta(x_t \mid x_{<t}),$$

where $m_t = 1$ for target tokens the model needs to generate, and 0 otherwise.

To reliably learn local substituent contributions, the model first requires a coherent global molecular context. We establish this prerequisite through *dual-view representation learning*, providing each molecule as both a SMILES and a SELFIES string in every turn. This joint formulation enforces structural consistency across complementary textual forms: SMILES (Weininger, 1988) compactly exposes atom connectivity, whereas SELFIES (Krenn et al., 2020) ensures chemical validity (Leon et al., 2024). By stabilizing the global representation, this dual-view alignment provides a robust basis for capturing fine-grained structure-property relationships.

4 Experiments

In this section, we demonstrate the efficacy of the substituent contribution information provided in `MolSC` through extensive experiments and analyses to answer the following questions:

Q1 Does `MolSC` offer a better understanding of substituent contribution? (§4.2)

Q2 Can the knowledge of `MolSC` be generalized to downstream tasks? (§4.3, §4.4)

Q3 Does `MolSC` enable models to capture structure-property relationships? (§4.5)

4.1 Experimental Settings

Backbone. We use Llama-3.2-1B-Instruct and Llama-3.2-3B-Instruct (Grattafiori et al., 2024) as backbone models, and fine-tune them on `MolSC` with LoRA (Hu et al., 2022) using the training strategy described in Section 3.3. Further implementation details are provided in Appendix A.

Dataset. For training and evaluating our models, we use `MolSC` and `MolSC-Bench`. We process all molecules using RDKit[4] and impose a maximum atom count of 100 due to computational constraints. For downstream tasks, we adopt the widely used benchmark, Mol-Instructions (Fang et al., 2023). Specifically, among its diverse tasks, we select those relevant to our study: forward reaction prediction, retrosynthesis, reagent prediction, and molecular property regression. To prevent data leakage, we exclude all molecules appearing in the downstream validation and test sets from the `MolSC` training set. The performance of our backbone model on the downstream tasks is provided in Appendix B.

[4] https://www.rdkit.org/

Method	# Params	Modality	Exact↑	BLEU↑	Lev↓	RDK↑	MAC↑	Mor↑	Validity↑
Forward Reaction Prediction									
Mol-Instructions (Fang et al., 2023)	8B	1D	0.503	0.883	13.410	0.756	0.863	0.708	1.000
InstructMol (Cao et al., 2025)	7B	1D + 2D	0.536	0.967	10.851	0.776	0.878	0.741	1.000
HIGHT (Chen et al., 2025)	7B	1D + 2D	0.037	0.869	23.759	0.590	0.394	0.340	0.993
UniMoT (Guo et al., 2025)	7B	1D + 2D	0.611	0.980	8.297	0.836	0.911	0.807	1.000
Omni-Mol (Hu et al., 2025)	2B	1D + 2D	0.733	0.980	5.550	0.895	0.947	0.870	1.000
KnowMol (Yang et al., 2025)	7B	1D + 2D	0.752	0.986	5.662	0.889	0.943	0.877	1.000
Ours-1B w/o SC, dual-view	1B	1D	0.762	0.985	5.311	0.895	0.946	0.878	1.000
Ours-1B w/o dual-view	1B	1D	0.790	0.986	4.916	0.917	0.955	0.899	1.000
Ours-1B	1B	1D	0.927	0.990	1.281	0.977	0.986	0.969	1.000
Ours-3B	3B	1D	**0.948**	**0.995**	**0.760**	**0.984**	**0.992**	**0.979**	1.000
Retrosynthesis									
Mol-Instructions (Fang et al., 2023)	8B	1D	0.333	0.842	17.642	0.704	0.815	0.646	1.000
InstructMol (Cao et al., 2025)	7B	1D + 2D	0.407	0.941	13.967	0.753	0.852	0.714	1.000
HIGHT (Chen et al., 2025)	7B	1D + 2D	0.008	0.863	28.912	0.564	0.340	0.309	1.000
UniMoT (Guo et al., 2025)	7B	1D + 2D	0.478	0.974	11.634	0.810	0.909	0.771	1.000
Omni-Mol (Hu et al., 2025)	2B	1D + 2D	0.570	0.960	8.970	0.864	0.909	0.830	1.000
KnowMol (Yang et al., 2025)	7B	1D + 2D	0.598	**0.975**	8.363	0.856	0.912	0.829	1.000
Ours-1B w/o SC, dual-view	1B	1D	0.581	0.960	9.031	0.854	0.907	0.823	1.000
Ours-1B w/o dual-view	1B	1D	0.595	0.963	8.077	0.876	0.919	0.840	1.000
Ours-1B	1B	1D	0.674	0.971	5.588	0.915	0.940	0.885	1.000
Ours-3B	3B	1D	**0.692**	0.972	**5.385**	**0.920**	**0.943**	**0.894**	1.000
Reagent Prediction									
Mol-Instructions (Fang et al., 2023)	8B	1D	0.101	0.648	18.326	0.412	0.521	0.375	1.000
InstructMol (Cao et al., 2025)	7B	1D + 2D	0.129	0.610	19.664	0.444	0.539	0.400	1.000
HIGHT (Chen et al., 2025)	7B	1D + 2D	0.050	0.462	28.970	0.441	0.314	0.275	1.000
UniMoT (Guo et al., 2025)	7B	1D + 2D	0.167	0.728	14.588	0.549	0.621	0.507	1.000
Omni-Mol (Hu et al., 2025)	2B	1D + 2D	0.230	0.736	14.590	**0.557**	0.627	0.520	1.000
KnowMol (Yang et al., 2025)	7B	1D + 2D	0.238	0.733	14.058	0.525	0.609	0.490	1.000
Ours-1B w/o SC, dual-view	1B	1D	0.230	0.728	14.717	0.506	0.603	0.488	1.000
Ours-1B w/o dual-view	1B	1D	0.245	0.745	14.144	0.538	0.623	0.513	1.000
Ours-1B	1B	1D	0.211	0.725	14.803	0.495	0.594	0.477	1.000
Ours-3B	3B	1D	**0.262**	**0.747**	**13.456**	0.548	**0.636**	**0.529**	1.000

Table 3: Results for reaction prediction tasks. **Bold** and underlined mark the best and second-best scores. 1D and 2D refer to 1D molecular representations and 2D molecular graphs.

4.2 MolSC-Bench

Baselines. We compare our models against two groups of baselines. The first group consists of molecular LLMs, including Mol-Instruction-8B (Fang et al., 2023), ChemLLM-20B (Zhang et al., 2024), and ChemDFM-v2-14B (Zhao et al., 2025). The second group consists of powerful proprietary models, GPT-5.2 and Gemini-3-Flash, accessed via their official APIs. All baselines receive the same prompts as shown in Figures 5 and 6.

Metrics. Across both tasks, we first report the Success Rate (SR), the proportion of outputs that adhere to the required format. To evaluate the model's predictions, we measure the Mean Absolute Error (MAE) for the physicochemical and bioactivity axes, and prediction accuracy for the structural-alert axis. Specifically, in Task 1, we evaluate the absolute profile of the input molecule. Thus, the MAE is computed between the predicted and actual property values (propMAE, bioMAE), and the accuracy is calculated over all alert flags (alertAcc). In Task 2, we evaluate the property changes induced by substituent attachment. Accordingly, the MAE is computed against the ground-truth property changes (ΔpropMAE, ΔbioMAE), and the accuracy is measured exclusively over the modified alert flags (ΔalertAcc). Additionally, Task 2 includes direction accuracy (dirAcc) to report the proportion of correctly predicted change directions.

Results. Models trained on `MolSC` outperform all baselines on `MolSC-Bench`, as shown in Table 2. While proprietary models perform better than existing molecular LLMs on Task 1, both groups perform poorly on Task 2, indicating that substituent contribution prediction remains difficult even for stronger general-purpose models. In particular, baselines often predict plausible magnitudes

Method	# Params	Modality	HOMO↓	LUMO↓	$\Delta\epsilon$ ↓	Avg↓
Mol-Instruction (Fang et al., 2023)	7B	1D	0.0210	0.0210	0.0203	0.0210
InstructMol (Cao et al., 2025)	7B	1D + 2D	0.0048	0.0050	0.0061	0.0050
HIGHT (Chen et al., 2025)	7B	1D + 2D	0.0056	0.0065	0.0077	0.0066
UniMoT (Guo et al., 2025)	7B	1D + 2D	0.0042	0.0047	0.0055	0.0049
Omni-Mol (Hu et al., 2025)	2B	1D + 2D	0.0038	0.0047	0.0049	0.0044
KnowMol (Yang et al., 2025)	7B	1D + 2D	0.0031	0.0037	**0.0035**	0.0034
Ours w/o SC, dual-view	1B	1D	0.0034	0.0036	0.0041	0.0037
Ours w/o Dual-view	1B	1D	0.0034	0.0036	0.0040	0.0037
Ours-1B	1B	1D	0.0033	**0.0030**	0.0045	0.0034
Ours-3B	3B	1D	**0.0029**	0.0031	0.0039	**0.0033**

Table 4: Results for molecular property regression tasks on the QM9 benchmark (Fang et al., 2023) evaluated using Mean Absolute Error (MAE). **Bold** and underline indicate the best and second-best results. 1D and 2D refer to 1D molecular representations and 2D molecular graphs.

but fail to capture the correct direction of change: molecular LLMs are near chance, and proprietary models reach only around 0.7 direction accuracy. In contrast, our 3B model achieves the best overall performance, with a propMAE of 0.307 on Task 1 and a direction accuracy of 0.923 on Task 2. Since `MolSC` and `MolSC-Bench` share no scaffolds, substituents, or original molecules, these results suggest that `MolSC` teaches generalizable substituent contribution prediction.

Ablation Study. We ablate the Substituent Contribution supervision (SC) and dual-view representation learning introduced in §3.3. Without SC, the model trained only on property profiling performs poorly on Task 2, showing that molecule-level property supervision alone is insufficient for substituent contribution prediction. Adding SC substantially improves Task 2 and also benefits Task 1, suggesting that substituent-level supervision improves property understanding. Finally, dual-view representation learning yields the best overall performance, indicating that it further helps fine-grained structure-property understanding.

4.3 Molecular Reaction Prediction

Baselines. We compare our model with a diverse set of competitive Molecule LLMs, including the model from Mol-Instructions (Fang et al., 2023), InstructMol (Cao et al., 2025), HIGHT (Chen et al., 2025), UniMoT (Guo et al., 2025), Omni-Mol (Hu et al., 2025), and KnowMol (Yang et al., 2025). In particular, we focus on comparisons with Omni-Mol, which shares the same backbone language model as our approach (Llama-3.2-1B-Instruct (Grattafiori et al., 2024)).

Metrics. We employ Exact Match (EM), BLEU-4 (Papineni et al., 2002), and Levenshtein distance (Lev) (Miller et al., 2009) to measure string-level similarity, along with Validity to assess the grammatical correctness of generated molecules. We also adopt molecular fingerprint-based similarity metrics, including MACCS FTS (MAC) (Durant et al., 2002), RDK FTS (RDK) (Schneider et al., 2015), and Morgan FTS (Mor) (Rogers and Hahn, 2010), to compare structural similarity with reference molecules. All evaluations are conducted using model-generated SELFIES representations.

Results. Models trained on `MolSC` demonstrate strong overall performance across reaction-related tasks, including forward reaction, retrosynthesis, and reagent prediction (Table 3). Notably, in forward reaction and retrosynthesis, our 1B model achieves an average Exact Match improvement of 14.9%p over Omni-Mol, while Ours-3B achieves the highest performance across all three tasks. Since molecular properties are known to be related to chemical reactivity (Ahneman et al., 2018; Sandfort et al., 2020), we interpret the substituent-level structure-property knowledge in `MolSC` as one factor contributing to these gains. Furthermore, our models achieve these results relying solely on 1D representations, demonstrating that the structural knowledge in `MolSC` effectively transfers to downstream tasks without requiring complex 2D graph encoders or larger backbones.

Ablation Study. Adding SC consistently improves the Exact Match score across all three tasks, from 0.762 to 0.790 on forward reaction prediction, 0.581 to 0.595 on retrosynthesis, and 0.230 to 0.245 on reagent prediction. Applying dual-view representation learning brings further gains, with especially large improvements on forward reaction prediction (0.790 → 0.927) and retrosynthesis (0.595 → 0.674). Taken together, these results

show that learning substituent contributions is effective for reaction-related tasks.

4.4 Molecular Property Prediction

Baselines and Metrics. We use the same baselines as in Section 4.3, and report MAE on the QM9 regression tasks of Mol-Instructions (Fang et al., 2023).

Results. As shown in Table 4, models trained on `MolSC` achieve strong performance on the property regression tasks. Notably, our Ours-1B performs on par with KnowMol on average (0.0034), despite using only 1B parameters against its 7B, and Ours-3B attains the best average MAE overall. This indicates that the structural knowledge learned from `MolSC` also benefits diverse property prediction tasks, even at a small model scale.

Ablation Study. Unlike the previous tasks, property prediction concerns the properties of the whole input molecule, where property profiling plays a larger role than substituent contribution supervision. Indeed, the model trained only on property profiling already performs well, and adding SC brings little change. Consistent with our findings on `MolSC-Bench`, however, applying the dual-view representation further improves performance, indicating that a robust molecular representation is important for property prediction.

4.5 Context-Dependent Contribution Analysis

To examine whether `MolSC` enables models to capture scaffold-dependent structure-property relationships, we analyze cases where the same substituent induces opposite property changes depending on the scaffold. These cases require modeling the interaction between the substituent and the scaffold, rather than the substituent alone. We attach morpholine to four scaffolds for which the ground-truth effect on `qed` is positive for two scaffolds and negative for the other two, and compare `Ours-3B` with GPT-5.2 in Table 5. These records are drawn from contributions excluded from `MolSC` during construction due to overlap with downstream molecules, and are therefore unseen during `MolSC` training.

The two models show a clear difference. On axes with scaffold-independent directions, such as `psa` and `alogp`, both models predict the correct direction across all four scaffolds, suggesting that they can capture simpler substituent-driven effects. The difference emerges on the context-dependent `qed` axis. GPT-5.2 predicts the wrong `qed` direction for all four scaffolds, failing to capture how the scaffold changes the effect of the same substituent. In contrast, `Ours-3B` correctly tracks the sign change across all four cases, recognizing that morpholine can either increase or decrease `qed` depending on where it is attached. This suggests that training on `MolSC` helps the model learn scaffold-dependent substituent contributions, rather than relying only on a fixed substituent-level prior.

Task: Predict property Δ upon adding morpholine ( )

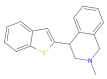
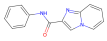
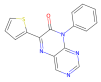
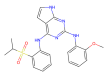
Scaffold	Axis	GT Δ		Ours-3B		GPT-5.2	
		Δ	dir	Δ	dir	Δ	dir
	psa	+12.47	↑	+12.47	↑	+28.96	↑
	alogp	-0.16	↓	-0.34	↓	-1.28	↓
	qed	+0.04	↑	+0.04	↑	-0.02	↓
	np_likeness	-0.34	↓	-0.49	↓	-0.12	↓
	psa	+12.47	↑	+12.47	↑	+32.80	↑
	alogp	-0.17	↓	-0.08	↓	-0.87	↓
	qed	+0.06	↑	+0.06	↑	-0.06	↓
	np_likeness	-0.09	↓	+0.01	↑	+0.74	↑
	psa	+12.47	↑	+12.47	↑	+27.83	↑
	alogp	-0.16	↓	-0.08	↓	-0.80	↓
	qed	-0.04	↓	-0.03	↓	+0.04	↑
	np_likeness	-0.15	↓	-0.14	↓	+0.43	↑
	psa	+12.47	↑	+12.47	↑	+25.00	↑
	alogp	-0.17	↓	-0.08	↓	-0.84	↓
	qed	-0.08	↓	-0.08	↓	+0.06	↑
	np_likeness	-0.36	↓	-0.28	↓	+0.23	↑

Table 5: Property changes and model predictions after attaching morpholine to four different scaffolds. Ground-truth (GT) values are from ChEMBL. dir denotes the predicted direction of change; blue and red indicate correct and incorrect dir predictions. Details of the used molecules are provided in Table 11.

5 Conclusion

In this paper, we explicitly model the contribution of individual substituents to molecular properties. To this end, we introduce `MolSC`, a dataset of substituent contributions spanning three property axes, together with `MolSC-Bench`, a held-out benchmark for evaluating substituent-level reasoning. We show that existing molecular LLMs and powerful proprietary models fail to reliably predict substituent contributions, whereas training on `MolSC` substantially improves this ability and transfers to downstream tasks, including reaction prediction and molecular property regression, using only 1D molecular representations. These results establish substituent contribution learning as a key component of fine-grained molecular understanding.

Limitations

We introduce MolSC, a dataset of substituent contributions that substantially improves the fine-grained molecular understanding of LLMs. Nevertheless, several limitations remain.

Limited property coverage. Molecular behavior spans a wide range of properties. In this work, MolSC focuses on three axes available from or derived using ChEMBL: structural-alert liability, target-specific bioactivity, and physicochemical descriptors. While these signals already support effective training and evaluation, properties beyond this scope, including those covered in MoleculeNet (Wu et al., 2018), are not captured. Expanding MolSC to broader property types would be an important direction for more comprehensive molecular understanding.

Reliance on measured properties. A substituent contribution is defined as the difference in a property value between a molecule and its corresponding scaffold, and can therefore only be computed when both have annotations for the same property. As such pairs are not always available, this restricts the set of molecule-scaffold pairs that MolSC can cover. Moreover, for the bioactivity axis, the measured differences may reflect assay-specific variability; although we compute each contribution from a pair measured within the same assay, experimental noise and assay-dependent effects may still remain. Nevertheless, even under these constraints, training on MolSC improves prediction on molecules, scaffolds, and substituents unseen during training (Table 2).

References

Derek T Ahneman, Jesús G Estrada, Shishi Lin, Spencer D Dreher, and Abigail G Doyle. 2018. Predicting reaction performance in c–n cross-coupling using machine learning. *Science*, 360(6385):186–190.

He Cao, Zijing Liu, Xingyu Lu, Yuan Yao, and Yu Li. 2025. InstructMol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 354–379, Abu Dhabi, UAE. Association for Computational Linguistics.

Yongqiang Chen, Quanming Yao, Juzheng Zhang, James Cheng, and Yatao Bian. 2025. Hierarchical graph tokenization for molecule-language alignment. In *Forty-second International Conference on Machine Learning*.

Artem Cherkasov, Eugene N Muratov, Denis Fourches, Alexandre Varnek, Igor I Baskin, Mark Cronin, John Dearden, Paola Gramatica, Yvonne C Martin, Roberto Todeschini, and 1 others. 2014. Qsar modeling: where have you been? where are you going to? *Journal of medicinal chemistry*, 57(12):4977–5010.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2020. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*.

Joseph L. Durant, Burton A. Leland, Douglas R. Henry, and James G. Nourse. 2002. Reoptimization of MDL keys for use in drug discovery. *J. Chem. Inf. Comput. Sci.*, 42(5):1273–1280.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. 2022. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 375–413, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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*.

Shikun Feng, Lixin Yang, Yanwen Huang, Yuyan Ni, Weiying Ma, and Yanyan Lan. 2023. Unimap: universal smiles-graph representation learning. *arXiv preprint arXiv:2310.14216*.

Spencer M Free and James W Wilson. 1964. A mathematical contribution to structure-activity studies. *Journal of medicinal chemistry*, 7(4):395–399.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shuhan Guo, Yatao Bian, Ruibing Wang, Nan Yin, Zhen Wang, and Quanming Yao. 2025. Unified molecule-text language model with discrete token representation. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 9205–9213.

Chengxin Hu, Hao Li, Yihe Yuan, Zezheng Song, Chenyang Zhao, and Haixin Wang. 2025. Omni-mol: Multitask molecular model for any-to-any modalities. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Jameed Hussain and Ceara Rea. 2010. Computationally efficient algorithm to identify matched molecular pairs (mmps) in large data sets. *Journal of chemical information and modeling*, 50(3):339–348.

Tuomo Kalliokoski, Christian Kramer, Anna Vulpetti, and Peter Gedeck. 2013. Comparability of mixed ic50 data–a statistical analysis. *PloS one*, 8(4):e61007.

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

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.

Miguelangel Leon, Yuriy Perezhohin, Fernando Peres, Aleš Popovič, and Mauro Castelli. 2024. Comparing smiles and selfies tokenization for enhanced chemical language modeling. *Scientific Reports*, 14(1):25016.

Xuan Lin, Qingrui Liu, Hongxin Xiang, Daojian Zeng, and Xiangxiang Zeng. 2025. Enhancing chemical reaction and retrosynthesis prediction with large language model and dual-task learning. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 7554–7562. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Xuan Liu, Siru Ouyang, Xianrui Zhong, Jiawei Han, and Huimin Zhao. 2026. FGBench: A dataset and benchmark for molecular property reasoning at functional group-level in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D White, and Philippe Schwaller. 2024. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6(5):525–535.

David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, and 1 others. 2019. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940.

Frederic P Miller, Agnes F Vandome, and John McBrewster. 2009. Levenshtein distance: Information theory, computer science, string (computer science), string metric, damerau? levenshtein distance, spell checker, hamming distance.

Eugene N Muratov, Jürgen Bajorath, Robert P Sheridan, Igor V Tetko, Dmitry Filimonov, Vladimir Poroikov, Tudor I Oprea, Igor I Baskin, Alexandre Varnek, Adrian Roitberg, and 1 others. 2020. Qsar without borders. *Chemical Society Reviews*, 49(11):3525–3564.

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.

Hyuntae Park, Yeachan Kim, and SangKeun Lee. 2025. Bridging the gap between molecule and textual descriptions via substructure-aware alignment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23470–23490.

Jinyoung Park, Minseong Bae, Dohwan Ko, and Hyunwoo J Kim. 2024a. Llamo: Large language model-based molecular graph assistant. *Advances in Neural Information Processing Systems*, 37:131972–132000.

Jun-Hyung Park, Hyuntae Park, Yeachan Kim, Woosang Lim, and SangKeun Lee. 2024b. Moleco: Molecular contrastive learning with chemical language models for molecular property prediction. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 408–420.

Qizhi Pei, Rui Yan, Kaiyuan Gao, Jinhua Zhu, and Lijun Wu. 2025. 3d-molt5: Leveraging discrete structural information for molecule-text modeling. In *The Thirteenth International Conference on Learning Representations*.

David Rogers and Mathew Hahn. 2010. Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, 50(5):742–754.

Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. 2022. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264.

Frederik Sandfort, Felix Strieth-Kalthoff, Marius Kühnemund, Christian Beecks, and Frank Glorius. 2020. A structure-based platform for predicting chemical reactivity. *Chem*, 6(6):1379–1390.

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. *J. Chem. Inf. Model.*, 55(10):2111–2120.

Jin-Mao Wei, Xiao-Jie Yuan, Qing-Hua Hu, and Shu-Qin Wang. 2010. A novel measure for evaluating classifiers. *Expert Systems with Applications*, 37(5):3799–3809.

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.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. 2018. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530.

Zaifei Yang, Hong Chang, Ruibing Hou, Shiguang Shan, and Xilin Chen. 2025. Knowmol: Advancing molecular large language models with multi-level chemical knowledge. *arXiv preprint arXiv:2510.19484*.

Botao Yu, Frazier N Baker, Ziqi Chen, Xia Ning, and Huan Sun. 2024a. Llasmol: Advancing large language models for chemistry with a large-scale, comprehensive, high-quality instruction tuning dataset. *arXiv preprint arXiv:2402.09391*.

Qiying Yu, Yudi Zhang, Yuyan Ni, Shikun Feng, Yanyan Lan, Hao Zhou, and Jingjing Liu. 2024b. Multimodal molecular pretraining via modality blending. In *The Twelfth International Conference on Learning Representations*.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, and 1 others. 2024. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*.

Yikun Zhang, Geyan Ye, Chaohao Yuan, Bo Han, Long-Kai Huang, Jianhua Yao, Wei Liu, and Yu Rong. 2025. Atomas: Hierarchical adaptive alignment on molecule-text for unified molecule understanding and generation. In *The Thirteenth International Conference on Learning Representations*.

Zihan Zhao, Da Ma, Lu Chen, Liangtai Sun, Zihao Li, Yi Xia, Bo Chen, Hongshen Xu, Zichen Zhu, Su Zhu, and 1 others. 2025. Developing chemdfm as a large language foundation model for chemistry. *Cell Reports Physical Science*, 6(4).

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Appendix

A Implementation Details

We use Llama-3.2-1B-Instruct and Llama-3.2-3B-Instruct as backbone models.[5] All models are fine-tuned with low-rank adaptation (LoRA; Hu et al., 2022) using LLaMA-Factory (Zheng et al., 2024). We use the AdamW optimizer, sequence packing for efficient training on variable-length conversations, and a cosine learning-rate schedule with linear warmup. The maximum sequence length is set to 2048 tokens, and training is performed in bf16 precision. The learning rate is set to 1×10^{-4} with a warmup ratio of 0.1. We use LoRA rank 64 and LoRA alpha 128. Each device processes one packed sequence per step, and gradients are accumulated over 8 steps. All experiments are conducted on a B200 GPU.

`MolSC` fine-tuning consists of a single supervised fine-tuning stage over 149,251 instruction conversations constructed from the forward and backward templates in Section C. We train this stage for 6 epochs. For downstream evaluation on Mol-Instructions (Fang et al., 2023), we further fine-tune the `MolSC`-trained checkpoint on each target task for 10 epochs, including forward reaction prediction, retrosynthesis, reagent prediction, and property prediction. In the dual-view setting, every molecule is represented with both SMILES and SELFIES in both inputs and outputs. For molecule generation tasks, we evaluate model outputs based on the generated SELFIES strings to match the setting of existing models. For inference, we use vLLM (Kwon et al., 2023) with greedy decoding under each model's native chat template. The same inference, parsing, and evaluation pipeline is used for all models and variants.

B Backbone Performance on Downstream Tasks

Model	HOMO↓	LUMO↓	$\Delta\epsilon$ ↓	Avg↓
Llama-3.2-1B-Instruct	0.0036	0.0038	0.0045	0.0040

Table 6: Results for property regression tasks of our backbone model.

This section reports the performance of the Llama-3.2-1B-Instruct backbone under our experimental setting. On `MolSC-Bench`, the backbone achieves an SR of 0 on both Task 1 and Task 2, indicating that it cannot produce correct responses. As shown in Table 6, it also underperforms our fine-tuned variants on property regression. For reaction prediction tasks, Table 7 shows that property profiling alone can degrade performance relative to the backbone, whereas adding substituent contribution supervision improves performance. These results further support the importance of explicitly learning substituent-level effects.

C Examples & Templates

[5] meta-llama/Llama-3.2-1B-Instruct; meta-llama/Llama-3.2-3B-Instruct.

Task	Model	Exact↑	BLEU↑	Lev↓	RDK↑	MAC↑	Mor↑	Validity↑
Forward Reaction Prediction	Llama-3.2-1B-Instruct	0.778	0.986	4.993	0.906	0.950	0.889	1.000
Retrosynthesis	Llama-3.2-1B-Instruct	0.574	0.962	8.707	0.858	0.909	0.825	1.000
Reagent Prediction	Llama-3.2-1B-Instruct	0.206	0.685	16.196	0.480	0.587	0.466	1.000

Table 7: Results for reaction prediction tasks of our backbone model.

Axis	Property	Unit / Range	Description
Physicochemical descriptors	PSA	$Å^2$	Topological polar surface area.
	ALogP	log unit	Atom-based octanol-water partition coefficient.
	HBD	count	Hydrogen bond donor count.
	QED	[0, 1]	Quantitative estimation of drug-likeness.
	NP-likeness	$\sim[-5, +5]$	Natural-product-likeness score.
	HBA	count	Hydrogen bond acceptor count.
	MW	g/mol	Molecular weight.
	Ro5 violations	count (0-4)	Number of Lipinski rule-of-five violations.
	Aromatic rings	count	Number of aromatic rings.
	Heavy atoms	count	Heavy atom count.
	Rotatable bonds	count	Number of rotatable bonds.
	Ro3 pass	binary	Astex rule-of-three pass indicator.
Structural-alert liability	PAINS	binary	Pan-assay interference compounds.
	BMS	binary	Bristol-Myers Squibb HTS deck filter.
	MLSMR	binary	NIH Molecular Libraries Small Molecule Repository excluded functionality filter.
	Dundee	binary	University of Dundee NTD screening library filter.
	Glaxo	binary	Glaxo Wellcome hard filter.
Target-specific bioactivity	IC50	pchembl	Half-maximal inhibitory concentration.
	Ki	pchembl	Inhibition constant.
	Kd	pchembl	Dissociation constant.
	EC50	pchembl	Half-maximal effective concentration.
	AC50	pchembl	Half-maximal activity concentration.
	XC50	pchembl	Generic half-maximal concentration.
	Potency	pchembl	Functional potency reported by the assay.
	ED50	pchembl	Half-maximal effective dose.

Table 8: All properties annotated in `MolSC`, organized along the three property axes. HBA, MW, Ro5 violations, aromatic rings, heavy atoms, rotatable bonds, and Ro3 pass are not considered when computing substituent contributions, since their values change monotonically under substituent attachment in our data. Bioactivity values are reported on the pchembl scale, $-\log_{10}$ of the molar concentration.

Forward conversation

T0 [human]
Tell me about physicochemical properties, structural alert flags, and target activities.

[SELFIES]
[C][C][C][C][C][C][Branch1][C][O][C][#C][C][#C][C][Branch1][C][O][C][C][C]

T1 [gpt]
The molecule has the following profile. Properties: psa=40.460, alogp=2.100, hba=2, hbd=2, mw=222.330, qed=0.530, ro5_viol=0, aromatic_rings=0, heavy_atoms=16, np_likeness=1.600, rtb=6, ro3_pass=False. Alerts: PAINS=no, Glaxo=no, BMS=no, Dundee=yes, MLSMR=yes. Bioactivity: IC50 on ADMET pchembl=4.72.

T2 [human]
Apply substituent
[SELFIES]
[C][C][C][C]
to the scaffold and tell me the resulting combined molecule. For the affected property axes (alogp, np_likeness, qed) and bioactivity stypes (IC50), report the signed change as combined − scaffold (positive means the fragment increased the axis). For every alert set (PAINS, Glaxo, BMS, Dundee, MLSMR), state whether it was introduced, removed, or preserved.

T3 [gpt]
Combined molecule:
[SELFIES]
[C][C][C][C][C][C][C][C][C][C][Branch1][C][O][C][#C][C][#C][C][Branch1][C][O][C][C][C]
Effect:
alogp: +1.560
qed: −0.060
np_likeness: −0.260
IC50 on ADMET: −0.20
PAINS: preserved
Glaxo: introduced
BMS: introduced
Dundee: preserved
MLSMR: preserved

Figure 3: A SFT example for the forward direction: starting from a scaffold profile, the model is then asked to attach a substituent and report the resulting original (combined) molecule together with the induced property changes.

Backward conversation

T0 [human]
Given
  [SELFIES]
  [C][=C][C][C][Branch1][=Branch1][C][C][C][C][C][N][C][Ring1][#Branch2][=O]
output its full property, alert, and bioactivity profile.

T1 [gpt]
The molecule has the following profile. Properties: psa=29.100, alogp=2.010, hba=1, hbd=1, mw=167.250, qed=0.500, ro5_viol=0, aromatic_rings=0, heavy_atoms=12, np_likeness=1.370, rtb=4, ro3_pass=False. Alerts: PAINS=no, Glaxo=no, BMS=yes, Dundee=yes, MLSMR=yes. Bioactivity: IC50 on L1210 (Mus musculus) pchembl=4.23; IC50 on NALM-6 (Homo sapiens) pchembl=4.08.

T2 [human]
Find a scaffold + fragment decomposition for this molecule and tell me what each piece is. For the affected property axes (alogp, np_likeness, qed) and bioactivity stypes (IC50), report the signed change as combined — scaffold (positive = the fragment increased that axis). For every alert set (PAINS, Glaxo, BMS, Dundee, MLSMR), state whether it was introduced, removed, or preserved.

T3 [gpt]
Scaffold:
  [SELFIES]
  [C][=C][C][C][Branch1][C][C][N][C][Ring1][=Branch1][=O]
 Fragment:
  [SELFIES]
  [C][C][C][C]
Effect:
  alogp: +1.560
  qed: +0.050
  np_likeness: +0.200
  IC50 on L1210 (Mus musculus): −0.47
  PAINS: preserved
  Glaxo: preserved
  BMS: preserved
  Dundee: preserved
  MLSMR: preserved

Figure 4: A SFT example for the backward direction: starting from a profile of the original molecule, the model is then asked to decompose it into a scaffold and a substituent and report the property changes attributable to the substituent.

Task 1 prompt

```
Predict the molecular properties of the following molecule.
Molecule (SELFIES):
  [C][N][Branch2][Ring1][=Branch1][C][C][C][C][C][N][Ring1][Branch1][C][=N][C][=N][C][NH1][C][
=C][C][Ring1][=Branch2][=Ring1][Branch1][C][=Branch1][C][=O][C][C][#N]
STRICT OUTPUT FORMAT — no reasoning, no commentary, no markdown, no preamble.
Respond with ONLY the lines below, one per line:
  <name>=<value>
Properties to predict (12 numeric): psa, alogp, hba, hbd, mw, qed, ro5_viol, aromatic_rings,
heavy_atoms, np_likeness, rtb, ro3_pass
Alerts to predict (5, yes/no): PAINS, Glaxo, BMS, Dundee, MLSMR
Bioactivity targets to predict (output pchembl value):
  - IC50 on Tyrosine-protein kinase JAK3 (Homo sapiens)
```

Ground truth.

```
Properties: psa=88.91, alogp=1.30, hba=5, hbd=1, mw=298.35, qed=0.92, ro5_viol=0,
aromatic_rings=2, heavy_atoms=22, np_likeness=−1.25, rtb=4, ro3_pass=no.
Alerts: PAINS=no, Glaxo=no, BMS=no, Dundee=no, MLSMR=yes.
Bioactivity: IC50 on Tyrosine-protein kinase JAK3 (Homo sapiens) pchembl=6.18.
```

Figure 5: An example of a MolSC-Bench Task 1 prompt used for existing model inference, with the ground-truth profile shown below it.

Task 2 prompt

```
A scaffold molecule (SELFIES):
  [C][C][C][C][C][N][Ring1][Branch1][C][=N][C][=N][C][NH1][C][=C][C][Ring1][=Branch2][=Ring1][
Branch1]

This scaffold has the following axis values:
  - psa = 44.81
  - alogp = 1.95
  - qed = 0.77
  - np_likeness = −0.95
  - MLSMR = no
  - IC50 on Tyrosine-protein kinase JAK3 (Homo sapiens) = 7.09

A substituent fragment (SELFIES):
  [C][N][C][=Branch1][C][=O][C][C][#N]
is attached to the scaffold to form a new molecule.

Predict the new values of the SAME axes for the resulting molecule.
STRICT OUTPUT FORMAT − no reasoning, no commentary, no markdown, no preamble.
Respond with ONLY one line per axis in this exact format:
  <axis>=<value>

Numeric axes: output a number. Alert axes (PAINS, Glaxo, BMS, Dundee, MLSMR): output yes or no.

Axes to predict:
  - psa
  - alogp
  - qed
  - np_likeness
  - MLSMR
  - IC50 on Tyrosine-protein kinase JAK3 (Homo sapiens)
```

Ground truth (signed contribution = combined − scaffold).

```
psa : 44.81 → 88.91 (Δ=+44.10, increase)
alogp : 1.95 → 1.30 (Δ=−0.65, decrease)
qed : 0.77 → 0.92 (Δ=+0.15, increase)
np_likeness : −0.95 → −1.25 (Δ=−0.30, decrease)
MLSMR (alert) : no → yes (introduced)
IC50 on JAK3 : 7.09 → 6.18 (Δ=−0.91, decrease)
```

Figure 6: An example of a MolSC-Bench Task 2 prompt used for existing model inference, with the ground-truth profile shown below it.

Turn	Paraphrasings
T0 (human) Profile request	(1) Predict the full molecular properties, structural alerts, and bioactivity of this molecule: {mol}. (2) What are the molecular properties, structural alerts, and target bioactivity of {mol}? (3) For the molecule {mol}, give a complete profile of its physicochemical properties, alert hits, and bioactivity. (4) I have a molecule: {mol}. Provide its properties, alerts, and bioactivity profile. (5) Describe the molecular properties, structural alerts, and bioactivity of {mol}.
T1 (model) Profile response	(1) Properties: {props}. Structural alerts: {alerts}. Bioactivity: {bio}. (2) Property profile - {props}. Alert flags - {alerts}. Bioactivity - {bio}. (3) Physicochemical: {props}. Alerts: {alerts}. Activity: {bio}. (4) Here is the full profile. Properties: {props}. Alerts: {alerts}. Bioactivity: {bio}. (5) Computed properties: {props}. Structural alert hits: {alerts}. Measured bioactivity: {bio}.
T2 (human) Attachment request	(1) Apply substituent {sub} to the scaffold and tell me the resulting combined molecule. For the affected property axes ({affected_props}) and bioactivity stypes ({affected_bio}), report the signed change as combined − scaffold. For every alert set (PAINS, Glaxo, BMS, Dundee, MLSMR), state whether it was introduced, removed, or preserved. (2) Attach fragment {sub} to the scaffold and give me the combined molecule. For each affected property axis ({affected_props}) and each affected bioactivity stype ({affected_bio}), list the signed change combined − scaffold. Report the 3-way direction (introduced / removed / preserved) for all five alert sets. (3) Bond fragment {sub} onto the scaffold and write out the combined molecule. Give the signed change (combined − scaffold) for: {affected_props}. Do the same for these bioactivity stypes: {affected_bio}. For each of the five alert sets, say introduced, removed, or preserved. (4) Combine the scaffold with fragment {sub} and output the resulting molecule. For each affected property axis ({affected_props}) and bioactivity stype ({affected_bio}), report combined − scaffold. Finish with the 3-way direction for all five alert sets. (5) Take the scaffold, attach {sub}, and describe the resulting combined molecule. For the affected property axes ({affected_props}) and bioactivity stypes ({affected_bio}), give the signed difference combined − scaffold. For every alert set, state whether the fragment introduced it, removed it, or preserved it.
T3 (model) Attachment response	(1) Combined molecule: {cm}. Effect: {chg} (2) After attachment, the molecule is {cm}. Effect: {chg} (3) The combined molecule is {cm}. Per-axis effect: {chg} (4) Result: {cm}. Per-axis effect: {chg} (5) Combined: {cm}. Effect: {chg}

Table 9: Forward direction templates: each turn lists five paraphrasings sampled uniformly at conversation construction time. Placeholders: {mol} is the anchor molecule, {sub} the substituent, {cm} the combined molecule, {props}/{alerts}/{bio} the absolute profile fields, {affected_props}/{affected_bio} comma-separated axis names only, and {chg} the signed-delta list followed by the five alert labels.

Turn	Paraphrasings
T0 (human) Profile request	(1) Predict the full molecular properties, structural alerts, and bioactivity of this molecule: {mol}. (2) What are the molecular properties, structural alerts, and target bioactivity of {mol}? (3) For the molecule {mol}, give a complete profile of its physicochemical properties, alert hits, and bioactivity. (4) I have a molecule: {mol}. Provide its properties, alerts, and bioactivity profile. (5) Describe the molecular properties, structural alerts, and bioactivity of {mol}.
T1 (model) Profile response	(1) Properties: {props}. Structural alerts: {alerts}. Bioactivity: {bio}. (2) Property profile — {props}. Alert flags — {alerts}. Bioactivity — {bio}. (3) Physicochemical: {props}. Alerts: {alerts}. Activity: {bio}. (4) Here is the full profile. Properties: {props}. Alerts: {alerts}. Bioactivity: {bio}. (5) Computed properties: {props}. Structural alert hits: {alerts}. Measured bioactivity: {bio}.
T2 (human) Decomposition request	(1) Find a scaffold + fragment decomposition for this molecule and tell me what each piece is. For the affected property axes ({affected_props}) and bioactivity stypes ({affected_bio}), report the signed change as combined − scaffold. For every alert set (PAINS, Glaxo, BMS, Dundee, MLSMR), state whether it was introduced, removed, or preserved. (2) Decompose this molecule into scaffold and fragment. List the signed change (combined − scaffold) for each affected property axis ({affected_props}) and each affected bioactivity stype ({affected_bio}). Report the 3-way direction for all five alert sets. (3) Break this molecule into a scaffold and a fragment. Give the signed change combined − scaffold for: {affected_props}. Do the same for these bioactivity stypes: {affected_bio}. For each of the five alert sets, say introduced, removed, or preserved. (4) Identify a scaffold + fragment decomposition for this molecule and write both out. For each affected property axis ({affected_props}) and bioactivity stype ({affected_bio}), give combined − scaffold. Finish by reporting all five alert sets as introduced, removed, or preserved. (5) Split this molecule into a scaffold and a fragment. Give the signed difference combined − scaffold for the affected property axes ({affected_props}) and the affected bioactivity stypes ({affected_bio}). For every alert set, state whether the fragment introduced it, removed it, or preserved it.
T3 (model) Decomposition response	(1) Scaffold: {sc}. Fragment: {sub}. Effect: {chg} (2) The scaffold is {sc} and the fragment is {sub}. Effect: {chg} (3) After decomposition: scaffold {sc}, fragment {sub}. Per-axis effect: {chg} (4) Decomposition — scaffold: {sc}; fragment: {sub}. Effect: {chg} (5) Decomposed: scaffold {sc}, fragment {sub}. Per-axis effect: {chg}

Table 10: Backward direction templates: each turn lists five paraphrasings. Placeholders follow Table 9, with {sc} denoting the scaffold (used in T3).

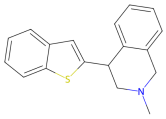
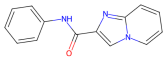
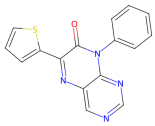
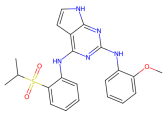
Label	Structure	Textual Representation
Substituent		**SMILES:** C1COCCN1 **SELFIES:** [C][C][O][C][C][N][Ring1][=Branch1]
Scaffold A		**SMILES:** CN1Cc2ccccc2C(c2cc3ccccc3s2)C1 **SELFIES:** [C][N][C][C][=C][C][=C][C][=C][Ring1][=Branch1][C][Branch1][=C][C][=C][C][=C][C][=C][C][=C][Ring1][=Branch1][S][Ring1][=Branch2][C][Ring2][Ring1][Ring1]
Scaffold B		**SMILES:** O=C(Nc1ccccc1)c1cn2ccccc2n1 **SELFIES:** [O][=C][Branch1][#Branch2][N][C][=C][C][=C][C][=C][Ring1][=Branch1][C][=C][N][C][=C][C][=C][C][Ring1][=Branch1][=N][Ring1][=Branch2]
Scaffold C		**SMILES:** O=c1c(-c2cccs2)nc2cncnc2n1-c1ccccc1 **SELFIES:** [O][=C][C][Branch1][Branch2][C][=C][C][=C][S][Ring1][Branch1][=N][C][=C][N][=C][N][=C][Ring1][=Branch1][N][Ring1][#C][C][=C][C][=C][C][=C][Ring1][=Branch1]
Scaffold D		**SMILES:** COc1ccccc1Nc1nc(Nc2ccccc2S(=O)(=O)C(C)C)c2cc[nH]c2n1 **SELFIES:** [C][O][C][=C][C][=C][C][=C][Ring1][=Branch1][N][C][=N][C][Branch2][Ring1][=Branch1][N][C][=C][C][=C][C][=C][Ring1][=Branch1][S][=Branch1][C][=O][=Branch1][C][=O][C][Branch1][C][C][C][=C][C][=C][NH1][C][Ring1][Branch1][=N][Ring2][Ring1][=Branch1]

Table 11: Molecules used in the context-dependent contribution analysis (Table 5).