MolecularGPT: Open Large Language Model (LLM) for Few-Shot Molecular Property Prediction

Anonymous ACL submission

Abstract

Molecular property prediction (MPP) is a fundamental and crucial task in drug discovery. However, prior methods are limited by the requirement for a large number of labeled molecules and their restricted ability to generalize for unseen and new tasks, both of which are essential for real-world applications. To address these challenges, we present MolecularGPT for few-shot MPP. From a perspective on instruction tuning, we fine-tune large language models (LLMs) based on curated molecular instructions spanning over 1000 property prediction tasks. This enables building a versatile and specialized LLM that can be adapted to novel MPP tasks without any fine-tuning through zero- and few-shot in-context learning (ICL). MolecularGPT exhibits competitive in-context reasoning capabilities across 10 downstream evaluation datasets, setting new benchmarks for few-shot molecular prediction tasks. More importantly, with just two-shot examples, MolecularGPT can outperform standard supervised graph neural network methods on 4 out of 7 datasets. It also excels state-of-the-art LLM baselines by up to 16.6% increase on classification accuracy and decrease of 199.17 on regression metrics (e.g., RMSE) under zero-shot. This study demonstrates the potential of LLMs as effective few-shot molecular property predictors. Our model and curated instruction set will be open-sourced.

1 Introduction

The discovery of molecules with desired functional properties is crucial for advancements in fields such as medicine (Stokes et al., 2020; Wong et al., 2024; Koscher et al., 2023; Abramson et al., 2024) and material (Merchant et al., 2023; Kang et al., 2023). Molecular property prediction (MPP), which employs deep learning techniques to predict molecules' functional properties, has proven effective in accelerating the drug discovery process and reducing associated costs (Wong et al., 2024; Merchant et al., 2023; Kang et al., 2023).

Among them, graph neural networks (GNNs)-based methods (Velickovic et al., 2017; Xu et al., 2019; Kipf and Welling, 2017; Gilmer et al., 2017; Hamilton et al., 2017) have achieved state-of-the-art results in the past few years. However, these methods (Li et al., 2022; Liu et al., 2022; Stärk et al., 2022) are limited in supervised settings, contradicting with practical needs as annotating molecules is both expensive and time-consuming. Furthermore, the task-specific supervised learning process may hurdle the model's adaptation to new tasks, limiting its generalization ability in open-world scenarios.

Inspired by this, several recent endeavors have aimed to enable zero-shot reasoning for MPP (Seidl et al., 2023; Zhao et al., 2024) by integrating both natural language and molecular representations. CLAMP (Seidl et al., 2023) is a text-molecule model that aligns pairs of chemical text (e.g., descriptions of molecular properties) and molecule graphs through contrastive learning. Subsequently, the bioactivity of a query molecule is classified by measuring the similarity between its molecular representation and corresponding bioassay description. While effective, CLAMP is limited to classification tasks and is not a generative model.

In contrast, another line of research in LLMs (Zhao et al., 2024) integrates molecule graphs and task descriptions into a unified generative LLM. This approach enables zero-shot reasoning for molecular property prediction across both classification and regression tasks. However, the inclusion of an additional architectural design restricts it from performing few-shot molecular property predictions, a capability naturally supported by standard LLMs.

To date, there's no LLM-based method in the molecular domain fully inherits the generalization and ICL abilities of LLMs as seen in the NLP field,

which raises a research question: *Can LLMs be fine-tuned for generic MPP, enabling the resultant model to generalize to a variety of unseen tasks and inherit LLMs' few-shot ICL ability?*

In this work, we aim to bridge the gap and present MolecularGPT, the first instructionally tuned LLM that can generalize to a variety of novel MPP tasks while retaining its zero-shot and few-shot in-context reasoning abilities. Specifically, MolecularGPT adopts the SMILES (Weininger, 1988) representation of molecules as a unified graph-to-string transformation for instruction construction, as it precisely translates molecules' chemical structures into a string of atomic symbols and chemical bonds based on a set of rules. To fully utilize the graph structures in molecules, we introduce structure-aware few-shot instructions, which incorporate the top-k neighbors, globally retrieved based on their similarities, of each molecule as complementary information for instruction design. This design aligns the instruction tuning and inference prompt format of MolecularGPT, making it naturally applicable for few-shot ICL. Additionally, to balance zero-shot and few-shot reasoning capabilities, we explore various combination options and empirically find that a hybrid instruction set, including both zero-shot and few-shot instructions, enables MolecularGPT to perform well in both zero-shot and few-shot property predictions. Our **main contributions** are summarized below:

- We study how to adapt pre-trained LLMs to molecular field, enabling effective few-shot MPP in the ICL fashion. Specifically, we propose MolecularGPT, the first instructionally fine-tuned LLM that supports few-shot property prediction on unseen tasks without any fine-tuning.

- We introduce the concept of structure-aware few-shot instruction to better adapt LLMs with molecular field. Unlike existing efforts (Seidl et al., 2023; Zhao et al., 2024; Zhang et al., 2023) that focus on fusing graph structures and SMILES representations in a model-centric perspective, we maliciously combine them in a data-centric manner by constructing global structure-aware few-shot demonstrations.

- We devise a hybrid instruction set to inherit the few-shot ICL capability of LLMs. This set is a mix of both few-shot and zero-shot instructions that span over 1000 MPP tasks including both classification and regression tasks across biological, chemical, and quantum mechanical domains, resulting in 3.5GB training tokens. This diversified instruction set has been empirically proved to be effective in adapting LLMs for MPP tasks.

- We extensively experimented on 10 molecular property benchmarks across different scales and tasks to validate the effectiveness of MoelcularGPT. Our empirical results demonstrate that MoelcularGPT outperforms the leading LLM baselines (e.g., GIMLET, LLaMA-7b (Touvron et al., 2023), and LLaMA-13B (Touvron et al., 2023)), with up to an average 16.6% improvement across all classification tasks. Additionally, with just two-shot examples, MolecularGPT surpass standard supervised GNN methods on 4 out of 7 datasets, setting new benchmarks for few-shot molecular property tasks.

2 Related work

GNNs-based MMP GNNs (Velickovic et al., 2017; Xu et al., 2019; Kipf and Welling, 2017; Stärk et al., 2022) perform MPP tasks by constructing models between molecular graphs and properties. Though have achieved great success (Gilmer et al., 2017; Hamilton et al., 2017; Li et al., 2022; Liu et al., 2022), these supervised models solely utilize structure information, neglecting the wealthy knowledge contained in texts derived from wet lab experiments. More importantly, they are implicitly trained for each task without explicit natural language instructions, which can not directly generalize to new tasks.

Pretrain-finetune based molecular language models To utilize the chemical knowledge in texts, molecular language models (Liu et al., 2023b; Edwards et al., 2022; Pei et al., 2023; Zhang et al., 2023; Liu et al., 2023d) aim to integrate natural language and molecular representations for joint reasoning. These models (Su et al., 2022; Zeng et al., 2022; Liu et al., 2023c, 2024; Li et al., 2024) involve two stages: pre-training and fine-tuning. The pre-training phase primarily focuses on learning molecular representations and their associated textual descriptions through masked language modeling, contrastive learning or next token prediction. However, they still require fine-tuning on particular MPP downstream tasks, thereby limiting their generalization abilities to new tasks.

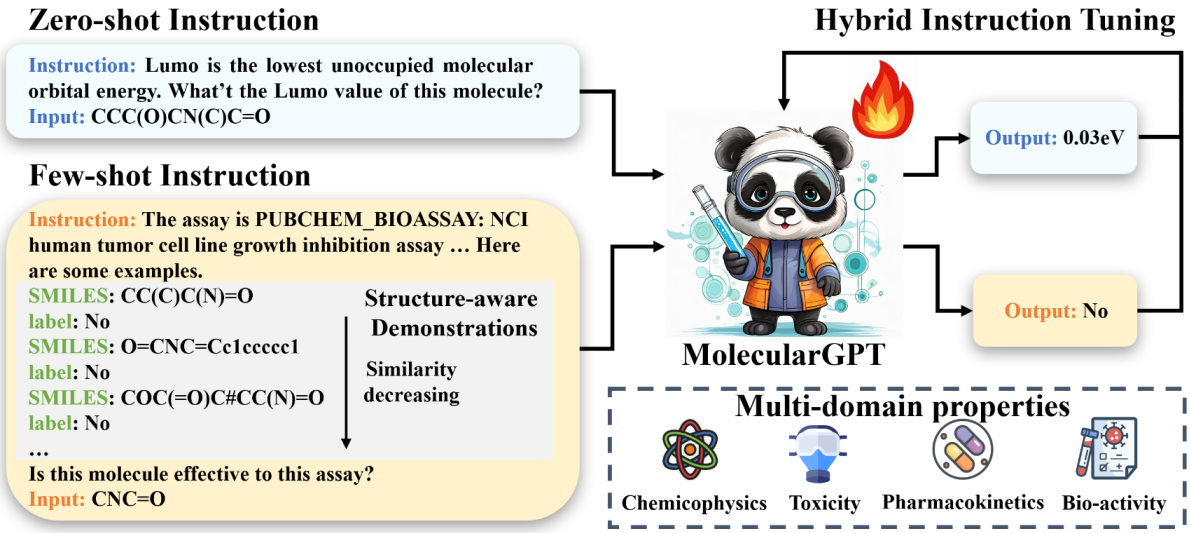

Figure 1: The proposed MolecularGPT framework. To instructionally fine-tune LLMs for MPP tasks, we construct a hybrid instruction set that includes both zero-shot and few-shot instructions across more than 1000 property tasks. Each few-shot instruction adaptively selects the query molecule's top-k neighboring molecules as labeled demonstrations for prompt design.

Instruction tuning based molecular language models To address this, recent efforts in molecular language modeling (Fang et al., 2023; Zhao et al., 2024) aim to explicitly align molecular graphs with their properties through instruction tuning (Longpre et al., 2023). For instance, GIMLET (Zhao et al., 2024) integrates molecular graphs with instruction languages for fine-tuning LLMs. GIMLET achieves effective zero-shot ICL for new tasks but lacks few-shot ICL capability due to its generalized position embedding and decoupled attention designs. Mol-Instructions (Fang et al., 2023) is a close work to us, but it fine-tunes LLMs with only three properties tasks and neglects inter-molecular correlations, significantly limiting its zero-shot and few-shot ICL performances. In contrast, we curate a diverse instruction set covering 1000 property tasks and introduce structure-aware few-shot instructions to significantly enhance the zero-shot and few-shot reasoning capabilities of LLMs in MPP tasks. More details about these property tasks can be found in Appendix A.1

3 Method

In this section, we present the proposed MolecularGPT, as shown in Fig. 1. First, we discuss the general instructional fine-tuning pipeline to adapt LLMs for MPP tasks (in Section 3.1). Next, we elaborate on a structure-aware few-shot instruction design strategy to effectively incorporate graph structures among molecules (in Section 3.2). Finally, we illustrate a hybrid instruction tuning approach that enhances both the zero-shot and few-shot reasoning capabilities of LLMs for MPP tasks (in Section 3.3).

Notations and Problem Formulation. Given a set of n molecular graphs $D = \{(G_i, y_i)|i \in 1, 2, ..., n\}$, where $G_i = (\mathcal{V}, \mathcal{E})$ represents the i-th molecule and y_i is the ground-truth property (e.g., categorical label or numerical score). Here, $\mathcal{V}$ and $\mathcal{E}$ denote the node set and edge set, respectively. The goal of molecular instruction tuning is to fine-tune a LLM model f_θ by fitting a set of training instructions S_D (i.e., (*input*, *output*) pairs) constructed from D, so that the fine-tuned LLM can be directly applied to make property predictions for unseen tasks or molecules, i.e., $D_{\text{test}} = \{(G_j, y_j)|j = 1, 2, ..., m\}$ with $D \cap D_{\text{test}} = \emptyset$.

While conceptually simple, successfully achieving molecular instruction tuning involves addressing several research challenges. **C1**: how can we unify molecules of varying sizes, densities, and domains into a consistent format, ensuring that important molecular information in D and D_{test} is consistently incorporated? **C2**: given that graph structures are crucial for molecular analysis, as verified in GNN studies, how can we effectively include these structures in molecular instruction tuning? **C3**: considering that molecule annotation is notoriously expensive and time-consuming, how can we enable the fine-tuned LLM to benefit from few-shot scenarios where only a few labeled

molecules are available in real-world applications?

3.1 SMILES-based Molecular Instruction Tuning: A Unified Step

To improve the generalization capability of fine-tuned LLM for MPP tasks (**C1**), prior models often utilize GNNs (Seidl et al., 2023) or graph transformer (Zhao et al., 2024) as encoders to map molecular graphs into hidden representations. When a graph encoder is well-trained, it can be used to map molecules in D or D_{test} into a shared hidden space, providing a unified hidden expression. However, as discussed above, this assumption may not hold in practice, as training a unified graph encoder for cross-domain molecules still remains an open-question (Liu et al., 2023a).

To address this, we aim to employ the well-known graph-derived linear strings (Weininger, 1988; Krenn et al., 2020) of molecular graphs, such as SMILES (Weininger, 1988), for instruction tuning. Unlike GNN encoders, SMILES translates molecules' chemical structure into a string of atomic symbols and chemical bonds (single, double, or triple) based on a set of rules (Qian et al., 2023). This precise translation not only accounts for the graph structure within each molecular graph, but also generalizes readily to arbitrary molecular graphs, providing a universal expression foundation for different types of molecules. Following standard instruction tuning protocol (Christofidellis et al., 2023; Fang et al., 2023; Zhang et al., 2023; Liu et al., 2024; Li et al., 2024), the molecular instruction set S_D can be generated by the following prompt template $T = \{Q, I, R\}$ based on D, regarding as a zero-shot instruction template.

```
### Instruction: {instruction}
### Input: {inputs}
### Response: {output}.
```

Here, the instruction question Q, SMILES strings of query molecule I, and property label R are mapped to the {instruction}, {inputs}, and {output} components, respectively.

3.2 Structure-Aware Molecular Instruction Tuning: Graph Structure Matters

So far, we have illustrated how to incorporate graph structure within each molecule into instruction via the zero-shot instruction template T. However, this approach may result in subpar prediction performance due to the neglect of correlations between molecules. To address this, we introduce structure-aware instruction tuning (**C2**), which aims to incorporate inter-molecular structures into the prompt template. The high-level idea is to utilize similar molecules as demonstrations to enhance LLM reasoning.

To achieve this, given a query molecule $G_i \in D$, we identify its top-K nearest molecules in D based on the following retrieval module.

$$N_{G_i} = \text{topK}(G_i, D, K), \tag{1}$$

where N_{G_i} is the retrieved neighborhood set with K molecules. topK() is a search algorithm based on the similarity between molecules. Specifically, we estimate the similarity between molecules by calculating their Tanimoto coefficient (Tanimoto, 1958) based on their MACCS Keys (Durant et al., 2002). Notably, MACCS Keys, comprising 166 binary keybits, provides a unified representation for molecules and has been widely adopted in many molecule retrieval systems, such as USearch (Vardanian, 2023).

Utilizing N_{G_i}, we can transform the zero-shot template into a few-shot version $T_{shot} = \{C, I, R\}$, where C represents the k-shot instruction question, extending Q with structurally similar molecule demonstrations extracted from N_{G_i}. Specifically, let (m_i, y_i) represents the i-th similar molecule-property pairs in N_{G_i}. Additionally, considering that the order of demonstrations may significant impact prompt design (Mosbach et al., 2023), we arrange these k demonstrations in a descending order based on their similarity scores. The C is formally expressed as:

$$C = \{Q, ((m_1, y_1), ..., (m_i, y_i), ..., (m_k, y_k))\}. \tag{2}$$

Similar to T, the extended question C in the few-shot instruction template T_{shot} will correspond to the {instruction} of the template in Section 3.1. In experiments, we empirically observed that including the target property of molecular neighbors as input in few-shot scenarios improves performance. This approach is reasonable because T_{shot} serves as a few-shot in-context prompt, akin to those widely used in the NLP domain, where the most similar neighbors are selected as demonstrations.

3.3 Hybrid Molecular Instruction Tuning: Better Few-Shot Learner

Given the advanced structure-aware instruction template T_{shot}, one can easily construct the instruction training set S_D by applying T_{shot} on each

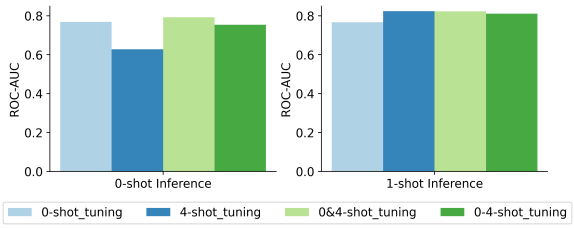

Figure 2: The performance on Cyp450 test dataset.

molecule in D. Then, we fine-tune a pre-trained LLM by optimizing the following training loss:

$$\mathcal{L}(\theta) = \sum_{(C_i,I_i,R_i))\in\mathcal{S}_D} -\log f_\theta(R_i|C_i,I_i). \quad (3)$$

Here, f_θ is a pre-trained LLM with parameter θ. In practice, we initialize f_θ as LLaMA2-7b-chat (Touvron et al., 2023) and adopt QLoRA (Dettmers et al., 2024) to speedup the training.

While T_{shot} appears effective, it may degrade the zero-shot reasoning capability of fine-tuned LLM due to the explicit graph structures among molecules. To verify this, we conducted a toy example by fine-tuning LLaMA2-7b-chat on different K-shot instruction sets. Specifically, Fig. 2 reports the zero-shot and one-shot inference results on the CYP450 dataset for $K = 0$ and $K = 4$.

In Fig. 2, we can observe an obvious trade-off between zero-shot and one-shot performance with respect to the instruction set. For example, when fine-tuning LLaMA2 on the 0-shot instruction set constructed using the T template, the resulting *0-shot_tuning* model performs well in zero-shot scenarios but underperforms in one-shot scenarios. Conversely, when fine-tuning on the 4-shot instruction set constructed using T_{shot} with $K = 4$, the resulting *4-shot_tuning* model excels in one-shot settings but underperforms in zero-shot cases.

This observation motivates us to introduce a **hybrid instruction set** S_D^h, combining the strengths of both the zero-shot instruction template T and the few-shot instruction template T_{shot}. Specifically, S_D^h is derived from a combination of 0, 1, 2, 3, and 4-shot instruction templates. In Fig. 2, we can see that our hybrid instruction tuned models, *0&4-shot_tuning* and *0-4-shot_tuning*, consistently outperforms others in both zero-shot and one-shot scenarios. Further details can be found in Section 4.3.

4 Experiment

In our experimental framework, we aim to answer three primary research questions: **RQ1:** Can MolecularGPT effectively and robustly handle new property prediction tasks through zero- and few-shot ICL? **RQ2:** What is the optimal design for in-context instruction set to improve MolecularGPT's generalization and ICL abilities during tuning? **RQ3:** How does the number, order, and diversity of in-context examples affect the performance of MolecularGPT?

4.1 Experimental Setup

Datastes Consistent with the GIMLET setting, we employ the MoleculeNet benchmark (Wu et al., 2018) and CYP450 (Li et al., 2018) datasets as our downstream datasets, totally 657 MMP tasks. More details about datasets can be found in Appendix A.1. We employ ROC-AUC as the evaluation metric for classification tasks, while the Root Mean Square Error (RMSE) for regression tasks.

Baselines Our baseline selection aligns with the approach used in GIMLET (Zhao et al., 2024), which can be categorized into two primary types: language models for directly inference and graph representation models totally fine-tuned on downstream tasks. The language models include XVPLM (Zeng et al., 2022), MoMu (Su et al., 2022), Galactica-125M (Taylor et al., 2022), Galactica-1.3B (Taylor et al., 2022), and GIMLET. And the finetuned molecular representation models comprise GCN (Kipf and Welling, 2017), GAT (Velickovic et al., 2017), GIN (Xu et al., 2019), Graphormer (Ying et al., 2021), and Graphormer-p, which is pretrained on Graphormer using datasets in GIMLET. We present the zero-shot results of these language models and the fine-tuned results of supervised models from GIMLET. Additionally, we consider general large language models: LLaMA-chat-7B and LLaMA-chat-13B as our baselines, which demonstrate ICL capabilities.

4.2 Performance Evaluation

As the results presented in Tab. 1, 2 respectively, MolecularGPT can achieve competitive performance on classification and regression tasks under both zero-shot and few-shot settings. We answer the **RQ1** with more details as follows.

① **The MolecularGPT demonstrates superior performance compared with other language models in zero-shot learning.** In comparison

Table 1: Performance over Bio-activity, Toxicity, and Pharmacokinetic classification tasks. Highlights are the **first**, **second**, and **third** best results of zero- and few-shot performances. In supervised finetuned models, we also mark the **highest** and **lowest** results.

Method	Model Size	Type	BACE	HIV	MUV	Avg.bio	Tox21	ToxCast	Avg.tox	BBBP	CYP450	Avg.pha
XVPLM	110M		0.5126	0.6120	0.6172	0.5806	0.4917	0.5096	0.5007	0.6020	0.5922	0.5971
MoMu	113M		0.6656	0.5026	0.6051	0.5911	0.5757	0.5238	0.5498	0.4981	0.5798	0.5390
Galactica-125M	125M	0-Shot	0.4451	0.3671	0.4986	0.4369	0.4964	0.5106	0.5035	0.6052	0.5369	0.5711
Galactica-1.3B	1.3B		0.5648	0.3385	0.5715	0.4916	0.4946	0.5123	0.5035	0.5394	0.4686	0.5040
GIMLET	64M		0.6957	0.6624	**0.6439**	0.6673	0.6119	0.5904	0.6011	0.5939	0.7125	0.6532
		0-shot	0.4911	0.6060	0.5554	0.5508	0.5481	0.4693	0.5087	0.3671	0.4198	0.3935
		1-shot	0.4911	0.6060	0.5554	0.5508	0.5481	0.4954	0.5218	0.3671	0.4198	0.3935
LLaMA2-chat-7B	7B	2-shot	0.6930	0.6587	0.5085	0.6201	0.6052	0.5010	0.5531	0.5459	0.5807	0.5633
		4-shot	**0.7685**	0.6781	0.4685	0.6384	0.6199	0.5025	0.5612	0.5423	0.6092	0.5758
		6-shot	0.7180	0.7058	0.5133	0.6457	0.6334	0.5228	0.5781	0.5161	0.6145	0.5653
		0-shot	0.6561	0.6797	0.4924	0.6094	0.5178	0.5382	0.5280	0.5630	0.4716	0.5173
		1-shot	**0.7534**	0.6419	0.4828	0.6260	0.6011	0.5591	0.5801	0.5372	0.5995	0.5684
LLaMA2-chat-13B	13B	2-shot	0.7454	0.6694	0.4886	0.6345	0.5907	0.5371	0.5639	0.4633	0.5784	0.5209
		4-shot	0.7471	0.7235	0.4792	0.6499	0.5750	0.5489	0.5620	0.5276	0.5555	0.5416
		6-shot	0.7412	0.6911	0.5267	0.6530	0.5650	0.5527	0.5589	0.5669	0.5787	0.5728
		0-Shot	0.6212	**0.7128**	0.6253	0.6531	0.5893	0.5669	0.5781	0.6373	0.8031	0.7202
		1-Shot	**0.7520**	**0.7172**	0.6327	**0.7006**	0.6529	**0.5968**	0.6249	0.6999	0.8229	0.7614
MolecularGPT(ours)	7B	2-Shot	0.7218	**0.7204**	0.6338	**0.6920**	**0.6573**	0.5945	**0.6259**	**0.7260**	**0.8275**	**0.7768**
		4-shot	0.7228	0.6893	**0.6419**	**0.6847**	**0.6577**	**0.5978**	**0.6278**	**0.7168**	**0.8252**	**0.7710**
		6-shot	0.7181	0.6554	**0.6561**	0.6765	**0.6629**	**0.5965**	**0.6297**	**0.7139**	**0.8289**	**0.7714**
GCN	0.5M		0.736	0.757	0.732	0.742	0.749	**0.633**	0.691	**0.649**	**0.8041**	**0.7266**
GAT	1.0M		**0.697**	**0.729**	**0.666**	**0.697**	0.754	0.646	0.700	0.662	0.8281	0.7451
GIN	1.8M	Finetuned	0.701	0.753	0.718	0.724	**0.740**	0.634	**0.687**	0.658	0.8205	0.7392
Graphormer	48M		0.7760	0.7452	0.7061	0.7424	0.7589	0.6470	0.7029	0.7015	0.8436	0.7725
Graphormer-p	48M		**0.8575**	**0.7788**	**0.7480**	**0.7948**	**0.7729**	**0.6649**	**0.7189**	**0.7163**	**0.8877**	**0.8020**

to language models under zero-shot inference, our model demonstrates enhanced performance across classification and regression tasks. In terms of GIMLET, MolecularGPT surpasses it in HIV, BBBP, and CYP450 classification datasets as well as in FreeSolv and Lipo regression datasets under zero-shot condition. Compared to the LLaMA-7B and LLaMA-13B, our models exhibit a significant improvement, an average improvement of 16.6% and 9.9% in ROC-AUC across all classification tasks and average decrease of 5.96 and 199.17 in RMSE across all regression tasks correspondingly, indicating the chemical knowledge have been effectively imbued into LLaMA through our tuning.

② **MolecularGPT establishes a new benchmark in few-shot ICL across all tasks and outperforms the SOTA supervised models in certain conditions.** When compared to the zero-shot learning, MolecularGPT demonstrates an average enhancement of 4.6% in ROC-AUC across all classification tasks under one-shot condition. Compared to GIMLET, MolecularGPT exhibits an average improvement of 5.5% and 5.8% on classification tasks under one-shot and two-shot settings respectively. Even by one-shot ICL, MolecularGPT displays comparable performance with GCN and GIN on 3 out of 7 classification datasets. By two-shot ICL, MolecularGPT matches the performance of GAT on 4 out of 7 classification datasets. Remarkably, MolecularGPT even outperforms the highest-performing finetuned model, Graphormer-p, on BBBP dataset under two-shot condition with ROC-AUC of 0.7260 compared to 0.7163.

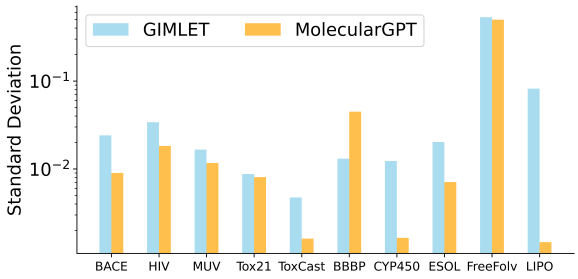

Figure 3: Standard deviation for GIMLET and MolecularGPT in response to 5 types of instructions.

③ **The exceptional robustness of MolecularGPT is validated across different tasks.** Given the diversity and flexibility of natural language, we aim to evaluate the robustness of MolecularGPT against various instructions. Adhering to the downstream datasets used in GIMLET, which provides five distinct types of instructions. We calculate the standard deviation of the ROC-AUC or RMSE

Table 2: Performance on Physicalchemical regression tasks. The highlight style is the same as Tab. 1

Method	Type	ESOL	FreeSolv	Lipo	Avg.phy
XVPLM		-	-	-	-
MoMu	0-Shot	-	-	-	-
GIMLET		**1.132**	**5.103**	1.345	**2.527**
	0-shot	7.227	15.912	2.329	8.489
	1-shot	1.819	525.478	1.204	176.167
LLaMA2-chat-7B	2-shot	3.856	41.168	1.128	15.384
	4-shot	5.940	66.593	1.112	24.548
	6-shot	7.569	55.933	1.112	21.538
	0-shot	281.617	321.313	2.194	201.708
	1-shot	9.405	11.356	1.427	7.396
LLaMA2-chat-13B	2-shot	27.717	39.254	1.420	22.797
	4-shot	643.408	9.589	1.462	218.153
	6-shot	6.481	154.635	1.363	54.160
	0-Shot	**1.471**	**4.975**	1.157	**2.534**
	1-Shot	1.496	5.248	1.058	2.601
MolecularGPT(ours)	2-Shot	1.489	5.226	**1.015**	2.577
	4-Shot	1.535	5.375	**1.045**	2.652
	6-Shot	**1.465**	**5.046**	**1.023**	**2.511**
GCN		**1.331**	2.119	0.760	1.403
GAT		1.253	2.493	0.770	1.505
GIN	Finetuned	1.243	**2.871**	**0.781**	**1.632**
Graphormer		0.901	2.210	0.740	1.284
Graphormer-p		**0.804**	**1.850**	**0.675**	**1.110**

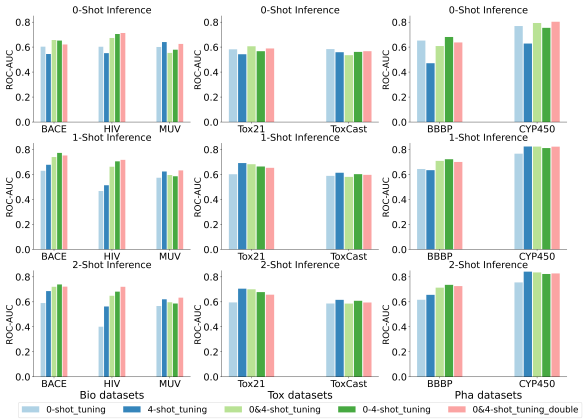

Figure 4: The performance of MolecularGPT on classifcation tasks tuning with different types of instruction datasets. We inference them with 0, 1, and 2-shot examples. (*0&4-shot* indicates hybrid of 0 and 4-shot. *0-4-shot* indicates mix of 0,1,2,3,4-shot. *tuning_double* indicates double the instruction set size.)

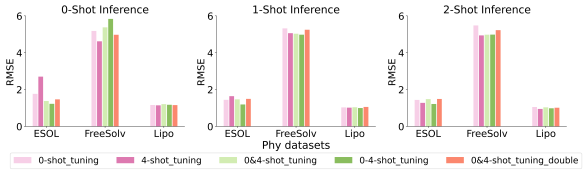

Figure 5: The performance of MolecularGPT on regression tasks tuning with different types of instruction datasets. The setting is same as in Fig. 4.

metrics derived from these five instruction datasets. Comparative results with GIMLET is presented in Fig. 3. It is evident that our model exhibits superior robustness compared to GIMLET across most tasks. This indicts the robustness of MolecularGPT that it genuinely comprehends complex instructions and can handle a range of property prediction tasks without requiring task-specific prompt designs.

4.3 Tuning on Hybrid Instruction Set

To investigate the **RQ2**, we conduct experiments to study the effect of hybrid instruction tuning set as presented in Fig. 4 and 5.

④ **Tuning on property descriptions without demonstrations can improve the zero-shot performance.** As shown in the *0-shot_tuning* in Fig. 4, 5, the model performed satisfactorily on some tasks under zero-shot inference but poorly on many tasks under few-shot inference. We speculated that the zero-shot instruction set imparts some knowledge to LLaMA without significantly enhancing the model's ICL ability.

⑤ **Providing the model with rich retrieved demonstrations would significantly improve its ICL ability.** To test this, we fine-tuned the model on a 4-shot instruction dataset, represented by the *4-shot_tuning* in Fig. 4 and 5. The results indicate an improvement in the model's ICL ability. However, the model's zero-shot generalization remained subpar on many tasks. We surmise that the model may learn shortcuts from the label words of the reference molecules rather than extracting the true relationships between the molecular representations and their properties.

⑥ **Mixed-shot instruction sets are promising to optimize both zero-shot generalization and ICL abilities.** We developed two mixed instruction datasets: a combined *0&4-shot* and a comprehensive mix of $0, 1, 2, 3, 4$-shot *(0-4-shot)* instruction datasets. As shown in *0&4-shot_tuning* and *0-4-shot_tuning* in Fig. 4 and 5, models fine-tuned on mixed-shot instruction datasets demonstrate a significant performance improvement compared to those fine-tuned on 0-shot or 4-shot instruction sets. This trend is consistently observed across various tested scenarios, indicating that our model derives the most benefit from mixed-shot instruction sets.

⑦ **Tuning on larger instruction set have exhibited superior performance across different tasks under both zero and few shot learning.** Models trained with larger datasets have exhibited superior performance on multi functional tasks, as evidenced by the improvements from GPT-2 (Rad-

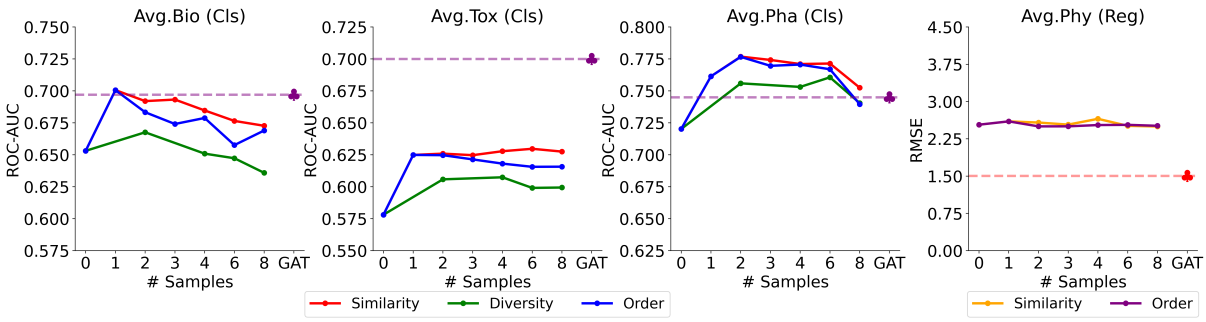

Figure 6: The performance of MolecularGPT on Classifcation (Cls) and Regrassion (Reg) tasks with different in-context inference strategies. To show our model's remarkable capability, we also add the performance of the finetuned model, GAT.

ford et al., 2019) to GPT3 (Brown et al., 2020) and LLaMA2 (Touvron et al., 2023) to LLaMA3 (Meta, 2024). To further enhance MolecularGPT, we double the size of the *0&4-shot* instruction sets. The results represented by the *0&4-shot_tuning_double* in Fig. 4 and 5 suggest that expanding the data scale enhances the model's performance across various tasks either by zero-shot or few-shot learning.

4.4 Hyperparameter Sensitivity Analysis

To fully utilize the ICL ability of MolecularGPT, we now pay attention to the impact of number, order and diversity of in-context demonstrations to discuss the **RQ3**. The results are depicted in Fig 6.

⑧ **MolecularGPT gains significant enhancement with up to 2 demonstrations, but the marginal benefit diminishes with additional retrieval molecules.** We investigate the impact of the number (Ye et al., 2024) of retrieval demonstrations, ranging from 0 to 8 examples based on similarity. The results indicate significant improvement when provided with up to 2 examples on many datasets. However, the performance does not get further improvement with more retrieval molecules. We hypothesize that: 1) More noise will be introduced with the increase of examples that has lower similarity with the query. 2) The maximum input length of 512 tokens with at most 4 examples in instructions constrains the model's capability while handling more examples.

⑨ **Ascending order of similarity for demonstrations is sub-optimal compared to descending order especially with more demonstrations.** We arrange the demonstrations (Lu et al., 2022; Zhao et al., 2021) in a ascending order, placing the most similar examples at the end of k-shot instructions. The results in Fig. 6 show that the ascending order is sub-optimal comparing to descending order, especially with more demonstrations which may be constrained by the model's long context capability. We also assume the model is more adaptable to reasoning with descending order by learning most related knowledge first.

⑩ **Similar retrieved molecule demonstrations provides better performance than diverse demonstrations.** To increase the diversity, we retrieve equal number of molecules from each category (Ma et al., 2024). When the same number of examples is provided within instructions, the retrieval approach based on similarity consistently outperforms the one based on diversity across all classification tasks as shown in Fig. 6. The similarity-based methodology tends to provide examples that align more coherently with the query molecules. In contrast, the diversity-based approach offers a mix of positive and negative examples, which potentially introduce noise and create ambiguity perplexing the language models.

5 Conclusion

In this study, we aim to equip the LLMs, particularly the LLaMA, with an expanded knowledge of molecular properties, enabling it to generalize to out-of-domain prediction tasks through zero-shot and few-shot ICL. We introduce MolecularGPT, a model that has been instruction tuned on over 1000 prediction tasks. Furthermore, we investigate the most effective types of instruction datasets for optimizing the model during both training and inference stages. Our findings demonstrate that MolecularGPT consistently outperforms baseline language models in few-shot scenarios and even surpasses supervised models on multiple datasets. In future work, we plan to incorporate additional molecular modalities and expand into other molecular-related tasks such as molecule captioning.

6 Limitation

In our research, we utilize SMILES strings to represent molecules. However, while effective, this approach overlooks the geometric structure information of real-world molecules, such as the 3D spatial position of each atom in a molecule. This limitation hinders our model's ability to represent molecular structures. Meanwhile, our work focuses solely on property prediction tasks and does not consider foundational tasks such as molecule optimization, molecule generation, and molecule captioning. This may restrict the potential applications of our model in practical settings. Lastly, although our model is compatible with supervised GNN models for classification tasks, we still have some gaps with them in regression tasks as directly generating numbers remains a challenge for nowadays foundational LLMs.

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. 2024. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

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*.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

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

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*.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Yeonghun Kang, Hyunsoo Park, Berend Smit, and Jihan Kim. 2023. A multi-modal pre-training transformer for universal transfer learning in metal–organic frameworks. *Nature Machine Intelligence*, 5(3):309–318.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

Brent A Koscher, Richard B Canty, Matthew A McDonald, Kevin P Greenman, Charles J McGill, Camille L Bilodeau, Wengong Jin, Haoyang Wu, Florence H Vermeire, Brooke Jin, et al. 2023. Autonomous, multiproperty-driven molecular discovery: From predictions to measurements and back. *Science*, 382(6677):eadi1407.

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.

Shuangli Li, Jingbo Zhou, Tong Xu, Dejing Dou, and Hui Xiong. 2022. Geomgcl: Geometric graph contrastive learning for molecular property prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4541–4549.

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

Xiang Li, Youjun Xu, Luhua Lai, and Jianfeng Pei. 2018. Prediction of human cytochrome p450 inhibition using a multitask deep autoencoder neural network. *Molecular pharmaceutics*, 15(10):4336–4345.

Jiawei Liu, Cheng Yang, Zhiyuan Lu, Junze Chen, Yibo Li, Mengmei Zhang, Ting Bai, Yuan Fang, Lichao Sun, Philip S Yu, et al. 2023a. Towards graph foundation models: A survey and beyond. *arXiv preprint arXiv:2310.11829*.

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

Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. 2023b. 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. 2022. Pre-training molecular graph representation with 3d geometry. In *International Conference on Learning Representations*.

Zequn Liu, Wei Zhang, Yingce Xia, Lijun Wu, Shufang Xie, Tao Qin, Ming Zhang, and Tie-Yan Liu. 2023c. Molxpt: Wrapping molecules with text for generative pre-training. In *Annual Meeting of the Association for Computational Linguistics*.

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

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098.

Huan Ma, Changqing Zhang, Yatao Bian, Lemao Liu, Zhirui Zhang, Peilin Zhao, Shu Zhang, Huazhu Fu, Qinghua Hu, and Bingzhe Wu. 2024. Fairness-guided few-shot prompting for large language models. *Advances in Neural Information Processing Systems*, 36.

Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. 2023. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85.

AI Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*.

OpenAI. 2023. GPT-4 technical report. *ArXiv preprint*, abs/2303.08774.

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*.

Chen Qian, Huayi Tang, Zhirui Yang, Hong Liang, and Yong Liu. 2023. Can large language models empower molecular property prediction? *arXiv preprint arXiv:2307.07443*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. 2014. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Philipp Seidl, Andreu Vall, Sepp Hochreiter, and Günter Klambauer. 2023. Enhancing activity prediction models in drug discovery with the ability to understand human language. In *International Conference on Machine Learning*, pages 30458–30490. PMLR.

Hannes Stärk, Dominique Beaini, Gabriele Corso, Prudencio Tossou, Christian Dallago, Stephan Günnemann, and Pietro Liò. 2022. 3d infomax improves gnns for molecular property prediction. In *International Conference on Machine Learning*, pages 20479–20502. PMLR.

Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. 2020. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702.

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.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti

Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ash Vardanian. 2023. USearch by Unum Cloud.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. 2017. Graph attention networks. *stat*, 1050(20):10–48550.

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.

Felix Wong, Erica J Zheng, Jacqueline A Valeri, Nina M Donghia, Melis N Anahtar, Satotaka Omori, Alicia Li, Andres Cubillos-Ruiz, Aarti Krishnan, Wengong Jin, et al. 2024. Discovery of a structural class of antibiotics with explainable deep learning. *Nature*, 626(7997):177–185.

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.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How powerful are graph neural networks? In *International Conference on Learning Representations*.

Seonghyeon Ye, Hyeonbin Hwang, Sohee Yang, Hyeongu Yun, Yireun Kim, and Minjoon Seo. 2024. Investigating the effectiveness of task-agnostic prefix prompt for instruction following. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19386–19394.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do transformers really perform badly for graph representation? *Advances in neural information processing systems*, 34:28877–28888.

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.

Weitong Zhang, Xiaoyun Wang, Weili Nie, Joe Eaton, Brad Rees, and Quanquan Gu. 2023. Moleculegpt: Instruction following large language models for molecular property prediction. In *NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development*.

Haiteng Zhao, Shengchao Liu, Ma Chang, Hannan Xu, Jie Fu, Zhihong Deng, Lingpeng Kong, and Qi Liu. 2024. Gimlet: A unified graph-text model for instruction-based molecule zero-shot learning. *Advances in Neural Information Processing Systems*, 36.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.

A Datasets

A.1 Details of datasets

We adhere to the dataset selections as outlined in GIMLET (Zhao et al., 2024). Moreover, considering the importance and extensive research in the field of quantum mechanical properties, we have included an additional two quantum mechanical properties: Highest Occupied Molecular Orbital(HOMO) and Lowest Unoccupied Molecular Orbital (LUMO), from the QM9 datasets (Ramakrishnan et al., 2014) as our instruction tuning datasets. To construct the instructions for these additional datasets, we employed the method in Mol-Instructions (Fang et al., 2023) and GIMLET (Zhao et al., 2024). Initially, we write a property description for each task according to Wikipedia and chemistry papers. Subsequently, we employ GPT-4.0 (OpenAI, 2023) to generate instructions based on these seed examples, resulting in various human question-framing styles instructions. The comprehensive list of tuning and downstream tasks are summarized in Tab. 3.

A.2 Details of instructions

Our instruction tuning datasets comprise three components: instruction, input, and output. The instruction component includes a description of the property along with some retrieval examples. The input is the SMILES string of the query molecule, while the output is the property label of query molecule. Here are a few examples of few-shot instructions from three tuning datasets: ChEMBL bioassay activity dataset, CHEMBL Property dataset, and QM9 dataset.

A 1-shot instruction tuning sample from CHEMBL Property datasets:
"### Instruction: Aromatic rings (also known as aromatic compounds or arenes) are hydrocarbons which contain benzene, or some other related ring structure. Here are some examples.
SMILES: Cc1ccc2ccccc2n1
label: 2
Please count how many aromatic rings exist in this molecule.
Input: Cc1ccnc2ccccc12
Response: 2"

A 3-shot instruction tuning sample from ChEMBL bioassay activity datasets:
"### Instruction: The assay is PUBCHEM_BIOASSAY: NCI human tumor cell line growth inhibition assay. Data for the DMS 273 Small Cell Lung cell line. (Class of assay: confirmatory), and it is Target assigned is non-molecular. The assay has properties: assay category is confirmatory; assay cell type is DMS-273; assay type description is Functional. Here are some examples.
SMILES: CC(C)C(N)=O
label: No
SMILES: O=CNC=Cc1ccccc1
label: No
SMILES: COC(=O)C#CC(N)=O
label: No
Is the molecule effective to this assay?
Input: CNC=O
Response: No"

A 4-shot instruction tuning sample from QM9 datasets:
"### Instruction: Lumo is the Lowest unoccupied molecular orbital energy. Here are some examples.
SMILES: CC
label: 0.1
SMILES: CC(C#C)C#CC#C
label: -0.02
SMILES: CC#CC#CC#C
label: -0.05
SMILES: CC(C#C)C#C
label: 0.03
What is Lumo value of this molecule?
Input: C1CC1
Response: 0.1"

B Training Setup

To efficiently finetune the LLaMA2-chat-7B, we employed QLoRA (Dettmers et al., 2024) approach. To enhance memory utilization and speed up the training process, we incorporated Deepspeed ZeRO stage 2 (Rasley et al., 2020), FlashAttention-2 (Dao, 2023), and BFloat16 mixed precision techniques. We set the learning rate to 3e-4 and the maximum inputs length to 512 tokens. All models were trained on 4 Tesla A800-80G GPUs and inferenced on 1 RTX 3090 GPU.

C Detailed Experiment Results

C.1 The robustness of MolecularGPT

To evaluate the robustness of MolecularGPT across diverse instructional phrasings, we adopt the instruction datasets constructed in GIMLET (Zhao

Table 3: The overview of datasets

Splitting	Data Class	Dataset	No. of Molecules	No. of Tasks	Task Type
Tuning tasks	Bioactivity assay	ChEMBL bioassay activity dataset	365065	1048	Classification
	Physico-chemical	CHEMBL Property	365065	13	Regression
	Quantum mechanical	QM9	267770	2	Regression
Downstream tasks	Pharmacokinetic	CYP inhibition	16896	5	Classification
		BBBB Blood-brain barrier penetration	2039	1	Classification
	Bio-activity	MUV PubChem bioAssay	93087	17	Classification
		BACE-1 benchmark set	1513	1	Classification
		HIV replication inhibition	41127	1	Classification
	Toxicity	Tox21Toxicology in the 21st century	7831	12	Classification
		Toxcast	8598	617	Classification
	Physico-chemical	ESOL Water solubility	1128	1	Regression
		FreeSolv Solvation free energy	642	1	Regression
		Lipo Lipophilicity	4200	1	Regression

Table 4: The zero-shot inference results under different types of instructions: the original, detailed, expanded, rewritten, and shortened instructions.

	Classification (AUC-ROC)							Regression (RMSE)		
Instruction type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
Original	0.6212	0.7128	0.6253	0.5893	0.5669	0.6373	0.8031	1.471	4.975	1.157
Detailed	0.6222	0.6754	0.6090	0.6047	0.5710	0.6600	0.8076	1.457	5.036	1.158
Expanded	0.6175	0.7134	0.6017	0.6110	0.5688	0.6511	0.8053	1.474	5.023	1.154
Rewritten	0.6351	0.6893	0.6172	0.5955	0.5666	0.6427	0.8050	1.457	5.018	1.157
Shortened	0.6409	0.6697	0.6348	0.5924	0.5692	0.5374	0.8032	1.462	6.258	1.158
Standard deviation	0.0090	0.0183	0.0117	0.0081	0.0016	0.0448	0.0016	0.0071	0.4984	0.0015

Table 5: The zero- and few-shot performances of model which was fine-tuned on 0-shot instruction datasets.

Tasks		Classification (AUC-ROC)							Regression (RMSE)		
Method	Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
0_examples	0-Shot	0.6033	0.6028	0.6010	0.5824	0.5839	0.6521	0.7684	1.767	5.185	1.163
	1-Shot	0.6297	0.4671	0.5740	0.6016	0.5886	0.6436	0.7667	1.442	5.324	1.032
	2-Shot	0.5903	0.4006	0.5665	0.5956	0.5867	0.6166	0.7556	1.438	5.482	1.053
	3-Shot	0.5344	0.4151	0.5705	0.5974	0.5757	0.6032	0.7457	1.379	5.617	1.016
	4-Shot	0.5334	0.4393	0.5675	0.5942	0.5828	0.6197	0.7367	1.249	5.555	1.010
	6-Shot	0.5314	0.3784	0.5312	0.5843	0.5723	0.5767	0.7374	1.241	5.961	0.979
	8-Shot	0.4388	0.3768	0.5637	0.5724	0.5672	0.5187	0.7050	1.131	5.852	0.984

Table 6: The zero- and few-shot performances of model which was fine-tuned on 4-shot instruction datasets.

Tasks		Classification (AUC-ROC)							Regression (RMSE)		
Method	Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
4_examples	0-Shot	0.5446	0.5514	0.6406	0.5425	0.5588	0.4709	0.6282	2.703	4.620	1.144
	1-Shot	0.6773	0.5135	0.6240	0.6911	0.6140	0.6342	0.8239	1.644	5.062	1.019
	2-Shot	0.6860	0.5626	0.6203	0.7053	0.6163	0.6563	0.8420	1.278	4.942	0.949
	3-Shot	0.7315	0.5577	0.6269	0.7096	0.6220	0.6533	0.8479	1.277	4.734	0.949
	4-Shot	0.7264	0.5624	0.6238	0.7233	0.6243	0.6644	0.8525	1.311	4.978	0.956
	6-Shot	0.7294	0.5768	0.6115	0.7339	0.6268	0.6553	0.8523	1.284	4.941	0.974
	8-Shot	0.7327	0.6234	0.6079	0.7396	0.6271	0.6430	0.8554	1.254	4.889	0.967

et al., 2024), which utilizes GPT-3.5-turbo to generate four distinct types of instructions based on the original instruction: detailed, expanded, rewritten, and shortened instructions. We present the zero-shot inference results derived from these diverse instructions and compute their ROC-AUC or RMSE standard deviation, as outlined in Tab. 4. Our findings suggest that MolecularGPT exhibits robust performance across different instructional variations.

C.2 The effect of instruction datasets

To find a model with superior zero-shot generalization and ICL capabilities, we assess the performance of models that have been fine-tuned by datasets that employ diverse mixture strategies. These strategies include single 0-shot instruction, single 4-shot instruction, combined 0&4-shot instruction, combined 0,1,2,3,4-shot (0-4 shot) instruction, and doubled scale of combined 0&4-shot instruction datasets.

In the combined 0&4-shot methodology, we merge the 0-shot and 4-shot instruction datasets in an equal ratio of 0.5: 0.5. For the comprehensive 0-4 shot mix, we integrate the 0,1,2,3, and 4-shot instruction datasets in a ratio of 0.6: 0.1: 0.1: 0.1: 0.1. During these procedures, we ensure the absence of duplicate query molecules and maintain the scale of the datasets. For the doubled scale of 0&4-shot, we amalgamate the 0-shot and 4-shot instruction datasets in an equal proportion of 1: 1. The results of the zero- and few-shot inferences are presented in the following Tab. 5, 6, 7, 8 and 9.

C.3 The effect of inference strategies

We examine the efficacy of the order of the demonstrations within instructions. Tab. 10 illustrates the performance of arranging retrieval demonstrations in ascending order. Notably, the phrasing in zero-shot or one-shot instruction is consistent in both ascending and descending order. Consequently, we present the results of 2-shot and above. Additionally, we examine the efficacy of retrieval based on diversity, comparing it with a strategy that prioritizes similarity, as illustrated in Tab. 11. It's important to note that to ensure an equal distribution of different class samples, evaluating even-numbered shot is essential. Moreover, this strategy is specifically designed for classification tasks, as regression tasks lack distinct classes.

Table 7: The zero- and few-shot performances of model which was fine-tuned on 0&4-shot instruction datasets.

Tasks		Classification (AUC-ROC)							Regression (RMSE)		
Method	Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
	0-Shot	0.6568	0.6728	0.5533	0.6067	0.5352	0.6086	0.7931	1.377	5.376	1.208
	1-Shot	0.7393	0.6620	0.5954	0.6817	0.5809	0.7087	0.8231	1.468	5.034	1.042
	2-Shot	0.7204	0.6485	0.5969	0.7004	0.5863	0.7135	0.8357	1.481	4.981	1.038
0,4_examples	3-Shot	0.7543	0.6459	0.6139	0.6964	0.5877	0.6997	0.8368	1.481	4.984	1.030
	4-Shot	0.7593	0.6363	0.6026	0.7074	0.5938	0.7130	0.8390	1.413	5.149	1.028
	6-Shot	0.7574	0.6150	0.5926	0.7156	0.5954	0.7145	0.8438	1.427	4.928	1.047
	8-Shot	0.7474	0.6197	0.5942	0.7182	0.5962	0.7029	0.8459	1.479	4.846	1.031

Table 8: The zero- and few-shot performances of model which was fine-tuned on 0,1,2,3,4-shot instruction datasets.

Tasks		Classification (AUC-ROC)							Regression (RMSE)		
Method	Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
	0-Shot	0.6521	0.7046	0.5788	0.5673	0.5612	0.6807	0.7539	1.228	5.835	1.176
	1-Shot	0.7728	0.7049	0.5859	0.6639	0.6026	0.7220	0.8115	1.192	4.979	0.996
	2-Shot	0.7393	0.6816	0.5866	0.6780	0.6085	0.7360	0.8232	1.218	4.985	0.983
0-4_examples	3-Shot	0.7793	0.6806	0.5993	0.6719	0.6066	0.7187	0.8323	1.223	4.979	0.960
	4-Shot	0.7743	0.6807	0.5849	0.6817	0.6148	0.7272	0.8394	1.167	5.247	0.983
	6-Shot	0.7724	0.6673	0.6044	0.6956	0.6179	0.7223	0.8452	1.165	5.219	0.976
	8-Shot	0.8102	0.6724	0.6170	0.7043	0.6190	0.7125	0.8418	1.163	5.033	0.992

Table 9: The zero- and few-shot performances of model which was fine-tuned on double scale 0&4-shot instruction datasets.

Tasks		Classification (AUC-ROC)							Regression (RMSE)		
Method	Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
	0-Shot	0.6212	0.7128	0.6253	0.5893	0.5669	0.6373	0.8031	1.471	4.975	1.157
	1-Shot	0.7520	0.7172	0.6327	0.6529	0.5968	0.6999	0.8229	1.496	5.248	1.058
	2-Shot	0.7218	0.7204	0.6338	0.6573	0.5945	0.7260	0.8275	1.489	5.226	1.015
0,4_examples_double	3-Shot	0.7350	0.7038	0.6408	0.6542	0.5951	0.7191	0.8293	1.494	5.082	1.032
	4-Shot	0.7228	0.6893	0.6419	0.6577	0.5978	0.7168	0.8252	1.535	5.375	1.045
	6-Shot	0.7181	0.6554	0.6561	0.6629	0.5965	0.7139	0.8289	1.465	5.046	1.023
	8-Shot	0.7331	0.6382	0.6469	0.6565	0.5985	0.6822	0.8228	1.433	5.033	1.028

Table 10: The few-shot inference results of MolecularGPT using a ICL template that organizes the retrieval demonstrations in a ascending order.

	Classification (AUC-ROC)							Regression (RMSE)		
Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450	ESOL	FreeSolv	Lipo
2-shot	0.7105	0.7126	0.6269	0.6553	0.5941	0.7245	0.8287	1.514	4.934	1.053
3-shot	0.7172	0.6884	0.6166	0.6489	0.5938	0.7090	0.8302	1.527	4.898	1.078
4-shot	0.7333	0.6732	0.6299	0.6474	0.5888	0.7130	0.8281	1.500	5.031	1.050
6-shot	0.7067	0.6423	0.6237	0.6447	0.5864	0.7040	0.8297	1.446	5.097	1.049
8-shot	0.7407	0.6311	0.6352	0.6452	0.5861	0.6555	0.8237	1.462	5.041	1.034

Table 11: The few-shot inference results of MolecularGPT, which retrieves demonstrations based on their diversity.

	Classification (AUC-ROC)						
Type	BACE	HIV	MUV	Tox21	ToxCast	BBBP	CYP450
2-shot	0.7039	0.6854	0.6135	0.6297	0.5819	0.7037	0.8081
4-shot	0.6688	0.6584	0.6255	0.6321	0.5826	0.6962	0.8100
6-shot	0.6782	0.6425	0.6213	0.6184	0.5797	0.7079	0.8133
8-shot	0.6832	0.6127	0.6118	0.6140	0.5848	0.6740	0.8070