MolVision: Molecular Property Prediction with Vision Language Models

Deepan Adak[1], Yogesh Singh Rawat[2], Shruti Vyas[2]
[1]NIT Kurukshetra, [2]Institute of AI, University of Central Florida

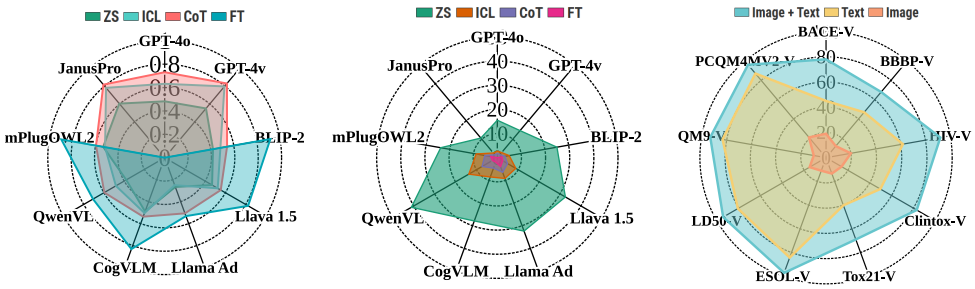

Figure 1: ***MolVision overview:*** Average performance comparison of models in zero-shot (ZS), in-context (ICL), chain-of-thoughts (CoT), and finetuning (FT) for classification (***Left*** ↑) and regression tasks (***Center*** ↓). (***Right:)*** Impact of using visual information on model performance (↑) (JanusPro).

Abstract

Molecular property prediction is a fundamental task in computational chemistry with critical applications in drug discovery and materials science. While recent works have explored Large Language Models (LLMs) for this task, they primarily rely on textual molecular representations such as SMILES/SELFIES, which can be ambiguous and structurally less informative. In this work, we introduce MolVision, a novel approach that leverages Vision-Language Models (VLMs) by integrating both molecular structure as images and textual descriptions to enhance property prediction. We construct a benchmark spanning ten diverse datasets, covering classification, regression and description tasks. Evaluating nine different VLMs in zero-shot, few-shot, and fine-tuned settings, we find that visual information improves prediction performance, particularly when combined with efficient fine-tuning strategies such as LoRA. Our results reveal that while visual information alone is insufficient, multimodal fusion significantly enhances generalization across molecular properties. Adaptation of vision encoder for molecular images in conjunction with LoRA further improves the performance. The code and data is available at : https://molvision.github.io/MolVision/.

1 Introduction

Recent advancements in Large Language Models (LLMs) have revolutionized natural language understanding and generation across multiple domains (1). Models such as GPT (2; 3), LLaMA (4), and Mistral (5) have demonstrated exceptional capabilities in reasoning, knowledge retrieval, and complex problem-solving. Extending beyond pure text-based reasoning, Vision-Language Models (VLMs) integrate visual and textual modalities (1; 6; 7; 8), enabling them to perform tasks such as image captioning, visual question answering, and multimodal retrieval with remarkable success. While VLMs have been extensively explored in computer vision and NLP applications, their potential in scientific domains—particularly in chemistry—remains largely unexplored. Given that molecular

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

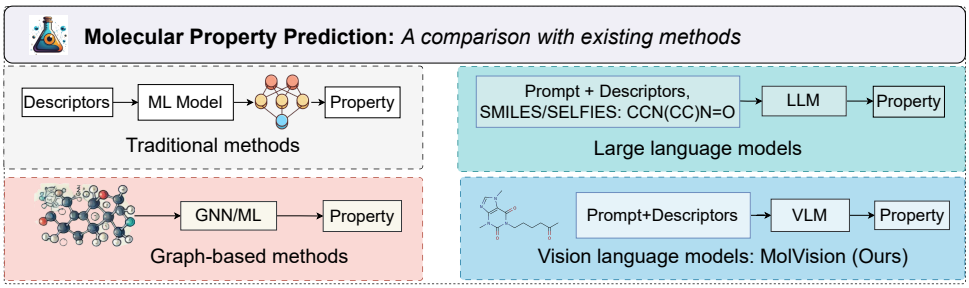

Figure 2: ***MolVision comparison:*** Comparison of relevant molecular property prediction approaches.

structures are inherently visual, leveraging vision in molecular analysis presents an exciting, yet underexplored, research direction.

Recent works such as ChemLLM (9) and ChemLLM-Bench (10) have begun to explore LLMs for molecular property prediction. These methods primarily rely on textual molecular representations, such as SMILES and SELFIES, which have been widely used in cheminformatics for decades. However, these representations have notable limitations, including their non-uniqueness and syntactic instability, where structurally identical molecules may have vastly different textual encodings. This ambiguity introduces challenges for LLMs, which process molecular structures as linear strings, potentially overlooking key structural relationships. While some approaches attempt to improve these representations through graph-based models (11), the integration of easily available visual molecular data remains largely unexplored in this domain.

Incorporating visual information has the potential to significantly enhance molecular property prediction. Chemists usually analyze molecular structures using bond-line or skeletal diagrams to infer properties such as reactivity, toxicity, and solubility. These visual representations inherently encode structural and spatial information that textual descriptors may fail to capture. For example, subtle differences in geometry, stereochemistry, or electron delocalization can have profound effects on molecular properties, yet are difficult to represent accurately in SMILES format alone. By leveraging VLMs, which are designed to process both visual and textual inputs, we aim to bridge this gap and improve predictive modeling in cheminformatics.

To this end, we introduce MolVision, a multimodal benchmark for molecular property prediction. In contrast to prior works MolVision integrates both textual and visual representations (Figure 2). Our benchmark spans ten diverse datasets, covering classification, regression and description tasks across a wide range of molecular properties, including toxicity, solubility, and bioactivity. We evaluate nine different VLMs in zero-shot, few-shot, and fine-tuned settings, providing a comprehensive analysis of their performance in this domain (Figure 1). We also propose a simple contrastive strategy to adapt visual component of VLMs for this domain and demonstrate its effectiveness for property prediction.

Through extensive experimentation, we uncover several key insights. First, while VLMs struggle in zero-shot settings, their performance improves significantly with in-context learning and fine-tuning. Second, efficient adaptation techniques such as LoRA enhance the predictive accuracy and generalization to unseen molecular properties. Third, while visual information alone is insufficient for accurate property prediction, combining molecular images with textual representations yields notable performance gains specifically for larger molecules. These findings suggest that vision-augmented molecular modeling presents a promising avenue for future research in AI-driven chemistry, with numerous potential applications. We make the following contributions:

- We introduce MolVision, a novel approach for molecular property prediction, integrating molecular structure images with textual representations.
- We present a multimodal benchmark and systematically assess nine state-of-the-art VLMs in zero-shot, few-shot, and fine-tuned settings across ten datasets highlighting their strengths and limitations for property prediction.
- We show that efficient adaptation of VLMs for property prediction enhances both performance and generalization, and that combining visual and textual data significantly improves molecular property prediction.
- We propose a simple contrastive strategy, implemented with LoRA, to efficiently adapt vision aspect of VLMs for this domain and demonstrate it effectiveness for property prediction.

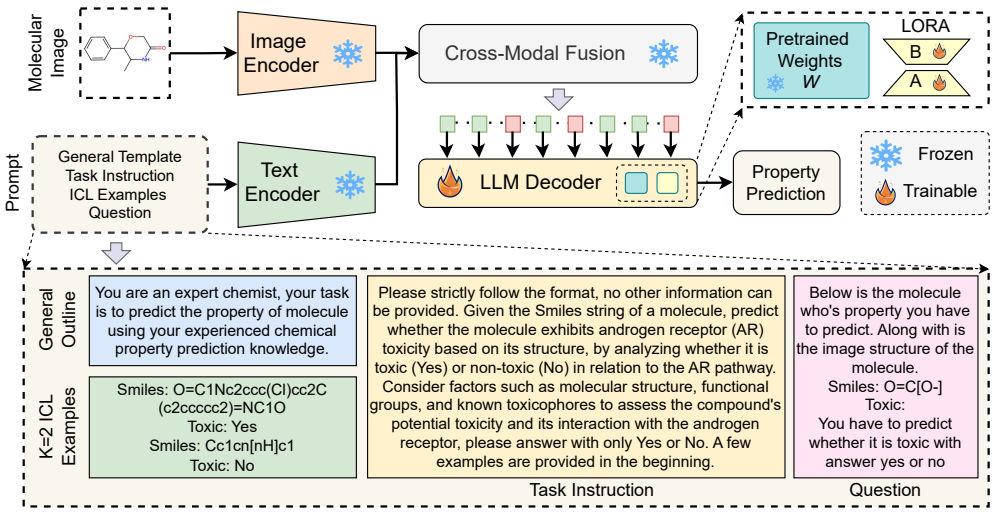

Figure 3: ***Overview of visual-textual approach for property prediction:*** The image representation along with textual description are used as input by the VLM where the image is encoded by a vision encoder and textual description is encoded by a text encoder. These multimodal features are used to generate the output with the help of a decoder. We show template prompt used for property prediction, including general outline, task instruction, in-context learning (ICL with k=2), and an image prompt.

2 Related works

Property prediction: Prior research in molecular property prediction has explored various methods and representations (12). Traditional approaches, like molecular fingerprints (13; 14) and descriptors (15), rely on expert knowledge but are limited in capturing complex data relationships. Recently, machine learning techniques (16; 17; 18), particularly graph-based methods like graph convolutional networks (GCNs) (19), have gained prominence for capturing molecular interactions. Additionally, deep learning models, including RNNs, CNNs, and transformers (20; 21; 22; 23; 24), have shown strong performance in modeling structural and sequential information from molecular data.

Multimodal foundational models: Recent advances in Foundational Models (5; 4; 25) have shown the ability of multimodal LLMs to process both vision and language. These models integrate vision encoders (26; 27) with LLMs (28; 4) for generating text responses. Models like Llama Adapter V2 (7) and Flamingo (29) explored multimodal structures. Typically, these models pre-train on image caption datasets (30; 31) and fine-tune on task-specific datasets (32). Models such as Llava 1.5 (33) and QwenVL (34) are designed for instruction-following tasks but may struggle with science-specific challenges like computational chemistry.

Foundation models for property prediction: Recent efforts have explored LLMs for property prediction, such as ChemLLM (9), ChemLLMBench (10), FS-Mol (35) and Nach0 (36). ChemLLM (9) utilizes ChemData, an instruction-tuning dataset, to address the need for specialized models in chemistry. Guo et al. (10) assess LLMs in chemistry, focusing on understanding and reasoning tasks with zero-shot and few-shot learning. In (37), the authors propose to utilize graphical structure with LLMs for molecule captioning, IUPAC name prediction, and molecule-text retrieval. In contrast, our work examines the role of multimodal vision-language models, incorporating visual and textual data for molecular property prediction, a first-of-its-kind exploration.

3 Visual language models for property prediction

We propose use of visual information, in the form of molecular images, alongside textual descriptions to improve property prediction. Images provide structural insights that are challenging to interpret from text alone. A vision-language model processes both the image and text prompt to generate a textual output, as shown in Figure 3. The input image is divided into patches, which are converted into tokens for the vision encoder (e.g., ViT (38)). The textual prompt is passed through a text encoder (e.g., BERT (38)), and the visual and textual features are fused via multi-modal learning (e.g., using Q-Former). LLM decoder (e.g., a transformer model) uses these fused features for text generation.

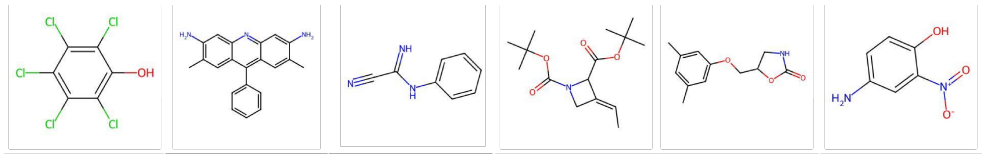

Figure 4: ***Sample visual and textual representation pairs:*** The images in top row shows skeletal structure of molecules and bottom row shows their corresponding SMILES representations.

Text prompt: The prompt consists of three components passed to the text encoder: 1) *General outline* provides an overview of the task, 2) *Task instruction* includes detailed task-specific guidance, and 3) *Question* requests the model's answer in a specific format. For in-context learning and chain-of-thoughts, additional information like examples or reasoning steps is included (Figure 3).

3.1 Model variants

We study three different setups: 1) zero-shot, 2) few-shot, and 3) fine-tuning on training data.

Zero-shot: The model is evaluated without fine-tuning or in-context examples.

Few-shot: We use two approaches: 1) *In-context Learning (ICL)*: traditional prompting (labeled ICL) and *Chain of Thought (CoT) Prompting*. Traditional ICL constructs prompts with examples similar to the input, enabling the model to learn relationships in the new domain. CoT prompting enhances reasoning by guiding the model through intermediate steps. In cheminformatics, molecular similarity is often quantified using methods like similarity and distance metrics. For selecting few-shot samples, we use the Tanimoto index, an effective parameter for similarity prediction (39).

Finetuning: Vision-language models (VLMs) possess a substantial number of trainable parameters, rendering traditional fine-tuning impractical as all model parameters undergo gradient updates simultaneously. In our study, we adopt LoRA (Low-Rank Adaptation) (40) for efficient fine-tuning, a technique that significantly reduce the number of trainable parameters. LoRA achieves this by updating weights through a pair of trainable rank decomposition matrices, which operate in parallel with existing weight matrices, while keeping the original pre-trained weights frozen during fine-tuning. We only adapt the LLM decoder keeping other components frozen during this finetuning to preserve the generalization capabilities of vision and text encoders (Figure 3).

3.2 Model architectures

We study nine different state-of-the-art visual language models in this study. This includes both closed-source and open-source models. In open-source, we experimented with Janus-Pro 7B (41), BLIP-2 (38), Llava 1.5 (33), Llama Adapter V2 (7), CogVLM (42), Qwen-VL (34), and mPLUGOWL2 (43). For closed-source, we experimented with GPT-4V and GPT-4o (1).

4 MolVision benchmark

In this section, we introduce the MolVision benchmark, which includes ten diverse datasets. These datasets cover a wide range of properties, such as molecular weight, topological polar surface area, and toxicity, and encompass classification, regression and description tasks. A summary in Table 1.

Tasks: VLMs generate textual outputs based on image and text prompts. For classification, we frame the task as a True/False question, where the model predicts whether a molecule inhibits a target property. For regression, the model generates a numerical value representing the target property and for description task, the model generates textual output.

Dataset curation: We incorporate both molecular skeletal structures as images and SMILES/SELFIES representations. Existing property prediction datasets primarily focus on textual representations like SMILES, lacking structural images. To address this, we augment these datasets with skeletal images generated using RDKit (44). Figure 4 illustrates examples of skeletal (bond-line) structures alongside their SMILES representations. RDKit also enables conversion between SMILES and SELFIES, allowing us to explore diverse molecular encodings and enhance model robustness.

4.1 Benchmark datasets

The curation process provides both image representations corresponding to each molecule along with a formatted prompt which is derived through manual engineering. The benchmark consists of the following datasets (more details in Appendix): **BACE-V** is derived from BACE (Binary Activity of Chemical Entities) dataset (45) which is widely used for binary classification in bioactivity prediction, particularly for BACE-1 inhibitors linked to Alzheimer's. **BBBP-V** is based on the Blood-Brain Barrier Penetration (BBBP) dataset (46), which provides binary labels for BBB penetration. **HIV-V** is based on the HIV (47) where we focus on predicting HIV replication inhibition. **Clintox-V** is derived from ClinTox dataset (45) and our focus is on predictions of clinical toxicity and FDA approval status. **Tox21-V** is based on the Tox21 dataset (48) and focuses on predicting chemical toxicity, critical for environmental safety. **ESOL-V** is based on the ESOL dataset (49), and focus on predicting aqueous solubility of organic compounds. **LD50-V** is based on the LD50 (50) and focuses on acute toxicity. **QM9-V** is derived from the QM9 (51) and focuses on quantum chemical properties. **PCQM4Mv2-V** is derived from the PCQM4Mv2 dataset (52) and focuses on predicting the HOMO-LUMO gap. **ChEBI20-V** is derived from the Chemical Entities of Biological Interest (ChEBI) (53)database and focuses on generating accurate textual descriptions of molecular structures.

Table 1: ***MolVision benchmark details*:** Statistics of datasets used in this study.

Dataset	Train	Test	Property
Classification			
BACE-V	1,210	303	Bioactivity
BBBP-V	1,640	410	BBB Pen.
HIV-V	32,902	8,225	HIV activity
ClinTox-V	1,193	298	Toxicity
Tox21-V	6,265	1,566	Toxicity
Regression			
ESOL-V	902	226	Solubility
LD50-V	5,908	1,477	Toxicity
QM9-V	107K	27K	Quantum
PCQM4Mv2-V	3.0M	0.7M	Quantum
Molecular Description			
ChEBI-V	32,000	8,000	Description

5 Experiments and results

Next, we provide evaluations on MolVision benchmark followed by some discussion and analysis.

Evaluation metrics: We evaluate classification performance using Accuracy and F1 Score. We evaluate classification using Accuracy and F1 Score, where Accuracy measures correct predictions, and F1 Score balances precision and recall. For regression, we use mean absolute error (MAE) and root mean square error (RMSE) to quantify prediction deviations. Molecular description tasks are assessed with BLEU-2, BLEU-4, ROUGE-1, ROUGE-2, ROUGE-L, and METEOR, capturing n-gram precision, sequence overlap, and semantic similarity.

5.1 Benchmarking results

All experiments are conducted with a temperature of 0 (unless stated) to reduce prediction volatility.

Zero-shot: VLMs are trained on large-scale datasets to learn associations between visual and textual features. However, property prediction presents a distinct challenge, differing from their training domain. Figures 1 (left and center) show zero-shot results across all datasets, where performance remains low for most models, except for proprietary models GPT-4o and GPT-4v. (More in Appendix.)

Few-shot: Tables 2 and 3 present few-shot ICL performance for classification and regression tasks, respectively. All models show performance gains over zero-shot, though Llama Adapter v2 7B and Qwen VL consistently underperform with classification accuracy below 48% and high regression errors. BBBP-V and QM9 remain challenging datasets, while Tox21-V and LD50 yield comparatively better results. As expected, closed models such as GPT-4o and GPT-4v achieved the best performance across most datasets however Janus-Pro 7B performed better on certain datasets. The performance was followed by Llava 1.5 13B and BLIP-2 as the second best open-source models for classification and regression, respectively. Table 3 also reports CoT prompting results for regression, showing further improvements, also seen in classification tasks (Figure 1, more info in Appendix).

Finetuning: Table 4 presents classification results after model adaptation, showing significant performance improvements with fine-tuning. A similar trend is observed for regression tasks, where adaptation reduces prediction error (Figure 1). Detailed results are provided in the Appendix. Overall, BLIP-2 achieves the best performance across both classification and regression tasks, while mPlugOWL2 remains competitive in classification but underperforms in regression. Table 5 show

Table 2: ***Few-shot performance for classification tasks:*** A comparison of accuracy (f1-score) on property prediction task using in-context learning (ICL with k=2). († - fully supervised training)

Models	**BACE-V ↑**	**BBBP-V↑**	**HIV-V ↑**	**ClinTox-V ↑**	**Tox21-V ↑**	**Average ↑**
GNN Models						
UniMol (11) †	0.78(0.67)	0.82(0.70)	0.82(0.73)	0.94(0.83)	0.77(0.65)	0.83(0.72)
Molca (37) †	0.79(0.73)	0.74(0.72)	0.89(0.84)	0.93(0.84)	0.80(0.72)	0.83(0.77)
LLM [ICL k=2]						
Guo et. al. (54)	0.49(0.40)	0.46(0.46)	0.86(0.80)	0.57(0.36)	0.57(0.52)	0.59(0.51)
Davinci-003	0.65(0.64)	0.39(0.37)	0.78(0.83)	0.84(0.85)	0.68(0.51)	0.67(0.64)
ChemLLM(9)	0.18(0.12)	0.12(0.08)	0.19(0.09)	0.21(0.13)	0.18(0.09)	0.18(0.10)
Gal-1.3B (55)	0.38(0.29)	0.42(0.26)	0.33(0.23)	0.40(0.34)	0.49(0.32)	0.40(0.29)
Gal-6.7B (55)	0.41(0.30)	0.44(0.28)	0.35(0.25)	0.43(0.36)	0.52(0.34)	0.44(0.31)
VLM [ICL k=2]						
GPT-4o	0.56(0.53)	**0.77(0.81)**	0.82(**0.56**)	0.59(0.44)	0.42(0.58)	0.63(0.58)
GPT-4v	0.72(0.66)	0.63(0.60)	**0.95**(0.44)	**0.96**(**0.94**)	0.72(0.52)	**0.80**(**0.63**)
Janus Pro 7B	**0.78(0.71)**	0.68(0.62)	0.92(0.52)	0.83(0.56)	0.69(0.49)	0.78(0.58)
BLIP-2	0.36(0.52)	0.37(0.29)	0.60(0.30)	0.34(0.36)	0.75(0.42)	0.48(0.38)
Llava 1.5 13B	0.49(0.48)	0.44(0.39)	0.24(0.34)	0.64(0.76)	**0.81**(0.31)	0.52(0.46)
Llama Ad v2 7B	0.28(0.29)	0.18(0.11)	0.19(0.17)	0.29(0.12)	0.31(0.21)	0.25(0.18)
CogVLM	0.48(0.51)	0.40(0.37)	0.31(0.21)	0.64(0.62)	0.69(**0.65**)	0.50(0.47)
QwenVL	0.69(0.46)	0.30(0.12)	0.28(0.36)	0.52(0.48)	0.62(0.63)	0.48(0.41)
mPlugowl2	0.59(0.32)	0.35(0.38)	0.62(0.29)	0.34(0.42)	0.69(0.56)	0.52(0.39)

Table 3: ***Few-shot performance for regression:*** A comparison of error in prediction using in-context learning (ICL k=2) and chain-of-thoughts (CoT) with traditional and LLM based approaches.

Model	**ESOL-V ↓**		**LD50-V ↓**		**QM9-V ↓**		**PCQM4M-V ↓**		**Average ↓**	
Traditional approaches										
GenRA(56)	-		0.58		-		-		-	
Unimol (11)	0.788		-		0.00467		0.070		-	
Large Language Models										
GPT-3.5	4.24		11.67		13.52		1.81		5.81	
Llama2 13B	27.71		49.22		78.92		102.92		64.69	
Mistral 13B	33.21		27.46		66.80		88.90		54.09	
ChemLLM (9)	23.42		33.91		147.10		29.01		58.36	
GAL-1.3B (55)	18.92		40.49		140.92		29.92		-	
Gal-6.7B (55)	13.47		38.02		128.90		28.48		-	
Molca (37)	1.849		0.982		4.889		0.802		-	
Vision-Language Models	ICL	CoT	ICL	CoT	ICL	CoT	ICL	CoT	ICL	CoT
GPT-4o	0.98	0.77	0.87	0.60	**8.38**	5.24	0.68	0.53	2.73	1.78
GPT-4v	0.99	**0.71**	**0.71**	0.59	8.62	4.66	0.77	0.66	2.78	1.66
Janus-Pro 7B	**0.61**	0.89	0.72	0.60	8.53	**4.42**	**0.62**	**0.38**	**2.52**	**1.57**
BLIP-2	1.99	1.07	0.73	**0.49**	16.01	10.09	1.30	1.25	5.01	3.23
Llava 1.5 13B	6.01	2.18	0.94	0.69	27.00	15.21	1.42	1.49	8.84	4.89
Llama Ad v2 7B	3.08	2.17	3.36	2.12	28.09	19.24	4.06	2.36	9.15	6.47
CogVLM	1.26	1.21	3.47	0.78	25.85	15.15	1.44	1.24	8.50	4.59
Qwen VL	3.96	2.89	1.06	0.63	38.92	18.08	10.61	9.56	13.64	7.29
mPlugOWL2	1.46	1.50	0.94	0.71	29.33	19.17	1.84	1.62	8.89	5.25

performance on molecular description task after finetuning. We observe that CogVLM outperforms other VLMs across all metrics and performs better than recent graph-based LLM.

Zero-shot generalization: Figure 5 shows results with zero-shot generalization where we adapted the model on one dataset and evaluated on others. As shown, the performance is better than few-shot (Table 2), however the performance is not as good as LoRA where the model was finetuned on the target dataset (Table 4). From Figure 5 we observe that after training on HIV-V dataset we get the best zero-shot average accuracy (61%) and best zero-shot avg F1-score is observed after training on BBBP-V. ClinTox-V and BBBP-V are the most difficult datasets for zero shot generalization with an average accuracy of 39% and 40%, respectively.

Table 4: ***Classification performance after finetuning:*** Accuracy (F1 score) comparison of models finetuned using LoRA. The best performing models are highlighted with bold text.

Models	BACE-V ↑	BBBP-V↑	HIV-V ↑	ClinTox-V ↑	Tox21-V↑	Average ↑
RF	0.79(0.76)	0.82(0.88)	0.87(0.52)	0.85(0.46)	0.83(0.26)	0.83(0.57)
XGBoost	0.81(0.77)	0.85(0.90)	0.87(0.55)	0.88(0.62)	0.84(0.33)	0.85(0.63)
ChemLLM (9)	0.18(0.12)	0.12(0.08)	0.19(0.09)	0.21(0.13)	0.18(0.09)	0.17(0.10)
Molca (37)	0.79(0.73)	0.74(0.72)	0.89(0.84)	0.93(0.84)	0.80(0.72)	0.83(0.77)
BLIP-2	**0.86(0.83)**	**0.93(0.96)**	**0.92**(0.76)	**0.89(0.93)**	**0.99**(0.80)	**0.92**(0.86)
Llava 1.5 13B	0.84(**0.83**)	0.86(0.88)	0.80(0.81)	0.70(0.72)	0.92(0.93)	0.82(0.83)
Llama Adapter v2 7B	0.52(0.48)	0.45(0.46)	0.43(0.42)	0.58(0.62)	0.68(0.69)	0.53(0.53)
CogVLM	0.72(0.71)	0.78(0.82)	0.85(0.83)	0.88(0.90)	0.93(0.93)	0.83(0.84)
Qwen VL	0.78(0.78)	0.70(0.72)	0.60(0.61)	0.71(0.64)	0.75(0.64)	0.71(0.68)
mPlugOWL2	**0.86**(0.82)	0.90(0.88)	0.90(**0.91**)	**0.89**(0.92)	0.94(**0.96**)	0.89(**0.89**)

Table 5: ***Molecular description performance after finetuning:*** Comparison of models finetuned using LoRA on the ChEBI dataset. The best performing models are highlighted with bold text.

Models	BLEU-2 ↑	BLEU-4 ↑	ROUGE-1 ↑	ROUGE-2 ↑	ROUGE-L ↑	METEOR ↑	Average ↑
MolT5 (57)	59.40	50.80	65.40	51.00	59.40	61.40	57.90
Molca (37)	62.00	53.10	68.10	53.70	61.80	65.10	60.60
BLIP-2	59.06	58.03	58.93	58.47	58.89	58.19	58.60
CogVLM	**63.00**	**60.01**	**62.39**	**61.16**	**62.00**	**60.60**	**61.52**
mPlugOWL2	51.93	49.64	51.56	50.67	51.33	50.06	50.87
Llava 1.5 13B	60.88	58.99	60.62	59.80	60.42	59.40	60.02
Llama Adapter v2 7B	46.60	44.65	46.27	45.50	46.07	45.00	45.68
Qwen VL	52.00	50.04	51.63	50.78	51.40	50.44	51.05

Table 6: ***Impact of visual information:*** A performance comparison showing the impact of visual information (molecular image) when used with textual description (SMILES). Accuracy is shown for classification (BACE-V, BBBP-V, HIV-V, Clintox-V (CV), Tox21-V (TV)), and MAE (LD50-V, QM9-V and PCQM4Mv2-V (PV)) and RMSE (ESOL-V) is shown for regression tasks.

Model	Input	BACE-V ↑	BBBP-V ↑	HIV-V ↑	CV ↑	TV ↑	ESOL-V ↓	LD50-V ↓	QM9-V ↓	PV ↓
BLIP-2	Text Only	0.71	0.76	0.69	0.64	0.78	9.89	7.80	31.76	11.31
	Image Only	0.15	0.09	0.10	0.13	0.18	32.16	31.23	149.12	36.96
	Image+Text	0.86	0.93	0.92	0.89	0.99	1.07	0.49	4.92	1.99
JanusPro	Text Only	0.45	0.47	0.62	0.50	0.40	1.23	1.16	20.94	2.09
	Image Only	0.19	0.12	0.20	0.14	0.13	21.57	12.49	125.03	16.20
	Image+Text	0.78	0.68	0.92	0.83	0.69	0.61	0.72	8.53	0.62

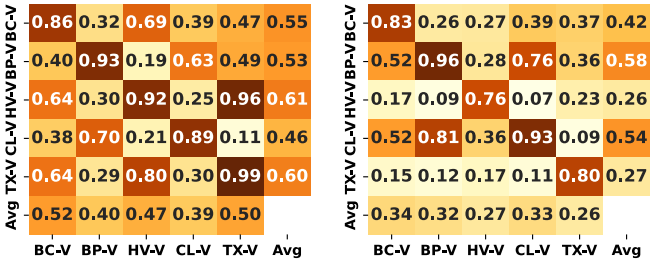

Figure 5: ***Zero-shot generalization:*** Visualization of accuracy (left) and F1-score (right) for zero-shot cross-dataset performance using BLIP-2. Each heatmap illustrates results from fine-tuning on one dataset (y-axis) and evaluating on others.

Table 7: ***Overall ranking:*** Performance in terms of average ranking across datasets (classification/regression/description).

Models	Zero	ICL	CoT	LoRA
GPT-4v	2/1/2	1/3/1	1/1/2	-
GPT-4o	5/3/1	3/2/2	3/2/1	-
JanusPro 7B	1/2/3	2/1/3	2/3/3	-
BLIP-2	7/5/8	7/4/6	7/3/7	1/1/3
Llava1.5 13B	4/8/4	4/6/7	6/6/4	4/4/2
Llama v2 7B	9/7/9	9/8/9	9/8/9	6/6/6
CogVLM	4/6/5	6/5/4	8/5/5	3/2/1
Qwen VL	8/9/7	8/9/5	5/9/6	5/3/4
mPlugOWL2	3/4/6	5/7/8	4/7/8	2/5/5

5.2 Comparison with existing methods

We compare our approach with traditional methods such as XGBoost, RF, GenRA (56), and Unimol (11), as well as recent LLM-based property prediction models that rely solely on text (ChemLLM-Bench (54) and ChemLLM (9)). Table 2 shows that incorporating visual information leads to better performance than LLMs using text alone. Table 3 compares VLMs with traditional models, where Janus-Pro, GPT-4o, and GPT-4v achieve competitive results on ESOL and LD50, though performance on QM9 and PCQM4M remains lower due to extensive training of traditional models on these

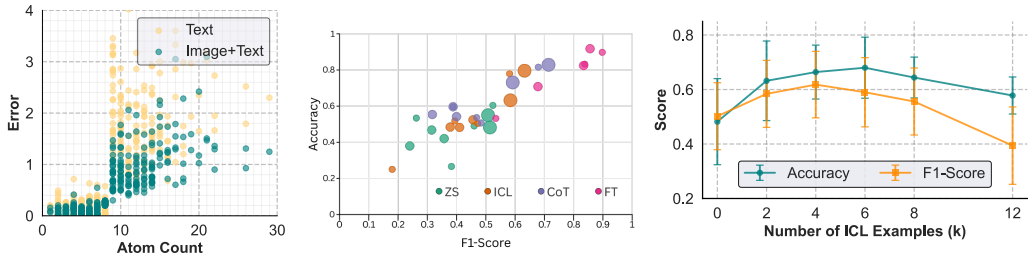

Figure 6: ***Analysis on molecular-size, model-size and effect of in-context examples:*** The first plot shows the impact of molecular size on regression error in LD50 with JanusPro, highlighting how visual data improves performance. The middle figure shows comparison of VLMs across datasets Accuracy vs F1 Score for zero shot (ZS), in-context (ICL), CoT and finetuning (FT). The bubble size represents the model's parameter scale. The right figure shows variation in accuracy and f1-scores in case of different ICL examples for GPT-4o model.

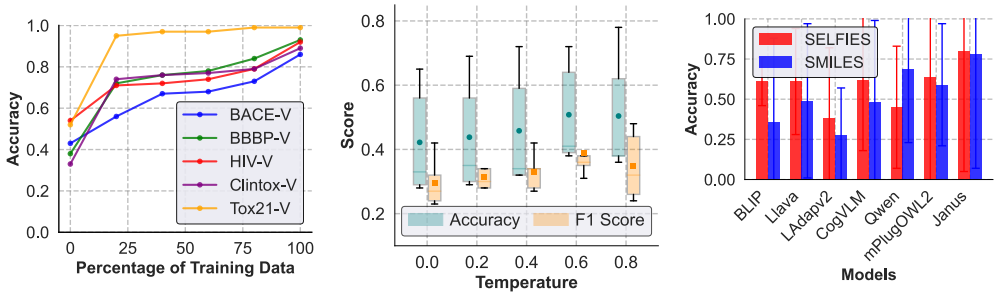

Figure 7: ***Analysis on finetuning, temperature and SELFIES vs. SMILES:*** The first plot shows the impact of percentage of finetuning data. The middle figure shows performance variation with temperature across datasets for BLIP2. The last figure shows analysis of SMILE vs SELFIES string for ICL k=2 across various models.

datasets. In classification tasks (Table 4), VLMs consistently outperform traditional approaches, with the largest gains on challenging datasets like BBBP-V. We also compare with a recent graph-based domain specific LLM approach Molca (37), and observe that our vision based approach provides better or comparable performance across all tasks and datasets (Table 3 and 5).

5.3 Discussion and analysis

This section provides further discussion and analysis for more insights into the benchmark.
Impact of visual data: Previously, we showed that VLMs outperform LLMs (Tables 2 and 3). Here, in Table 6, we analyze the impact of visual information within the same model by evaluating BLIP-2 and JanusPro in three configurations: (1) Image+Text (molecular structure images + SMILES), (2) Text Only (SMILES), and (3) Image Only (molecular images). We see that image-only inputs are insufficient, but augmenting text with visual data improves performance, showing the benefits of multimodal learning. Limitations and ethical considerations are discussed in Appendix.

Molecular size: We analyze the impact of molecular size on performance and find that larger molecules are more challenging, suggesting that longer representations are harder for models to interpret (Figure 6 (a)). We also observe that incorporating visual representations improves performance on larger molecules, further reinforcing the value of structural images in property prediction.

Capability of different models: After fine-tuning (Table 4), BLIP-2 achieves the highest accuracy across classification tasks, except for ClinTox-V, where mPlugOWL2 performs comparably. mPlugOWL2 ranks second overall, followed by CogVLM and Llava 1.5 13B, as confirmed in Table 7. BLIP-2 also excels in regression tasks.

Impact of model size on performance: Since VLMs are trained on large-scale datasets, their size generally correlates with performance. In our analysis of property prediction, we observe a similar trend, where larger models consistently outperform their smaller counterparts (Figure 6 (c)). Notably, while larger models perform better post-ICL, smaller models surpass them after fine-tuning.

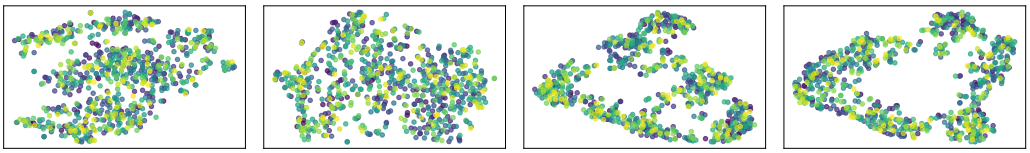

Figure 8: ***Analyzing visual features:*** The left two plots show t-SNE visualizations of visual encodings of BLIP-2 before and after cross-modal fusion respectively. The right two plots show corresponding t-SNE plots with the proposed contrastive loss using Tanimoto augmentation (T-Aug).

Table 8: ***Performance comparison for proposed contrastive learning:*** Evaluation of different contrastive learning approaches across multiple molecular datasets.

Method	BACE-V Acc(F1)	BBBP-V Acc(F1)	HIV-V Acc(F1)	Clintox-V Acc(F1)	Tox21-V Acc(F1)	ESOL-V (RMSE)	LD50-V (MAE)	QM9-V (MAE)	PCQM-V	Chebi-V (Average)
LoRA	0.86(0.83)	0.93(0.96)	0.92(0.76)	0.89(0.93)	0.99(0.80)	1.76	0.78	4.92	0.24	58.59
Aug	0.87(0.85)	0.94(0.95)	0.93(0.78)	0.90(0.94)	0.97(0.83)	0.90	0.20	4.09	0.21	60.98
T-Aug	**0.91(0.88)**	**0.95(0.96)**	**0.95(0.84)**	**0.93(0.93)**	**0.98(0.89)**	**0.58**	**0.10**	**2.95**	**0.12**	**63.73**

Effect of number of ICL examples: Figure 6 (a) shows that performance improves with more in-context examples but degrades beyond a certain point. This can be accredited to VLMs' limitations in processing long prompts with excessive tokens.

Effect of amount of finetuning data: Figure 7 (left) show the impact of increasing finetuning data from 0% to 100% in 20% increments. We observe best performance with 100% data in all datasets, resulting in $\sim$40% increase in performance.

Effect of temperature: Figure 7 shows how accuracy (F1-score) averaged across datasets vary with change in temperature. With most of the models (in supplementary) usually highest accuracy and F1-score is observed at lower temperatures (0.0-0.4) (in appendix), however, BLIP-2 showed better performance at higher temperatures (0.8).

SELFIES vs SMILES. SELFIES are more robust molecular representations, adhering to valence and ring constraints, thus avoiding invalid molecule generation (58). In Figure 7 (right) we observe that with few exceptions, SELFIES provides better scores on most datasets.

5.4 Adaptation of vision encoder to molecular structures

To enhance the visual representation capability of VLMs in the molecular domain, we analyzed the vision embeddings of BLIP-2 using t-SNE and found them to be poorly clustered and non-discriminative—likely due to pretraining on natural images (Fig. 8). To address this, we fine-tuned the vision encoder using a contrastive learning objective (NT-Xent loss (59)) along with LoRA finetuning. We follow two strategies for identifying the positive pairs and use them in separate approaches. First approach uses augmented views of the same molecule (Aug, Table 8), and second approach uses structurally similar molecules identified via Tanimoto similarity ($>$0.85) (T-Aug). As shown in Fig. 8, the similarity-based approach leads to more distinct and meaningful clusters in the embedding space, capturing pharmacophoric patterns more effectively. This method significantly improves performance—reducing ESOL RMSE by 35%, LD50 MAE by 51%, and boosting classification accuracy and F1 scores by 2–4% over the base model. These results underscore the importance of domain-aware vision adaptation, and demonstrate that Tanimoto-guided contrastive learning offers a simple yet powerful enhancement for VLMs in molecular property prediction.

6 Conclusion

We present MolVision, a multimodal approach for molecular property prediction using vision-language models. We analyze zero-shot, few-shot, and fine-tuned models, combining 2D molecular structure images with textual representations. We provide evaluations on a wide variety of datasets covering classification, regression and description tasks, and demonstrate the benefits of visual information for molecular property prediction. Adaptation of vision encoder of VLMs to molecular data makes them more promising. This study will serve as a benchmark for further research exploring the use of easily available 2D visual information in multimodal molecular modeling.

7 Acknowledgment

This research has benefitted from the Microsoft Accelerating Foundation Models Research (AFMR) grant program.

References

[1] OpenAI, J. Achiam, and et. al., "Gpt-4 technical report," 2024.

[2] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[3] L. Floridi and M. Chiriatti, "Gpt-3: Its nature, scope, limits, and consequences," *Minds and Machines*, vol. 30, pp. 681–694, 2020.

[4] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, "Llama 2: Open foundation and fine-tuned chat models," 2023.

[5] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7b," 2023.

[6] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," 2023.

[7] P. Gao, J. Han, R. Zhang, Z. Lin, S. Geng, A. Zhou, W. Zhang, P. Lu, C. He, X. Yue, H. Li, and Y. Qiao, "Llama-adapter v2: Parameter-efficient visual instruction model," 2023.

[8] G. Team and et. al., "Gemini: A family of highly capable multimodal models," 2024.

[9] D. Zhang, W. Liu, Q. Tan, J. Chen, H. Yan, Y. Yan, J. Li, W. Huang, X. Yue, W. Ouyang, D. Zhou, S. Zhang, M. Su, H.-S. Zhong, and Y. Li, "Chemllm: A chemical large language model," 2024.

[10] T. Guo, K. Guo, B. Nan, Z. Liang, Z. Guo, N. V. Chawla, O. Wiest, and X. Zhang, "What can large language models do in chemistry? a comprehensive benchmark on eight tasks," 2023.

[11] S. Lu, Z. Gao, D. He, L. Zhang, and G. Ke, "Highly accurate quantum chemical property prediction with uni-mol+," *arXiv preprint arXiv:2303.16982*, 2023.

[12] C. Hasselgren and T. I. Oprea, "Artificial intelligence for drug discovery: Are we there yet?," *Annual Review of Pharmacology and Toxicology*, vol. 64, p. 527–550, Jan. 2024.

[13] R. N. Tazhigulov, J. Schiller, J. Oppenheim, and M. Winston, "Molecular fingerprints for robust and efficient ml-driven molecular generation," 2022.

[14] I. Kumar and P. K. Jha, "Coarse-grained configurational polymer fingerprints for property prediction using machine learning," 2023.

[15] R. Todeschini and V. Consonni, *Handbook of molecular descriptors*. John Wiley & Sons, 2008.

[16] K. Choudhary and M. L. Kelley, "Chemnlp: A natural language-processing-based library for materials chemistry text data," *The Journal of Physical Chemistry C*, vol. 127, p. 17545–17555, Aug. 2023.

[17] H. Öztürk, A. Özgür, P. Schwaller, T. Laino, and E. Ozkirimli, "Exploring chemical space using natural language processing methodologies for drug discovery," *Drug Discovery Today*, vol. 25, p. 689–705, Apr. 2020.

[18] G. M. Hocky and A. D. White, "Natural language processing models that automate programming will transform chemistry research and teaching," *Digital Discovery*, vol. 1, no. 2, p. 79–83, 2022.

[19] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International conference on machine learning*, pp. 1263–1272, PMLR, 2017.

[20] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chemical science*, vol. 9, no. 2, pp. 513–530, 2018.

[21] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[22] B. Tang, S. T. Kramer, M. Fang, Y. Qiu, Z. Wu, and D. Xu, "A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility," *Journal of cheminformatics*, vol. 12, pp. 1–9, 2020.

[23] G. B. Goh, N. O. Hodas, and A. Vishnu, "Deep learning for computational chemistry," *Journal of computational chemistry*, vol. 38, no. 16, pp. 1291–1307, 2017.

[24] D. C. Elton, Z. Boukouvalas, M. D. Fuge, and P. W. Chung, "Deep learning for molecular design—a review of the state of the art," *Molecular Systems Design & Engineering*, vol. 4, no. 4, pp. 828–849, 2019.

[25] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[26] Y. Fang, W. Wang, B. Xie, Q. Sun, L. Wu, X. Wang, T. Huang, X. Wang, and Y. Cao, "Eva: Exploring the limits of masked visual representation learning at scale," 2022.

[27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021.

[28] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging llm-as-a-judge with mt-bench and chatbot arena," 2023.

[29] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan, "Flamingo: a visual language model for few-shot learning," 2022.

[30] S. Changpinyo, P. Sharma, N. Ding, and R. Soricut, "Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts," 2021.

[31] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015.

[32] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Batra, and D. Parikh, "Vqa: Visual question answering," 2016.

[33] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved baselines with visual instruction tuning," 2024.

[34] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou, "Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond," 2023.

[35] M. Stanley, J. F. Bronskill, K. Maziarz, H. Misztela, J. Lanini, M. Segler, N. Schneider, and M. Brockschmidt, "Fs-mol: A few-shot learning dataset of molecules," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[36] M. Livne, Z. Miftahutdinov, E. Tutubalina, M. Kuznetsov, D. Polykovskiy, A. Brundyn, A. Jhunjhunwala, A. Costa, A. Aliper, A. Aspuru-Guzik, and A. Zhavoronkov, "nach0: Multimodal natural and chemical languages foundation model," 2024.

[37] Z. Liu, S. Li, Y. Luo, H. Fei, Y. Cao, K. Kawaguchi, X. Wang, and T.-S. Chua, "Molca: Molecular graph-language modeling with cross-modal projector and uni-modal adapter," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 15623–15638, 2023.

[38] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.

[39] D. Bajusz, A. Rácz, and K. Héberger, "Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?," *Journal of cheminformatics*, vol. 7, pp. 1–13, 2015.

[40] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[41] X. Chen, Z. Wu, X. Liu, Z. Pan, W. Liu, Z. Xie, X. Yu, and C. Ruan, "Janus-pro: Unified multimodal understanding and generation with data and model scaling," *arXiv preprint arXiv:2501.17811*, 2025.

[42] W. Wang, Q. Lv, W. Yu, W. Hong, J. Qi, Y. Wang, J. Ji, Z. Yang, L. Zhao, X. Song, J. Xu, B. Xu, J. Li, Y. Dong, M. Ding, and J. Tang, "Cogvlm: Visual expert for pretrained language models," 2024.

[43] Q. Ye, H. Xu, G. Xu, J. Ye, M. Yan, Y. Zhou, J. Wang, A. Hu, P. Shi, Y. Shi, C. Li, Y. Xu, H. Chen, J. Tian, Q. Qian, J. Zhang, F. Huang, and J. Zhou, "mplug-owl: Modularization empowers large language models with multimodality," 2024.

[44] G. Landrum, "Rdkit documentation," *Release*, vol. 1, no. 1-79, p. 4, 2013.

[45] G. Subramanian, B. Ramsundar, V. Pande, and R. A. Denny, "Computational modeling of -secretase 1 (bace-1) inhibitors using ligand based approaches," *Journal of Chemical Information and Modeling*, vol. 56, no. 10, pp. 1936–1949, 2016. PMID: 27689393.

[46] O. T. Sakiyama H, Fukuda M, "Prediction of blood-brain barrier penetration (bbbp) based on molecular descriptors of the free-form and in-blood-form datasets. molecules.," 2021.

[47] Y. SS, "The nci's aids antiviral drug screening program," 1995.

[48] R. Huang, M. Xia, D.-T. Nguyen, T. Zhao, S. Sakamuru, J. Zhao, S. A. Shahane, A. Rossoshek, and A. Simeonov, "Tox21challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs," *Frontiers in Environmental Science*, vol. 3, p. 85, 2016.

[49] J. S. Delaney, "Esol: estimating aqueous solubility directly from molecular structure," *Journal of chemical information and computer sciences*, vol. 44, no. 3, pp. 1000–1005, 2004.

[50] A. Karmaus, J. Fitzpatrick, D. Allen, G. Patlewicz, N. Kleinstreuer, and W. Casey, "Variability of ld50 values from rat oral acute toxicity studies: implications for alternative model development," *Society of Toxicology, San Antonio, TX*, vol. 3, pp. 11–15, 2018.

[51] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, *et al.*, "Commentary: The materials project: A materials genome approach to accelerating materials innovation," *APL materials*, vol. 1, no. 1, 2013.

[52] W. Hu, M. Fey, H. Ren, M. Nakata, Y. Dong, and J. Leskovec, "Ogb-lsc: A large-scale challenge for machine learning on graphs," *arXiv preprint arXiv:2103.09430*, 2021.

[53] C. Edwards, C. Zhai, and H. Ji, "Text2Mol: Cross-modal molecule retrieval with natural language queries," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, eds.), (Online and Punta Cana, Dominican Republic), pp. 595–607, Association for Computational Linguistics, Nov. 2021.

[54] T. Guo, B. Nan, Z. Liang, Z. Guo, N. Chawla, O. Wiest, X. Zhang, *et al.*, "What can large language models do in chemistry? a comprehensive benchmark on eight tasks," *Advances in Neural Information Processing Systems*, vol. 36, pp. 59662–59688, 2023.

[55] R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic, "Galactica: A large language model for science," *arXiv preprint arXiv:2211.09085*, 2022.

[56] G. Helman, I. Shah, and G. Patlewicz, "Transitioning the generalised read-across approach (genra) to quantitative predictions: a case study using acute oral toxicity data," *Computational Toxicology*, vol. 12, p. 100097, 2019.

[57] C. Edwards, T. Lai, K. Ros, G. Honke, K. Cho, and H. Ji, "Translation between molecules and natural language," 2022.

[58] M. Krenn, Q. Ai, S. Barthel, N. Carson, A. Frei, N. C. Frey, P. Friederich, T. Gaudin, A. A. Gayle, K. M. Jablonka, *et al.*, "Selfies and the future of molecular string representations," *Patterns*, vol. 3, no. 10, 2022.

[59] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, PmLR, 2020.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Section 1

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Appendix

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.

- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Appendix

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: See Abstract

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4. Additional experiment settings details are available at code documentation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: See Section 5.3; Zero temperature

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: appendix

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research adheres to the NeurIPS Code of Ethics

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: See Appendix

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Previous Datasets used have been cited in the manuscript.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Code, Datasets have been released as a part of the manuscript.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Evaluation of LLMs for certain tasks

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.