MolXPT: Wrapping Molecules with Text for Generative Pre-training

Zequn Liu[1]*, **Wei Zhang**[2]*, **Yingce Xia**[3]†, **Lijun Wu**[3], **Shufang Xie**[4],
Tao Qin[3], **Ming Zhang**[1]† **and Tie-Yan Liu**[3]
[1] Peking University; [2] University of Science and Technology of China
[3] Microsoft Research AI4Science; [4] Renmin University of China
{zequnliu,mzhang_cs}@pku.edu.cn; weizhang_cs@mail.ustc.edu.cn
{yingce.xia, lijunwu, taoqin, tyliu}@microsoft.com
shufangxie@ruc.edu.cn

Abstract

Generative pre-trained Transformer (GPT) has demonstrates its great success in natural language processing and related techniques have been adapted into molecular modeling. Considering that text is the most important record for scientific discovery, in this paper, we propose MolXPT, a unified language model of text and molecules pre-trained on SMILES (a sequence representation of molecules) wrapped by text. Briefly, we detect the molecule names in each sequence and replace them to the corresponding SMILES. In this way, the SMILES could leverage the information from surrounding text, and vice versa. The above wrapped sequences, text sequences from PubMed and SMILES sequences from PubChem are all fed into a language model for pre-training. Experimental results demonstrate that MolXPT outperforms strong baselines of molecular property prediction on MoleculeNet, performs comparably to the best model in text-molecule translation while using less than half of its parameters, and enables zero-shot molecular generation without finetuning.

1 Introduction

Generative pre-trained Transformer (GPT), like GPT-3 (Brown et al., 2020) and ChatGPT (OpenAI, 2022), have obtained great success in natural language processing. They usually have billions of parameters and are trained on large corpus (Taylor et al., 2022; Singhal et al., 2022). By witnessing their great power, people start transferring language models to chemical (Bagal et al., 2022) and biological domains (Ferruz et al., 2022). For example, a small molecule (e.g., an oral drug) can be represented using simplified molecular-input line-entry system (SMILES) (Weininger, 1988), which is a sequence obtained by traversing the molecular graph using depth-first-search and several rules for branching, aromaticity, etc. After serializing molecules, people pre-train language models on SMILES (Bagal et al., 2022; Tong et al., 2021; Frey et al., 2022) and obtain promising results for molecular generation.

Text is the most important record for molecular science and more generally, scientific discovery (Beltagy et al., 2019). It describes detailed properties of molecules, like how to synthesize the molecule (Feng et al., 2016), whether the molecule is toxic (Juurlink et al., 2003), etc. BioGPT (Luo et al., 2022) and PubMedGPT (Bolton et al., 2022) are two language models trained on biomedical literature. Recently, a new trend is to jointly model SMILES and scientific text so as to obtain shared representations across the two modalities. MolT5 is a T5-like (Raffel et al., 2020) model, where several spans of the text/SMILES are masked in the encoder and they should be reconstructed in the decoder. Galactica (Taylor et al., 2022) is a GPT-like (Brown et al., 2020) model pre-trained on various types of inputs, like text, SMILES, protein sequences, etc. Although those models demonstrate progress in prediction and generation tasks, they do not explicitly leverage the relation between molecules and text. An intuition is that, in scientific literature, when a molecule name appears in a sentence, the surrounding context could be a description of the molecule. This should be useful information for joint training but is ignored in those models.

To leverage such relations, in this work, we propose a novel molecule-text language model (MolXPT), which is trained on "wrapped" sequences: Given a sentence, we detect the molecular names with named entity recognition tools, and if any, replace them to the corresponding SMILES and obtain the "wrapped" sequence between SMILES and text. We pre-train a 24-layer MolXPT (with $350M$ parameters) on 8M wrapped sequences, as well as 30M SMILES from PubChem

[*] Equal contribution. This work was done when Z. Liu and W. Zhang were interns at Microsoft Research AI4Science.
[†] Corresponding authors.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1606–1616
July 9-14, 2023 ©2023 Association for Computational Linguistics

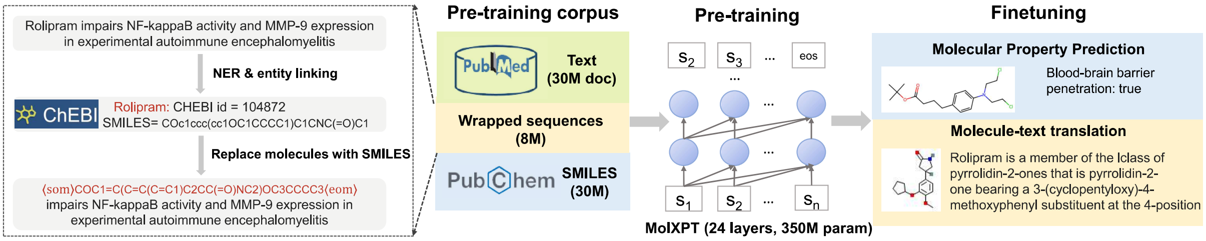

Rolipram impairs NF-kappaB activity and MMP-9 expression in experimental autoimmune encephalomyelitis
NER & entity linking
ChEBI
Rolipram: CHEBI id = 104872
SMILES= COc1ccc(cc1OC1CCCC1)C1CNC(=O)C1
Replace molecules with SMILES
⟨som⟩COC1=C(C=C(C=C1)C2CC(=O)NC2)OC3CCCC3⟨eom⟩ impairs NF-kappaB activity and MMP-9 expression in experimental autoimmune encephalomyelitis
Pre-training corpus
PubMed
Text (30M doc)
Wrapped sequences (8M)
PubChem
SMILES (30M)
Pre-training
s_2
s_3
...
eos
s_1
s_2
s_n
MolXPT (24 layers, 350M param)
Finetuning
Molecular Property Prediction
Blood-brain barrier penetration: true
Molecule-text translation
Rolipram is a member of the lclass of pyrrolidin-2-ones that is pyrrolidin-2-one bearing a 3-(cyclopentyloxy)-4-methoxyphenyl substituent at the 4-position

Figure 1: Framework of MolXPT. MolXPT is pretrained on text from PubMed, SMILES from PubChem and wrapped sequences between SMILES and text. The wrapped sequences are obtained by applying NER and entity linking to text and then replacing matched molecular mentions with SMILES. MolXPT can be finetuned for various text and molecular downstream tasks, like molecular property prediction and molecule-text translation.

(Kim et al., 2022) and 30M titles and abstracts from PubMed (a popular biomedical literature search engine).

After pre-training, we finetune MolXPT on MoleculeNet (a benchmark about molecular property prediction) (Wu et al., 2018) and molecule-text translation (Edwards et al., 2022) using prompt-based finetuning. On MoleculeNet, MolXPT outperforms strong baselines with sophisticated design like GEM (Fang et al., 2022). On text-molecule translation, MolXPT performs comparably with the state-of-the-art model, MolT5-large (Edwards et al., 2022). MolT5-large has $800M$ parameters while MolXPT only uses 44% of its parameters. We also verify that MolXPT has the zero-shot ability on text-to-molecule generation.

2 Our Method

MolXPT is a language model pre-trained on heterogeneous data including scientific text, SMILES sequences, and "wrapped" sequences between SMILES and text. Due to the flexible input, we can finetune it for various text and molecular tasks. The framework of MolXPT is in Figure 1.

2.1 Pre-training corpus

For scientific text, we use the titles and abstracts of 30M papers from PubMed[1]. For molecular SMILES, we randomly choose 30M molecules from PubChem[2] (Kim et al., 2022).

The wrapped sequences are constructed via a "detect and replace" pipeline. We first use BERN2 (Sung et al., 2022), a widely used named entity recognition (NER) tool for biomedical purpose, to detect all mentions of molecules and link them to the entities in public knowledge bases like ChEBI (Hastings et al., 2016). After that, we can retrieve the molecular SMILES of the matched entities. Finally, we replace the molecular mentions to their corresponding SMILES. An example is shown in the left panel of Figure 1. The wrapped sequences must contain at least one molecular SMILES. We eventually obtain 8M wrapped sequences in total.

Text and SMILES are tokenized separately. For text, we use byte-pair encoding (BPE) (Sennrich et al., 2016) to split the words into subwords. The number of BPE merge operation is 40k. For SMILES sequences (including those in wrapped sequences), we tokenize them with the regular expression from (Schwaller et al., 2018). For each SMILES sequence S, we add a start-of-molecule token $\langle\text{som}\rangle$ at the beginning of S and append an end-of-molecule token $\langle\text{eom}\rangle$ at the end of S.

2.2 Model and training

Model architecture: MolXPT has the same architecture as the GPT models (Radford et al., 2019). Due to computational resource limitation, in this paper, we follow the GPT-2_{medium} configuration with 24 layers, 1024 hidden size and 16 attention heads. The maximum length of input we can process is 2048 and the vocabulary size is 44536. In total, our model has 350M parameters.

Pre-training: The pre-training objective function of MolXPT is the negative log-likelihood. Mathematically, let $\mathcal{D} = \{x_i\}_i$ denote the collection of sequences of the three types of the data, and $x_i = (s_{i,1}, s_{i,2}, \cdots, s_{i,n_i})$ is the i-th sequence with n_i tokens. The training objective function is:

$$\min -\frac{1}{|\mathcal{D}|}\sum_{i=1}^{|\mathcal{D}|}\sum_{j=1}^{n_i}\log P(s_{i,j}|s_{i,j-1}, s_{i,j-2}, \cdots, s_1).$$

The pre-training details are left in Appendix B.

[1] https://ftp.ncbi.nlm.nih.gov/pubmed/
[2] https://pubchem.ncbi.nlm.nih.gov/

Dataset **#Molecules**	BBBP 2039	Tox21 7831	ClinTox 1478	HIV 41127	BACE 1513	SIDER 1478	Avg
G-Contextual	70.3 ± 1.6	75.2 ± 0.3	59.9 ± 8.2	75.9 ± 0.9	79.2 ± 0.3	58.4 ± 0.6	69.8
G-Motif	66.4 ± 3.4	73.2 ± 0.8	77.8 ± 2.0	73.8 ± 1.4	73.4 ± 4.0	60.6 ± 1.1	70.9
GROVER$_{\text{base}}$	70.0 ± 0.1	74.3 ± 0.1	81.2 ± 3.0	62.5 ± 0.9	82.6 ± 0.7	64.8 ± 0.6	72.6
GROVER$_{\text{large}}$	69.5 ± 0.1	73.5 ± 0.1	76.2 ± 3.7	68.2 ± 1.1	81.0 ± 1.4	65.4 ± 0.1	72.3
GraphMVP	72.4 ± 1.6	75.9 ± 0.5	79.1 ± 2.8	77.0 ± 1.2	81.2 ± 0.9	63.9 ± 1.2	74.9
MGSSL	70.5 ± 1.1	76.5 ± 0.3	80.7 ± 2.1	79.5 ± 1.1	79.7 ± 0.8	61.8 ± 0.8	74.8
GEM	72.4 ± 0.4	**78.1 ± 0.1**	90.1 ± 1.3	**80.6 ± 0.9**	85.6 ± 1.1	67.2 ± 0.4	79.0
KV-PLM	74.6 ± 0.9	72.7 ± 0.6	–	74.0 ± 1.2	–	61.5 ± 1.5	–
Galactica	66.1	68.9	82.6	74.5	61.7	63.2	69.5
MoMu	70.5 ± 2.0	75.6 ± 0.3	79.9 ± 4.1	76.2 ± 0.9	77.1 ± 1.4	60.5 ± 0.9	73.3
MolXPT	**80.0 ± 0.5**	77.1 ± 0.2	**95.3 ± 0.2**	78.1 ± 0.4	**88.4 ± 1.0**	**71.7 ± 0.2**	**81.9**

Table 1: Results on MoleculeNet. The evaluation metric is ROC-AUC. Bold fonts indicate the best results.

Prompt-based finetuning: MolXPT can be finetuned for downstream tasks about molecules and text. Adding classification or regression heads to pre-trained backbone models introduces the gap between pre-training and finetuning (Brown et al., 2020; Chen et al., 2022; Gu et al., 2022). Therefore, we adopt prompt-based finetuning (Gao et al., 2021) to unify different tasks into a sequence generation task, which is consistent with the pre-training objective. Briefly, given a task, we convert the input and output into text and/or SMILES sequences, equip the sequences with task-specific prompts and finetune using language modeling loss. Prompts for MoleculeNet and text-molecule translation are introduced in the Section 3.1 and 3.2 respectively.

Discussion: Some works also try to jointly model text and molecules. Zeng et al. (2022) propose KV-PLM, where SMILES sequences are appended after molecule names for pre-training. Su et al. (2022) use contrastive learning between text and molecular graphs. Our MolXPT is a generative model while the above two models are not. Both of them are built upon SciBERT (Beltagy et al., 2019), a BERT model (Devlin et al., 2019) for scientific literature. MolXPT is complementary to them.

3 Experiments

We evaluated MolXPT on two downstream tasks: (1) molecular property prediction on MoleculeNet (Wu et al., 2018), which is to predict whether the given molecule has specific properties; (2) the generation between text descriptions and molecules (Edwards et al., 2022), where both molecules and text should be considered. In this section, we focus on introducing task definition, prompt design and results while leaving the detailed finetuning hyper-parameters in Appendix C.

3.1 Results on MoleculeNet

MoleculeNet (Wu et al., 2018) is a widely-used benchmark for molecular modeling, which has more than $700k$ compounds for various different properties. We choose six molecular classification tasks for evaluation, which are BBBP, Tox21, ClinTox, HIV, BACE and SIDER. Details are left in Appendix A. We follow GEM (Fang et al., 2022) to split the data into training/validation/test sets based on the scaffold. For these tasks, the input is a SMILES and the output is a binary label.

Finetuning strategy: Previous molecular property prediction models mainly use SMILES sequences or molecular graphs as input, while we can use the "wrapped" sequences. For example, one task is to predict the blood-brain barrier penetration (BBBP) of a molecule. Therefore, the prompt is "*We can conclude that the BBB penetration of* ⟨som⟩ ⟨`SMILES`⟩ ⟨eom⟩ is ⟨tag⟩", where ⟨`SMILES`⟩ denotes the molecular SMILES, and ⟨tag⟩ denotes the classification result. For the BBBP task, we design ⟨tag⟩ as "true" or "false", indicating whether the compound can or cannot cross BBB.

Different tasks have different prompts (see Appendix C.1), but we put the tags to the last token of the prompt for all tasks. Let $(s_{i,1}, s_{i,2}, \cdots, s_{i,T_i})$ denote the i-th wrapped sequence for the downstream task with T_i tokens, where s_{i,T_i} is the tag of the sequence. Denote that there are N samples for finetuning. The finetuning strategy could be either

$$\min -\frac{1}{N}\sum_{i=1}^{N}\log P(s_{i,T_i}|s_{i,<T_i}), \qquad (1)$$

indicating that we finetune the tags only, or

$$\min -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{T_i}\sum_{j=1}^{T_i}\log P(s_{i,j}|s_{i,<j}), \quad (2)$$

indicating that we finetune the full prompts. According to our exploration, Eqn.(1) achieves slightly better results and we use it for all tasks (see Appendix C.4 for the results).

Let p_{true} and p_{false} denote the probabilities of tags "true" and "false" after encoding the prefix "*We can conclude that the BBB penetration of* $\langle$som$\rangle$ $\langle$`SMILES`$\rangle$ $\langle$eom$\rangle$ *is*". The probabilities that $\langle$`SMILES`$\rangle$ can and cannot cross blood-brain barrier are normalized as $p_{\text{true}}/(p_{\text{true}}+p_{\text{false}})$ and $p_{\text{false}}/(p_{\text{true}}+p_{\text{false}})$ respectively. The finetuning hyper-parameters are in Appendix C.2.

We compare MolXPT with two types of baselines: (1) pre-trained language model baselines including KV-PLM (Zeng et al., 2022), Galactica (Taylor et al., 2022) and MoMu (Su et al., 2022). (2) pre-trained Graph Neural Network (GNN) baselines including G-Contextual (Rong et al., 2020), G-Motif (Rong et al., 2020), GROVER$_{\text{base}}$ (Rong et al., 2020), GROVER$_{\text{large}}$ (Rong et al., 2020), GraphMVP (Liu et al., 2022), MGSSL (Zhang et al., 2021) and GEM (Fang et al., 2022). The evaluation metric is the ROC-AUC score. The results are in Table 1.

MolXPT outperforms the GNN baselines pre-trained on pure molecular data, indicating the effectiveness of pre-training with scientific text corpus. Compared with Galactica which also uses both SMILES and text for pre-training GPT-like model, MolXPT obtains better performance. Note that Galactica does not purposely build and train on the "wrapped" sequences, whose importance is demonstrated via our empirical results. A possible explanation of the superior performance is that the SMILES describes the component and structural information of molecules, while the text describes the general properties. They are complementary to each other, and joint training on them brings more effective representations.

3.2 Results on text-molecule translation

We evaluated the performance of MolXPT on CheBI-20 (Edwards et al., 2021), a bidirectional text-molecule translation dataset. It consists of 33,010 molecule-description pairs. We use the data split provided by MolT5 (Edwards et al., 2022), where the training, validation and test sets account 80%, 10% and 10% of total data. For molecule-to-text generation, given a molecular SMILES S, the prompt is: "*The description of* $\langle$som$\rangle$ S $\langle$eom$\rangle$ *is: The molecule is*", followed by the text description of S. For text-to-molecule generation, given a text description T, the prompt is: "T. *The compound is* $\langle$som$\rangle$", and the model will generate the molecular SMILES ended with $\langle$eom$\rangle$. We compare our method with MolT5 (Edwards et al., 2022).

For molecule-to-text generation, the results are evaluated by NLP metrics including BLEU (Papineni et al., 2002), Rouge (Lin, 2004) and METEOR (Banerjee and Lavie, 2005). "Text2mol" is a deep learning based metric proposed by Edwards et al. (2022) to measure the similarity of the text-molecule pairs. For text-to-molecule generation, we evaluate the following metrics: the proportion of the generated SMILES that exactly match the reference SMILES (denoted as "Exact"); the Tanimoto similarity of three types of fingerprints: MACCS (Durant et al., 2002), RDK (Schneider et al., 2015) and Morgan (Rogers and Hahn, 2010); the FCD score (Preuer et al., 2018), which measures the molecule distances by a pre-trained model; the percentage of the valid generated SMILES. The results are reported in Table 2.

We observe that MolXPT achieves significantly better performance than MolT5-small and MolT5-base, and has comparable performance with MolT5-large. Note that MolT5-large has $800M$ parameters while MolXPT only uses 44% of its parameters. For both tasks, our model performs the best on Text2Mol metric, indicating that MolXPT captures the alignment between text and molecule better. We attribute it to the wrapped sequences, by which the model can learn the relation between molecule and text explicitly.

We further verify the zero-shot text-to-molecule generation ability of MolXPT. The pre-trained MolXPT takes the text as input and directly generates molecules without finetuning. The top-1 and top-5 fingerprint similarity is in Table 3. Indeed, compared with the full data setting, the performance drops, but still reasonable numbers. In addition, the zero-shot MolXPT successfully recovers 33 molecules based on the text (see Appendix D).

4 Conclusions and Future Work

We propose MolXPT, a generative model pre-trained on scientific text, molecular SMILES and

Molecule-to-text	BLEU-2	BLEU-4	Rouge-1	Rouge-2	Rouge-L	METEOR	Text2Mol
MolT5-small (77M)	0.519	0.436	0.620	0.469	0.563	0.551	0.540
MolT5-base (250M)	0.540	0.457	0.634	0.485	0.578	0.569	0.547
MolT5-Large (800M)	**0.594**	**0.508**	0.654	0.510	0.594	0.614	0.582
MolXPT (350M)	**0.594**	0.505	**0.660**	**0.511**	**0.597**	**0.626**	**0.594**

Text-to-molecule	Exact↑	MACCS↑	RDK↑	Morgan↑	FCD↓	Text2mol↑	Validity↑
MolT5-small	0.079	0.703	0.568	0.517	2.49	0.482	0.721
MolT5-medium	0.081	0.721	0.588	0.529	2.18	0.496	0.772
MolT5-large	**0.311**	0.834	0.746	**0.684**	1.20	0.554	0.905
MolXPT	0.215	**0.859**	**0.757**	0.667	**0.45**	**0.578**	**0.983**

Table 2: Results of molecule-to-text (top) and text-to-molecule generation (bottom). For FCD, the smaller, the better. For the remaining metrics, the larger, the better. MolT5 results are from Table 1 and 2 of (Edwards et al., 2022). MolT5 parameters are from `https://github.com/blender-nlp/MolT5`. Bold fonts indicate the best results.

	MACCS	RDK	Morgan
Zero-shot (Top-1)	0.540	0.383	0.228
Zero-shot (Top-5)	0.580	0.423	0.423
Full data (Top-1)	0.841	0.746	0.660

Table 3: Zero-shot text-to-molecule generation.

their wrapped sequences. We train a 24-layer MolXPT with 350M parameters. By prompt-based finetuning, it improves strong baselines on MoleculeNet and achieves comparable results with the best model on molecule-text translation but using much fewer parameters.

For future work, first, we will train larger MolXPT to further verify the performances across different tasks and the zero-shot/in-context (Xie et al., 2022) learning ability. Second, how to further enhance the interaction between molecules and text (e.g., using contrastive learning to enhance consistency) should be studied. Third, how to effectively adapt MolXPT into other molecule and text tasks such as text-guided molecule optimization is another direction to explore.

Limitations

One limitation of our method is that when training larger models, it requires more computation resources, whose cost is relatively high. However, after pre-training, we will release our models so that readers can directly use them without pre-training again.

Broader Impacts

We provide a new generative pre-trained model on molecules and text. On one hand, the model can be used to speed up scientific discovery, like molecule design, drug optimization, etc. On the other hand, once the model is trained on clinical data (which also describes the usage of drug molecules), it might lead to personal information leaky. We will enhance data filtration to anonymize all personal information, and will design new models to protect the information.

Acknowledgement

The authors Zequn Liu and Ming Zhang are partially supported by National Natural Science Foundation of China (NSFC Grant Number 62276002).

References

Viraj Bagal, Rishal Aggarwal, P. K. Vinod, and U. Deva Priyakumar. 2022. Molgpt: Molecular generation using a transformer-decoder model. *Journal of Chemical Information and Modeling*, 62(9):2064–2076.

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

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Elliot Bolton, David Hall, Michihiro Yasunaga, Tony Lee, Chris Manning, and Percy Liang. 2022. PubMedGPT 2.7B.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yulong Chen, Yang Liu, Li Dong, Shuohang Wang, Chenguang Zhu, Michael Zeng, and Yue Zhang. 2022. Adaprompt: Adaptive model training for prompt-based nlp. *arXiv preprint arXiv:2202.04824*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, and Heng Ji. 2022. Translation between molecules and natural language. *arXiv preprint arXiv:2204.11817*.

Carl Edwards, ChengXiang Zhai, and Heng Ji. 2021. Text2mol: Cross-modal molecule retrieval with natural language queries. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 595–607.

Xiaomin Fang, Lihang Liu, Jieqiong Lei, Donglong He, Shanzhuo Zhang, Jingbo Zhou, Fan Wang, Hua Wu, and Haifeng Wang. 2022. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 4(2):127–134.

Minghao Feng, Bingqing Tang, Steven H Liang, and Xuefeng Jiang. 2016. Sulfur containing scaffolds in drugs: synthesis and application in medicinal chemistry. *Current topics in medicinal chemistry*, 16(11):1200–1216.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. 2022. Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications*, 13(1):4348.

Nathan Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gomez-Bombarelli, Connor Coley, and Vijay Gadepally. 2022. Neural scaling of deep chemical models. *ChemRxiv*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. Ppt: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423.

Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. 2016. Chebi in 2016: Improved services and an expanding collection of metabolites. *Nucleic acids research*, 44(D1):D1214–D1219.

David N Juurlink, Muhammad Mamdani, Alexander Kopp, Andreas Laupacis, and Donald A Redelmeier. 2003. Drug-drug interactions among elderly patients hospitalized for drug toxicity. *Jama*, 289(13):1652–1658.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E Bolton. 2022. PubChem 2023 update. *Nucleic Acids Research*, 51(D1):D1373–D1380.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

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

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*.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. Technical blog.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318.

Kristina Preuer, Philipp Renz, Thomas Unterthiner, Sepp Hochreiter, and Gunter Klambauer. 2018. Frechet chemnet distance: a metric for generative models for molecules in drug discovery. *Journal of chemical information and modeling*, 58(9):1736–1741.

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.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. 2020. Self-supervised graph transformer on large-scale molecular data. *Advances in Neural Information Processing Systems*, 33:12559–12571.

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

Philippe Schwaller, Theophile Gaudin, David Lanyi, Costas Bekas, and Teodoro Laino. 2018. "found in translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chemical science*, 9(28):6091–6098.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2022. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*.

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*.

Mujeen Sung, Minbyul Jeong, Yonghwa Choi, Donghyeon Kim, Jinhyuk Lee, and Jaewoo Kang. 2022. Bern2: an advanced neural biomedical named entity recognition and normalization tool. *arXiv preprint arXiv:2201.02080*.

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*.

Xiaochu Tong, Xiaohong Liu, Xiaoqin Tan, Xutong Li, Jiaxin Jiang, Zhaoping Xiong, Tingyang Xu, Hualiang Jiang, Nan Qiao, and Mingyue Zheng. 2021. Generative models for de novo drug design. *Journal of Medicinal Chemistry*, 64(19):14011–14027.

David Weininger. 1988. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36.

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

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

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.

Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee. 2021. Motif-based graph self-supervised learning for molecular property prediction. *Advances in Neural Information Processing Systems*, 34:15870–15882.

Appendix

A Datasets and Baselines of MoleculeNet

We choose the following tasks of MoleculeNet for evaluation:
(1) BBBP contains compounds with binary labels on blood-brain barrier penetration.
(2) Tox21 is a dataset for predicting the human toxicity of compounds on 12 different targets.
(3) ClinTox contains drugs approved by the FDA and those that have failed clinical trials for toxicity reasons.
(4) HIV aims to predict whether a drug can inhibit HIV replication.
(5) BACE describes binding results for a set of inhibitors of human β-secretase 1.
(6) SIDER has compounds used in marketed medicines with 27 categories of side effects.
We compare MolXPT with the following baselines:

(1) GROVER is a self-supervised pre-trained graph Transformer model. G-Contextual and G-Motif are two variants of it pre-trained with contextual property prediction task and motif prediction task.
(2) GraphMVP is a self-supervised pre-trained GNN model using both 2D topological structures and 3D geometric views of molecules.
(3) MGSSL leverages a retrosynthesis-based algorithm BRICS and additional rules to find the motifs and combines motif layers with atom layers.
(4) GEM is a geometry-enhanced pre-trained GNN model.
(5) Galactica is a GPT-like model trained on a large scientific corpus and many natural sequences like SMILES. We report the result of Galactica-120B.
(6) KV-PLM is a BERT-like model where SMILES sequences are appended after molecule names for pre-training.
(7) MoMu uses contrastive learning to jointly pre-train a BERT model for text and a GNN model for molecules.

B Pre-training hyper-parameters

MolXPT is pre-trained for 200k steps on eight A100 GPUs. The batchsize is 2048 tokens per GPU. The gradients are accumulated for 16 steps before updating. We use Adam (Kingma and Ba, 2015) optimizer for optimization. The peak learning rate is 0.0005 and the warm-up steps are 20000. The learning rate scheduler is inverse square root decay scheduler. The dropout is 0.1.

C Finetuning details of downstream tasks

C.1 Prompts for finetuning MoleculeNet

(1) BBBP: "*We can conclude that the BBB penetration of* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *is true/false.*"
(2) Tox21: "*We can conclude that the* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *activity outcome on* ⟨*target*⟩ *is active/inactive.* " where ⟨*target*⟩ refers to corresponding receptor or enzyme for each subtask, e.g. the ⟨*target*⟩ of subtask "AR" is "Androgen Receptor".
(3) ClinTox:"*We can conclude that the clinical trial toxicity of* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *is true/false.*" for subtask CT_TOX and "*We can conclude that the FDA approval status of* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *is true/false.*" for subtask FDA_APPROVED.
(4) HIV: "*We can conclude that the screening result of ability to inhibit HIV replication of* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *is active/inactive.*"
(5) BACE: "*We can conclude that the binding result on beta-secretase 1 of* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *is true/false.*"
(6) SIDER:"*We can conclude that the* ⟨som⟩ ⟨SMILES⟩ ⟨eom⟩ *can bring about the side effect of* ⟨*side-effect*⟩ *is true/false.*" where ⟨*side-effect*⟩ refers to corresponding side-effect for each subtask.

C.2 Details of finetuning MoleculeNet

We grid search the following hyper-parameters: learning rate in $\{3 \times 10^{-5}, 5 \times 10^{-5}\}$; dropout in $\{0.1, 0.3\}$; total epochs from $\{30, 50\}$. The model is selected according to validation performance.

C.3 Details of finetuning text-molecule generation

For text-molecule generation, MolXPT is finetuned for 100 steps on one P40 GPU with 1024 tokens and 16 accumulated steps per device. Models are finetuned for 100 epochs. The learning rate is 0.0001 and the dropout rate is grid searched from $[0.1, 0.2, 0.3, 0.4, 0.5]$. Setting dropout rate as 0.4 and 0.5 achieves the best validation performance on molecule-to-text generation and text-to-molecule generation respectively. We use the corresponding models for testing.

C.4 MoleculeNet finetuning strategy selection

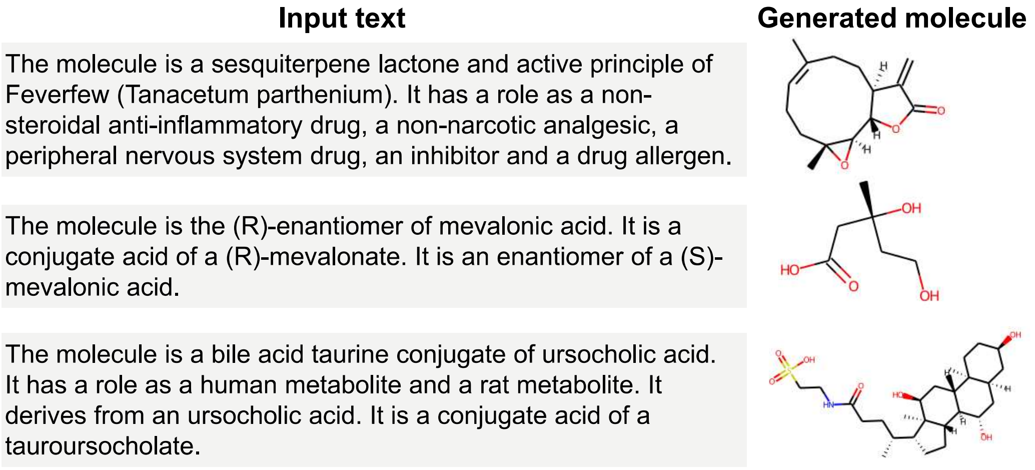
We provide two finetune strategies in Eqn.(1) and Eqn.(2). Their results are reported in Table 4. Their results are similar and Eqn.(1) is slightly better.

D Zero-shot text-to-molecule generation

Given K generated molecule $\hat{m}_1, \hat{m}_2, \cdots, \hat{m}_K$ and the reference molecule m, the top-K fingerprint similarity is

$$\max_{i \in [K]} \texttt{similarity}(m, \hat{m}_i). \quad (3)$$

MolXPT generates 33 molecules that can exactly match the reference molecules without finetuning. Figure 2 shows three of the cases.



The molecule is a sesquiterpene lactone and active principle of Feverfew (Tanacetum parthenium). It has a role as a non-steroidal anti-inflammatory drug, a non-narcotic analgesic, a peripheral nervous system drug, an inhibitor and a drug allergen.

The molecule is the (R)-enantiomer of mevalonic acid. It is a conjugate acid of a (R)-mevalonate. It is an enantiomer of a (S)-mevalonic acid.

The molecule is a bile acid taurine conjugate of ursocholic acid. It has a role as a human metabolite and a rat metabolite. It derives from an ursocholic acid. It is a conjugate acid of a tauroursocholate.

Figure 2: Examples for zero-shot text-to-molecule generation. We randomly pick up three cases that MolXPT can successfully generate the reference molecules without finetuning.

Dataset	BBBP	Tox21	ClinTox	HIV	BACE	SIDER	Avg
$\text{Dev}_{\text{full prompt}}$	98.8 ± 0.2	78.8 ± 0.1	98.8 ± 0.1	82.9 ± 1.0	78.4 ± 0.3	67.7 ± 0.7	84.2
$\text{Dev}_{\text{tags only}}$	98.9 ± 0.3	78.8 ± 0.2	97.7 ± 0.1	85.3 ± 0.2	75.8 ± 0.8	69.4 ± 0.6	84.3
$\text{Test}_{\text{full prompt}}$	78.1 ± 0.4	77.2 ± 0.1	93.4 ± 0.1	78.1 ± 0.9	87.9 ± 0.3	70.0 ± 0.2	80.8
$\text{Test}_{\text{tags only}}$	80.0 ± 0.5	77.1 ± 0.2	95.3 ± 0.2	78.1 ± 0.4	88.4 ± 1.0	71.7 ± 0.2	81.9

Table 4: Comparison of different finetuning strategies on MoleculeNet. “Dev” and “Test” denote validation set and test set respectively. Subscripts represent finetuning full prompts (Eqn.(2)) or tags only respectively (Eqn.(1)). The evaluation metric is ROC-AUC.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
The section called "Limitations".

☑ A2. Did you discuss any potential risks of your work?
The section called "Ethnics statement".

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Introduction (Section 1)

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Section 2.2 and Appendix B

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 2.2 and Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 3.1, Section 3.2 and Appendix C

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 3.1, Section 3.2

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 3.2.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.