Studying various training approaches for the MoLFormer model on the Lipophilicity dataset

Anchit Gupta
Matr. No. 7035463

Lei Huang
Matr. No. 2575683

Ruoxuan Liu
Matr. No. 7047285

Abstract

In this paper, we explored and implemented various data selection and fine-tuning strategies to adapt the pre-trained chemical language model, MoLFormer, to the regression task of predicting lipophilicity values. First, we trained our modified MoLFormer model with a regression head on the lipophilicity dataset. To enhance the robustness of our model's performance, we selected external data points for training. For this purpose, we employed various data selection strategies: (1) applying influence scores with the LISSA approximation, (2) prediction error-based data selection, (3) aligning the distribution of the external dataset's lipophilicity values with the distribution of the original training set, and (4) constructing clusters using the K-means clustering method and silhouette scores to identify the most suitable data points. Subsequently, we implemented three fine-tuning strategies — BitFit, LoRA, and iA3, to fine-tune the model in hopes of improving the performance.

1 Introduction

Lipophilicity (LogP) measures the oil-to-water solubility of a substance (Chandrasekaran et al., 2018) and is crucial in drug discovery, agrochemicals, and environmental chemistry (Bhal, 2007; Johnson et al., 2018). Machine learning and deep learning have been widely used for its prediction (Datta et al., 2021; Isert et al., 2023; Schroeter et al., 2007; Jia et al., 2022), but challenges remain, including limited data, diverse molecular structures, and algorithm selection (Wu et al., 2018).

To address these issues, we trained a deep learning model using the MoleculeNet Lipophilicity dataset, which provides diverse datasets and standardized evaluation (Wu et al., 2018). We adapted the pre-trained MoLFormer model with a regression head and employed data selection strategies, including LISSA-based influence scores, maximum error samples, distribution matching, and K-means clustering. Fine-tuning with iA3, LoRA, and BitFit further enhanced performance.

The following sections cover related work, methodology, results, and conclusions, providing insight for future research.

2 Related Work

Machine learning methods like Random Forest, XGBoost, and Bayesian approaches have been used for lipophilicity prediction (Singh and Sunoj, 2023; Isert et al., 2023; Schroeter et al., 2007), though deep learning often outperforms them (LeCun et al., 2015; Li et al., 2022).

Graph-based models (e.g., attention-enhanced GCNs) show promise but struggle with graph isomorphism; DIN mitigates this using D-MPNN and GIN (Yang et al., 2019; Xu et al., 2018; Wieder et al., 2021).

Sequential models like seq2seq and Seq3seq rely on semi-supervised learning but face RNN inefficiencies (Mswahili and Jeong, 2024), while transformer-based models trained on molecular data improve SMILES-based predictions (Liu et al., 2023; Fabian et al., 2020).

Recent studies highlight that combining trainable molecular representations with simple regression models performs well (Lukashina et al., 2020), an approach we adopt alongside external data selection and fine-tuning.

3 Methodology

3.1 Dataset

We used the MoleculeNet lipophilicity dataset with SMILES strings as input and lipophilicity values as labels.

The dataset is divided into 68% training, 12% validation and 20% test sets to provide a good balance in our training approach.

3.2 Base Model

In this study, we used MoLFormer as our foundational model, which had been trained on SMILES string representations encompassing approximately 1.1 billion molecules sourced from the ZINC (Irwin et al., 2012) and PubChem (Wang et al., 2009) databases (Ross et al., 2022).

3.3 Model Modifications

To better adapt Molformer to our Lipophilicity Value Prediction Task, we added a regression head with two hidden layers (256 and 64 units, ReLU, dropout 0.1) to it.

3.4 Training Setup

During training, the model uses the Adam optimizer, coupled with a `ReduceLROnPlateau` scheduler for Task 1 and Task 2 to dynamically adjust the learning rate. For Task 3, we used `get_linear_schedule_with_warmup` scheduler as it provided a better learning rate decay mechanism for our use case. Training runs up to 200 epochs with early stopping (patience = 10) to prevent overfitting.

3.5 External Data Points Selection Strategies

Leveraging external information is widely encouraged in data integration and transfer learning frameworks, as joint learning often yields more accurate inference compared to the analysis of datasets in isolation (Rognon-Vael et al., 2025). To enhance our model training process, we implemented the following external data selection strategies:

3.5.1 Influence Scores with LiSSA Approximation

We use influence functions to select high-impact external data for the Lipophilicity dataset. Since computing the exact Hessian inverse is impractical, we estimate the inverse Hessian vector product (iHVP) using LiSSA.

Our approach consists of:

1. **Test Gradient**: Compute the test loss gradient for trainable parameters to obtain $v = \nabla L_{test}$.
2. **LiSSA Approximation**: Iteratively compute Hessian vector products with `autograd.grad` (using `create\_graph=True`) to approximate $H^{-1}v$, applying damping and scaling for stability.
3. **Influence Calculation**: For each external sample, compute its gradient and dot it with the approximated iHVP to assess its impact.

We replace `None` gradients with zero tensors and manage memory with `zero_grad()` and `empty_cache()`. This method, based on (Koh and Liang, 2017) and (Agarwal et al., 2016), enables scalable influence estimation for data selection.

3.5.2 Prediction error-based data selection

We applied prediction error-based data selection to extract the top-k external samples that could best enhance our training set. The hypothesis is that samples with high prediction error likely contain novel or challenging information that is underrepresented in the current training set.

We first applied a preliminary model (the fine-tuned model from Task 1) to compute the absolute error for each sample in the external dataset. The samples were then ranked by error, and the top-k high-error points were added to the original training set to improve both its size and quality.

This approach not only increases data volume, but also forces the model to address previously unmodeled variations, enhancing its generalization on unseen data. Similar methods in active learning have shown improvements in sample efficiency and model accuracy (Settles, 2009; Cohn et al., 1996).

3.5.3 Distribution Alignment Between External and Original Datasets

A useful way to select data from an extra dataset is by analyzing its distribution. A Gaussian Mixture Model (GMM) helps learn the distribution of the original dataset and identifies relevant data points from the extra dataset based on log-likelihood scores. Data points that do not fit well into the learned distribution are selected, improving model generalization.

Since SMILES strings encode rich molecular structures, we first convert them into Morgan fingerprints (Rogers and Hahn, 2010), which represent structural features in a fixed-length binary format. To handle high dimensionality, we apply Principal Component Analysis (PCA) to reduce them to three components. A GMM with a single cluster is then trained on the original dataset, and the extra dataset is tested against this model. We select 100 points with the lowest likelihood, as they are the most informative for enhancing model robustness. Adding these points to the training set improves data diversity, leading to better generalization and accuracy

3.5.4 K-means Clustering with Silhouette Scores

To enhance the robustness of our model, we implemented an external data selection strategy leveraging embedding extraction and clustering. First, embeddings were extracted from the external dataset using the trained regression model, transforming each data point into a low-dimensional representation. These embeddings were then reshaped into a 2D format for clustering. The optimal number of clusters was determined using the Silhouette Score, which evaluates clustering quality by measuring the separation and cohesion of clusters (Shahapure and Nicholas, 2020). K-Means clustering was applied with the optimal number of clusters, and data points were selected from each cluster based on their proximity to the cluster centroid. To ensure a balanced distribution, a fixed number of points were chosen from each cluster, with additional points randomly selected from the remaining data if necessary. This approach ensures a diverse and representative subset of the external dataset for model training.

3.6 Model Fine-tuning Strategies

3.6.1 iA3

Infused Adapter Averaging (iA3) is a parameter-efficient fine-tuning method that adapts large pre-trained models while keeping most parameters frozen. Instead of modifying weights, iA3 introduces trainable scaling factors on intermediate activations, reducing the learnable parameters. Unlike LoRA, which applies low-rank updates to weight matrices, iA3 scales activations in attention and feedforward layers, preserving pre-trained knowledge while allowing targeted adaptation.

iA3's efficiency minimizes memory and computational overhead, making it ideal when full fine-tuning is impractical. By updating only a small fraction of parameters, it ensures stable gradient flow and mitigates catastrophic forgetting.

In our project, we explored iA3 as a lightweight alternative to full fine-tuning, leveraging its efficiency for integrating new data. This aligns with recent advances in parameter-efficient learning, inspired by works such as (Houlsby et al., 2019) and (Liu et al., 2022).

3.6.2 Lora

LoRA (Low-Rank Adaptation) is a technique that helps us to efficiently fine-tune large-language models by introducing a low-rank decomposition into weight updates (Hu et al., 2022). Instead of updating the full model weights, LoRA inserts small trainable matrices (low-rank adapters) that capture task-specific knowledge while keeping the original model frozen. This leads to lower memory usage and faster training.

In our project, we added the Lora parameters only to the key, value, and key layers, significantly reducing the number of trainable parameters. Meanwhile, we also kept our regression head trainable as it is the extra layer we added to the original model. Our method ensured that while the core transformer remained frozen, the model could still effectively learn task-specific representations through lightweight LoRA updates and a fully trainable regression head.

3.6.3 BitFit

In the Bias-term Fine-tuning approach (Zaken et al., 2021), all base model parameters were frozen except for the bias terms, which were allowed to be fine-tuned. Additionally, a task-specific regression head was added similar to Task 1, consisting of fully connected layers with ReLU activations and dropout for regularization. This regression head was fully trainable, enabling the model to adapt to the specific task of predicting lipophilicity.

The training process optimized only the bias terms and the regression head parameters using the Adam optimizer (LR = 1e-5), while employing early stopping (patience = 10 epochs) to prevent overfitting. This approach balances the preservation of pre-trained knowledge with task-specific adaptation, enabling efficient and effective fine-tuning for the regression task.

3.7 Evaluation Metrics

We used RMSE and R^2 score to evaluate our model, following the dataset authors' recommendation for RMSE. Since PyTorch lacks a native RMSE function, we implemented our own.

4 Results and Analysis

In our experiments, as seen in Table 1, we adapted MoLFormer to predict lipophilicity values using various data selection and fine-tuning strategies. In the baseline experiments (Task 1), training a

Task	Data selection	Fine-tuning	Epochs	R^2 Score	Average test RMSE
1	Default	-	37	0.7191	0.6325
1	Default	Regression Model	17	0.7327	0.6159
2	Influence scores	Regression Model	55	**0.7383**	**0.6137**
3	Error-based	iA3	74	0.5505	0.8056
3	Error-based	LoRA	49	0.6190	0.7297
3	Error-based	BitFit	125	0.6002	0.7579
3	Distribution-based	iA3	76	0.5340	0.8190
3	Distribution-based	LoRA	53	0.6439	0.7151
3	Distribution-based	BitFit	93	0.5789	0.7806
3	Clustering-based	iA3	41	0.4985	0.8487
3	Clustering-based	LoRA	42	0.6182	0.7411
3	Clustering-based	BitFit	87	0.5697	0.7921
3^*	Error-based	Regression Model	67	0.7265	0.6262
3^*	Distribution-based	Regression Model	39	0.7322	0.6202
3^*	Clustering-based	Regression Model	36	0.7180	0.6381

Table 1: Results for the various experiments performed in this project ($*$ experiments were done to have a direct comparison between all the data selection strategies)

regression head in MoLFormer with default data selection yielded a $\mathbf{R^2}$ score of 0.7191 and an average test RMSE of 0.6325. After performing unsupervised fine-tuning using the Masked Language Modeling (MLM) objective, and then fine-tuning our regression model, we observe a $\mathbf{R^2}$ score of 0.7327 and an RMSE of 0.6159, demonstrating that even simple fine-tuning can enhance regression performance.

Task 2 utilised influence scores with the LiSSA approximation for external data selection. This approach further enhanced performance, achieving an $\mathbf{R^2}$ score of 0.7383 and an RMSE of 0.6137. The improvement suggests that influence-based data selection helps identify more informative external samples, thereby refining the training set.

Task 3 explored three fine-tuning strategies—iA3, LoRA, and BitFit—across different data selection methods: error-based, distribution-based, and clustering-based.

Across all data selection methods, LoRA consistently outperformed the other strategies. The lower performance of iA3 might be due to its sensitivity to the quality of selected data or challenges in optimizing the scaling factors under the chosen scheduler. In contrast, LoRA's robust low-rank updates seem to provide a more stable adaptation, leading to better generalization. BitFit falls in between, offering moderate improvements but not matching LoRA's performance.

We can also observe that the datapoints selected by the error-based and distribution-based methods were a little better than the clustering-based method. This can be due to the unsupervised nature of the clustering approach and how true these clusters are with respect to the underlying data patterns.

5 Conclusions

In this work, we explored various data selection and fine-tuning strategies to enhance the prediction of lipophilicity using the MoLFormer model. Our experiments demonstrated that selecting high-quality external data—via influence scores, distribution matching, or clustering—combined with robust fine-tuning methods like LoRA, can significantly improve regression performance. While baseline and regression head approaches provided a solid starting point, advanced strategies further reduced RMSE and increased $\mathbf{R^2}$ scores.

These findings highlight the importance of both effective data selection and fine-tuning in adapting large pre-trained models for specialized tasks. Future research may further optimize these techniques or extend them to other molecular properties. Additionally, incorporating domain-specific knowledge of molecular lipophilicity for feature engineering could further enhance the robustness and predictive accuracy of the model.

References

Anshumali Agarwal et al. 2016. Second order optimization for deep learning: An overview. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Sanjivanjit K Bhal. 2007. Logp—making sense of the value. *Adv. Chem. Dev*, pages 1–4.

Balakumar Chandrasekaran, Sara Nidal Abed, Omar Al-Attraqchi, Kaushik Kuche, and Rakesh K. Tekade. 2018. Computer-aided prediction of pharmacokinetic (admet) properties. *Dosage Form Design Parameters*, pages 731–755.

David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. 1996. Active learning with statistical models. In *Proceedings of the Thirteenth International Conference on Machine Learning*.

Riya Datta, Dibyendu Das, and Srinjoy Das. 2021. Efficient lipophilicity prediction of molecules employing deep-learning models. *Chemometrics and Intelligent Laboratory Systems*, 213:104309.

Benedek Fabian, Thomas Edlich, Héléna Gaspar, Marwin Segler, Joshua Meyers, Marco Fiscato, and Mohamed Ahmed. 2020. Molecular representation learning with language models and domain-relevant auxiliary tasks. *arXiv preprint arXiv:2011.13230*.

Neil Houlsby, Alex Giurgiu, Stanislaw Jastrzebski, et al. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*.

John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. 2012. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768.

Clemens Isert, Jimmy C Kromann, Nikolaus Stiefl, Gisbert Schneider, and Richard A Lewis. 2023. Machine learning for fast, quantum mechanics-based approximation of drug lipophilicity. *ACS omega*, 8(2):2046–2056.

Qingqing Jia, Yifan Ni, Ziteng Liu, Xu Gu, Ziyi Cui, Mengting Fan, Qiang Zhu, Yi Wang, and Jing Ma. 2022. Fast prediction of lipophilicity of organofluorine molecules: deep learning-derived polarity characters and experimental tests. *Journal of Chemical Information and Modeling*, 62(20):4928–4936.

Ted W Johnson, Rebecca A Gallego, and Martin P Edwards. 2018. Lipophilic efficiency as an important metric in drug design. *Journal of medicinal chemistry*, 61(15):6401–6420.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521(7553):436–444.

Zhen Li, Mingjian Jiang, Shuang Wang, and Shugang Zhang. 2022. Deep learning methods for molecular representation and property prediction. *Drug Discovery Today*, 27(12):103373.

Haotian Liu, Barret Zoph, Jonathon Shlens, and Wei Hua. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *arXiv preprint arXiv:2205.05638*.

Zequn Liu, Wei Zhang, Yingce Xia, Lijun Wu, Shufang Xie, Tao Qin, Ming Zhang, and Tie-Yan Liu. 2023. Molxpt: Wrapping molecules with text for generative pre-training. *arXiv preprint arXiv:2305.10688*.

Nina Lukashina, Alisa Alenicheva, Elizaveta Vlasova, Artem Kondiukov, Aigul Khakimova, Emil Magerramov, Nikita Churikov, and Aleksei Shpilman. 2020. Lipophilicity prediction with multitask learning and molecular substructures representation. *arXiv preprint arXiv:2011.12117*.

Medard Edmund Mswahili and Young-Seob Jeong. 2024. Transformer-based models for chemical smiles representation: A comprehensive literature review. *Heliyon*.

David Rogers and Mathew Hahn. 2010. Extended-connectivity fingerprints. *Journal of chemical information and modeling*.

Paul Rognon-Vael, David Rossell, and Piotr Zwiernik. 2025. Improving variable selection properties by using external data. *arXiv preprint arXiv:2502.15584*.

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

Timon Schroeter, Anton Schwaighofer, Sebastian Mika, Antonius Ter Laak, Detlev Suelzle, Ursula Ganzer, Nikolaus Heinrich, and Klaus-Robert Müller. 2007. Machine learning models for lipophilicity and their domain of applicability. *Molecular pharmaceutics*, 4(4):524–538.

Burr Settles. 2009. Active learning literature survey. Technical Report CS-Technical Report 1648, University of Wisconsin–Madison.

Ketan Rajshekhar Shahapure and Charles Nicholas. 2020. Cluster quality analysis using silhouette score. In *2020 IEEE 7th international conference on data science and advanced analytics (DSAA)*, pages 747–748. IEEE.

Sukriti Singh and Raghavan B. Sunoj. 2023. Molecular machine learning for chemical catalysis: Prospects and challenges. *Accounts of Chemical Research*, 56(3):402–412.

Yanli Wang, Jewen Xiao, Tugba O Suzek, Jian Zhang, Jiyao Wang, and Stephen H Bryant. 2009. Pubchem: a public information system for analyzing bioactivities of small molecules. *Nucleic acids research*, 37(suppl_2):W623–W633.

Oliver Wieder, Mélaine Kuenemann, Marcus Wieder, Thomas Seidel, Christophe Meyer, Sharon D Bryant, and Thierry Langer. 2021. Improved lipophilicity and aqueous solubility prediction with composite graph neural networks. *Molecules*, page 6185.

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

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

Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. 2019. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388.

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. 2021. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.