How Well Do Large-Scale Chemical Language Models Transfer to Downstream Tasks?

Tatsuya Sagawa[1,2] and **Ryosuke Kojima**[2,3]
[1] Graduate School of Pharmaceutical Sciences, Kyoto University
[2] RIKEN BDR
[3] Graduate School of Medicine, Kyoto University
sagawa.tatsuya.82s@st.kyoto-u.ac.jp, kojima.ryosuke.8e@kyoto-u.ac.jp

Abstract

Chemical Language Models (CLMs) pre-trained on large-scale molecular data are widely used for molecular property prediction. However, the common belief that increasing training resources, such as model size, dataset size, and training compute, improves both pre-training loss and downstream task performance has not been systematically validated in the chemical domain. In this work, we evaluate this assumption by pre-training CLMs while scaling training resources and measuring transfer performance across diverse molecular property prediction (MPP) tasks. We find that while pre-training loss consistently decreases with increased training resources, downstream task performance shows limited improvement. Moreover, alternative metrics based on the Hessian or loss landscape also fail to estimate downstream performance in CLMs. We further identify conditions under which downstream performance saturates or degrades despite continued improvements in pre-training metrics, and analyze the underlying task-dependent failure modes through parameter space visualizations. These results expose a gap between pre-training-based evaluation and downstream performance, and emphasize the need for model selection and evaluation strategies that explicitly account for downstream task characteristics. The code is available at https://github.com/sagawatatsuya/MolScaleTransfer

1 Introduction

Chemical language models (CLMs), which are pre-trained on string representations of molecules, have become increasingly important as foundation models that can transfer to a wide range of molecular property prediction (MPP) tasks in biology, chemistry, and drug discovery [Park et al., 2024; Ross et al., 2022; Edwards et al., 2022]. This is largely because labeled data are expensive to obtain, while the rapid growth of molecular databases has made billions of unlabeled structures available. By representing molecules as sequences of atomic and bond symbols, we can treat them as a form of language. This formulation makes it natural to adopt techniques from natural language processing (NLP), and a variety of NLP methods have been adopted in this line of work [Chithrananda et al., 2020; Li and Jiang, 2021; Irwin et al., 2022]. Motivated by scaling practices in NLP, CLMs have also been rapidly scaled, following the expectation that increasing model size, data size, and compute lead to better performance [Soares et al., 2025a, 2025b; Cai et al., 2025]. In NLP, pre-training loss has been shown to follow a power law with respect to training resources, and this relationship has served as a practical guideline for model design and training resource allocation [Kaplan et al., 2020; Hoffmann et al., 2022].

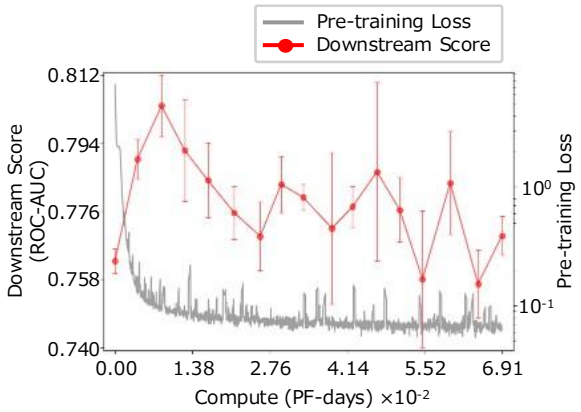

Figure 1: Training dynamics of the pre-training cross-entropy loss in a CLM and the downstream performance (ROC-AUC) obtained by fine-tuning checkpoints from different pre-training stages on the HIV benchmark.

In recent years, studies in NLP have reported that improvements in pre-training loss do not necessarily translate into better downstream performance and can even lead to negative transfer in some cases [Zoph et al., 2020; Isik et al., 2024; Lourie et al., 2025]. This observation suggests that the implicit assumption that minimizing the pre-training loss is equivalent to acquiring useful representations may not be universal. To address this issue, prior work has proposed alternatives to pre-training loss, including Hessian and loss landscape-based measures, and has shown that these metrics can serve as indicators of transfer performance on downstream NLP tasks [Liu et al., 2023]. However, whether negative transfers occur and whether such alternative metrics to pre-

training loss are expected to be effective depend on the downstream tasks and application domains. In particular, these questions have not been sufficiently examined in the context of CLMs.

Figure 1 illustrates that, for a typical Transformer-based CLM, lower pre-training loss does not necessarily translate into better downstream performance. As pre-training progresses, the loss decreases monotonically, but downstream performance after fine-tuning model states from different stages on the HIV benchmark [Wu et al., 2018] is non-monotonic. This suggests that optimizing the pre-training objective can diverge from learning representations that transfer well. It also indicates that common CLM practices, such as using pre-training loss as the primary criterion for early stopping, model selection, and training resource allocation, may not be fully justified from the perspective of downstream applications.

Motivated by this observation, we investigate (i) how well pre-training loss correlates with downstream performance on MPP tasks, and (ii) how this relationship changes as we scale model size, data size, and training compute. To address these questions, we focus on encoder-based CLMs that are commonly used for MPP, pre-train them under different resource budgets and evaluate transfer on MPP benchmarks. We then evaluate downstream performance under practical transfer settings that are widely used in CLM applications, including fine-tuning and linear probe, to identify when improvements in pre-training align with downstream gains versus when they diverge. We further analyze task-dependent factors underlying this phenomenon in a comprehensive evaluation across 36 MPP tasks.

Our contributions are summarized as follows:

- We show that scaling model size, data size, and training compute consistently reduces pre-training loss, suggesting scaling-law behavior for CLMs at least with respect to loss.
- We systematically evaluate the relationship between pre-training loss and downstream performance on MPP benchmarks and demonstrate that lower loss does not reliably imply better downstream performance in CLMs.
- We show that, under standard CLM training and transfer settings, the relationship between pre-training loss and downstream performance depends strongly on the task and the training setup, which establishes the limitations of using pre-training loss as a single criterion for early stopping, model selection, and resource allocation. These limitations persist even when using alternatives to pre-training loss based on Hessian information and loss landscapes. Finally, we analyze this behavior using visualizations in parameter space.

2 Related Work

2.1 Chemical Language Models for Molecular Property Prediction

Chemical language models (CLMs) are typically pre-trained on large-scale unlabeled corpora of molecular strings and use representations such as SMILES (Simplified Molecular-Input Line-Entry System) as input [Weininger, 1988; Park et al., 2024; Ross et al., 2022; Yüksel et al., 2023]. This allows molecules to be treated as text. Accordingly, CLMs are commonly pre-trained with standard NLP objectives, including masked language modeling (MLM), which predicts masked tokens, and autoregressive (next-token) language modeling. The same framework is widely used for downstream molecular property prediction (MPP) tasks, covering a broad range of classification and regression problems.

Input representations for CLMs have been studied extensively. Beyond SMILES, alternative notations such as Self-Referencing Embedded Strings (SELFIES) [Krenn et al., 2020] and DeepSMILES [O'Boyle and Dalke, 2018] have been developed and have also been used in CLM pre-training and modeling [Yüksel et al., 2023]. However, prior work reports that the choice of notation has little impact on downstream performance for large-scale CLMs [Rajan et al., 2026]. In light of this observation, our scaling study focuses on SMILES, which remains widely used in large-scale CLMs.

2.2 Scaling Laws in Chemical Language Models

Scaling laws have been observed across domains such as NLP, computer vision, and speech recognition, where performance improves predictably as model size, data size, and training compute increase [Zhai et al., 2021; Chen et al., 2025; Kaplan et al., 2020]. Similar observations have been reported in the chemistry domain. For example, prior work shows that pre-training loss in decoder-based CLMs follows a power law with respect to model size and data size [Ramos et al., 2025; Frey et al., 2023]. For encoder-based models, which are often pre-trained and then fine-tuned for MPP tasks, existing studies have typically examined pre-training or downstream performance in isolation [Chen et al., 2023]. As a result, how strongly improvements in pre-training loss translate into downstream gains remains unclear.

Scaling-law studies also extend beyond string-based CLMs. In graph-based molecular representation learning, performance has been reported to scale consistently with downstream dataset size [Chen et al., 2023]. Another line of work studies models that incorporate atomic coordinates and reports monotonic improvements on certain quantum chemistry benchmarks as model size increases [Wang et al., 2024]. While these approaches go beyond simple string representations of molecules, combining their insights with our comprehensive downstream-task results on standard string representation may help clarify the mechanisms underlying these scaling phenomena.

2.3 When Pre-training Loss Fails to Reflect Transferability

In research on pre-trained models, the empirical rule that lower pre-training loss implies better downstream performance has guided practical choices such as model selection, early stopping, and resource allocation [Kaplan et al., 2020]. Evidence from large language models (LLMs) has further reinforced this practice. As scale increases, loss often decreases alongside improvements in downstream performance, which

encourages the use of loss as a proxy for downstream performance.

However, recent studies have shown that improvements in pre-training loss do not necessarily yield downstream gains and can even lead to negative transfer [Zoph et al., 2020; He et al., 2019]. For example, under substantial distribution shift between pre-training and downstream data, downstream performance may not scale predictably with increasing amounts of pre-training data [Isik et al., 2024]. In addition, a meta-analysis of LLMs found that the conditions under which downstream scaling laws hold are limited and that scaling behavior can fail to generalize even under small changes in experimental settings [Lourie et al., 2025]. These findings suggest that making pre-training-stage decisions based solely on loss can be unreliable in some settings. To better anticipate downstream performance beyond what is captured by loss, prior work has proposed alternatives to pre-training loss, such as metrics derived from the loss-landscape curvature [Liu et al., 2023] and gradient-based measures that quantify similarity between parameter vectors before and after learning [Yao et al., 2024].

Negative transfer has also been reported in molecular modeling. Prior work on CLMs shows that, in certain tasks and data regimes, pre-training contributes little to downstream performance and can even degrade transfer [Chen et al., 2023; Hormazabal et al., 2024]. Studies of graph-based molecular representations likewise report that pre-training can induce negative transfer, highlighting the importance of pre-training task design and data conditions [Xia et al., 2022]. In such cases, improvements in pre-training loss do not necessarily indicate that the model has learned representations useful for downstream tasks. Nevertheless, pre-training loss remains a widely used metric for monitoring pre-training progress and selecting models. This may be partly because much of the literature examines pre-training and downstream performance in isolation and reports results in relatively specific settings. As a result, comprehensive evaluations that systematically vary model size, data size, and compute under a consistent pre-training protocol are still limited. Moreover, alternatives to pre-training loss that have shown promise in NLP have not yet been systematically examined for CLMs.

3 Method

We systematically vary model size, data size, and training compute under a controlled pre-training protocol to examine scaling behavior and downstream transfer. We first describe the pre-training and downstream evaluation procedures, and then define metrics quantifying downstream transfer performance.

3.1 Pre-training and Downstream Evaluation

Pre-training. We pre-train CLMs with MLM on a large-scale corpus of SMILES strings. We randomly mask a subset of tokens and train the model to predict the masked tokens. We adopt a Transformer encoder architecture [Vaswani et al., 2017] (see Appendix A for details). The pre-training objective is the MLM cross-entropy loss, which we track throughout training.

Downstream tasks. We evaluate downstream transfer using two standard protocols: fine-tuning and linear probe. For fine-tuning, we attach a task-specific prediction head to the pre-trained encoder and update both the encoder and head to optimize the downstream objective. This setting reflects performance after task-specific adaptation of the pre-trained representations. For linear probe, we freeze the encoder and train only a linear head on top of the encoder representations, measuring the linear separability of the learned representations. For each benchmark, we use task-specific losses and evaluation metrics (Appendix B).

3.2 Downstream Transferability

We quantitatively evaluate how well pre-training loss correlates with downstream performance. We compare model states saved at different stages of pre-training and test whether pre-training loss exhibits a monotonic relationship with downstream metrics via correlation analysis.

We periodically save the model during pre-training and refer to each saved snapshot as a checkpoint.

For each checkpoint, we compute the following values:

- L_{pre}: the MLM cross-entropy loss on the validation split of the pre-training dataset
- L_{down}: the MLM cross-entropy loss computed on downstream inputs without any parameter updates (i.e. zero-shot evaluation)
- P^{FT}: downstream performance after fine-tuning
- P^{LP}: downstream performance under linear probe

We define consistency as the monotonic association between L_{pre} and downstream metrics and quantify it using Spearman's rank correlation coefficient [Spearman, 1904], denoted by ρ. Specifically, we compute ρ between L_{pre} and each of L_{down}, P^{FT}, and P^{LP} across checkpoints. To keep the direction consistent across tasks, we transform downstream metrics only for correlation computation so that smaller values indicate better performance; for classification tasks, we use $1 - \mathrm{ROC\text{-}AUC}$.

3.3 Alternative Metrics to Pre-training Loss

In addition to the loss-based analysis in Section 3.2, we consider two alternatives to pre-training loss as predictors of downstream performance: the trace of the Hessian ($\mathrm{Tr}(\boldsymbol{H})$) [Liu et al., 2023] and the Principal Gradient-based Measurement (PGM) distance [Yao et al., 2024]. $\mathrm{Tr}(\boldsymbol{H})$ captures the curvature of the pre-training loss landscape, while the PGM distance assesses transferability via the misalignment between gradients induced by the pre-training objective and those induced by downstream objective. We expect smaller values of these metrics to correspond to better downstream performance. Definitions and implementation details are provided in Appendix C.

For each training run with fixed model size and training data size, we compute these metrics on checkpoints saved at regular intervals. We quantify the relationship between metric values and downstream performance using Spearman's rank correlation.

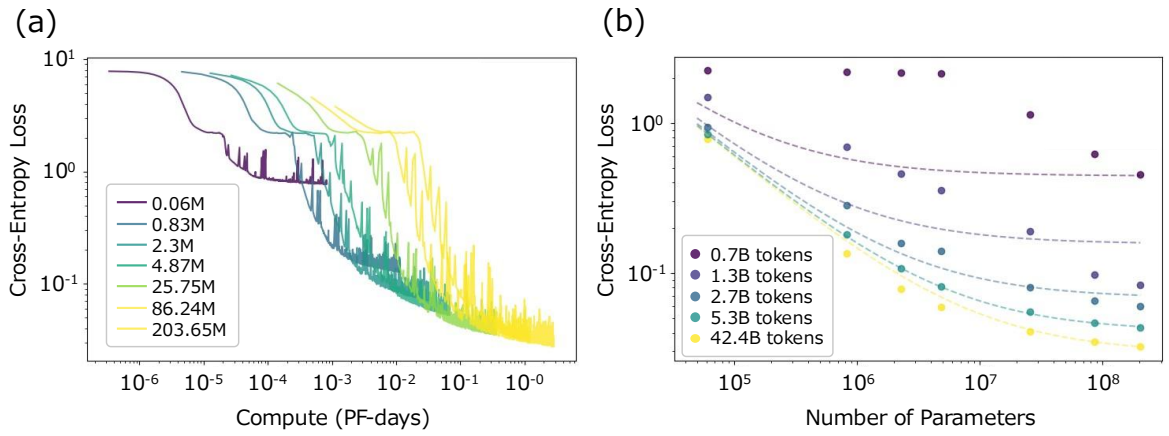

Figure 2: Pre-training scaling laws in pre-training. (a) Pre-training loss as a function of training compute for different model sizes. (b) Final pre-training loss as a function of model size and data size, with a power-law fit (exponents $\alpha = 0.686, \beta = 1.702$).

4 Experiments

We examine how scaling affects pre-training loss and downstream transfer performance. Section 4.1 confirms that previously reported pre-training scaling laws hold in our setting. Section 4.2 evaluates models pre-trained under different scaling regimes on downstream tasks and quantifies the relationship between pre-training loss and downstream performance using rank correlation and identifies when loss improvements translate into downstream gains. Finally, Section 4.3 evaluates alternatives to pre-training loss in settings where loss alone fails to predict downstream outcomes and analyzes task-dependent failure modes when using loss as a proxy for transfer.

4.1 Pre-training Evaluation

We evaluate how pre-training loss changes as we scale model size, data size, and compute. Our models are Transformer encoders with from 0.06 to 203.65 million parameters excluding embeddings (Appendix A), covering a range comparable to prior encoder-based CLMs [Ross et al., 2022; Park et al., 2024; Yüksel et al., 2023]. We pre-train on a large-scale dataset of SMILES strings compiled from ZINC-15 [Sterling and Irwin, 2015] and PubChem [Kim et al., 2019]. To assess the effect of data size, we subsample the training data to obtain subsets ranging from 0.7 to 42.4 billion tokens.

Figure 2(a) shows learning curves for seven model sizes trained for four epochs with the data size fixed at the largest subset. For all models, pre-training loss decreases throughout training with diminishing returns later in training, and larger models consistently achieve lower loss.

Figure 2(b) shows the final pre-training loss while varying both model size and training data size. Pre-training loss decreases monotonically as model size and data size increase, consistent with scaling-law behavior over our experimental range (Appendix D).

4.2 Downstream Transferability Evaluation

We evaluate how improvements during pre-training relate to downstream gains along three axes: model size, data size, and compute. For each axis, we vary only that factor while holding the others fixed. We then plot L_{pre}, L_{down}, P^{FT}, and P^{LP} across the resulting checkpoints (Figure 3) and quantify the monotonic relationship between L_{pre} and downstream metrics using Spearman's rank correlation (Figure 4).

For downstream evaluation, we use MoleculeNet [Wu et al., 2018], a widely used benchmark suite for MPP. We evaluate 36 tasks in total: seven classification tasks, five non-quantum regression tasks, and 24 quantum-chemistry regression tasks. Unless otherwise noted, P^{FT} and P^{LP} denote performance averaged over each task group. We use ROC-AUC for classification, MAE for quantum-chemistry regression tasks, and RMSE for non-quantum regression tasks (see Appendix B for details).

To disentangle the contributions of training resources to downstream performance, we conduct three scaling experiments. In the model-size experiment, we fix the training data size at 42.4 billion tokens and train for four epochs, comparing seven models with different numbers of parameters. In the data-size experiment, we fix the model at 4.87 million parameters and train for four epochs across five data-size settings ranging from 0.7 to 42.4 billion tokens. In the compute experiment, we fix the model (4.87M parameters) and training data size (42.4 billion tokens) and save checkpoints every 0.25 epoch over the 4-epoch training and compare 17 checkpoints in total, including the 0-step initialization. We measure compute in PF-days [Kaplan et al., 2020; Appendix E].

Figure 3 shows that L_{pre} decreases consistently as we scale model size, data size, or compute. We also observe the same monotonic decrease for L_{down}, the MLM loss computed on downstream inputs. In other words, despite distributional differences between pre-training and downstream data, improvements under the pre-training objective carry over to downstream inputs, consistent with prior findings [Frey et al., 2023; Isik et al., 2024].

In contrast, downstream performance shows qualitatively different trends depending on which factor is scaled. Scaling model size tends to improve both P^{FT} and P^{LP} (Figure 3(a)–

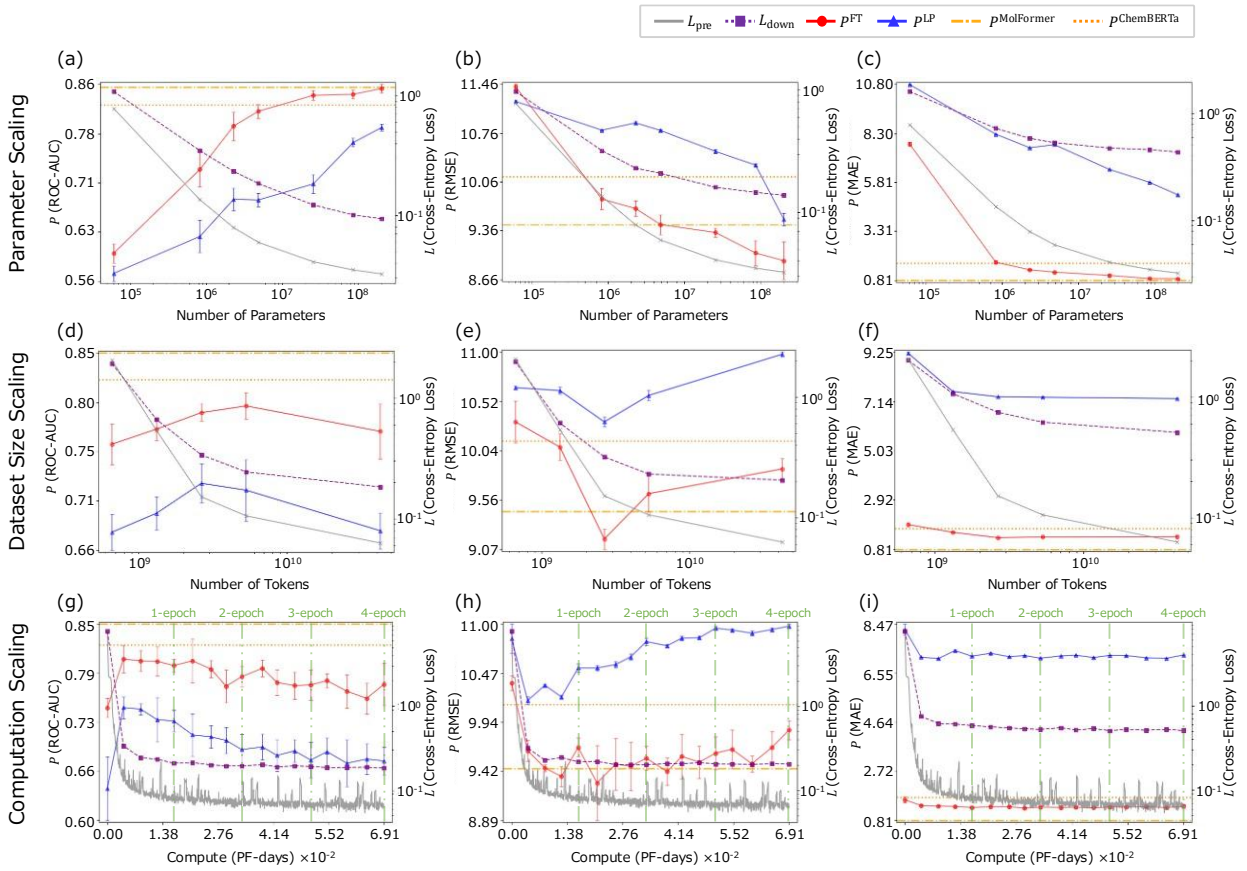

Figure 3: Trends of L_{pre}, L_{down}, P^{FT}, and P^{LP} under (a–c) model size scaling, (d–f) data size scaling, and (g–i) compute scaling. Panels correspond to task groups: (a,d,g) classification, (b,e,h) non-quantum regression, and (c,f,i) quantum-chemistry regression (averaged over tasks). In (g–i), dashed vertical lines mark epoch boundaries during pre-training. For reference, we include fine-tuned performance of MolFormer-XL ($P^{\text{MolFormer}}$) [Ross et al., 2022] and ChemBERTa ($P^{\text{ChemBERTa}}$) [Chithrananda et al., 2020].

(c)). When scaling data size, increasing the number of training tokens from 0.7B to 2.7B typically improves downstream performance; however, further scaling to 5.3B led to diminishing returns and performance degradation (Figure 3(d)–(f)).

Comparing P^{FT} and P^{LP}, the benefits of scaling training resources are generally more pronounced under fine-tuning. In particular, as compute increases, P^{FT} saturates early, while P^{LP} tends to deteriorate as pre-training proceeds (Figure 3(g)–(i)). In this setting, downstream performance reaches its peaks before completing one epoch (1.73×10^{-2} PF-days), i.e., before the model completes a full pass through the training data. Although CLMs are often pre-trained for multiple epochs [Ross et al., 2022; Frey et al., 2023], these results suggest that longer pre-training can further reduce pre-training loss while potentially causing over-optimizing for the pre-training objective, leading to worse transfer.

As shown in Figure 4, we compute Spearman's rank correlation between L_{pre} and downstream metrics P^{FT} and P^{LP} for each task, as described in Section 3.2, after transforming metrics, when necessary, so that smaller values indicate better performance. The spread and sign of the correlations depend on which factor is scaled. When scaling model size, points lie mostly in the first quadrant, indicating that

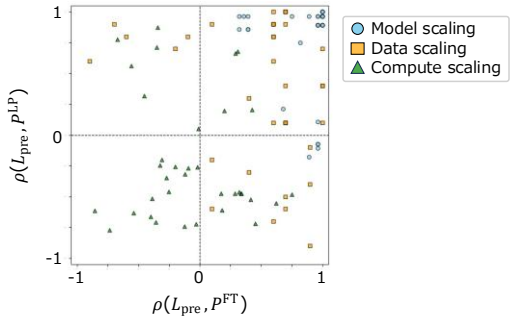

Figure 4: Consistency evaluation using Spearman's rank correlation. When scaling model size, data size, or compute in isolation, the x-axis shows $\rho(L_{\text{pre}}, P^{\text{FT}})$ and the y-axis shows $\rho(L_{\text{pre}}, P^{\text{LP}})$. Each point corresponds to a task.

$\rho(L_{\text{pre}}, P^{\text{FT}})$ and $\rho(L_{\text{pre}}, P^{\text{LP}})$ are typically positive. When scaling data size, points are concentrated in the first quadrant, but several also appear in the second and fourth quadrants. When scaling compute, points are more broadly dispersed, and 26 out of 36 tasks have negative $\rho(L_{\text{pre}}, P^{\text{LP}})$.

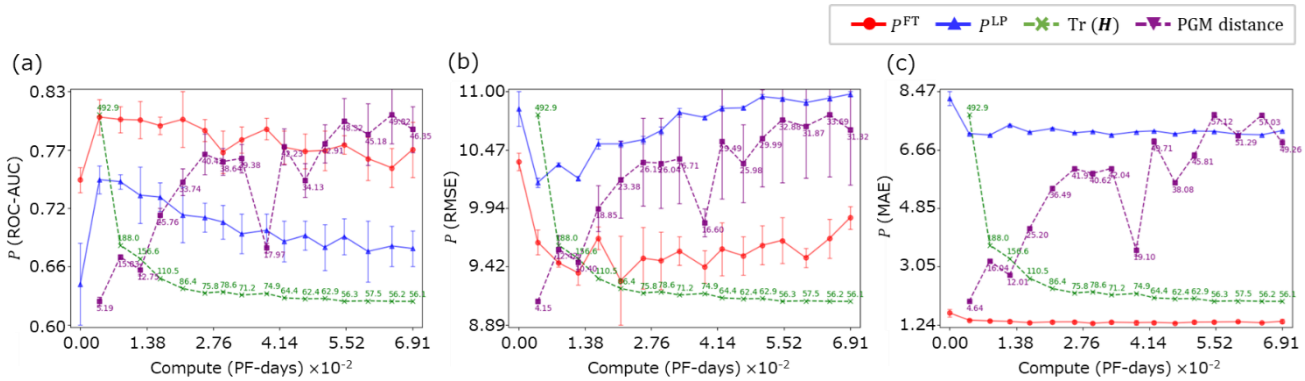

Figure 5: Trajectories of metric values and downstream performance as a function of pre-training compute. The x-axis indicates pre-training progress measured in compute. Subplots correspond to task groups: (a) classification, (b) non-quantum regression, and (c) quantum-chemistry regression. For each group, we plot the task-averaged downstream performance under fine-tuning and linear probe (P^{FT} and P^{LP}) and overlay the task-averaged PGM distance. We also overlay Tr($\boldsymbol{H}$), which is task-agnostic and thus shared across task groups.

Combined with the results in Section 4.1, these findings show that, under commonly used CLM training setups, pre-training loss decreases as we scale model size, data size, or compute, whereas downstream performance varies substantially across tasks and scaling settings. This implies that pre-training loss alone is an unreliable indicator for consistently predicting downstream gains. More broadly, our results suggest that common CLM practices such as early stopping and checkpoint selection based solely on pre-training loss, as well as resource allocation guided by loss-based scaling laws, may fail to deliver consistent downstream improvements.

4.3 Transferability Evaluation Using Alternatives to Pre-training Loss

Following the evaluation protocol in Section 3.3, we examine how two metrics beyond pre-training loss, Tr($\boldsymbol{H}$) and the PGM distance, relate to downstream metrics (P^{FT} and P^{LP}). We compute these metrics on checkpoints saved every 0.25 epoch during pre-training and report their trajectories alongside downstream performance in Figure 5 (see Appendix I for per-task results). Note that Tr($\boldsymbol{H}$) is task-agnostic and yields a single value per checkpoint. As a result, when evaluated on the same checkpoint sequence, it is not expected to capture task-specific differences in downstream performance trajectories. We further report per-task Spearman correlations between each metric and downstream performance (Figure 6).

In Figure 5, Tr($\boldsymbol{H}$) decreases approximately monotonically as pre-training proceeds, whereas downstream performance shows saturation and fluctuations, highlighting intervals where the two curves are not well aligned. This is reflected in Figure 6, where the Spearman correlation between Tr($\boldsymbol{H}$) and P^{FT} varies from negative to positive across tasks. For P^{LP}, we also observe negative correlations, which may indicate that Tr($\boldsymbol{H}$) does not capture the late-stage decline in P^{LP}.

Taken together, these results indicate that although Tr($\boldsymbol{H}$) and the PGM distance change as pre-training proceeds, neither provides a reliable predictor of downstream performance. Tr($\boldsymbol{H}$) primarily reflects progress in optimizing the pre-training objective and decreases in tandem with L_{pre} and L_{down}.

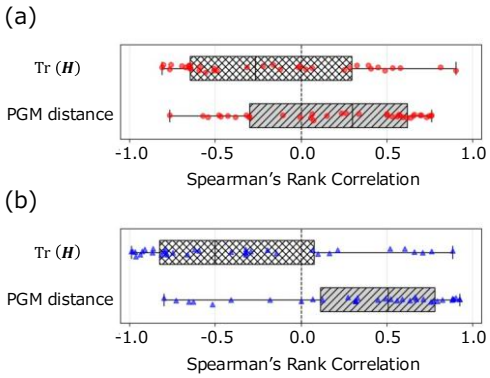

Figure 6: Per-task relationships between metrics computed during pre-training and downstream performance. Spearman correlations between Tr($\boldsymbol{H}$) and the PGM distance and (a) P^{FT} and (b) P^{LP}.

However, its correspondence to task-dependent performance metrics such as P^{FT} and P^{LP} is limited. In contrast, the PGM distance shows a weak correlation with P^{LP}, while its trajectory is similar across tasks, suggesting limited sensitivity to task-specific variation in downstream performance.

Overall, our results suggest that relying on a single metric beyond pre-training loss may be insufficient for predicting downstream performance or selecting checkpoints. Accurately estimating downstream performance may instead require lightweight task-aware evaluation, such as periodically evaluating candidate checkpoints or scaling configurations on a small validation set from the target downstream task. Although such heuristic workarounds by task-aware evaluation require access to target-task validation data in advance, this approach offers a practical way to select model size, dataset size, and compute budget.

4.4 Visualization of Parameter Space

In this section, we visualize the parameter space to gain insights into why the PGM distance does not reliably predict P^{FT} and why the consistency between pre-training loss and downstream performance varies by task. We consider 16

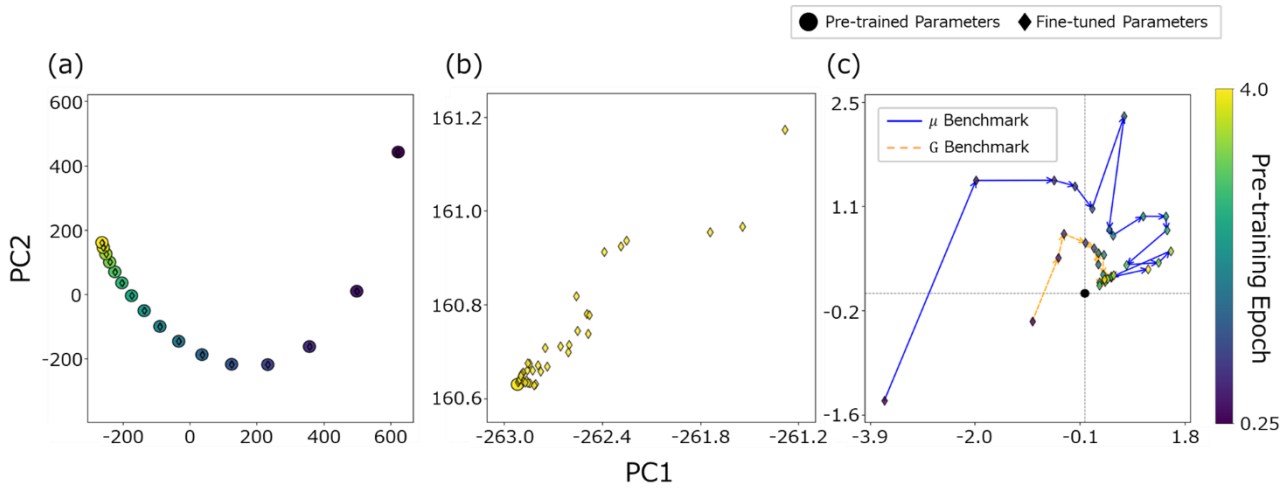

Figure 7: Visualization of the parameter space. (a) Sixteen pre-training checkpoints and the models obtained by fine-tuning each checkpoint separately on each of the 36 downstream tasks are plotted in the same 2D PCA space. (b) A zoomed-in view focusing on the checkpoint after 4 epochs of pre-training and the corresponding fine-tuned models across tasks. (c) Visualization in relative coordinates, obtained by translating each fine-tuned model's point in the projected space so that its initialization checkpoint is mapped to the origin.

checkpoints saved at 0.25-epoch intervals over a 4-epoch pre-training run (excluding the 0-step checkpoint). For each checkpoint, we fine-tune separate models for each of the 36 downstream tasks, resulting in task-specific models. For both the pre-trained checkpoints and the fine-tuned models, we extract the parameters from the final Transformer block and project them into two dimensions using principal component analysis (PCA) (Figure 7(a)). Hereafter, we refer to the pre-training checkpoint used to initialize fine-tuning as the initialization checkpoint. We focus on the final Transformer block because layers closer to the prediction head tend to exhibit more task-specific changes [Mosbach et al., 2020]. We further focus on the checkpoint after 4 epochs and the corresponding fine-tuned models to examine the parameter changes in more detail (Figure 7(b)).

Figures 7(a) and 7(b) show that the fine-tuned models remain close to their initialization checkpoints in the projected space. The PGM distance is motivated by the assumption that the pre-training update direction is indicative of a global direction toward a good downstream optimum. However, the visualization suggests that fine-tuning updates remain largely local to each initialization checkpoint, and that the fine-tuned solutions vary substantially across initializations and tasks. As a result, the assumption underlying the PGM distance may not hold in our setting, which makes it difficult to predict downstream performance consistently using a single metric.

Next, to understand the task dependence of the correspondence between loss and downstream performance, we visualize changes induced by fine-tuning as displacements from the initialization checkpoint. As in Figures 7(a) and 7(b), we use the parameters of the final Transformer block and first project the corresponding parameter vectors into a two-dimensional space using PCA. We then compute the displacement of each fine-tuned model from its corresponding initialization checkpoint by translating coordinates so that the initialization checkpoint is mapped at the origin (Figure 7(c)). As a concrete example, we compare two QM9 tasks, G and μ, which exhibit high and low correlations ($\rho = 0.78$ and $\rho = 0.07$), respectively.

As shown in Figure 7(c), G exhibits smaller displacements than μ, indicating that the fine-tuned models remain closer to their initialization checkpoints in the projected space. Although the fine-tuned models for both tasks show decreasing displacement as pre-training proceeds, G remains closer to its initialization checkpoint than μ at the final stage. We observe qualitatively similar trends across other downstream tasks, although the magnitude varies by task (see Appendix J for details).

5 Conclusions

In this study, we systematically examined whether the conventional wisdom that scaling training resources improves performance also holds for downstream transfer performance in chemical language models (CLMs). By independently varying model size, data size, and compute within practical ranges and evaluating on multiple molecular property prediction (MPP) tasks, we found that although pre-training loss consistently decreases as training resources increase, these loss reductions do not reliably translate into downstream gains; in particular, scaling data size and compute often yields limited benefits and can even induce negative transfer under certain conditions. We further showed that neither pre-training loss nor alternative proxy metrics reliably predict downstream performance, implying that common CLM practices such as early stopping, model selection, and compute budgeting based solely on pre-training loss may be suboptimal for downstream transfer. Overall, our findings highlight the need to rethink evaluation metrics and development guidelines for scaling CLMs as foundation models with downstream transfer in mind, and suggest that task-aware evaluation that accounts for task-specffific characteristics of downstream chemical tasks is essential for more efficient and practical foundation-model design.

Acknowledgments

This work was supported by JST Moonshot R&D (No. JPMJMS2024), JSPS KAKENHI (Nos. JP21H05207, JP21H05221), RIKEN TRIP initiative (AGIS) and CREST (No. JPMJCR22D3).

References

[Cai et al., 2025] Feiyang Cai, Katelin Zacour, Tianyu Zhu, Tzuen-Rong Tzeng, Yongping Duan, Ling Liu, Srikanth Pilla, Gang Li, and Feng Luo. ChemFM as a scaling law guided foundation model pre-trained on informative chemicals. Communications Chemistry, 2025.

[Chen et al., 2023] Dingshuo Chen, Yanqiao Zhu, Jieyu Zhang, Yuanqi Du, Zhixun Li, Qiang Liu, Shu Wu, and Liang Wang. Uncovering Neural Scaling Laws in Molecular Representation Learning. In Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track. 2023.

[Chen et al., 2025] William Chen, Jinchuan Tian, Yifan Peng, Brian Yan, Chao-Han Huck Yang, and Shinji Watanabe. OWLS: Scaling laws for multilingual speech recognition and translation models. arXiv [cs.CL], 2025.

[Chithrananda et al., 2020] Seyone Chithrananda, Gabe Grand, and Bharath Ramsundar. ChemBERTa: Large-scale self-supervised pretraining for molecular property prediction. arXiv [cs.LG], 2020.

[Edwards et al., 2022] Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2022.

[Frey et al., 2023] Nathan C. Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gómez-Bombarelli, Connor W. Coley, and Vijay Gadepally. Neural scaling of deep chemical models. Nature machine intelligence, 5(11), 1297–1305, 2023.

[He et al., 2019] Kaiming He, Ross Girshick, and Piotr Dollar. Rethinking ImageNet Pre-Training. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 4918–4927. 2019.

[Hoffmann et al., 2022] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models. arXiv [cs.CL], 2022.

[Hormazabal et al., 2024] Rodrigo Hormazabal, Seung Woo Ko, Inwan Yoo, Sehui Han, and Paul Bertens. Exploring Neural Scaling Laws in Molecular Pretraining with Synthetic Tasks. In ICML 2024 AI for Science Workshop. 2024.

[Irwin et al., 2022] Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. Machine learning: science and technology, 3(1), 015022, 2022.

[Isik et al., 2024] Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. Scaling Laws for Downstream Task Performance in Machine Translation. In The Thirteenth International Conference on Learning Representations. 2024.

[Kaplan et al., 2020] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. arXiv [cs.LG], 2020.

[Kim et al., 2019] 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. PubChem 2019 update: improved access to chemical data. Nucleic Acids Research, 47(D1), D1102–D1109, 2019.

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

[Li and Jiang, 2021] Juncai Li, and Xiaofei Jiang. Mol-BERT: An effective molecular representation with BERT for molecular property prediction. Wireless Communications and Mobile Computing, 2021(1), 1–7, 2021.

[Liu et al., 2023] Hong Liu, Sang Michael Xie, Zhiyuan Li, and Tengyu Ma. Same Pre-training Loss, Better Downstream: Implicit Bias Matters for Language Models. In Proceedings of the 40th International Conference on Machine Learning, 22188–22214. PMLR, 23--29 Jul 2023.

[Lourie et al., 2025] Nicholas Lourie, Michael Y. Hu, and Kyunghyun Cho. Scaling laws are unreliable for downstream tasks: A reality check. arXiv [cs.CL], 2025.

[Mosbach et al., 2020] Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. On the stability of fine-tuning BERT: Misconceptions, explanations, and strong baselines. arXiv [cs.LG], 2020.

[O'Boyle and Dalke, 2018] Noel O'Boyle, and Andrew Dalke. DeepSMILES: An adaptation of SMILES for use in machine-learning of chemical structures. ChemRxiv, 2018.

[Park et al., 2024] Jun-Hyung Park, Yeachan Kim, Mingyu Lee, Hyuntae Park, and Sangkeun Lee. MolTRES: Improving chemical language representation learning for molecular property prediction. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 14241–14254. Association for Computational Linguistics, 2024.

[Rajan et al., 2026] Kohulan Rajan, Achim Zielesny, and Christoph Steinbeck. Comparative analysis of chemical structure string representations for neural machine translation. Lecture Notes in Computer Science, Springer Nature Switzerland, 2026.

[Ramos et al., 2025] Mayk Caldas Ramos, Christopher J. Collison, and Andrew D. White. A review of large language models and autonomous agents in chemistry. Chemical Science (Royal Society of Chemistry: 2010), 16(6), 2514–2572, 2025.

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

[Soares et al., 2025a] Eduardo Soares, Emilio Vital Brazil, Victor Shirasuna, Dmitry Zubarev, Renato Cerqueira, and Kristin Schmidt. A Mamba-based foundation model for materials. NPJ Artificial Intelligence, 1(1), 2025.

[Soares et al., 2025b] Eduardo Soares, Emilio Vital Brazil, Victor Shirasuna, Dmitry Zubarev, Renato Cerqueira, and Kristin Schmidt. An open-source family of large encoder-decoder foundation models for chemistry. Communications Chemistry, 8(1), 193, 2025.

[Spearman, 1904] Charles Spearman. The proof and measurement of association between two things. The American Journal of Psychology, 15(1), 72–101, 1904.

[Sterling and Irwin, 2015] Teague Sterling, and John J. Irwin. ZINC 15--ligand discovery for everyone. Journal of Chemical Information and Modeling, 55(11), 2324–2337, 2015.

[Vaswani et al., 2017] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In Advances in Neural Information Processing Systems 30 (NeurIPS 2017), 5998–6008, 2017.

[Wang et al., 2024] Zhen Wang, Zhifeng Gao, Hang Zheng, Linfeng Zhang, Guolin Ke, and Others. Exploring molecular pretraining model at scale. Advances in neural information processing systems, 37, 46956–46978, 2024.

[Weininger, 1988] David Weininger. 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, 1988.

[Wu et al., 2018] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: a benchmark for molecular machine learning. Chemical science (Royal Society of Chemistry: 2010), 9(2), 513–530, 2018.

[Xia et al., 2022] Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z. Li. Mole-BERT: Rethinking Pre-training Graph Neural Networks for Molecules. In The Eleventh International Conference on Learning Representations. 2022.

[Yao et al., 2024] Shaolun Yao, Jie Song, Lingxiang Jia, Lechao Cheng, Zipeng Zhong, Mingli Song, and Zunlei Feng. Fast and effective molecular property prediction with transferability map. Communications chemistry, 7(1), 85, 2024.

[Yüksel et al., 2023] Atakan Yüksel, Erva Ulusoy, Atabey Ünlü, and Tunca Doğan. SELFormer: molecular representation learning via SELFIES language models. Machine learning: science and technology, 4(2), 025035, 2023.

[Zhai et al., 2021] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling Vision Transformers. arXiv [cs.CV], 2021.

[Zoph et al., 2020] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking Pre-training and Self-training. Advances in Neural Information Processing Systems, 33, 3833–3845, 2020.