Self-Distillation for Continual Learning in Masked Language Models

Ali Saadat [1] **Jacques Fellay** [1]

Abstract

Masked language modeling is a widely used pretraining objective for learning reusable representation backbones. In practical settings, models pretrained with masked language modeling often need to be adapted sequentially to new supervised tasks without losing performance on previously learned ones. We introduce SDFT-MLM, a self-distillation fine-tuning method for continual adaptation of such pretrained models. SDFT-MLM combines a supervised task loss with a token-level distillation penalty from an exponential-moving-average teacher. As a case study, we apply SDFT-MLM to protein language models, where new experimental assays often become available over time and models must learn new protein property predictors while retaining performance on earlier tasks. We evaluate continual protein property regression across four tasks using ESM2 as the encoder. To assess robustness to task order, we fine-tune the model under three dataset orderings. Across all orderings, SDFT-MLM improves final test Spearman correlation on earlier tasks, which are most susceptible to forgetting, while maintaining performance on later tasks comparable to standard supervised fine-tuning. These results suggest that token-level self-distillation provides a simple and effective mechanism for improving retention during sequential adaptation of models pretrained with masked language modeling.

1. Introduction

Masked language modeling is a pretraining technique in which a model learns contextual representations by reconstructing masked tokens from their surrounding context (Devlin et al., 2019). Models pretrained with this objective have become effective reusable backbones across many domains, including natural language and biological sequences (Gardazi et al., 2025; Wang et al., 2025; Consens et al., 2025). After pretraining, these models are commonly adapted to downstream tasks through supervised fine-tuning (Howard & Ruder, 2018; Saadat & Fellay, 2025; Schmirler et al., 2024). In many practical settings, however, adaptation is not a one-time procedure. New datasets, labels, or objectives may arrive sequentially, and a pretrained model must learn from new supervision while preserving representations that remain useful for tasks encountered earlier (Delange et al., 2021; Wang et al., 2024).

A common response to each new task is either to retrain the model from scratch or to fine-tune it only on the newest dataset. Retraining after every update is computationally expensive and operationally impractical, while naive fine-tuning on the latest task can overwrite representations needed for previous tasks. This tension makes continual adaptation important for building models that accumulate task-specific knowledge over time rather than treating each new task as an isolated fine-tuning problem (Patil et al., 2026).

We investigate whether self-distillation can improve retention during sequential adaptation of models pretrained with masked language modeling. Motivated by Self-Distillation Fine-Tuning (SDFT) (Shenfeld et al., 2026), originally proposed for autoregressive language models, we adapt the underlying principle to masked language models. We propose SDFT-MLM, a continual fine-tuning objective that regularizes supervised adaptation with token-level self-distillation. During sequential training, an exponential-moving-average teacher tracks the student model. The student is optimized for the current supervised task while being encouraged to preserve the teacher's output distribution on the same inputs. This objective allows the model to adapt to new labels while discouraging abrupt changes to token-level predictions induced by pretraining and prior tasks.

We study this setting using protein language models as a concrete case study (Lin et al., 2023). Protein modeling often involves multiple experimental measurements of the same or related proteins such as fluorescence, Stability, binding, activity, and localization which are relevant for protein engineering, functional annotation, disease studies, and biological discovery, and are often collected at different times or for different protein subsets (Schmirler et al., 2024;

[1]School of Life Sciences, Ecole Polytechnique Fédérale de Lausanne, Switzerland. Correspondence to: Jacques Fellay <jacques.fellay@epfl.ch>.

Accepted as a workshop paper at ICML 2026.

Patil et al., 2026). This makes protein property regression a natural instance of sequential supervised adaptation, in which models pretrained with masked language modeling are adapted over time.

Our results show that SDFT-MLM mitigates forgetting on earlier tasks, which are especially vulnerable during sequential fine-tuning, while maintaining competitive performance on later tasks. These findings suggest that token-level self-distillation offers a simple and broadly applicable mechanism for continual adaptation of models pretrained with masked language modeling.

2. Methods

2.1. Tasks and Data

We evaluate SDFT-MLM on continual protein property regression, a representative downstream adaptation setting for masked language models. The benchmark contains four protein property prediction tasks: AAV packaging (Dallago et al., 2021), protein stability (Rocklin et al., 2017), GFP fluorescence (Sarkisyan et al., 2016), and Meltome thermostability (Dallago et al., 2021). Each task consists of amino-acid sequences paired with a scalar experimental score. We use the data splits from Schmirler et al. (2024), which are summarized in Table 1. Although these tasks differ in assay type, dataset size, and sequence length, they are not fully independent: all can reflect broader notions of protein fitness, stability, or functional tolerance. Thus, the benchmark should not be interpreted as a perfectly isolated test of catastrophic forgetting across unrelated tasks. Instead, we view it as a realistic biological setting for sequential fine-tuning, where a model must retain useful predictors while adapting across related but heterogeneous properties and input distributions.

Table 1. Dataset sizes and average sequence lengths used in the continual protein regression experiments.

Task	Train	Validation	Test	Mean length
AAV	28,626	3,181	50,776	736.3
Stability	53,614	2,512	12,851	45.0
GFP	21,446	5,362	27,217	237.0
Meltome	22,335	2,482	3,134	544.5

2.2. Model

All experiments use the 35M-parameter ESM2 masked protein language model (Lin et al., 2023). For each task, we attach a task-specific regression head to the final-layer `[CLS]` embedding. Each head is implemented as a two-layer multilayer perceptron followed by a scalar output layer. The ESM2 backbone and all task heads are trainable during both supervised fine-tuning and SDFT-MLM.

Before the continual-learning runs, we initialize the task heads by fitting them on top of a frozen ESM2 backbone for 500 steps per task. This provides a common starting point for both SFT and SDFT-MLM. This yields a frozen-backbone baseline performance of 0.30 Spearman for AAV, 0.40 for Stab, 0.29 for GFP, and 0.51 for Meltome. The continual-learning experiments then start from the same initialized model and update both the backbone and task heads sequentially.

2.3. Continual-Learning Study Design

We evaluate three four-stage task orders:

1. AAV → Stability → GFP → Meltome,
2. AAV → GFP → Stability → Meltome,
3. Stability → Meltome → AAV → GFP.

At each stage, the model is trained only on the current task. This protocol measures both acquisition of the active task and retention of tasks encountered earlier in the sequence. It also captures order effects: for a given task, final performance may depend on which datasets were presented before and after that task.

The nominal training budget is maximum 1,500 gradient steps per stage. The resulting four-stage runs contain 5,738 gradient steps in total: 1,500 for AAV, 1,500 for Stab, 1,342 for GFP, and 1,396 for Meltome, arranged according to the corresponding task order. Evaluation is performed every 50 gradient steps and once more at the end of training.

2.4. SFT and SDFT-MLM Objectives

The supervised fine-tuning (SFT) baseline optimizes the model only for the downstream scalar prediction task. Given a protein sequence x_i, the model predicts a scalar value

$$\hat{y}_{i,t} = f_{\theta,t}(x_i)$$

for task t. For a mini-batch of size B, the supervised loss is the mean squared error between the measured and predicted scores:

$$\mathcal{L}_{\mathrm{MSE}}^{(t)}(\theta) = \frac{1}{B}\sum_{i=1}^{B}\left(\hat{y}_{i,t} - y_{i,t}\right)^2 . \tag{1}$$

SDFT-MLM augments this supervised objective with a self-distillation term applied at the masked-language-modeling output layer. The original SDFT method was developed for autoregressive large language models (LLMs) trained from demonstrations (Shenfeld et al., 2026), where a demonstration-conditioned model distribution serves as the teacher and the same model conditioned only on the input

serves as the student. This construction relies on the in-context learning capabilities of autoregressive LLMs. In our setting, however, the downstream protein tasks produce scalar property predictions rather than response sequences, and demonstration-conditioned prompting is not natural for pretrained masked language models. We therefore adapt the self-distillation idea by distilling token-level distributions, which preserves the regularizing effect of self-distillation.

At the beginning of each task stage, we initialize a teacher model as a copy of the student model. The teacher is not updated by gradient descent. Instead, after each student update, its parameters are updated using an exponential moving average (EMA) of the student parameters:

$$\theta_{\mathrm{T}} \leftarrow \beta\theta_{\mathrm{T}} + (1-\beta)\theta_{\mathrm{S}}. \tag{2}$$

In our experiments, we set the EMA decay coefficient to $\beta = 0.99$, so the teacher acts as a slowly moving reference model for the student.

For each input sequence, both the student and teacher produce logits over the vocabulary at each non-padding position. Let $q_{\mathrm{T}}^{i,\ell}$ and $q_{\mathrm{S}}^{i,\ell}$ denote the teacher and student token distributions for example i at sequence position ℓ, respectively. Let $a_{i,\ell} \in \{0,1\}$ denote the attention mask, where $a_{i,\ell} = 1$ indicates a non-padding token. The distillation loss is the average teacher-to-student KL divergence over non-padding positions:

$$\mathcal{L}_{\mathrm{KL}}^{(t)} = \frac{\sum_{i=1}^{B}\sum_{\ell=1}^{L} a_{i,\ell} D_{\mathrm{KL}}\left(q_{\mathrm{T}}^{i,\ell} \,\middle\|\, q_{\mathrm{S}}^{i,\ell}\right)}{\sum_{i=1}^{B}\sum_{\ell=1}^{L} a_{i,\ell}}. \tag{3}$$

The full SDFT-MLM objective combines supervised regression with masked-language-model self-distillation:

$$\mathcal{L}_{\text{SDFT-MLM}}^{(t)} = \mathcal{L}_{\mathrm{MSE}}^{(t)} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}^{(t)}. \tag{4}$$

We set $\lambda_{\mathrm{KL}} = 1$ in our experiments, giving equal weight to the supervised and distillation terms, although other weightings could also be explored. The MSE term adapts the model to the current downstream property, while the KL term discourages abrupt changes in the masked-language-model token distribution represented by the backbone. In the protein setting, this regularizes the amino-acid sequence representation without storing examples from previous tasks.

2.5. Optimization and Evaluation

Both SFT and SDFT-MLM are trained with AdamW using a learning rate of 5×10^{-5}, weight decay of 0.01, gradient clipping with a maximum norm of 1.0, and a cosine learning-rate schedule with 10% warmup. Unless otherwise specified, all experiments use a batch size of 32, a maximum sequence length of $1{,}024$, random seed 42, and bfloat16 automatic mixed precision. The primary evaluation metric is the Spearman rank correlation between predicted and measured scores on the test set. We evaluate the model every 50 gradient steps and perform an additional evaluation at the end of training.

3. Results

3.1. Final Performance After Continual Learning

Table 2 reports the final full-test Spearman correlations after all four stages of continual training. The main trend is that SDFT-MLM is particularly beneficial for tasks learned earlier in the sequence. In the first order, AAV is trained first and is followed by three additional tasks; its final score improves substantially from 0.61 with SFT to 0.72 with SDFT-MLM. In the second order, AAV is again learned first, and SDFT-MLM improves its final score from 0.66 to 0.69. Similarly, in the third order, where Stability and Meltome are learned before AAV and GFP, SDFT-MLM improves Stability from 0.65 to 0.69. These results suggest that SDFT-MLM mainly improves retention of previously learned tasks rather than only improving adaptation to the current task.

For later tasks, the two methods perform comparably. When Meltome or GFP is the final task, SFT and SDFT-MLM achieve similar results, with only minor differences across task orders. This pattern suggests a retention–adaptation trade-off: the distillation objectives help preserve previously learned representations, but they may slightly limit adaptation to the most recently trained task.

The results also show that task ordering matters. The first two orders differ only in the relative order of Stability and GFP, yet this change affects not only those tasks but also AAV, which is trained first in both cases. For example, the final AAV score under SDFT-MLM decreases from 0.72 in AAV → Stability → GFP → Meltome to 0.69 in AAV → GFP → Stability → Meltome. This indicates that interference is not determined only by a task's position in the sequence, but also by which subsequent tasks are learned and in what order.

3.2. Training Dynamics

Figures 1 to 3 show Spearman trajectories for all tasks during sequential training. Vertical dashed lines mark task boundaries. These trajectories complement the final results by showing when performance changes occur and how each method behaves during later fine-tuning stages.

Overall, the trajectories of different tasks are not completely independent. Although the datasets measure different protein properties, they are still related enough that training on one task can sometimes improve performance on another. For example, in the last stage of Figure 1, training

Table 2. Final test-set Spearman correlations after continual training on four regression tasks. Values are rounded to two decimal places. Bold values indicate the best method within each task order and task; when the two methods are tied after rounding, both values are bolded.

Model	Method	AAV	Stability	GFP	Meltome
Frozen backbone	–	0.30	0.40	0.29	0.51
AAV → Stability → GFP → Meltome	SFT	0.61	0.73	**0.64**	**0.64**
AAV → Stability → GFP → Meltome	SDFT-MLM	**0.72**	**0.74**	**0.64**	0.63
AAV → GFP → Stability → Meltome	SFT	0.66	0.71	0.59	**0.64**
AAV → GFP → Stability → Meltome	SDFT-MLM	**0.69**	**0.75**	**0.63**	0.63
Stability → Meltome → AAV → GFP	SFT	**0.74**	0.65	**0.68**	0.54
Stability → Meltome → AAV → GFP	SDFT-MLM	**0.74**	**0.69**	**0.68**	**0.55**

on Meltome also improves AAV and Stability performance. This suggests that some stages provide shared useful signal rather than only causing interference.

At the same time, the curves indicate that SDFT-MLM can be more tolerant to forgetting than standard SFT. An example appears in Figure 2: after AAV is trained in the first stage, the model is trained on GFP in the second stage. During this GFP stage, the SFT performance on AAV drops from roughly 0.8 to about 0.6, whereas SDFT-MLM decreases more mildly, from roughly 0.8 to about 0.7. This pattern suggests that the MLM distillation term helps preserve previously learned representations while the model adapts to the new task.

Some datasets also produce less smooth training dynamics, especially Stability dataset. When Stability task is introduced, the curves often show sharp changes not only for Stability itself but also for other tasks. For instance, in the third stage of Figure 2, training on Stability introduces abrupt fluctuations in both Stability and GFP performance for SFT and SDFT-MLM. Similar behavior is visible in Figure 3. SDFT-MLM appears somewhat more robust to these unstable transitions, with fewer or smaller degradations in the retained-task trajectories, improving the dynamics of sequential fine-tuning.

4. Discussion

These experiments suggest that self-distillation can improve the continual adaptation of pretrained masked language models. In our protein language model case study, SDFT-MLM improves average final ranking performance relative to supervised fine-tuning, with the largest gains appearing on tasks learned earlier in the training (Table 2). This pattern supports the interpretation that SDFT-MLM mainly improves retention by preserving the backbone's masked-token distribution through an EMA teacher, thereby reducing destructive representation drift during later fine-tuning stages. The benefit, however, is not uniform. SDFT-MLM does not consistently improve performance on the final task and slightly underperforms supervised fine-tuning on Meltome in the first two task orders, suggesting a trade-off between retention and adaptation: the distillation term can protect earlier tasks by discouraging large changes in the masked-language-model output distribution, but it may also constrain useful updates for the most recent task. Task order also plays an important role. AAV is trained at the start of the first two orders, yet its final performance differs depending on whether Stability or GFP follows, indicating that interference depends not only on task position but also on dataset similarity and transfer structure. SDFT-MLM is not a direct implementation of prompt-based SDFT. Rather, it adapts the broader self-distillation principle to masked language models by using an EMA teacher over token-level distributions.

Our study has several limitations. We evaluate only three task orders, a single EMA coefficient ($\beta = 0.99$), a single distillation weight ($\lambda_{\mathrm{KL}} = 1$), one model size, and one model family, and we do not systematically tune other hyperparameters such as learning rate, training budget, batch size, or head architecture. The empirical scope is also limited to four protein regression datasets, which is insufficient to characterize behavior across broader variation in assay type, sequence length, label noise, dataset size, and domain shift. Another practical limitation is that SDFT-MLM requires storing and evaluating a teacher copy of the model. This overhead is manageable for the 35M-parameter ESM2 model used here, but it may become challenging for larger foundation models because it increases memory use and compute cost during fine-tuning. These results should therefore be interpreted as preliminary evidence from a controlled case study rather than as a fully optimized or comprehensive benchmark.

Future work should evaluate SDFT-MLM across a broader range of datasets, task orders, model sizes, and model families. It should also quantify dataset and task similarity more explicitly, as these factors may help predict when self-distillation improves retention and when it instead impedes transfer. Stronger continual-learning baselines such

as LoRA-style task modules (Hu et al., 2021) is also needed to isolate the specific contribution of self-distillation. More broadly, this approach should be tested in other foundation-model settings, including natural language and biological sequence modeling (Lee et al., 2019; Nguyen et al., 2024). Despite its current limitations, this study provides preliminary evidence that preserving token-level distributions with an EMA teacher is a simple and promising mechanism for improving retention during sequential fine-tuning of pre-trained masked language model backbones.

References

Consens, M. E., Dufault, C., Wainberg, M., Forster, D., Karimzadeh, M., Goodarzi, H., Theis, F. J., Moses, A., and Wang, B. Transformers and genome language models. *Nat. Mach. Intell.*, March 2025.

Dallago, C., Mou, J., Johnston, K. E., Wittmann, B. J., Bhattacharya, N., Goldman, S., Madani, A., and Yang, K. K. Flip: Benchmark tasks in fitness landscape inference for proteins. *bioRxiv*, 2021. URL https://api.semanticscholar.org/CorpusID:244060611.

Delange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., and Tuytelaars, T. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021. ISSN 1939-3539. doi: 10.1109/tpami.2021.3057446. URL http://dx.doi.org/10.1109/TPAMI.2021.3057446.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. URL https://arxiv.org/abs/1810.04805.

Gardazi, N. M., Daud, A., Malik, M. K., Bukhari, A., Alsahfi, T., and Alshemaimri, B. BERT applications in natural language processing: a review. *Artif. Intell. Rev.*, 58(6), March 2025.

Howard, J. and Ruder, S. Universal language model fine-tuning for text classification. In Gurevych, I. and Miyao, Y. (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1031. URL https://aclanthology.org/P18-1031/.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2019. ISSN 1367-4811. doi: 10.1093/bioinformatics/btz682. URL http://dx.doi.org/10.1093/bioinformatics/btz682.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., Dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., and Rives, A. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023.

Nguyen, E., Poli, M., Durrant, M. G., Kang, B., Katrekar, D., Li, D. B., Bartie, L. J., Thomas, A. W., King, S. H., Brixi, G., Sullivan, J., Ng, M. Y., Lewis, A., Lou, A., Ermon, S., Baccus, S. A., Hernandez-Boussard, T., Ré, C., Hsu, P. D., and Hie, B. L. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, November 2024.

Patil, D., Malviya, P., Reymond, M., Fournier, Q., and Chandar, S. Copep: Benchmarking continual pretraining for protein language models, 2026. URL https://arxiv.org/abs/2603.00253.

Rocklin, G. J., Chidyausiku, T. M., Goreshnik, I., Ford, A., Houliston, S., Lemak, A., Carter, L., Ravichandran, R., Mulligan, V. K., Chevalier, A., et al. Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science*, 357(6347):168–175, 2017.

Saadat, A. and Fellay, J. Fine-tuning protein language models to understand the functional impact of missense variants. *Comput. Struct. Biotechnol. J.*, 27:2199–2207, May 2025.

Sarkisyan, K. S., Bolotin, D. A., Meer, M. V., Usmanova, D. R., Mishin, A. S., Sharonov, G. V., Ivankov, D. N., Bozhanova, N. G., Baranov, M. S., Soylemez, O., et al. Local fitness landscape of the green fluorescent protein. *Nature*, 533(7603):397, 2016.

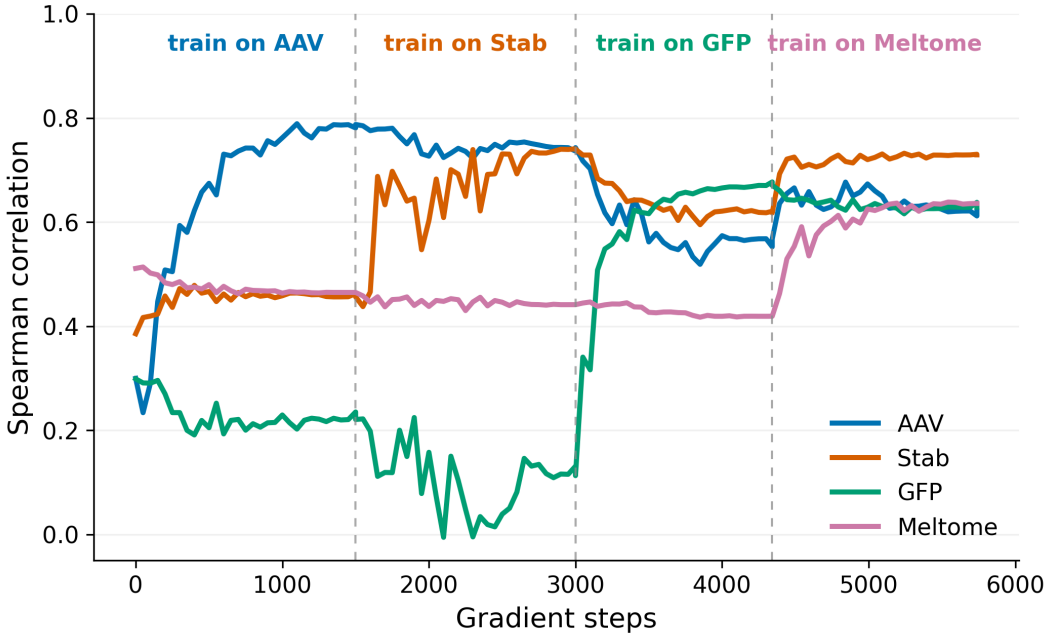
Schmirler, R., Heinzinger, M., and Rost, B. Fine-tuning protein language models boosts predictions across diverse tasks. *Nat. Commun.*, 15(1):7407, August 2024.

Shenfeld, I., Damani, M., Hübotter, J., and Agrawal, P. Self-distillation enables continual learning, 2026. URL https://arxiv.org/abs/2601.19897.

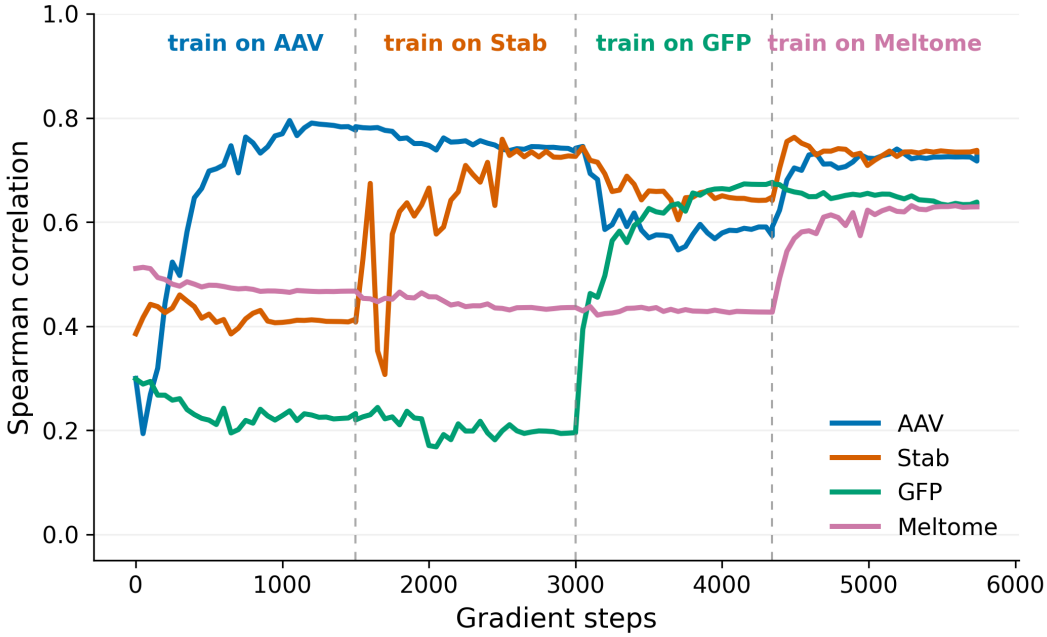
Wang, L., Zhang, X., Su, H., and Zhu, J. A comprehensive survey of continual learning: Theory, method and application, 2024. URL https://arxiv.org/abs/2302.00487.

Wang, Z., Wang, Z., Jiang, J., Chen, P., Shi, X., and Li, Y. Large language models in bioinformatics: A survey. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 3602–3615, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.184. URL https://aclanthology.org/2025.findings-acl.184/.

A. Supplementary Figures

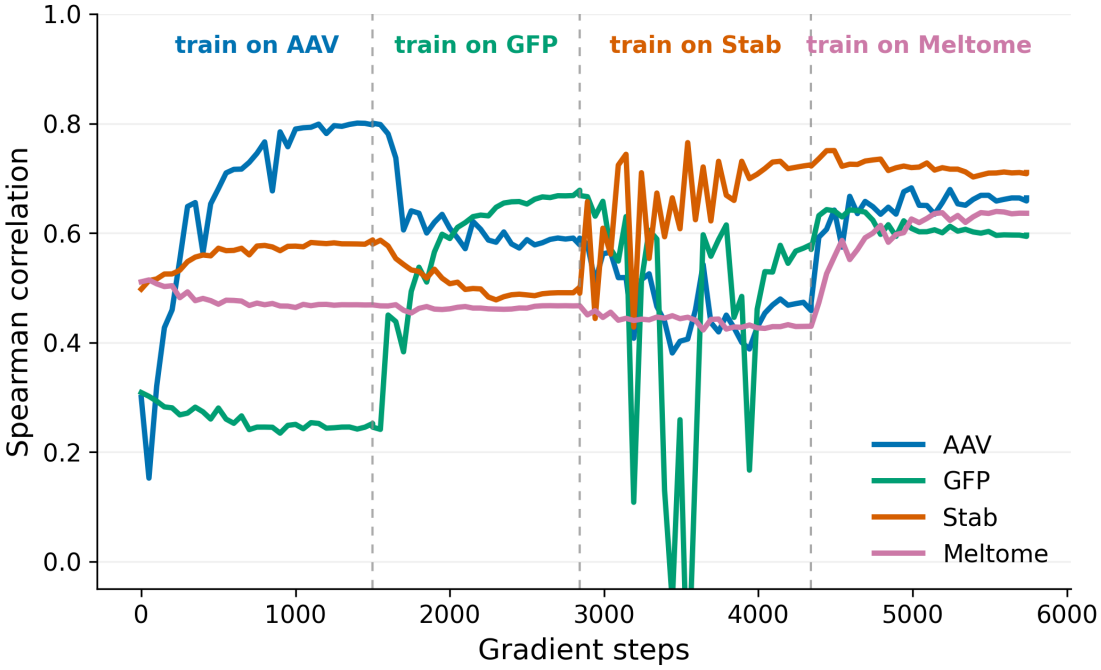


(a) SFT

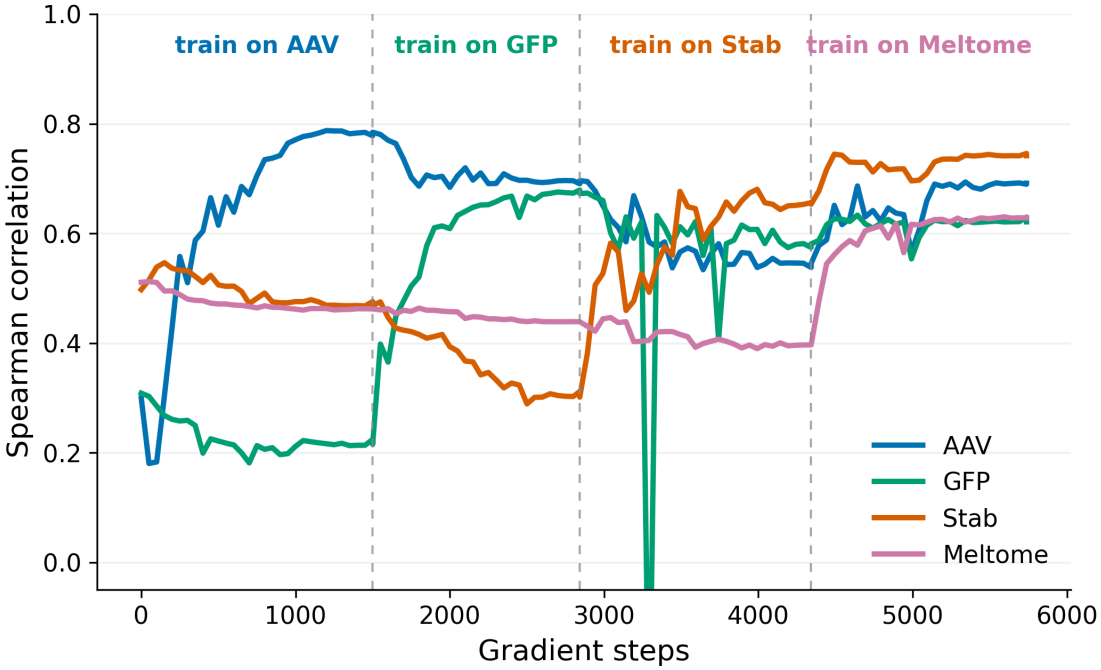


(b) SDFT-MLM

Figure 1. Spearman trajectories for AAV → Stability → GFP → Meltome.

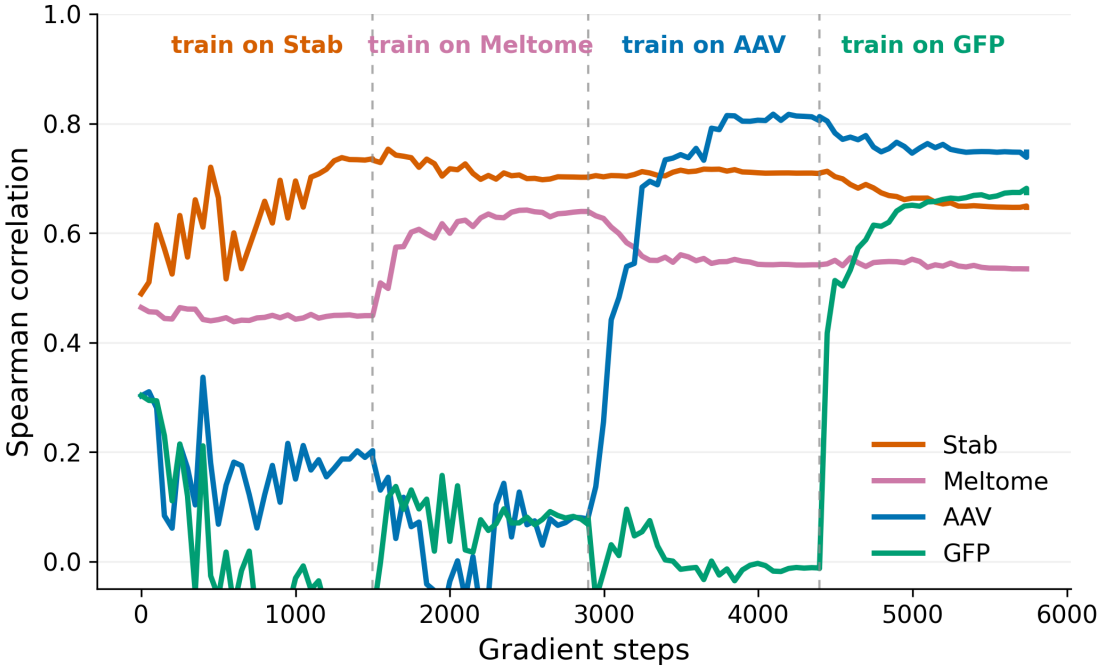


(a) SFT

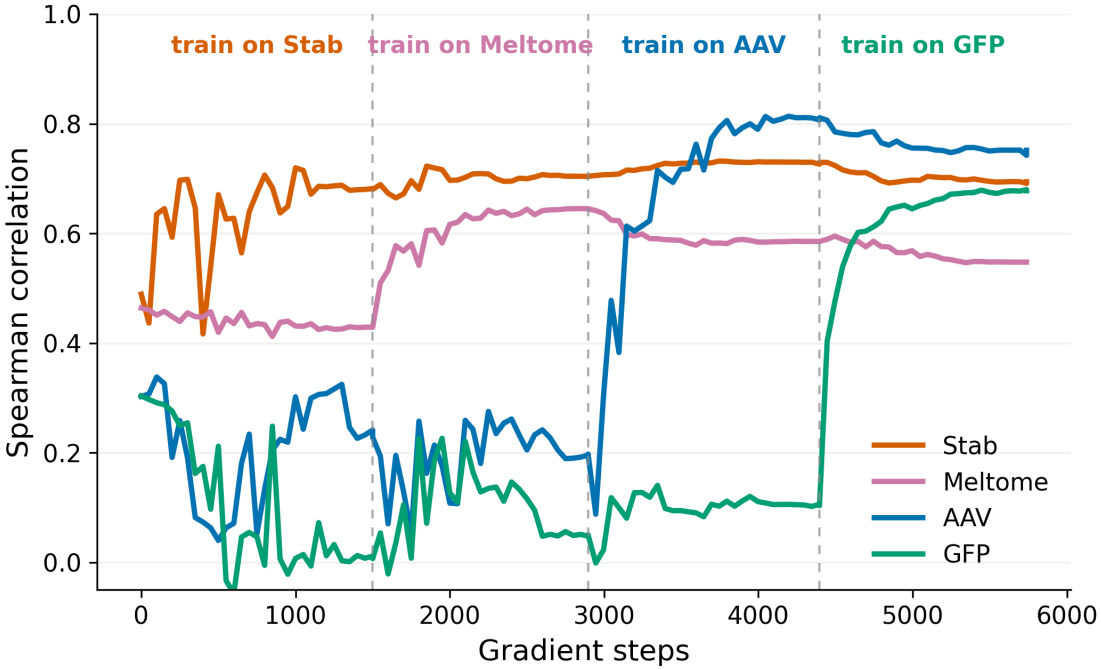


(b) SDFT-MLM

Figure 2. Spearman trajectories for AAV → GFP → Stability → Meltome.



(a) SFT



(b) SDFT-MLM

Figure 3. Spearman trajectories for Stability → Meltome → AAV → GFP.