TadABench-1M: A Large-Scale Wet-Lab Protein Activity Dataset from Real-World Applications

Anonymous Author(s)
Affiliation
Address
email

Abstract

Large language models trained on biomolecular sequences—DNA, RNA, and proteins—exhibit impressive in silico scaling trends, yet their practical utility in laboratory protein engineering remains under-explored. We assemble a million-example, wet-lab-validated dataset comprising 31 rounds of directed evolution on the tRNA-specific adenosine deaminase (TadA) that underlies adenine base editors. To harmonize labels across rounds, we introduce Seq2Graph, a scalable graph-based reconciliation algorithm that mitigates sequencing noise. Leveraging this resource, we propose TadABench-1M, an application-oriented benchmark that tasks models with ranking candidate variants for the next evolutionary round, given data from previous rounds. State-of-the-art biological language models achieve a Spearman correlation of only $\rho \approx 0.1$ under this realistic setup, contrasting sharply with $\rho \approx 0.8$ on a random split of this dataset, revealing a striking gap between computational metrics and wet-lab success. Controlled ablations show that sequence diversity and round coverage, rather than raw data density, dominate performance, pinpointing key bottlenecks for next-generation biological language models. TadABench-1M provides a large-scale, realistic foundation for developing and evaluating pre-trained language models. We will release the data and code.

1 Introduction

Language models pre-trained on biomolecular sequences, DNA [13], RNA [28], and proteins [37, 35], have recently exhibited scaling behaviour analogous to that observed in natural-language processing [3, 24, 7, 17, 18, 43]. As parameter count and corpus size grow, the computational evaluation, such as perplexity and masked-token accuracy, steadily improves. We refer to these models collectively as biological language models (BLMs). Despite their impressive in silico results, the real-world value of BLMs remains unclear. In a recent crowdsourced antibody-design challenge [11], models with state-of-the-art computational metrics nonetheless produced low-affinity binders, highlighting a troubling gap between numerical scores and wet-lab success.

In this work, we interrogate the practical utility of BLMs for protein engineering. Our case study is tRNA-specific adenosine deaminase (TadA), the catalytic core of widely used adenine base editors [30, 21]. We first assemble a rigorously standardized, million-scale dataset of TadA variants and their in-cell editing activities. The dataset is generated over 31 rounds of directed evolution by coupling phage-assisted non-continuous evolution (PANCE) with degenerate sequence synthesis (see Appendix A). To harmonize labels across rounds, we introduce Seq2Graph, a scalable graph-based algorithm that reconciles replicate measurements and suppresses sequencing noise.

Leveraging this dataset, we formulate TadABench-1M, an application-oriented benchmark that mirrors how BLMs are deployed in practice: given data from previous rounds, predict the relative activity of variants synthesised in the next round. Strikingly, state-of-the-art BLMs achieve only a

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

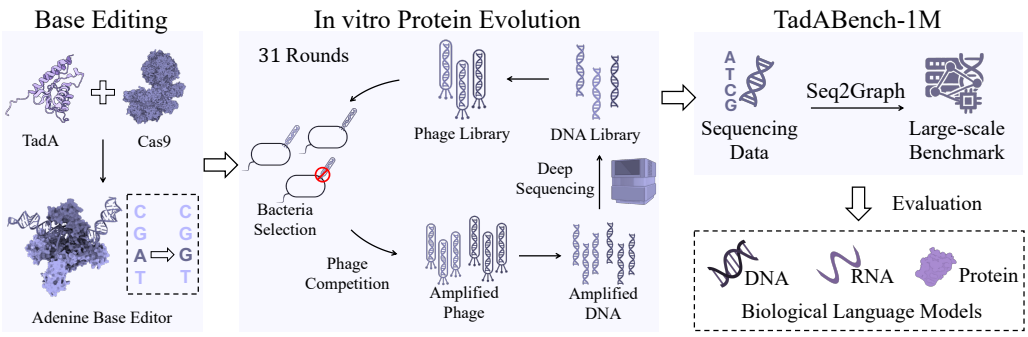

Figure 1: TadABench-1M is derived from extensive wet-lab protein evolution experiments encompassing 31 iterative rounds. The left panel illustrates the functional role of TadA as a key enzyme in base editing. The central panel presents the in vitro evolution process of TadA, which enabled the generation of comprehensive deep sequencing data. The right panel outlines the construction of our dataset using the Seq2Graph method, followed by its evaluation with biological language models.

Spearman correlation of $\rho \approx 0.1$ under this realistic setting, whereas the same models reach $\rho \approx 0.8$ on a conventional i.i.d. random split of TadABench-1M. These results echo the antibody-design failure [11] and reveal that standard in silico evaluations may overestimate the performance of BLMs.

To probe the origin of this gap, we conduct a controlled study that systematically varies the *density* (number of randomly retained variants), *diversity* (sequence-similarity-based sampling), and *round coverage* of the training data. We find that the largest gains arise from incorporating additional evolutionary rounds or maximizing sequence diversity, rather than merely increasing the number of training examples. These results suggest that BLMs benefit most from training data with broad sequence diversity to effectively capture the attributes required for downstream tasks. From a data curation perspective, this highlights the importance of expanding experimental coverage and ensuring diversity across the vast sequence space, rather than focusing solely on dataset size.

Contributions.

- We release a million-example, wet-lab-validated dataset of TadA activity obtained from 31 standardized rounds of directed evolution. It is built by our Seq2Graph, a scalable graph-based algorithm that enforces cross-round label consistency and mitigates sequencing noise.
- We establish TadABench-1M, an application-oriented benchmark spanning protein, DNA, and RNA language models. Experiments reveal a large gap between in-silico evaluation and real-world performance, where the dataset split for a practical scenario is much harder than the random split.
- Through controlled ablation, we show that BLMs need higher sequence diversity and round coverage, rather than raw data density, to dominate downstream performance.

2 Related Work

2.1 Protein Activity Benchmark

While structural benchmarks in protein research have achieved notable success [69, 41, 23, 4], functional benchmarks are still in nascent stages. These benchmarks are primarily categorized into two groups [62], biophysical properties and deep mutational scanning (DMS) data. The benchmarks for biological properties [1, 65, 73, 44, 49, 58] include metrics like enzymatic activity, fluorescence, thermodynamics, and solubility; however, their broad focus limits their utility in precise evaluations. DMS benchmarks [20, 29, 22], which utilize large-scale mutagenesis and high-throughput sequencing, offer detailed insights into fitness landscapes for protein mutations. Researchers [45, 14, 51] also leverage diverse DMS datasets to construct the comprehensive benchmark, but may introduce inconsistency since the way data is treated varies widely within the community [45]. In conclusion, the above benchmarks are for general purposes without a specific application. In contrast, TadABench-1M is from a real-world application scenario. Besides, it has a significantly higher sequence diversity

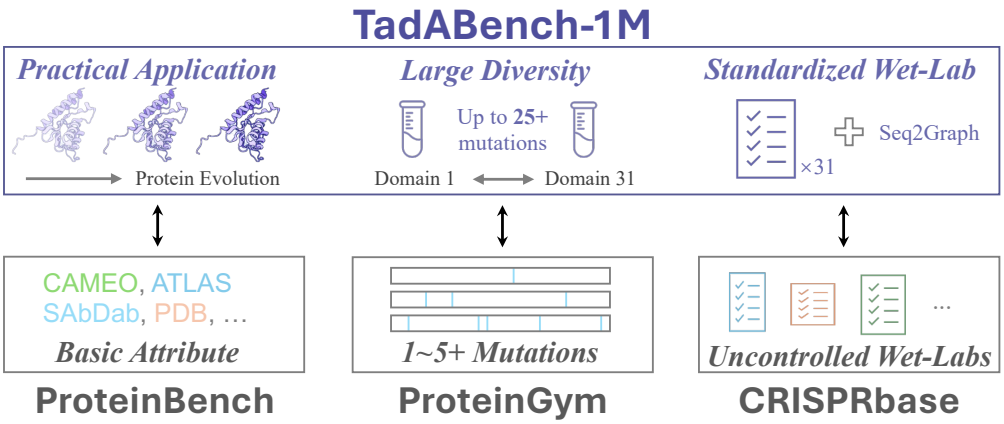

Figure 2: Our TadABench-1M dataset provides three key advantages over related benchmarks. First, it is purposefully curated from practical TadA evolution. Second, it comprises 1,027,200 unique TadA variants spanning 31 diverse rounds, with sequence differences reaching up to 25 amino acid mutations (and over 150 at the DNA level). Third, our standardized wet-lab protocols and algorithmic pipeline (Seq2Graph) ensure data consistency suitable for machine learning applications.

than the traditional mutation-based dataset. Finally, it is guaranteed with strict consistency by standardized wet-lab experiments and our novel algorithm Seq2Graph, as illustrated in Figure 2.

2.2 Base Editing Dataset

Current datasets focusing on deaminase enzyme optimization in base editing are relatively sparse and often fragmented. Most available resources [15, 66, 38, 56, 64, 32] emphasize the optimization of base editors through the lens of their interactions with CRISPR-associated proteins (Cas) and single-guide RNAs (sgRNAs), rather than through a systematic exploration of the deaminase variants themselves. These datasets are typically derived from narrow experimental conditions, thereby limiting their generalizability and scalability for machine learning (ML)-based modeling and prediction. Several research groups have published improved or novel deaminase protein sequences through directed evolution or rational design [50, 34, 57, 48, 10], but with various wet-lab experimental conditions. Recent aggregation platforms such as CRISPRbase [19] aim to centralize base editing datasets across various publications and labs. While this is a significant step forward in data accessibility, it introduces significant batch effects due to diverse experimental protocols [45], compromising the accuracy of models under real experimental conditions. In contrast, we introduce a large-scale wet-lab TadA dataset, TadABench-1M, by our standardized in vitro experiments and our cross-round consistency control algorithm Seq2Graph. It is designed to simulate real-world laboratory conditions consistently, thereby enhancing the precision in deaminase evolution and evaluating biological language models.

3 TadABench-1M Construction by Seq2Graph

This section presents our dataset construction method, Seq2Graph, shown in Figure 3. It processes 31 large-scale deep sequencing results collected in our wet lab. The pipeline comprises three key stages: directed graph construction (Section 3.1), inconsistency elimination (Section 3.2), and activity assignment (Section 3.3). Finally, we visualize the benchmark and dataset in Appendix C.1.

3.1 Directed Graph Construction

In our wet experiments, phages encoding TadA variants with enhanced activity propagate faster, resulting in increased read counts during next-generation sequencing (NGS)[1] of the final populations, shown on the left of Figure 3. For each NGS data, researchers build a dataset with normalized read

[1]Next-Generation Sequencing (NGS) is a broad technology platform that enables deep sequencing, which refers to generating high coverage of sequencing reads for a target region. In this paper, we use NGS and deep sequencing interchangeably.

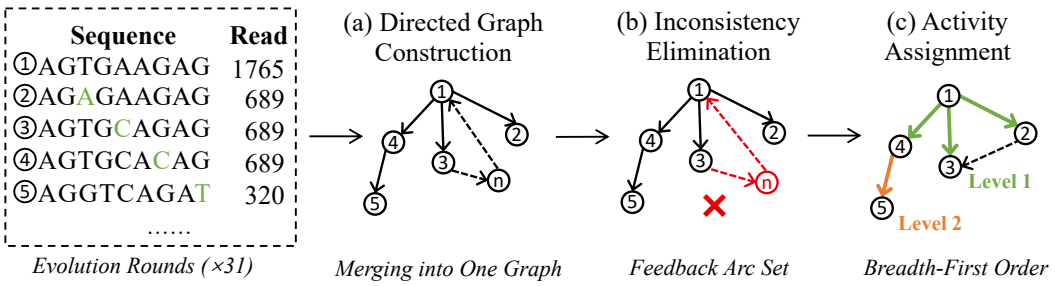

Figure 3: Pipeline of Seq2Graph. (a) We integrate deep sequencing data from 30 rounds of protein evolution into a directed acyclic graph. (b) We apply the Feedback Arc Set algorithm to resolve inconsistencies across experimental rounds. (c) We assign activity values to each TadA variant following a breadth-first traversal order to minimize sequencing noise along relative relationships.

counts for each NGS result [59, 36, 52]. However, normalization cannot merge NGS data from 31 rounds into one consistent super-large dataset. Although each round uses the same standardized wet experiment protocol, the inherent randomness of biological experiments leads to inconsistencies.

To solve such limitations, we propose to model the relative activity of variants in a directed graph, rather than the common practice of absolute activity after normalization, shown in Figure 3 (a). In the directed graph, $G = (V, E)$, the DNA sequence of each variant obtained from NGS is taken as a node v_i. For the list of growth multiples for read counts, the number of edges in G can be up to $|V|^2$. Considering the computational complexity, we require a sparse and weakly connected graph.

Specifically, we sort the list of growth multiples for read counts and only add edges for nodes with adjacent count values along the list. Besides, we perform average edge pruning with no more than 100,000 edges between nodes of two adjacent growth multiples. Each edge points from nodes with higher activity to those with lower activity, which means the edge weight is always greater than 1. The weight of edge $e_{i \to j}$ represents the relative activity of v_i over v_j, $w_{ij} = \frac{C(v_i)}{C(v_j)}$, where $C()$ is the growth multiples for read counts.

In our protein evolution, we conduct a new round based on the best-performing variants in previous rounds. Hence, the graph constructed from all rounds of NGS is connected such that it can capture the activity relationships between each pair of sequence variants. Besides, record the experimental round information in the node attributes, where each node belongs to 1.58 rounds on average.

3.2 Inconsistency Elimination

Owing to the inherent randomness of biological experiments, inconsistencies occur among 31 rounds, though each round uses the same standardized wet experiment protocol. As shown in Figure 3 (b), the red node indicates the presence of strongly connected components in the directed graph, caused by conflicting relative activity relationships across diverse rounds. To eliminate inconsistency, we should remove cycles in the current directed graph. Since the read count of NGS data is accumulated with detected sequences during sequencing, the higher count represents higher reliability concerning individual sequencing error. In other words, edges with higher weights are considered more reliable. Hence, it can be formalized as a weighted feedback arc set problem in Equation (1).

$$\begin{aligned} \min_{F \subseteq E} \quad & \sum_{e \in F} w_e \\ \text{s.t.} \quad & G' = (V, E \setminus F) \text{ is acyclic} \end{aligned} \tag{1}$$

Since our graph has more than one million nodes, we choose a fast heuristic algorithm [16] to solve this problem. It recursively selects vertices based on the difference between their in-degree and out-degree to construct an acyclic ordering, approximating the minimum feedback arc set. This method efficiently reduces cycles while maintaining scalability. Furthermore, we apply it on the

subgraph built from each strongly connected component for speedup, rather than on the whole graph. Hence, we formulate a directed acyclic graph with the node number unchanged but fewer edges.

3.3 Activity Assignment

In protein engineering, the practical demand is to select sequences with higher activity among massive candidates. Therefore, we use relative activity rather than the absolute value. To assign activity values to each sequence variant, we designate the relative activity of the initial reference sequence (TadA8e [50]) as 1.0. Subsequently, as shown in Figure 3 (c), we assign relative activity values to the sequence variants based on the ratios of growth multiples for read counts between sequences, *i.e.*, the edge weights. We take the logarithm level assignment since the phage replication roughly follows an exponential growth.

To assign relative activity values to sequence variants, we take the breadth-first order since paths with more intermediate nodes may accumulate more biological noise and uncertainty. This approach is necessary because there may exist multiple paths between any two variants, potentially leading to conflicting activity estimates. If a node is encountered again through an alternative path, its value remains unchanged, as the initial assignment is assumed to have higher confidence. Hence, we obtain the DNA version of TadABench-1M, with 1,027,200 DNA sequences with consistent activity labels.

In our evolution experiments, the functioning of proteins occurs after the DNA transcription and protein translation. Since the activity of folded protein is hard to directly capture, the common observation of one protein is at the DNA level. To obtain the activity at the protein level, we average over all of its DNA variants in our graph. Notably, DNA variants of one protein have different observed activities owing to multiple complex effects, such as synonymous codon usage on sequence activity. Finally, we obtain a protein dataset consisting of 409,869 annotated protein sequences. We provide the visualization of TadABench-1M in Figure 6. The left protein structure is one variant in our TadABench-1M, folded using ESMFold [35].

3.4 Robustness of Seq2Graph

To demonstrate the rigorous uncertainty quantification for Seq2Graph, we perform two bootstrapping analyses. We randomly remove 50% of the rounds (15 of 31) and re-run the entire Seq2Graph construction pipeline to create a new dataset, TadABench-half. We then identify the sequences common to both the full TadABench-1M and TadABench-half datasets and calculate the correlation of their assigned activity labels. The Spearman's ρ between the activity labels of the common sequences is 0.90, with a p-value of 0.0000 (SCIPY.STATS.SPEARMANR). Furthermore, removing 50% of sequencing reads within each round and re-running our full pipeline yield a Spearman's ρ of 0.95 and a p value of 0.0000. This extremely high correlation demonstrates that our Seq2Graph construction method is robust and stable, even when a substantial portion of the input data is removed.

4 Experiment

This section involves the evaluation on our TadABench-1M. We elaborate on detailed experimental settings in Section 4.1. A comprehensive evaluation is conducted among biological language models in Section 4.2. Furthermore, we provide the result of a random split in Section 4.3 to demonstrate that our dataset is learnable, but hard in a practical scenario. Finally, we systematically explore which matters more in data fractions among density, diversity, and round in Section 4.4.

4.1 Experimental Settings

Dataset TadABench-1M is curated from NGS data of evolved TadA variants, yielding both DNA/RNA and protein sequence datasets (Section 3.3). We emulate a practical protein engineering scenario by splitting the data chronologically: rounds 1–27 for training, round 28 for validation, and rounds 29–31 for testing. The nucleic acid (DNA/RNA) dataset contains 729,302 training sequences, 148,014 validation sequences, and 149,884 test sequences. The protein dataset comprises 256,429 training sequences, 45,208 validation sequences, and 108,232 test sequences.

Table 1: Performance on TadABench-1M (DNA version). Data from rounds 1–27 is used for training, round 28 for validation, and rounds 29–31 for testing. For each model, the best result is selected from three different learning rates. The **Bold** numbers indicate the highest performance. We here report the RNA language model, OmniGenome (OG), since the dataset is the same after changing T to U.

Model	Validation			Test		
	Spearman	Recall@10%	nDCG@10%	Spearman	Recall@10%	nDCG@10%
Evo-7B	0.0490	0.1097	0.2604	**0.0707**	0.1005	0.3236
Evo-40B	**0.0980**	**0.1157**	**0.2702**	0.0675	0.1003	**0.3244**
NT-50M	0.0401	0.0959	0.2464	0.0166	0.0950	0.3109
NT-100M	0.0520	0.0982	0.2485	0.0045	0.0870	0.3048
NT-250M	0.0470	0.0858	0.2137	0.0006	0.0971	0.3085
NT-500M	0.0361	0.0985	0.2225	0.0189	0.1005	0.3079
OG-46M	0.0555	0.0911	0.2192	0.0079	**0.1063**	0.3158
OG-418M	0.0078	0.0949	0.2391	0.0048	0.0859	0.3042

Task and Evaluation While the training labels correspond to activity values, TadABench-1M 's objective is not pure regression. Instead, the task focuses on predicting relative activity trends across test variants, reflecting realistic protein engineering workflows. We evaluate models using Spearman's rank correlation coefficient, normalized Discounted Cumulative Gain at the top 10% (nDCG@10%), and Recall@10%, adopting evaluation protocols from prior work [45]. Spearman's coefficient captures the overall relative ordering of activities across test sequences. Recall@10% measures the fraction of true top 10% variants correctly identified in the top 10% of predicted scores. Complementing these, nDCG@10% assesses whether the predicted top variants are correctly ranked by activity within the top decile (see Appendix D.1 for details).

Model We evaluate pre-trained biological language models (BLMs) spanning DNA, RNA, and protein domains. For DNA, we use models from the EVO2 [3] and NucleotideTransformer (NT) [13] families. For RNA, we include the OmniGenome (OG) [67] family. For protein, we consider ESM2 [35], ProtTrans [17], and ESMC [18] families. We extract representations from the final layer (or logits for EVO2 models) and perform linear probing to efficiently assess the encoded biological knowledge. Due to varying representation dimensions across models, the regression head's parameter count differs substantially. To ensure fair comparison, we employ a two-layer MLP with hidden layer sizes tailored per model for an equal number of trainable parameters, with the ReLU activation between layers. More details are shown in Appendix C.3. Note that the DNA/RNA language models can not be compared with protein language models directly, owing to the dataset difference.

4.2 Evaluation in Practical Scenarios

Biological language models underperform in splits based on practical scenarios. We evaluate the performance of various models on the TadABench-1M dataset, considering both DNA and protein versions, as shown in Table 1 and Table 2. All biological language models exhibit poor performance in this realistic setting, with a Spearman correlation of only $\rho \approx 0.1$. To further investigate, we include an experiment using one-hot encoding on the DNA version, which also yields low correlations, 0.0707 on the validation set and 0.0459 on the test set. These results highlight the difficulty of accurately predicting variants in practical, out-of-distribution scenarios.

4.3 Evaluation under Random Split

Results from the random split validate TadABench-1M and highlight the need for practical, application-level evaluation. Due to the relatively low absolute values observed in the real-world split, we additionally evaluate performance on a random split of TadABench-1M, where the sequences are divided into training, validation, and test sets in an 8:1:1 ratio. As shown in Table 3, biological language models (BLMs) achieve a Spearman correlation coefficient of $\rho \approx 0.8$ on both the validation and test sets. Their Recall@10% exceeds 0.2, significantly higher than the 0.1 level observed under

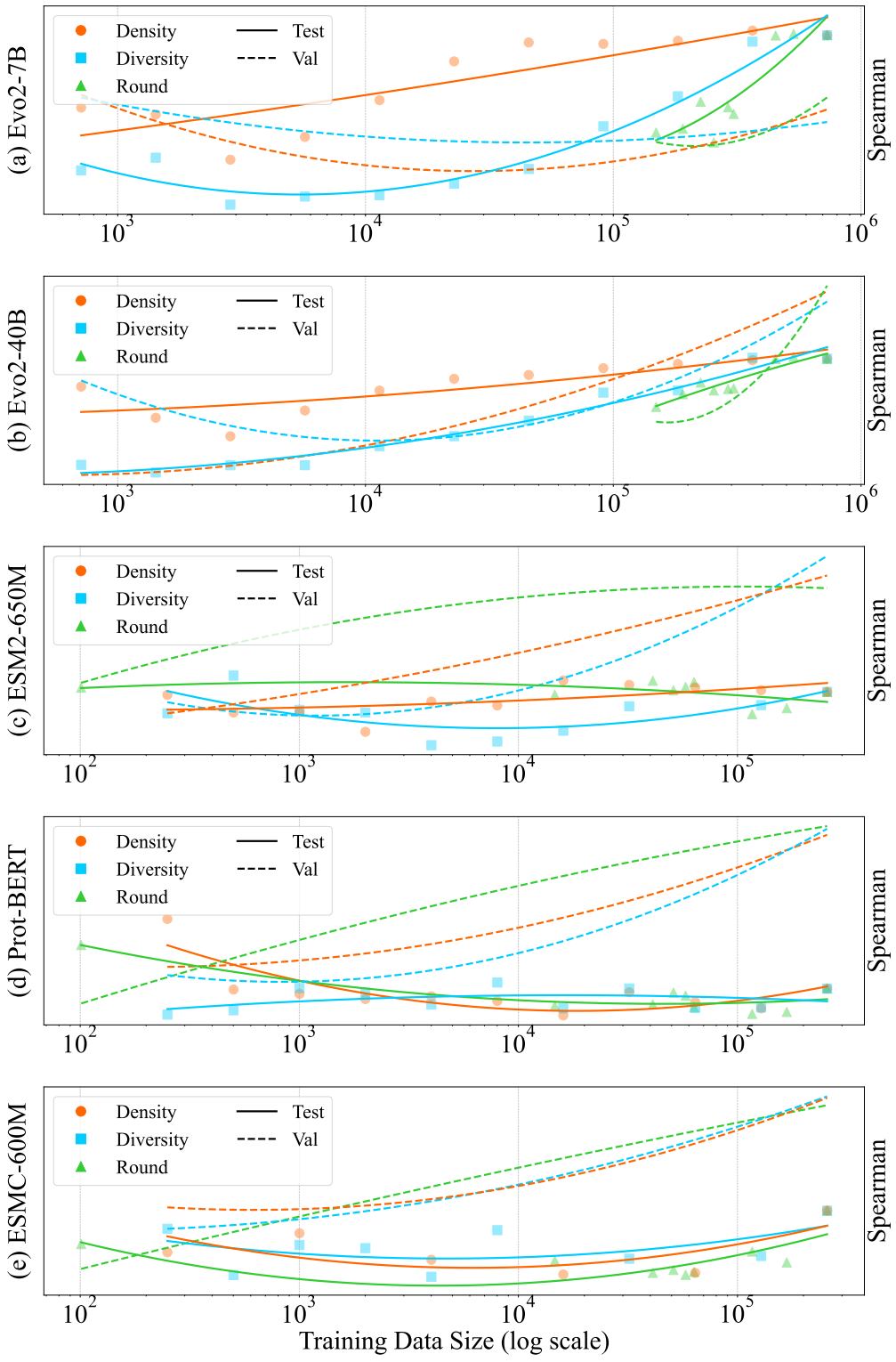

Figure 4: Performance by data scales in different modes of segmentation on diverse models. The rough data scaling trend can be observed among all models, especially on the domain. Note that the dataset of the DNA version (a, b) and protein version (c, d, e) is different.

Table 2: Performance on TadABench-1M (protein version). Data from rounds 1–27 is used for training, round 28 for validation, and rounds 29–31 for testing. For each model, the best result is selected from three different learning rates. **Bold** numbers indicate the highest performance.

Model	Validation			Test		
	Spearman	Recall@10%	nDCG@10%	Spearman	Recall@10%	nDCG@10%
ESM2-150M	0.1458	0.1420	0.6569	0.0416	**0.1230**	**0.3068**
ESM2-650M	0.1423	**0.1473**	0.6530	0.0479	0.1120	0.2791
Prot-BERT	0.1280	0.1128	0.6534	0.0214	0.1162	0.2980
Prot-XLNET	**0.1570**	0.1261	**0.6589**	0.0342	0.1175	0.2895
ESMC-300M	0.1498	0.1199	0.6495	0.0355	0.1151	0.2867
ESMC-600M	0.1452	0.1206	0.6397	**0.0509**	0.1180	0.2860

the practical-scenario split. It means that BLMs effectively capture the i.i.d. pattern under this setting, in stark contrast to the results presented in Table 2. These results suggest that the dataset is indeed learnable and that the underlying sequence patterns are detectable under idealized conditions. Consequently, the poor performance of BLMs in real-world scenarios stems from the substantial gap between conventional evaluation setups and the demands of real-world applications. In addition, the graph directly encodes the relative activity ranking among sequence variants, making it valuable to use BLMs for predicting sequence rankings (in Appendix D).

4.4 Ablation Study on Data

In prior experiments, we demonstrated the limitations of conventional evaluation setups and the mismatch between standard metrics and the requirements of real-world applications. Motivated by this gap, we investigate how characteristics of the training data influence downstream prediction performance in the next-round prediction. To this end, we partition the TadABench-1M dataset using three strategies—*density*, *diversity*, and *round*—to isolate which aspects most impact data scaling. The *density* strategy randomly subsamples the training data. The *diversity* strategy selects sequences with the highest similarity to those in the validation set, aiming to maximize coverage of the relevant functional space. The *round* strategy either retains or discards entire experimental rounds based on their aggregate similarity to the validation round. The validation and test sets remain fixed across all configurations. For consistency, we train with the same number of iterations across partitions.

Data diversity, not density, drives performance in protein engineering tasks. As shown in Figure 4, the x-axis denotes training data size (log scale), and the y-axis reports test set Spearman correlation. Subplots (a) and (b) correspond to Evo2 family models and exhibit clear scaling behavior predominantly on the test set. In contrast, subplots (c)–(e), which include ESM2-650M, Prot-BERT, and ESMC-600M, show stronger scaling patterns in the validation set. Notably, the *diversity* and *round*-based splits consistently outperform random *density*-based ones, indicating a greater benefit from strategic data selection. These findings suggest that biologically pre-trained language models need more diverse data to capture the functional landscape relevant to downstream tasks. From a data curation standpoint, this underscores the importance of broadening experimental coverage across the sequence space, rather than merely increasing dataset size.

5 Conclusion

We present the million-example, wet-lab-validated dataset of TadA variants, a scalable label-reconciliation algorithm (Seq2Graph), and TadABench-1M, an application-oriented benchmark that exposes a stark gap between in silico metrics and laboratory reality for biological language models. Our systematic ablations reveal that sequence *diversity* and evolutionary *round coverage*—rather than brute-force data density—are the key levers for closing this gap. These findings suggest that next-generation BLMs should move beyond static, randomly sampled corpora toward evolution-aware, diversity-oriented training data. Although comprehensive for TadA, our dataset is restricted to a single protein family and assay. Extending the framework to additional protein activity landscapes will be essential for broader generalization for the development of BLMs.

References

[1] Amos Bairoch. The enzyme database in 2000. *Nucleic acids research*, 28(1):304–305, 2000.

[2] Nicholas Boyd, Brandon M Anderson, Brent Townshend, Ryan Chow, Connor J Stephens, Ramya Rangan, Matias Kaplan, Meredith Corley, Akshay Tambe, Yuzu Ido, et al. Atom-1: A foundation model for rna structure and function built on chemical mapping data. *bioRxiv*, pages 2023–12, 2023.

[3] Garyk Brixi, Matthew G Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A Gonzalez, Samuel H King, David B Li, Aditi T Merchant, et al. Genome modeling and design across all domains of life with evo 2. *BioRxiv*, pages 2025–02, 2025.

[4] Martin Buttenschoen, Garrett M Morris, and Charlotte M Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15(9):3130–3139, 2024.

[5] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136, 2007.

[6] Chai Discovery team. Chai-1 technical report. Technical report, Chai Discovery, September 2024.

[7] Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, et al. xtrimopglm: unified 100b-scale pre-trained transformer for deciphering the language of protein. *arXiv preprint arXiv:2401.06199*, 2024.

[8] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*, 2022.

[9] Ken Chen, Yue Zhou, Maolin Ding, Yu Wang, Zhixiang Ren, and Yuedong Yang. Self-supervised learning on millions of pre-mrna sequences improves sequence-based rna splicing prediction. *bioRxiv*, pages 2023–01, 2023.

[10] Peng Cheng, Cong Mao, Jin Tang, Sen Yang, Yu Cheng, Wuke Wang, Qiuxi Gu, Wei Han, Hao Chen, Sihan Li, et al. Zero-shot prediction of mutation effects with multimodal deep representation learning guides protein engineering. *Cell Research*, pages 1–18, 2024.

[11] Tudor-Stefan Cotet, Igor Krawczuk, Filippo Stocco, Noelia Ferruz, Anthony Gitter, Yoichi Kurumida, Lucas de Almeida Machado, Francesco Paesani, Cianna N Calia, Chance A Challacombe, et al. Crowdsourced protein design: Lessons from the adaptyv egfr binder competition. *bioRxiv*, pages 2025–04, 2025.

[12] David Benjamin Turitz Cox, Randall Jeffrey Platt, and Feng Zhang. Therapeutic genome editing: prospects and challenges. *Nature medicine*, 21(2):121–131, 2015.

[13] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P de Almeida, Hassan Sirelkhatim, et al. The nucleotide transformer: Building and evaluating robust foundation models for human genomics. *BioRxiv*, pages 2023–01, 2023.

[14] Christian Dallago, Jody Mou, Kadina E Johnston, Bruce J Wittmann, Nicholas Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K Yang. Flip: Benchmark tasks in fitness landscape inference for proteins. *bioRxiv*, pages 2021–11, 2021.

[15] Shriniket Dixit, Anant Kumar, Kathiravan Srinivasan, PM Durai Raj Vincent, and Nadesh Ramu Krishnan. Advancing genome editing with artificial intelligence: opportunities, challenges, and future directions. *Frontiers in Bioengineering and Biotechnology*, 11:1335901, 2024.

[16] Peter Eades, Xuemin Lin, and William F Smyth. A fast and effective heuristic for the feedback arc set problem. *Information processing letters*, 47(6):319–323, 1993.

[17] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: towards cracking the language of life's code through self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:7112–7127, 2021.

[18] ESM Team. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning, 2024. URL `https://evolutionaryscale.ai/blog/esm-cambrian`.

[19] Jibiao Fan, Leisheng Shi, Qi Liu, Zhipeng Zhu, Fan Wang, Runxian Song, Jimeng Su, Degui Zhou, Xiao Chen, Kailong Li, et al. Annotation and evaluation of base editing outcomes in multiple cell types using crisprbase. *Nucleic Acids Research*, 51(D1):D1249–D1256, 2023.

[20] Douglas M Fowler and Stanley Fields. Deep mutational scanning: a new style of protein science. *Nature methods*, 11(8):801–807, 2014.

[21] Nicole M Gaudelli, Alexis C Komor, Holly A Rees, Michael S Packer, Ahmed H Badran, David I Bryson, and David R Liu. Programmable base editing of a• t to g• c in genomic dna without dna cleavage. *Nature*, 551(7681):464–471, 2017.

[22] Vanessa E Gray, Ronald J Hause, Jens Luebeck, Jay Shendure, and Douglas M Fowler. Quantitative missense variant effect prediction using large-scale mutagenesis data. *Cell systems*, 6(1): 116–124, 2018.

[23] Jürgen Haas, Alessandro Barbato, Dario Behringer, Gabriel Studer, Steven Roth, Martino Bertoni, Khaled Mostaguir, Rafal Gumienny, and Torsten Schwede. Continuous automated model evaluation (cameo) complementing the critical assessment of structure prediction in casp12. *Proteins: Structure, Function, and Bioinformatics*, 86:387–398, 2018.

[24] Tomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, et al. Simulating 500 million years of evolution with a language model. *bioRxiv*, pages 2024–07, 2024.

[25] Yong He, Pan Fang, Yongtao Shan, Yuanfei Pan, Yanhong Wei, Yichang Chen, Yihao Chen, Yi Liu, Zhenyu Zeng, Zhan Zhou, et al. Lucaone: Generalized biological foundation model with unified nucleic acid and protein language. *bioRxiv*, pages 2024–05, 2024.

[26] Isaac B Hilton and Charles A Gersbach. Enabling functional genomics with genome engineering. *Genome research*, 25(10):1442–1455, 2015.

[27] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48, 2000.

[28] Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

[29] Kaiyi Jiang, Zhaoqing Yan, Matteo Di Bernardo, Samantha R Sgrizzi, Lukas Villiger, Alisan Kayabolen, Byungji Kim, Josephine K Carscadden, Masahiro Hiraizumi, Hiroshi Nishimasu, et al. Rapid protein evolution by few-shot learning with a protein language model. *bioRxiv*, pages 2024–07, 2024.

[30] Alexis C Komor, Yongjoo B Kim, Michael S Packer, John A Zuris, and David R Liu. Programmable editing of a target base in genomic dna without double-stranded dna cleavage. *Nature*, 533(7603):420–424, 2016.

[31] Rohith Krishna, Jue Wang, Woody Ahern, Pascal Sturmfels, Preetham Venkatesh, Indrek Kalvet, Gyu Rie Lee, Felix S Morey-Burrows, Ivan Anishchenko, Ian R Humphreys, et al. Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384(6693):eadl2528, 2024.

[32] Ryan T Leenay, Amirali Aghazadeh, Joseph Hiatt, David Tse, Theodore L Roth, Ryan Apathy, Eric Shifrut, Judd F Hultquist, Nevan Krogan, Zhenqin Wu, et al. Large dataset enables prediction of repair after crispr–cas9 editing in primary t cells. *Nature biotechnology*, 37(9): 1034–1037, 2019.

[33] Hongzhao Li, Alexander Bello, Greg Smith, Dominic MS Kielich, James E Strong, and Bradley S Pickering. Degenerate sequence-based crispr diagnostic for crimean–congo hemorrhagic fever virus. *PLoS neglected tropical diseases*, 16(3):e0010285, 2022.

[34] Jianan Li, Wenxia Yu, Shisheng Huang, Susu Wu, Liping Li, Jiankui Zhou, Yu Cao, Xingxu Huang, and Yunbo Qiao. Upgraded adenine base editor (uabe) with minimized rna off-targeting activity. *Nature Portfolio*, 2020.

[35] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[36] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15:1–21, 2014.

[37] Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos Jr, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Large language models generate functional protein sequences across diverse families. *Nature biotechnology*, 41 (8):1099–1106, 2023.

[38] Kim F Marquart, Ahmed Allam, Sharan Janjuha, Anna Sintsova, Lukas Villiger, Nina Frey, Michael Krauthammer, and Gerald Schwank. Predicting base editing outcomes with an attention-based deep learning algorithm trained on high-throughput target library screens. *Nature communications*, 12(1):5114, 2021.

[39] Javier Mendoza-Revilla, Evan Trop, Liam Gonzalez, Maša Roller, Hugo Dalla-Torre, Bernardo P de Almeida, Guillaume Richard, Jonathan Caton, Nicolas Lopez Carranza, Marcin Skwark, et al. A foundational large language model for edible plant genomes. *Communications Biology*, 7(1):835, 2024.

[40] Shannon M Miller, Tina Wang, and David R Liu. Phage-assisted continuous and non-continuous evolution. *Nature protocols*, 15(12):4101–4127, 2020.

[41] John Moult, Krzysztof Fidelis, Andriy Kryshtafovych, Torsten Schwede, and Maya Topf. Critical assessment of techniques for protein structure prediction, fourteenth round. CASP 14 Abstract Book, 2020. URL `https://www.predictioncenter.org/casp14/doc/CASP14_Abstracts.pdf`.

[42] Eric Nguyen, Michael Poli, Matthew G Durrant, Armin W Thomas, Brian Kang, Jeremy Sullivan, Madelena Y Ng, Ashley Lewis, Aman Patel, Aaron Lou, et al. Sequence modeling and design from molecular to genome scale with evo. *BioRxiv*, pages 2024–02, 2024.

[43] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in neural information processing systems*, 36, 2024.

[44] Rahul Nikam, A Kulandaisamy, K Harini, Divya Sharma, and M Michael Gromiha. Prothermdb: thermodynamic database for proteins and mutants revisited after 15 years. *Nucleic acids research*, 49(D1):D420–D424, 2021.

[45] Pascal Notin, Aaron Kollasch, Daniel Ritter, Lood Van Niekerk, Steffanie Paul, Han Spinner, Nathan Rollins, Ada Shaw, Rose Orenbuch, Ruben Weitzman, et al. Proteingym: Large-scale benchmarks for protein fitness prediction and design. *Advances in Neural Information Processing Systems*, 36, 2024.

[46] Carlos Outeiral and Charlotte M Deane. Codon language embeddings provide strong signals for use in protein engineering. *Nature Machine Intelligence*, 6(2):170–179, 2024.

[47] Rafael Josip Penić, Tin Vlašić, Roland G Huber, Yue Wan, and Mile Šikić. Rinalmo: General-purpose rna language models can generalize well on structure prediction tasks. *arXiv preprint arXiv:2403.00043*, 2024.

[48] Ramiro Martin Perrotta, Svenja Vinke, Raphael Ferreira, Michael Moret, Ahmed Mahas, Anush Chiappino-Pepe, Lisa Maria Riedmayr, Louisa Lehmann, Anna-Therese Mehra, and George Church. Machine learning and directed evolution of base editing enzymes. *bioRxiv*, pages 2024–05, 2024.

[49] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with tape. *Advances in neural information processing systems*, 32, 2019.

[50] Michelle F Richter, Kevin T Zhao, Elliot Eton, Audrone Lapinaite, Gregory A Newby, B W Thuronyi, Christopher Wilson, Luke W Koblan, Jing Zeng, Daniel E Bauer, et al. Phage-assisted evolution of an adenine base editor with improved cas domain compatibility and activity. *Nature biotechnology*, 38(7):883–891, 2020.

[51] Adam J Riesselman, John B Ingraham, and Debora S Marks. Deep generative models of genetic variation capture the effects of mutations. *Nature methods*, 15(10):816–822, 2018.

[52] Mark D Robinson and Alicia Oshlack. A scaling normalization method for differential expression analysis of rna-seq data. *Genome biology*, 11:1–9, 2010.

[53] Philip Sedgwick. Spearman's rank correlation coefficient. *Bmj*, 349, 2014.

[54] Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. Saprot: Protein language modeling with structure-aware vocabulary. *bioRxiv*, pages 2023–10, 2023.

[55] Shunsuke Sumi, Michiaki Hamada, and Hirohide Saito. Deep generative design of rna family sequences. *Nature Methods*, 21(3):435–443, 2024.

[56] Francisco J Sánchez-Rivera, Bianca J Diaz, Edward R Kastenhuber, Henri Schmidt, Alyna Katti, Margaret Kennedy, Vincent Tem, Yu-Jui Ho, Josef Leibold, Stella V Paffenholz, et al. Base editing sensor libraries for high-throughput engineering and functional analysis of cancer-associated single nucleotide variants. *Nature biotechnology*, 40(6):862–873, 2022.

[57] Tianxiang Tu, Zongming Song, Xiaoyu Liu, Shengxing Wang, Xiaoxue He, Haitao Xi, Jiahua Wang, Tong Yan, Haoran Chen, Zhenwu Zhang, et al. A precise and efficient adenine base editor. *Molecular Therapy*, 30(9):2933–2941, 2022.

[58] Yann Vander Meersche, Gabriel Cretin, Aria Gheeraert, Jean-Christophe Gelly, and Tatiana Galochkina. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research*, 52(D1):D384–D392, 2024.

[59] Günter P Wagner, Koryu Kin, and Vincent J Lynch. Measurement of mrna abundance using rna-seq data: Rpkm measure is inconsistent among samples. *Theory in biosciences*, 131: 281–285, 2012.

[60] Ning Wang, Jiang Bian, Yuchen Li, Xuhong Li, Shahid Mumtaz, Linghe Kong, and Haoyi Xiong. Multi-purpose rna language modelling with motif-aware pretraining and type-guided fine-tuning. *Nature Machine Intelligence*, pages 1–10, 2024.

[61] Xi Wang, Ruichu Gu, Zhiyuan Chen, Yongge Li, Xiaohong Ji, Guolin Ke, and Han Wen. Uni-rna: universal pre-trained models revolutionize rna research. *bioRxiv*, pages 2023–07, 2023.

[62] Jacob West-Roberts, Joshua Kravitz, Nishant Jha, Andre Cornman, and Yunha Hwang. Diverse genomic embedding benchmark for functional evaluation across the tree of life. *bioRxiv*, pages 2024–07, 2024.

[63] Yang Wu, Masayuki Mukunoki, Takuya Funatomi, Michihiko Minoh, and Shihong Lao. Optimizing mean reciprocal rank for person re-identification. In *2011 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 408–413. IEEE, 2011.

[64] Xi Xiang, Giulia I Corsi, Christian Anthon, Kunli Qu, Xiaoguang Pan, Xue Liang, Peng Han, Zhanying Dong, Lijun Liu, Jiayan Zhong, et al. Enhancing crispr-cas9 grna efficiency prediction by data integration and deep learning. *Nature communications*, 12(1):3238, 2021.

[65] Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Ma Chang, Runcheng Liu, and Jian Tang. Peer: a comprehensive and multi-task benchmark for protein sequence understanding. *Advances in Neural Information Processing Systems*, 35:35156–35173, 2022.

[66] Jifang Yan, Dongyu Xue, Guohui Chuai, Yuli Gao, Gongchen Zhang, and Qi Liu. Benchmarking and integrating genome-wide crispr off-target detection and prediction. *Nucleic acids research*, 48(20):11370–11379, 2020.

[67] Heng Yang and Ke Li. Omnigenome: Aligning rna sequences with secondary structures in genomic foundation models. *arXiv preprint arXiv:2407.11242*, 2024.

[68] Yuning Yang, Gen Li, Kuan Pang, Wuxinhao Cao, Xiangtao Li, and Zhaolei Zhang. Deciphering 3'utr mediated gene regulation using interpretable deep representation learning. *Advanced Science*, page 2407013, 2023.

[69] Fei Ye, Zaixiang Zheng, Dongyu Xue, Yuning Shen, Lihao Wang, Yiming Ma, Yan Wang, Xinyou Wang, Xiangxin Zhou, and Quanquan Gu. Proteinbench: A holistic evaluation of protein foundation models. *arXiv preprint arXiv:2409.06744*, 2024.

[70] Weijie Yin, Zhaoyu Zhang, Liang He, Rui Jiang, Shuo Zhang, Gan Liu, Xuegong Zhang, Tao Qin, and Zhen Xie. Ernie-rna: An rna language model with structure-enhanced representations. *bioRxiv*, pages 2024–03, 2024.

[71] Emily Zhang, Monica E Neugebauer, Nicholas A Krasnow, and David R Liu. Phage-assisted evolution of highly active cytosine base editors with enhanced selectivity and minimal sequence context preference. *Nature Communications*, 15(1):1697, 2024.

[72] Yikun Zhang, Mei Lang, Jiuhong Jiang, Zhiqiang Gao, Fan Xu, Thomas Litfin, Ke Chen, Jaswinder Singh, Xiansong Huang, Guoli Song, et al. Multiple sequence alignment-based rna language model and its application to structural inference. *Nucleic Acids Research*, 52(1):e3–e3, 2024.

[73] Naihui Zhou, Yuxiang Jiang, Timothy R Bergquist, Alexandra J Lee, Balint Z Kacsoh, Alex W Crocker, Kimberley A Lewis, George Georghiou, Huy N Nguyen, Md Nafiz Hamid, et al. The cafa challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome biology*, 20:1–23, 2019.

[74] Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.

[75] Zhihan Zhou, Weimin Wu, Harrison Ho, Jiayi Wang, Lizhen Shi, Ramana V Davuluri, Zhong Wang, and Han Liu. Dnabert-s: Learning species-aware dna embedding with genome foundation models. *ArXiv*, 2024.

[76] Maxim Zvyagin, Alexander Brace, Kyle Hippe, Yuntian Deng, Bin Zhang, Cindy Orozco Bohorquez, Austin Clyde, Bharat Kale, Danilo Perez-Rivera, Heng Ma, et al. Genslms: Genome-scale language models reveal sars-cov-2 evolutionary dynamics. *The International Journal of High Performance Computing Applications*, 37(6):683–705, 2023.

Appendix

In Appendix B we introduce PANCE, our wet-lab data-collection framework. Appendix C supplies additional experimental settings and results. Finally, Appendix D describes the ranking track for a subset of TadABench-1M, provided only to showcase further results.

A Preliminary

TadA Gene editing is revolutionizing genetic disease treatment, agriculture, and personalized medicine. Among gene editing approaches, base editing provides a safer alternative to traditional CRISPR methods [12, 26] by enabling single-nucleotide conversions without inducing double-strand breaks, thereby enhancing editing precision [30, 21]. Base editors are divided into cytosine base editors, which convert C–G to T–A, and adenine base editors, which convert A–T to G–C. ABEs employ engineered variants of tRNA-specific adenosine deaminase (TadA) to catalyze adenosine-to-inosine (A-to-I) deamination, interpreted as guanine during DNA replication. TadA8e [50] is a high-efficiency variant commonly used for precise A-to-G editing and consists of 167 amino acids (501 nucleotides). Our TadABench-1M is built on TadA8e variants and encompasses large-scale mutagenesis, introducing over 25 amino-acid mutations (exceeding 150 nucleotide mutations).

PANCE Phage-Assisted Non-Continuous Evolution (PANCE) [40, 71] is a directed evolution strategy that employs discrete manual dilutions of bacteriophage populations to impose controlled selection pressures, thereby enabling the rapid evolution of proteins with desired traits. In our implementation, phages encoding TadA variants with enhanced activity outcompete less active variants, leading to their preferential enrichment and higher read counts in next-generation sequencing (NGS) of the final populations (see Appendix B). Our TadABench-1M includes 31 rounds of PANCE, each comprising an independent evolution, where each corresponds to a distinct TadA variant library.

Degenerate Sequence A degenerate sequence [33] is a DNA sequence that includes ambiguous nucleotide codes at defined positions, allowing a single sequence to represent tens of thousands of distinct variants simultaneously. Although PANCE provides automatic activity labeling via competition among TadA variants, constructing such expansive libraries through individual sequences is resource- and time-intensive. To address this, we employ degenerate sequence synthesis, in which one degenerate sequence generates a vast variant library at the cost of a single DNA synthesis.

B Phage-Assisted Non-Continuous Evolution (PANCE)

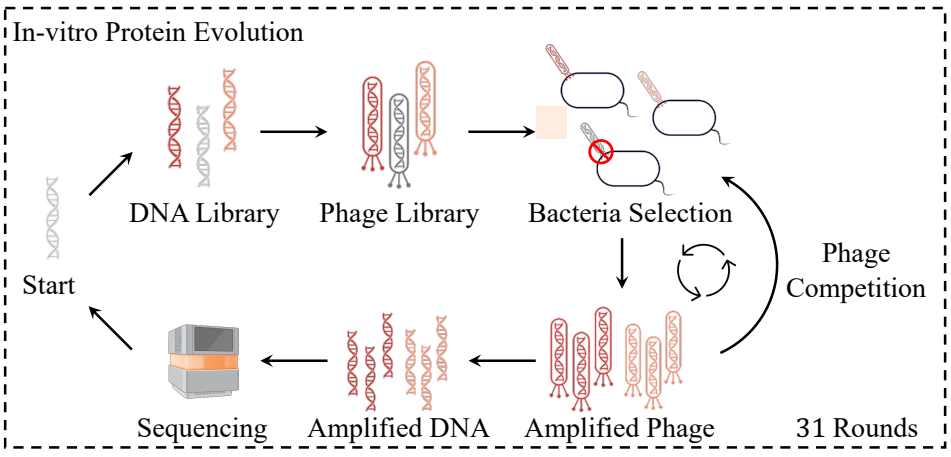

Figure 5: Overview of the PANCE workflow used to obtain TadA activity data. A large library of TadA mutants is screened by an AI predictor, and high-activity candidates are selected for phage-assisted evolution. Variants with higher activity trigger gIII expression, leading to phage propagation.

Phage-Assisted Non-Continuous Evolution (PANCE) [40, 71] represents a sophisticated platform for the directed evolution of biomolecules. This methodology builds upon the principles of Darwinian evolution and leverages the powerful selection capabilities of bacteriophages. The strength of this approach lies in its high-throughput ability to identify the highest-activity protein variants from vast AI-generated starting sequences. In our study, we employed PANCE to evolve TadA, a critical enzyme used in CRISPR base editing, by systematically selecting variants with enhanced activity from vast AI-generated libraries. The convergence of advanced artificial intelligence for library design and PANCE for evolutionary selection represents a frontier in protein engineering, offering a high-throughput and scalable approach to optimize enzymatic functions.

The core of this method is selecting protein variants with improved activity by coupling their function to the replication of bacteriophages. Phages lacking a key gene required for propagation are engineered to rely on the activity of the target protein within host cells to trigger their replication. Through iterative rounds of serial dilution, phages linked to protein variants with higher activity maintain their population, while low-activity counterparts are washed out, allowing for the gradual enrichment of high-performance variants.

We engineered the M13 phage, a filamentous virus that propagates within Escherichia coli (E. coli), to lack the essential gene gIII, which encodes the phage protein pIII, responsible for facilitating the release of new virions from the host. The expression of gIII was made contingent on the activity of TadA within E. coli, such that TadA variants with sufficient activity would trigger gIII expression, enabling phage replication. Each round of PANCE involved the serial dilution of bacterial cultures. Over multiple cycles, variants with superior activity outcompeted their lower-performing counterparts, resulting in a highly refined population of phage-encoded TadA variants. This iterative process ensures that even minimal gains in activity are captured and amplified across generations, gradually evolving TadA to a high-performance state. Finally, we observe the activity of different TadA variants by analyzing the read counts from the NGS sequencing results of lysed Escherichia coli cultures. We are unable to release the detailed protocols due to company licensing restrictions.

Figure 6: Visualization of TadABench-1M. The left part shows the folded structure of TadA obtained from our wet-lab experiments, predicted using ESMFold [35]. The right part presents a t-SNE visualization of the clustering results across 31 rounds of protein evolution.

C Details of TadABench-1M

C.1 Visualization

In summary, we propose an algorithm, Seq2Graph, to construct our dataset TadABench-1M based on the 31 rounds of NGS sequencing results. We construct a graph reflecting the relative activity relationships among sequence variants based on their read counts in the NGS data. To prevent conflicts in activity relationships, we apply a fast heuristic algorithm to eliminate inconsistencies. Finally, we assign each variant a relative activity value by incorporating the biological principle of exponential growth in the breadth-first order to preserve the most confident assignment strategy.

We provide the visualization of TadABench-1M in Figure 6. The left protein structure is one variant in our TadABench-1M, folded using ESMFold [35]. To demonstrate the difference in our 31 rounds of protein evolution, we select 50 sequences from each round and show the t-SNE result on the right. Variants from diverse rounds have notable differences at the sequence level. We provide a rough tendency of sequence similarity across experimental rounds in the appendix (Figure 7).

C.2 Similarity across Rouds

Figure 7 depicts the sequence similarity dynamics across evolutionary rounds. We present a similarity matrix capturing the pairwise average sequence similarity between TadA protein variants across 31 rounds of directed evolution in our TadABench-1M dataset. Each matrix entry reflects the average sequence similarity between all pairs of sequences from two rounds, normalized between 0 and 1. Notably, distinct blocks of higher intra-round similarity (diagonal) and variable inter-round similarities highlight the heterogeneous nature of sequence evolution. These observations underline the diverse mutational trajectories during the evolutionary process and provide a quantitative foundation for benchmarking protein sequence models under varying evolutionary pressures.

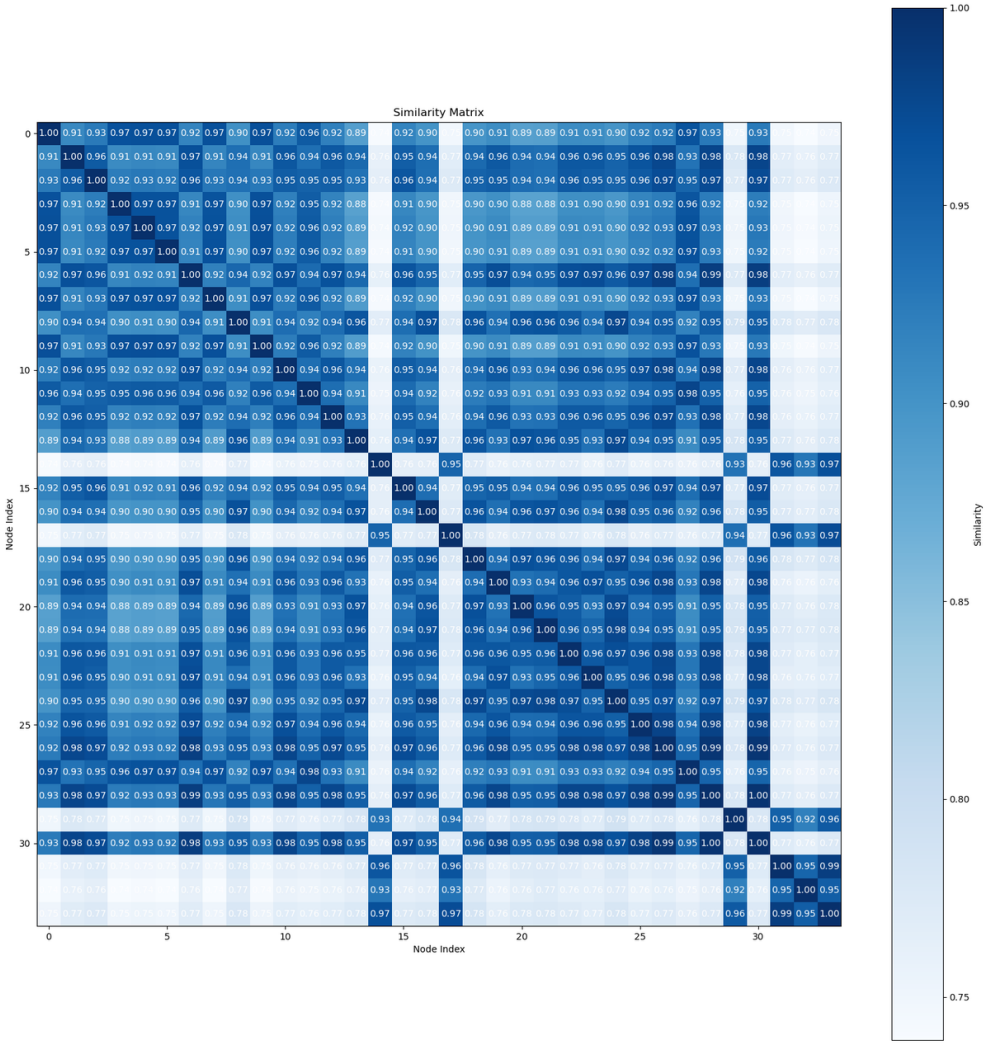

Figure 7: We show the overall sequence similarity matrix of each round. TadABench-1M contains TadA variants from 31 rounds of protein evolution. Owing to the application, the sequence similarity differs by rounds. The average of the sequence similarities is reported for each pair of rounds.

C.3 Experimental Settings

Dataset TadABench-1M is curated from NGS data of evolved TadA variants, yielding both DNA/RNA and protein sequence datasets (Section 3.3). For both modalities, we emulate a practical protein engineering scenario by splitting the data chronologically: rounds 1–27 for training, round 28 for validation, and rounds 29–31 for testing. The nucleic acid (DNA/RNA) dataset contains 729,302 training sequences, 148,014 validation sequences, and 149,884 test sequences. The protein dataset comprises 256,429 training sequences, 45,208 validation sequences, and 108,232 test sequences.

Task and Evaluation While the training labels correspond to activity values, TadABench-1M 's objective is not pure regression. Instead, the task focuses on predicting relative activity trends across test variants, reflecting realistic protein engineering workflows. We evaluate models using Spearman's rank correlation coefficient, normalized Discounted Cumulative Gain at the top 10% (nDCG@10%), and Recall@10%, adopting evaluation protocols from prior work [45]. Spearman's coefficient captures the overall relative ordering of activities across test sequences. Recall@10% measures the fraction of true top 10% variants correctly identified in the top 10% of predicted scores.

Complementing these, nDCG@10% assesses whether the predicted top variants are correctly ranked by activity within the top decile (see Appendix D.1 for details).

Biological Language Models We evaluate a diverse set of pre-trained biological language models (BLMs) spanning DNA, RNA, and protein domains. For DNA, we use models from the EVO2 [3] and NucleotideTransformer (NT)[13] families. For RNA, we include the OmniGenome [67] family. For protein, we consider ESM2 [35], ProtTrans [17], and ESMC [18] families. We utilize DNA and RNA language models in the same manner by mapping T to U. The Evo2 40B model is accessed through the API[2], while other models are deployed by the NVIDIA GeForce RTX 4090 GPU.

Hyperparameters We extract representations from the final layer (or logits for EVO2 models) and perform linear probing to efficiently assess the encoded biological knowledge. Due to varying representation dimensions across models, the regression head's parameter count differs substantially. To ensure fair comparison, we employ a two-layer MLP with hidden layer sizes tailored per model for an equal number of trainable parameters, with the ReLU activation between layers. For the Evo2 family, we first take all the logits as input. However, the training is relatively unstable. Therefore, we use normalization on embedding-based models and only take the A/T/C/G four dimensions of logits as representations, largely facilitating the training stability. Besides, we keep the representations of all tokens to conserve more information, which has a similar performance to the average on tokens in our setting. Each head is trained for 20 epochs with a cosine learning rate scheduler and a 1-epoch warmup. We evaluate three learning rates (3e-5, 1e-4, 3e-4) and select the best-performing based on validation performance.

C.4 Performance of Random Split

We benchmark protein language models on the random split of TadABench-1M (protein version) in Table 3. We evaluate a range of protein language models using an 8:1:1 split for training, validation, and test sets. Performance is reported for the best result over three learning rates per model. Metrics include Spearman correlation and ranking-based metrics (Recall@10% and nDCG@10%) on both validation and test sets. ESMC-600M achieves the highest scores across most evaluation criteria, including the best Spearman correlation (0.8079), and competitive Recall@10% (0.2317)nDCG@10% (0.4949) on the test set, demonstrating superior ranking and correlation performance in protein variant prediction. Notably, smaller models such as ESM2-35M show competitive performance on correlation metrics, but lag in ranking-based retrieval.

All biological language models (BLMs) achieve a Spearman correlation coefficient of approximately $\rho \approx 0.8$ on both the validation and test sets. Their Recall@10% exceeds 0.2, which is significantly higher than the 0.1 level observed under the practical-scenario split. This indicates that BLMs effectively capture the i.i.d. pattern in this setting, in stark contrast to the results shown in Table 2. These findings suggest that the dataset is indeed learnable and that the underlying sequence patterns are detectable under idealized conditions. Accordingly, the poor performance of BLMs in real-world scenarios arises from a substantial gap between standard evaluation setups and the demands of real-world applications.

D Ranking Task of TadABench-1M

Apart from the experiments in TadABench-1M mentioned in the main text, we also run more experiments on a part of our dataset, TadABench-100K. Considering the practical application of protein engineering needs relative activity comparison among massive candidates, we support ranking-based evaluation to examine model performance without relying on the absolute activity scales. Specifically, we extract lists of sequences sorted from their activities and require the model to make correct rankings. In the evaluation process, we similarly use embeddings generated by BLMs and train a head module to predict the activity ranking of different sequence variants.

[2]https://build.nvidia.com/arc/evo2-40b

Table 3: Performance on TadABench-1M (protein version). The training, validation, and test sets were obtained via an 8:1:1 random split. For each model, the best result is selected from three different learning rates. **Bold** numbers indicate the highest performance.

Model	Validation			Test		
	Spearman	Recall@10%	nDCG@10%	Spearman	Recall@10%	nDCG@10%
ESM2-35M	0.8032	0.1830	0.4824	0.8014	0.1617	0.4814
ESM2-150M	0.7386	0.2290	0.4364	0.7371	0.2324	0.4437
ESM2-650M	0.5360	0.1793	0.4740	0.5348	0.1710	0.4779
Prot-BERT	0.7910	0.2230	0.4879	0.7883	0.2262	0.4918
Prot-XLNET	0.8054	0.2264	0.4912	0.8030	0.2193	0.4965
ESMC-300M	0.8102	0.2439	0.4959	0.8067	**0.2363**	**0.4995**
ESMC-600M	**0.8127**	**0.2446**	**0.5006**	**0.8079**	0.2317	0.4949

D.1 Evaluation Metric: nDCG

Normalized Discounted Cumulative Gain (nDCG) is a commonly used metric to evaluate the ranking quality of algorithms, particularly in information retrieval and recommendation systems [27]. It focuses on both the relevance of the ranked items and the position of these items in the ranking list. The relevance score of each item is assigned based on its importance or utility to the user. The gain is discounted logarithmically as the rank increases, meaning that highly relevant items appearing earlier in the ranking list contribute more to the overall score.

The nDCG is normalized by dividing the DCG of the actual ranking by the DCG of the ideal ranking (IDCG), ensuring the score falls within the range of 0 to 1. The DCG (Discounted Cumulative Gain) is calculated as:

$$\text{DCG}_p = \sum_{i=1}^{p} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)} \tag{2}$$

where p represents the position in the ranking (typically the top p items are evaluated) and i is the rank of the item in the list. The rel_i is the relevance score of the item at position i, which is the reverse ranking in our setting, *i.e.*, the ranking list $1, 2, 3, \ldots$ with a length of N has the rel_i as $N, N-1, N-2, \ldots$. The $\log_2(i+1)$ is A logarithmic discounting factor that reduces the contribution of lower-ranked items.

The normalized version, nDCG, is calculated as:

$$\text{nDCG}_p = \frac{\text{DCG}_p}{\text{IDCG}_p} \tag{3}$$

where IDCG_p is the ideal DCG for a perfect ranking.

The nDCG is especially valuable for evaluating ranked retrieval systems because it accounts for the importance of the placement of relevant items within the list. This metric assigns greater weight to items at higher-ranked positions, ensuring that the ranking system's effectiveness is measured more accurately by prioritizing top results, which are typically more relevant to the user. It is particularly suitable for our task of ranking protein activities because we focus more on the top-ranked proteins.

D.2 Experimental Results

On our TadABench-100K, a subset of TadABench-1M, we conduct additional experiments under the ranking-based setting, which reduces the impact of round-specific wet experiment noise. Even in this more controlled scenario, we observed that models still struggle to predict the next round's outcomes based on data from the previous round, which further highlights the inherent difficulty and significance of this real-world task. We begin by introducing the models, data, and experimental settings used in the ranking-based experiments. Following the structure of the main paper, we first demonstrate the learnability of the dataset using randomly split data and highlight several properties of BLMs. We then present results under the more realistic evolutionary round setting, where BLMs still fall short in fully capturing application-level protein evolution tasks.

Table 4: **Diverse BLMs are evaluated using linear probing in the ranking-based and random data split setting.** The top, middle, and bottom groups are protein, DNA, and RNA BLMs. * indicates that a smaller batch size is employed due to the large size of the embeddings.

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
One-hot	0.826	0.764	0.058	0.820	0.322	0.079	0.854	0.057	-0.009
Chai1	0.847	0.792	0.169	0.857	0.322	0.194	0.900	0.095	0.138
ESM2	0.831	0.771	0.082	0.844	0.322	0.175	0.907	0.050	0.252
ESM3	0.840	0.783	0.133	0.860	0.335	0.214	0.892	0.100	0.103
RFAA	0.838	0.780	0.120	0.858	0.323	0.205	0.890	0.050	0.158
SaProt	0.831	0.771	0.083	0.839	0.322	0.144	0.864	0.042	0.085
LucaOne	0.830	0.770	0.078	0.839	0.313	0.146	0.901	0.029	0.218
ProtTrans	0.831	0.771	0.085	0.844	0.325	0.172	0.886	0.038	0.185
One-hot	0.819	0.754	0.017	0.822	0.281	0.072	0.854	0.052	0.027
NT	0.836	0.777	0.109	0.845	0.307	0.180	0.884	0.030	0.182
EVO*	0.830	0.770	0.080	0.850	0.317	0.190	0.895	0.007	0.153
Chai1	0.848	0.794	0.175	0.868	0.322	0.224	0.901	0.086	0.222
AgroNT	0.831	0.772	0.086	0.839	0.304	0.156	0.868	0.096	0.123
GenSLM*	0.836	0.777	0.109	0.857	0.327	0.204	0.904	0.043	0.233
LucaOne*	0.835	0.776	0.106	0.843	0.303	0.165	0.888	0.037	0.165
HyenaDNA	0.831	0.771	0.085	0.848	0.314	0.178	0.883	0.029	0.132
DNABERT-2	0.816	0.750	0.001	0.814	0.295	0.038	0.861	0.026	0.018
DNABERT-S	0.817	0.752	0.007	0.812	0.301	0.028	0.853	0.040	0.017
DNABERT-1	0.835	0.776	0.105	0.845	0.299	0.163	0.893	0.075	0.236
One-hot	0.819	0.754	0.017	0.822	0.281	0.072	0.854	0.052	0.027
Chai1	0.845	0.790	0.161	0.867	0.316	0.225	0.897	0.124	0.217
CaLM	0.834	0.775	0.099	0.847	0.309	0.178	0.882	0.062	0.146
RNA-FM	0.830	0.770	0.079	0.846	0.315	0.187	0.880	0.026	0.117
RiNALMo*	0.843	0.787	0.148	0.870	0.326	0.235	0.904	0.049	0.198
RNAErnie*	0.837	0.780	0.119	0.867	0.326	0.230	0.906	0.056	0.237
RNA-MSM	0.832	0.773	0.090	0.850	0.317	0.197	0.902	0.045	0.253
SpliceBERT	0.833	0.774	0.095	0.844	0.308	0.167	0.890	0.051	0.203
3UTRBERT*	0.840	0.784	0.135	0.870	0.324	0.244	0.908	0.044	0.256
ERNIE-RNA*	0.836	0.778	0.113	0.860	0.327	0.230	0.909	0.041	0.213
OmniGenome*	0.838	0.781	0.122	0.868	0.320	0.239	0.910	0.059	0.207

Model The evaluation is based on protein, DNA, and RNA modalities, since these 3 forms play important roles in the natural transcription and translation process, and all contain important information. In TadABench-1M, DNA sequences obtained from biological sequencing data are translated into RNA and protein sequences according to biological principles. These transformed sequences are inputs for the corresponding biological language models (BLMs).

As for protein modality, we test the ESM2 [35], ESM3 [24], ProtTrans [17], SaProt [54], and RFAA [31]. Our DNA modality evaluation involves the EVO [42], NucleotideTansformer (NT) [13], AgroNT [39], GenSLMs [76], HyenaDNA [43], DNABERT-1 [28], DNABERT-2 [74], and DNABERT-S [75]. For RNA modality, TadABench-1M tests the RNA-FM [8], SpliceBERT [9], 3UTRBERT [68], OmniGenome [67], CaLM [46], ERNIE-RNA [70], RNAErnie [60], RNA-MSM [72], and RiNALMo [47]. We also include LucaOne [25] and Chai1 [6] as representatives of multimodal BLMs, reflecting the popular concept of multimodality in the foundation models domain.

Overall, we test 80 models across 24 papers [3]. For linear probing, we use multimodal BLMs such as LucaOne and Chai1 to tackle three modalities of input sequence input independently, referred to as three BLMs for convenience. We extract features of the last trunk for folding models such as Chai1 and RoseTTAFold-All-Atom. Owing to space limitations, we only report one model for each paper in Table 4.

[3]There are some other BLMs that we do not include, such as Atom-1 [2], UNI-RNA [61], and RFamGen [55], since their codebases or model weights have not been released.

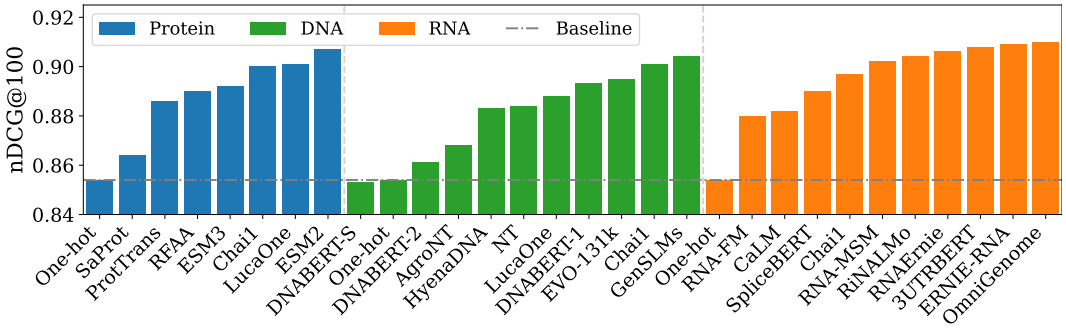

Figure 8: **BLMs using different modalities perform comparably using linear probing on the @100 ranking track.** The performance gap between the 3 modalities of BLMs is not obvious, which means the knowledge of DNA and RNA BLMs is also important in the protein evolution task.

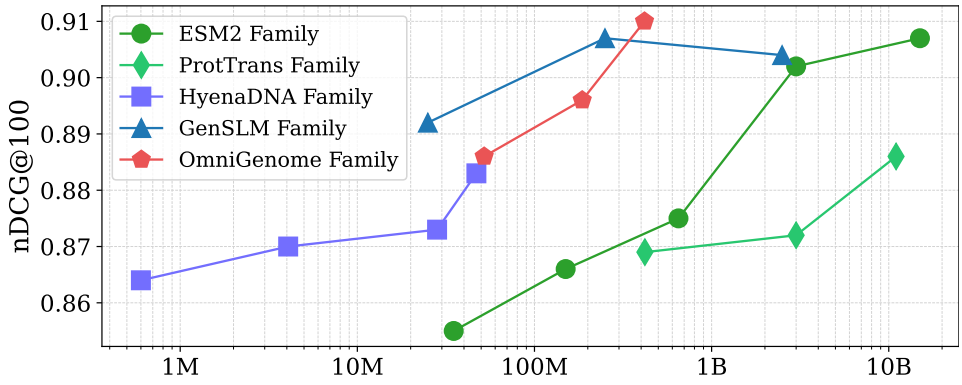

Figure 9: **The scaling law behavior is demonstrated in the ranking-based task for selected BLM families among three modalities.** We select BLM families across three modalities, protein, DNA, and RNA. The x-axis represents the parameter number and the y-axis reflects the nDCG@100 score.

Model and Data Consistent with our main paper, our evaluation spans a diverse set of BLMs, including models specialized for DNA, RNA, and protein modalities, along with several multimodal architectures. We take a 3-layer fully connected network with a hidden size of 128 as the head module, using a cross-entropy ListNet loss [5]. We adopt linear probing and fine-tuning to evaluate the performance of various BLMs without introducing complexity. We experiment on TadABench-100K, a subset of TadABench-1M, comprising 100,000 sequences with annotated activities.

Ranking Metrics We offer three tracks, @2, @10, and @100 for different ranking lists with corresponding lengths. We adopt three common ranking evaluation metrics to assess the effectiveness of the predicted rankings within a population of size x, normalized discounted cumulative gain (nDCG@x) [27], mean Reciprocal Rank (mRR@x) [63], and Spearman's Rank Correlation (SP@x) [53]. The nDCG measures the accuracy of ranking results, with greater emphasis placed on higher-ranked items. The mRR focuses exclusively on the accuracy of predictions for the top-ranked sample. SP evaluates the predicted rankings' overall distribution.

D.2.1 Random Data Split

Consistent with our main paper, we randomly shuffle the data from the first and second rounds of evolution and split it into training and test sets using a standard 7:3 ratio. Each data point is a list of x sequences, and the objective is to predict their correct activity ranking. In Table 4, we present the activity ranking prediction results for a subset of BLMs, with ranking list lengths of 2, 10, and 100.

Here, we primarily take the linear probing with a batch size of 64, freezing the parameters of BLMs, and using the output embeddings to train head modules. Given the influence of different embedding

Table 5: **Under the random data split setting, BLMs perform well on the ranking-based task using linear probing and fine-tuning.** The table shows the result of @100 track with a batch size of 1×100 sequences. For most selected models except 3UTRBERT-6mer, fine-tuning provides better results than linear probing.

Modality	Model	Random Initialization		Linear Probing		Fine-tuning	
		nDCG↑	SP↑	nDCG↑	SP↑	nDCG↑	SP↑
Protein	ESM2-650M	0.844	-0.050	0.875	0.138	0.902	0.208
	ESM2-150M	0.856	0.050	0.866	0.075	0.898	0.187
DNA	DNABERT-1-6mer	0.856	0.017	0.893	0.236	0.908	0.226
	HyenaDNA-T-d256	0.865	0.053	0.886	0.200	0.908	0.205
RNA	RNA-Ernie	0.855	0.003	0.906	0.237	0.907	0.239
	3UTRBERT-6mer	0.836	-0.015	0.908	0.256	0.902	0.210

lengths on the learning rate, we specify 3 learning rates for each experiment, 1e-5, 1e-4, and 1e-3, and choose the optimal result as its reported result. We use one-hot vectors of the sequences as the baseline to compare with the embeddings of BLMs.

Under the ranking-based setting, we observe conclusions consistent with those reported in our main paper. Compared to training classification heads directly using sequence one-hot vectors, using embeddings extracted from pre-trained BLMs significantly enhances the test performance. This demonstrates that BLMs are well-suited for protein activity prediction tasks on TadABench-1M, aligning with experiences in the language model field. The complete evaluation of 80 models can be found in Tables 6 to 8. We have also fine-tuned the BLMs (as shown in Table 5), which further improves performance. This aligns well with a general understanding of language models, while it is not the main focus of this paper.

Modality DNA and RNA BLMs demonstrate performance comparable to protein BLMs on nDCG@100, as shown in Figure 8. It demonstrates that the nucleotide BLMs also gain knowledge about protein functionality on DNA or RNA sequences. Since proteins, DNA, and RNA fundamentally form an integrated system within organisms and each plays a crucial role in protein expression, models across all three modalities significantly outperform those trained on one-hot vectors of the sequences.

Additionally, the performance of multimodal models is consistent with the conclusions reported in the main paper. LucaOne achieves nDCG@100 scores of 0.901 for protein and 0.888 for nucleotide, whereas Chai1 attains scores of 0.900 and 0.897, respectively. These results indicate that Chai1 achieves better modality unification, as its performance across modalities is more consistent, while LucaOne shows a notable advantage in protein performance over nucleotide.

D.2.2 Scaling Law

The Scaling Law behavior of BLMs is also observed under the ranking-based setting. We observe that most BLM model families in Figure 9 demonstrate the scaling law.

D.2.3 K-mer

K-mer in BLMs sequence of k consecutive nucleotides used to capture local sequence patterns and the context in biological modeling analysis. 3UTRBERT is an RNA BLM model family composed of different k-mer models. Considering the test nDCG@10 in in-domain ranking, the results for 6-mer, 5-mer, 4-mer, and 3-mer are respectively 0.870, 0.860, 0.869, and 0.870. We observe that the results for 3-mer and 6-mer are higher than those for 4-mer and 5-mer. In biological terms, a protein is encoded by three nucleotides, demonstrating that TadABench-1M aligns well with the actual biological k-mer patterns. It also indicates that RNA BLMs are significant in protein-related tasks, provided that an appropriate k-mer is selected.

Table 6: **Evaluation on protein BLMs using the linear probing for random-data-split ranking.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
ESM2-8M	0.828	0.767	0.069	0.836	0.321	0.136	0.874	0.053	0.176
ESM2-35M	0.829	0.769	0.074	0.833	0.314	0.133	0.855	0.048	0.012
ESM2-150M	0.828	0.768	0.070	0.839	0.322	0.154	0.866	0.084	0.075
ESM2-650M	0.830	0.770	0.080	0.840	0.324	0.162	0.875	0.075	0.138
ESM2-3B	0.832	0.772	0.090	0.842	0.318	0.163	0.902	0.046	0.222
ESM2-15B	0.831	0.771	0.082	0.844	0.322	0.175	0.907	0.050	0.252
ESM3	0.840	0.783	0.133	0.860	0.335	0.214	0.892	0.100	0.103
SaProt-650M-AF2	0.828	0.766	0.066	0.837	0.307	0.137	0.862	0.034	0.067
SaProt-650M-PDB	0.831	0.771	0.083	0.839	0.322	0.144	0.864	0.042	0.085
SaProt-35M-AF2	0.832	0.773	0.091	0.835	0.311	0.134	0.877	0.069	0.144
SaProt-35M-AF2-Seq	0.830	0.769	0.077	0.837	0.323	0.143	0.870	0.058	0.066
LucaOne	0.830	0.770	0.078	0.839	0.313	0.146	0.901	0.029	0.218
RosettaFold-STATE	0.823	0.760	0.040	0.817	0.299	0.058	0.851	0.078	0.010
RosettaFold-MSA	0.838	0.780	0.120	0.858	0.323	0.205	0.890	0.050	0.158
ProstT5	0.827	0.766	0.064	0.842	0.320	0.156	0.865	0.087	0.081
ProstT5-fp16	0.827	0.765	0.060	0.840	0.329	0.166	0.872	0.052	0.147
Prot-T5-XL-U50	0.832	0.773	0.090	0.835	0.320	0.144	0.880	0.053	0.160
Prot-T5-XL-Half	0.834	0.775	0.102	0.835	0.309	0.143	0.868	0.048	0.098
Chai1	0.844	0.788	0.152	0.858	0.322	0.203	0.896	0.035	0.140
Chai1-ESM	0.847	0.792	0.169	0.857	0.322	0.194	0.900	0.095	0.138
Prot-Bert	0.824	0.761	0.046	0.828	0.303	0.098	0.871	0.068	0.134
Prot-ss3	0.823	0.760	0.039	0.825	0.301	0.084	0.867	0.030	0.050
Prot-Membrane	0.830	0.770	0.079	0.829	0.316	0.119	0.862	0.062	0.028
Prot-Localization	0.826	0.765	0.060	0.829	0.314	0.114	0.858	0.021	0.055
Prot-T5-XXL-U50	0.832	0.773	0.090	0.842	0.320	0.177	0.882	0.045	0.154
Prot-Generator	0.830	0.770	0.079	0.842	0.322	0.169	0.882	0.062	0.148
Prot-Discriminator	0.831	0.771	0.084	0.844	0.322	0.174	0.881	0.137	0.140
Prot-T5-XL-BFD	0.831	0.772	0.086	0.839	0.319	0.162	0.882	0.102	0.170
Prot-Bert-BFD	0.827	0.766	0.065	0.839	0.308	0.141	0.869	0.086	0.141
Prot-T5-XXL-BFD	0.831	0.771	0.085	0.844	0.325	0.172	0.886	0.038	0.185
Prot-Xlnet	0.830	0.769	0.077	0.833	0.314	0.141	0.880	0.060	0.110
Prot-Albert	0.831	0.771	0.086	0.836	0.311	0.132	0.883	0.049	0.105

Fine-tuned BLMs In the ranking-based task, we also report the fine-tuning performance under the random data split setting, shown in Table 5. Firstly, linear probing and fine-tuning effectively surpass the random init in nDCG@100 and SP. Secondly, fine-tuning provides better results than linear probing for most selected models except 3UTRBERT-6mer. Thirdly, SP and n@DCG can provide different tendencies, demonstrating the different concentrations for distinct metrics, shown in Appendix D.2. For example, the nDCG of linear probing in 3UTRBERT-6mer is higher than fine-tuning, while the SP is the opposite. The linear probing performs better at the top sequences, while the fine-tuning shows better rankings on 100 sequences.

Full Results of Random Data-Split Ranking-Based Task In this paragraph, we report the performance of a broad range of BLMs on the ranking-based task with randomly split data. In Tables 6 to 8, we report the performance of protein, RNA, and DNA BLMs separately.

D.2.4 Real-World Evolution Scenario

Consistent with our main paper, for the ranking-based task, we also assess the real-world cross-round evolution by training a linear probe on first-round evolution data and testing it on second-round data. The train-test data splitting is shown in Table 9.

As discussed in our main paper, although the real-world evolution setting is a challenging task, it aligns well with the underlying logic of real-world protein evolution processes. Despite extensive hyperparameter tuning and fine-tuning efforts, this setting remains highly challenging for BLMs.

Table 7: **Evaluation on RNA BLMs using the linear probing for random-data-split ranking.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
mRNA-FM	0.830	0.770	0.079	0.846	0.315	0.187	0.880	0.026	0.117
RNA-FM	0.831	0.771	0.084	0.838	0.293	0.146	0.879	0.026	0.161
RNA-MSM	0.832	0.773	0.090	0.850	0.317	0.197	0.902	0.045	0.253
RNA-Ernie	0.837	0.780	0.119	0.867	0.326	0.230	0.906	0.056	0.237
RiNaLMo	0.843	0.787	0.148	0.870	0.326	0.235	0.904	0.049	0.198
ERNIERNA	0.836	0.778	0.113	0.860	0.327	0.230	0.909	0.041	0.213
ERNIERNA.ss	0.837	0.779	0.115	0.860	0.323	0.226	0.906	0.031	0.234
Chai1	0.845	0.790	0.161	0.867	0.316	0.225	0.897	0.124	0.217
OmniGenome-418M	0.838	0.781	0.122	0.868	0.320	0.239	0.910	0.059	0.207
OmniGenome-186M	0.839	0.782	0.127	0.861	0.313	0.210	0.896	0.061	0.213
OmniGenome-52M	0.831	0.771	0.083	0.846	0.327	0.184	0.886	0.041	0.207
3UTRBERT-6mer	0.840	0.784	0.135	0.870	0.324	0.244	0.908	0.044	0.256
3UTRBERT-5mer	0.834	0.775	0.102	0.861	0.323	0.231	0.906	0.046	0.232
3UTRBERT-4mer	0.840	0.784	0.134	0.869	0.326	0.243	0.906	0.047	0.234
3UTRBERT-3mer	0.841	0.785	0.138	0.870	0.323	0.246	0.906	0.041	0.226
SpliceBERT	0.829	0.768	0.072	0.836	0.307	0.140	0.872	0.058	0.114
SpliceBERT-H.510nt	0.833	0.774	0.095	0.844	0.308	0.167	0.890	0.051	0.203
SpliceBERT.510nt	0.830	0.770	0.080	0.838	0.310	0.152	0.874	0.036	0.121
CaLM	0.834	0.775	0.099	0.847	0.309	0.178	0.882	0.062	0.146

This indicates that even in the ranking-based task designed to reduce wet experimental noise, model performance does not improve significantly, suggesting the presence of more fundamental limitations. We will report the experimental results in detail below.

Experimental Results We benchmark all of the BLMs using the linear probing approach here. The results are presented separately for protein, DNA, and RNA BLMs in Tables 10 to 12, respectively. Across all modalities, most BLMs perform poorly in the real-world evolution ranking task, with results barely surpassing those of random guess ranking, which is highly consistent with the conclusions presented in the main paper. This suggests that even under the ranking-based setting with lower wet-experiment noise, the models show limited improvement, pointing to deeper, underlying limitations.

This poor performance stands in stark contrast to the outcomes observed in the random-split setting, where nearly all BLMs achieve results consistent with expectations. These results confirm that the embeddings generated by BLMs are meaningful and effective in in-domain tasks, demonstrating no apparent issues related to the curse of dimensionality or loss of information during the embedding process. The disparity between in-domain ranking and out-of-domain ranking performance suggests that the challenges faced by BLMs in out-of-domain ranking are not due to the embeddings themselves but are likely attributed to the difficulty of generalizing to out-of-domain data. While the embeddings remain useful within the context of in-domain ranking tasks, their transferability and robustness across varying experimental conditions in out-of-domain ranking are limited. This emphasizes the need for more advanced strategies to enhance the generalization ability of BLMs when faced with out-of-domain ranking tasks.

Table 8: **Evaluation on DNA BLMs using the linear probing for random-data-split ranking.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
EVO-8k	0.829	0.769	0.075	0.851	0.308	0.183	0.891	0.045	0.153
EVO-131k	0.830	0.770	0.080	0.850	0.317	0.190	0.895	0.007	0.153
LucaOne	0.835	0.776	0.106	0.843	0.303	0.165	0.888	0.037	0.165
Chai1	0.848	0.794	0.175	0.868	0.322	0.224	0.901	0.086	0.222
NT-2-50M	0.833	0.774	0.097	0.845	0.305	0.179	0.879	0.057	0.137
NT-2-100M	0.836	0.777	0.109	0.843	0.306	0.171	0.872	0.068	0.118
NT-2-250M	0.830	0.770	0.078	0.845	0.312	0.171	0.866	0.081	0.088
NT-2-500M	0.834	0.776	0.102	0.849	0.313	0.202	0.880	0.027	0.176
NT-500M-human-ref	0.836	0.777	0.109	0.840	0.292	0.154	0.892	0.098	0.195
NT-500M-1000G	0.833	0.774	0.097	0.847	0.314	0.191	0.864	0.034	0.107
NT-2B5-1000G	0.836	0.777	0.109	0.845	0.307	0.180	0.884	0.030	0.182
NT-2B5-multi-species	0.828	0.767	0.069	0.838	0.291	0.145	0.872	0.035	0.110
AgroNT	0.831	0.772	0.086	0.839	0.304	0.156	0.868	0.096	0.123
GenSLMs 2.5B	0.836	0.777	0.109	0.857	0.327	0.204	0.904	0.043	0.233
GenSLMs 250M	0.836	0.777	0.109	0.856	0.326	0.204	0.907	0.040	0.249
GenSLMs 25M	0.831	0.771	0.084	0.837	0.322	0.160	0.892	0.072	0.178
DNABERT-2-117M	0.816	0.750	0.001	0.814	0.295	0.038	0.861	0.026	0.018
DNABERT-S	0.817	0.752	0.007	0.812	0.301	0.028	0.853	0.040	0.017
DNABERT-1-3mer	0.830	0.770	0.081	0.841	0.303	0.163	0.879	0.144	0.085
DNABERT-1-4mer	0.830	0.770	0.080	0.836	0.299	0.138	0.872	0.043	0.089
DNABERT-1-5mer	0.837	0.779	0.114	0.849	0.313	0.179	0.874	0.043	0.142
DNABERT-1-6mer	0.835	0.776	0.105	0.845	0.299	0.163	0.893	0.075	0.236
HyenaDNA-T	0.832	0.773	0.092	0.844	0.314	0.178	0.864	0.032	0.037
HyenaDNA-T-d256	0.835	0.776	0.104	0.848	0.325	0.195	0.886	0.043	0.200
HyenaDNA-T-d128	0.830	0.770	0.079	0.843	0.313	0.166	0.864	0.025	0.112
HyenaDNA-S	0.830	0.770	0.081	0.842	0.306	0.177	0.870	0.069	0.098
HyenaDNA-M-160k	0.831	0.771	0.083	0.848	0.314	0.186	0.884	0.037	0.115
HyenaDNA-M-450k	0.832	0.772	0.089	0.845	0.309	0.172	0.873	0.064	0.113
HyenaDNA-L	0.831	0.771	0.085	0.848	0.314	0.178	0.883	0.029	0.132

Table 9: **The real-world evolution ranking task is highly challenging as it is based on actual in-vitro evolution rounds.** We provide three tracks, @2, @10, and @100, where the lengths of ranking lists are 2, 10, and 100, respectively.

Track	#List		#DNA		#Protein	
	Train	Test	Train	Test	Train	Test
@100	7	99	682	9822	661	9159
@10	1155	4563	8745	41264	5398	24906
@2	27754	44322	38114	63445	16461	28800

Fine-tuned BLMs for Real-World Evolution Ranking We also fine-tuned selected BLMs on the fine-tuned BLMs for the real-world evolution ranking task, training the BLM backbones and their ranking heads to ensure that performance limitations are not solely due to linear probing. We report the results in Table 13. Although fine-tuning provides improvements over the random init, most BLMs do not show substantial performance gains. This indicates that when BLMs face real-world evolution ranking tasks in our benchmark, *i.e.*, predicting the outcomes of the next round of protein evolution based on results from the current round, they are almost incapable. This reflects the considerable challenge posed by our benchmark in real-world evolution ranking tasks with existing BLMs. Such challenges align with the logic of actual biological experiments and represent real difficulties that need resolution in practical applications.

Table 10: **Protein BLMs fail to solve the real-world evolution ranking task using linear probing.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
ESM2-8M	0.811	0.744	-0.023	0.815	0.310	0.061	0.856	0.053	0.014
ESM2-35M	0.810	0.743	-0.028	0.803	0.298	-0.005	0.858	0.055	0.060
ESM2-150M	0.811	0.744	-0.024	0.808	0.293	0.023	0.856	0.057	0.028
ESM2-650M	0.814	0.747	-0.010	0.802	0.293	-0.004	0.845	0.049	0.016
ESM2-3B	0.815	0.750	-0.001	0.808	0.308	0.027	0.838	0.037	-0.018
ESM2-15B	0.815	0.749	-0.002	0.801	0.300	0.000	0.834	0.064	-0.071
ESM3	0.819	0.755	0.018	0.802	0.298	-0.004	0.864	0.054	0.069
SaProt-650M-AF2	0.813	0.747	-0.011	0.797	0.284	-0.036	0.853	0.055	0.060
SaProt-650M-PDB	0.817	0.752	0.006	0.802	0.294	-0.009	0.836	0.063	-0.046
SaProt-35M-AF2	0.817	0.751	0.006	0.812	0.306	0.046	0.845	0.057	-0.029
SaProt-35M-AF2-Seq	0.821	0.757	0.029	0.802	0.297	-0.016	0.854	0.079	0.025
LucaOne	0.823	0.761	0.044	0.798	0.291	-0.021	0.846	0.069	0.007
RosettaFold-STATE	0.812	0.746	-0.017	0.801	0.282	-0.018	0.837	0.044	-0.055
RosettaFold-MSA	0.811	0.745	-0.022	0.800	0.285	-0.022	0.844	0.077	0.001
ProstT5	0.817	0.752	0.006	0.804	0.289	-0.004	0.843	0.041	-0.038
ProstT5-fp16	0.816	0.750	0.002	0.801	0.300	-0.006	0.852	0.043	0.012
Prot-T5-XL-U50	0.815	0.749	-0.003	0.807	0.307	0.019	0.852	0.053	0.040
Prot-T5-XL-Half	0.810	0.742	-0.031	0.803	0.287	-0.010	0.855	0.038	0.008
Chai1	0.814	0.748	-0.008	0.802	0.290	-0.008	0.857	0.055	0.062
Chai1-ESM	0.808	0.740	-0.042	0.803	0.296	-0.008	0.842	0.033	-0.050
Prot-Bert	0.819	0.755	0.021	0.817	0.304	0.059	0.843	0.047	-0.024
Prot-ss3	0.813	0.747	-0.012	0.804	0.290	0.000	0.840	0.047	-0.035
Prot-Membrane	0.822	0.758	0.033	0.805	0.298	-0.001	0.853	0.067	0.035
Prot-Localization	0.807	0.738	-0.048	0.802	0.296	-0.005	0.841	0.050	-0.033
Prot-T5-XXL-U50	0.816	0.751	0.003	0.799	0.305	-0.020	0.857	0.062	0.039
Prot-Generator	0.818	0.754	0.015	0.804	0.301	0.003	0.849	0.073	0.013
Prot-Discriminator	0.816	0.750	0.000	0.801	0.290	-0.090	0.851	0.051	-0.005
Prot-T5-XL-BFD	0.815	0.750	0.000	0.799	0.297	-0.016	0.844	0.069	-0.012
Prot-Bert-BFD	0.814	0.748	-0.010	0.803	0.291	-0.001	0.837	0.039	-0.044
Prot-T5-XXL-BFD	0.813	0.746	-0.015	0.808	0.299	0.024	0.841	0.049	-0.025
Prot-Xlnet	0.813	0.747	-0.012	0.803	0.292	0.004	0.839	0.054	0.003
Prot-Albert	0.812	0.745	-0.020	0.796	0.288	-0.037	0.838	0.044	-0.036

Table 11: **RNA BLMs fail to solve the real-world evolution ranking task using linear probing.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
mRNA-FM	0.814	0.748	-0.006	0.809	0.291	0.015	0.847	0.045	0.003
RNA-FM	0.813	0.747	-0.013	0.814	0.303	0.045	0.852	0.037	0.020
RNA-MSM	0.821	0.757	0.029	0.811	0.299	0.021	0.844	0.060	0.007
RNA-Ernie	0.815	0.750	-0.001	0.803	0.298	-0.007	0.837	0.056	-0.039
RiNALMo	0.817	0.751	0.006	0.807	0.291	0.017	0.832	0.046	-0.037
ERNIE-RNA	0.816	0.750	0.001	0.803	0.293	-0.010	0.853	0.065	0.050
ERNIE-RNA.ss	0.817	0.751	0.006	0.807	0.296	0.007	0.864	0.076	0.070
Chai1	0.814	0.748	-0.006	0.809	0.293	0.018	0.853	0.075	0.044
OmniGenome-418M	0.817	0.752	0.009	0.799	0.287	-0.037	0.840	0.042	-0.017
OmniGenome-186M	0.819	0.755	0.019	0.810	0.297	0.017	0.842	0.088	-0.036
OmniGenome-52M	0.814	0.749	-0.006	0.812	0.296	0.036	0.835	0.076	-0.040
3UTRBERT-6mer	0.812	0.745	-0.019	0.811	0.293	0.029	0.849	0.074	0.025
3UTRBERT-5mer	0.819	0.755	0.020	0.811	0.293	0.026	0.842	0.044	0.028
3UTRBERT-4mer	0.820	0.756	0.024	0.800	0.285	-0.029	0.850	0.055	0.017
3UTRBERT-3mer	0.815	0.750	0.000	0.809	0.299	0.299	0.842	0.045	-0.056
SpliceBERT	0.814	0.748	-0.008	0.802	0.298	-0.011	0.856	0.049	0.035
SpliceBERT-H.510nt	0.814	0.748	-0.008	0.805	0.297	0.004	0.839	0.045	-0.077
SpliceBERT.510nt	0.817	0.752	0.007	0.801	0.294	-0.015	0.847	0.065	-0.005
CaLM	0.817	0.752	0.009	0.800	0.285	-0.031	0.840	0.080	-0.056

Table 12: **DNA BLMs fail to solve the real-world evolution ranking task using linear probing.**

Model	@2			@10			@100		
	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑	nDCG↑	mRR↑	SP↑
EVO-8k	0.809	0.741	-0.036	0.799	0.286	-0.022	0.831	0.043	-0.079
EVO-131k	0.809	0.741	-0.037	0.802	0.293	-0.016	0.833	0.054	-0.080
LucaOne	0.816	0.750	0.001	0.808	0.289	0.006	0.839	0.055	0.013
Chai1	0.820	0.756	0.025	0.802	0.292	-0.015	0.851	0.063	0.009
NT-2-50M	0.812	0.746	-0.017	0.802	0.291	-0.019	0.837	0.035	-0.025
NT-2-100M	0.818	0.753	0.011	0.800	0.290	-0.024	0.857	0.070	0.052
NT-2-250M	0.818	0.753	0.013	0.805	0.288	0.004	0.849	0.037	0.007
NT-2-500M	0.816	0.751	0.005	0.804	0.289	-0.013	0.843	0.047	-0.044
NT-500M-human	0.812	0.745	-0.021	0.806	0.291	0.005	0.829	0.051	-0.108
NT-500M-1000G	0.816	0.751	0.004	0.804	0.295	0.001	0.841	0.059	-0.025
NT-2B5-1000G	0.815	0.749	-0.003	0.805	0.301	0.005	0.856	0.034	0.046
NT-2B5	0.820	0.757	0.027	0.803	0.297	-0.008	0.840	0.026	-0.033
AgroNT	0.815	0.749	-0.003	0.815	0.298	0.038	0.830	0.056	-0.083
GenSLMs-2.5B	0.810	0.743	-0.029	0.810	0.300	0.027	0.857	0.043	0.066
GenSLMs-250M	0.812	0.746	-0.018	0.799	0.289	-0.028	0.853	0.049	0.020
GenSLMs-25M	0.819	0.755	0.020	0.807	0.298	0.007	0.841	0.050	0.002
DNABERT-2	0.813	0.747	-0.015	0.802	0.285	-0.024	0.863	0.062	0.072
DNABERT-S	0.812	0.745	-0.019	0.801	0.288	-0.026	0.851	0.043	0.036
DNABERT1-3mer	0.818	0.753	0.014	0.801	0.289	-0.023	0.851	0.041	0.039
DNABERT1-4mer	0.815	0.749	-0.002	0.804	0.288	-0.007	0.840	0.043	-0.026
DNABERT1-5mer	0.818	0.753	0.011	0.806	0.297	0.007	0.850	0.062	-0.001
DNABERT1-6mer	0.811	0.744	-0.025	0.809	0.296	0.019	0.843	0.060	-0.049
HyenaDNA-T	0.816	0.751	0.004	0.808	0.294	0.006	0.828	0.046	-0.124
HyenaDNA-T-d128	0.817	0.753	0.011	0.800	0.286	-0.044	0.845	0.039	-0.024
HyenaDNA-T-d256	0.816	0.750	0.002	0.803	0.286	-0.007	0.851	0.038	0.029
HyenaDNA-S	0.817	0.752	0.006	0.816	0.292	0.047	0.857	0.076	0.027
HyenaDNA-M-160k	0.817	0.752	0.010	0.800	0.282	-0.023	0.850	0.042	-0.003
HyenaDNA-M-450k	0.819	0.755	0.021	0.803	0.284	-0.027	0.844	0.060	-0.052
HyenaDNA-L	0.814	0.748	-0.008	0.804	0.288	-0.019	0.861	0.045	0.061

Table 13: **BLMs struggle to solve the real-world evolution tasks using linear probing and fine-tuning.** The table shows the result of @100 track with a batch size of 4×100 sequences. Although fine-tuning can help a little, most BLMs cannot solve the task well with a similar performance of random initialization.

Modality	Model	Random Initialization		Linear Probing		Fine-tuning	
		nDCG↑	SP↑	nDCG↑	SP↑	nDCG↑	SP↑
Protein	ESM2-650M	0.847	0.017	0.845	0.016	0.869	0.114
	ESM2-150M	0.851	0.010	0.846	0.007	0.859	0.051
DNA	DNABERT-1-6mer	0.845	-0.057	0.843	-0.049	0.846	0.009
	HyenaDNA-T-d256	0.854	0.018	0.851	0.029	0.860	0.068
RNA	RNAErnie	0.841	0.007	0.837	-0.039	0.844	-0.003
	3UTRBERT-6mer	0.842	-0.042	0.849	0.025	0.845	0.007