ProTrek: Navigating the Protein Universe through Tri-Modal Contrastive Learning

Jin Su[1], Xibin Zhou[1], Xuting Zhang[1], Fajie Yuan[1]

[1]Westlake University.

Correspondence to: yuanfajie@westlake.edu.cn;

Abstract

ProTrek, a tri-modal protein language model, enables contrastive learning of protein sequence, structure, and function (SSF). Through its natural language search interface, users can navigate the vast protein universe in seconds, accessing nine distinct search tasks that cover all possible pairwise combinations of SSF. Additionally, ProTrek serves as a general-purpose protein representation model, excelling in various downstream prediction tasks through supervised transfer learning, thereby providing extensive support for protein research and analysis.

Proteins, the fundamental building blocks of life, are essential for numerous biological processes. Understanding their intricate structures, functions, and interactions is pivotal across various scientific domains, including biochemistry, molecular biology, and drug discovery. Consequently, biologists often seek to decipher the three dimensions of a protein - sequence, structure, and function (SSF) - to advance our understanding of protein mechanisms.

Alignment-based tools have advanced protein analysis, utilizing either amino acid (AA) sequence-based [2, 17, 6] or structure-based [21] alignment methods. Yet, these tools primarily focus on local alignment, potentially overlooking global insights. Furthermore, existing methods typically handle data from a single modality - either sequence or structure - limiting their ability to align multiple modalities. Recently, neural network-based functional annotation tools have emerged [3, 25, 16], capable of identifying corresponding annotations for a given protein. However, these annotation tools, being based on predefined labels, lack the ability to understand natural language, thereby hindering their capability to provide accurate textual descriptions of protein functions or identify proteins from lengthy textual descriptions.

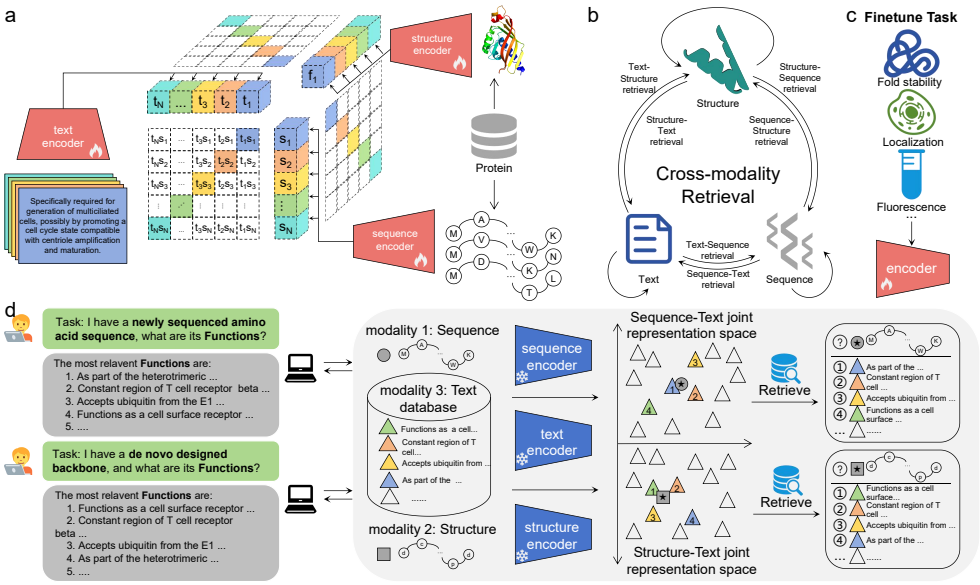

Fig. 1: **Illustration of ProTrek. a,** ProTrek architecture and tri-modal contrast learning. **b,** Cross-modal and intra-modal retrieval. ProTrek supports 9 searching tasks, including sequence-structure, sequence-function, sequence-sequence, structure-structure, structure-sequence, structure-function, function-function, function-sequence, and function-structure. **c,** After the tri-modal contrast learning, the protein sequence encoder encodes almost universal representation of proteins, which can be fine-tuned to predict diverse downstream tasks, such as protein fitness and stability prediction. **d,** An example to demonstrate sequence-function and structure-function retrieval.

Meanwhile, large language models like ChatGPT [5] and Llama [20] have demonstrated remarkable performance in numerous natural language processing tasks. Concurrently, protein language models (PLMs) have emerged as a notable area of exploration within the field of proteins[12, 10, 1, 14]. Motivated by these advancements, we ask whether it is feasible to establish a foundation protein model that can jointly represent protein SSF using such language modeling techniques. Such a model holds the promise of bolstering a multitude of prediction and analysis tasks within the realm of proteins

Here, we introduce ProTrek, a tri-modal PLM that jointly models protein SSF. ProTrek employs contrastive learning [15] with three core alignment strategies (Fig 1a): (1) using structure as the supervision signal for AA sequences and vice versa, (2) mutual supervision between sequences and functions, and (3) mutual supervision between structures and functions. This tri-modal alignment training enables ProTrek to tightly associate SSF by bringing genuine sample pairs (sequence-structure,

sequence-function, and structure-function) closer together while pushing negative samples farther apart in the latent space.

ProTrek employs a pre-trained ESM encoder [12] for its AA sequence language model and a pre-trained BERT encoder [11] for its textual function language model. Protein structures here are transformed into discrete 3Di token sequences via Foldseek [21]. The structural language model still uses a BERT architecture, trained on sequential structure data with a vocabulary comprising 20 3Di structural alphabets. ProTrek is trained on a dataset comprising 40 million examples, which includes 14 million precise protein-function pairs collected from Swiss-Prot [4] database and 25 million relatively noisy pairs filtered from 300 million Uniprot [7] samples. The ProTrek model incorporates 8 loss functions - 6 for alignment across modalities and 2 masked language modeling losses to maintain recognition at the amino acid and 3Di token levels.

We provide two versions of ProTrek: a large version with a 650M AA sequence encoder, 150M structure encoder, and 130M function encoder, and a smaller version with a 35M sequence encoder, 35M structure encoder, and 130M function encoder. During inference, ProTrek employs a max-inner product search (MIPS) algorithm [9] to swiftly retrieve the most relevant protein sequences, structures, and functions from the corresponding database. This allows ProTrek to complete the search in just one or two seconds, even when querying a database of billions of entries.

ProTrek is a near-universal protein model, characterized by its two primary capabilities: (1) ProTrek is a potent retrieval-enabled language model that allows researchers to explore the complex interplay of protein sequence-structure-function through 9 distinct search tasks across the entire protein universe (Fig. 1b). Specifically, when given an AA sequence, ProTrek can identify related structures, provide functional descriptions, and find other AA sequences that encode similar structures and functions. Conversely, it can retrieve the corresponding AA sequence and functional description when given a protein structure. Additionally, ProTrek can discover protein sequences and structures associated with specific functions based on textual descriptions. Unlike typical protein annotation tasks, ProTrek supports detailed, long-text natural language descriptions for protein functions (Fig. 1d). (2) ProTrek also serves as an exceptional representation model, enabling the transfer and fine-tuning of its protein sequence or structure encoders on new data for various downstream prediction tasks (Fig. 1c), similar to other popular PLMs such as ESM [12], ProtTrans [10] and Saprot [18, 19]. With its universal search and representation capabilities, ProTrek enhances the exploration and understanding of the protein universe, making it invaluable for protein research.

First, we evaluated ProTrek on four search tasks, including pairwise searches between protein sequences/structures and textual functional descriptions - specifically, sequence-to-text, structure-to-text, text-to-sequence, and text-to-structure (Fig. 2a). We compared ProTrek with two state-of-the-art methods, ProteinDT [13] and ProtST [24] on the human-reviewed Swiss-Prot [4] dataset, which contains 4,000 proteins and their associated functions as the text set. For a fair evaluation, proteins sampled in the test set have less than 50% sequence identity with those in the training set. Additionally, for text-to-protein (sequence/structure) tasks, we included 100,000

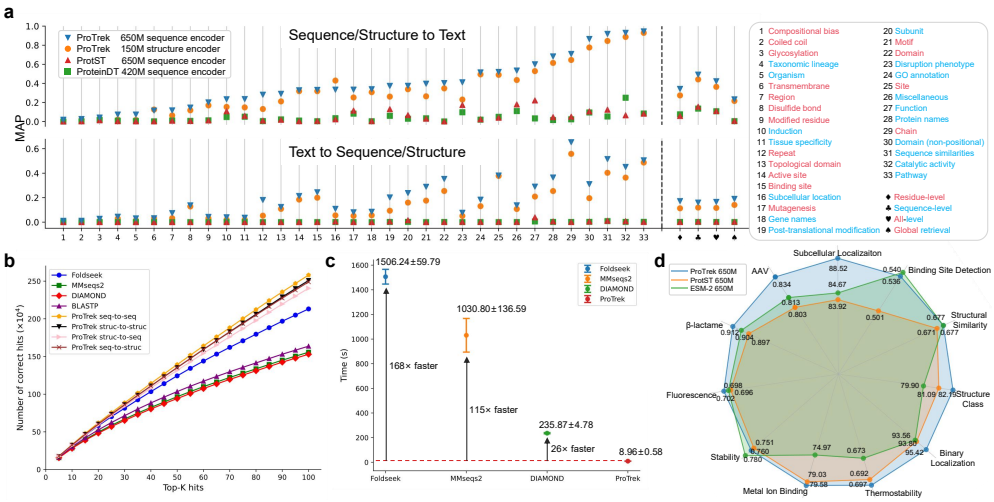

Fig. 2: **ProTrek performance on protein search and representation tasks.** **a,** The top chart indicates searching for protein function descriptions based on AA sequences or structures, while the bottom chart indicates finding corresponding protein sequences/structures based on textual function descriptions. The x-axis shows protein function categories to the left of the dashed line and aggregated categories (residue-level, protein-level, and all-level) to the right. "Global retrieval" indicates a search across the entire database, not within individual categories. The y-axis is MAP (mean average precision), a commonly used metric for information retrieval tasks. **b**, Searching proteins with similar functions using protein sequence or structure as input. **c**, Comparing alignment speed (CPU time) for 100 query proteins on UniRef50 with 50 million candidate proteins, utilizing 24 CPU cores. **d**, Evaluating the protein representation ability of the ProTrek AA sequence encoder.

randomly sampled proteins from UniProt as unknown negative examples to assess ProTrek's generalization to other databases.

Notably, whether using the AA sequence encoder or structure encoder, ProTrek substantially outperforms ProteinDT and ProtST across most functional categories and achieves over 30× and 60× improvements on the global retrieval task (0.233 for ProTrek vs. 0.007 for ProtST vs. 0.004 for ProteinDT in sequence-to-text, and 0.190 for ProTrek vs. 0.001 for ProtST vs. 0.003 for ProteinDT in text-to-sequence) (Fig. 2a). Clearly, ProTrek's exceptional performance stems from its extremely large training dataset, which is two orders of magnitude larger than that of ProteinDT and ProtST. With this advantage, ProTrek will be a highly practical biological tool and can also serve as an important aid in text-based protein design and protein question-answer generation systems. Regarding ProTrek itself, the AA sequence encoder generally outperforms the structure encoder, a difference that may be related to the size of the encoder model and whether it is pre-trained.

Second, we evaluated ProTrek's performance in detecting protein homology by comparing it with several renowned alignment-based tools, including MMseqs2 [17], DIAMOND [6], BLASTP [2], and the latest structure aligner, Foldseek [21] (Fig. 2b). Unlike these classical methods that rely on local alignment through sequence or structure similarity, ProTrek employs global alignment via cross-modal contrastive learning. To demonstrate this, we evaluated the functions of the retrieved proteins through another four search tasks: two cross-modal searches between sequences and structures (sequence-to-structure and structure-to-sequence), and two homomodal searches between sequences (sequence-to-sequence) and between structures (structure-to-structure) (Fig. 2b). We performed an all-versus-all search on the same Swiss-Prot test dataset with 4,000 proteins and compared their performance for finding members of the same protein GO annotation (see Methods).

ProTrek significantly outperforms all sequence alignment tools and even surpasses Foldseek in terms of the number of correct hits. Among ProTrek's four search tasks, sequence-to-sequence search performs the best, followed by structure-to-structure and sequence-to-structure searches. However, each search task has its unique applications. For example, although structure-to-sequence search is slightly less effective, it can help filter or rank inaccurately generated protein sequences by AI-based protein design models (such as ProteinMPNN [8]).

Another advantage of ProTrek is its inference speed, supported by MIPS, enabling rapid search and sorting of billion-scale databases within seconds, outpacing Foldseek and MMseq2 by over 100× (Fig. 2c). It currently holds the title of the fastest alignment algorithm available. ProTrek's exceptional search speed empowers users to explore and analyze the entire protein universe swiftly. Moreover, conventional alignment tools lack cross-modal search capabilities, further highlighting ProTrek's significance as a valuable addition to existing bioinformatics tools.

In addition to its search ability, the amino acid and 3Di sequence encoders of ProTrek can also serve as general-purpose representation models, owing to their two unsupervised loss functions (contrastive learning and mask-token-prediction losses). We evaluated these models on over 10 diverse downstream protein tasks, including both protein-level and residue-level, regression, and classification tasks. Through downstream supervise fine-tuning, ProTrek outperformed the classical ESM-2 model and the state-of-the-art ProtST model in 9 out of 11 tasks (Fig. 2d). In some tasks, it even exceeded the performance of ESM-2 or ProtST by a relative margin of 5% to 7%, showcasing its strong transfer learning capability as a representation model.

We have released the ProTrek model weights (https://huggingface.co/westlake-repl/ProTrek_650M_UniRef50) and will soon develop a web server (https://huggingface.co/spaces/westlake-repl/Demo_ProTrek_650M_UniRef50) for conducting multi-database searches, which includes Swiss-Prot, AlphaFoldDB, UniRef50, and the Protein Data Bank (PDB). Additionally, we have made all protein sequence embeddings available for proteins in these databases. Furthermore, we are working on developing a Google Colab version of ProTrek, following the approach of SaprotHub [19].

ProTrek demonstrates potent and rapid cross-modal search capabilities, poised to revolutionize the analysis of emerging sequencing databases. This enables biologists

to quickly delve into the vast expanse of the protein universe. Its robust natural language understanding facilitates nuanced searches, overcoming the limitations of traditional keyword-matching approaches. Moreover, ProTrek serves as a sturdy foundation model, pivotal in facilitating intricate protein-text interactions, such as text-guided protein design [13, 23] and protein question-answer systems [22].

References

[1] Ethan C Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

[2] Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, 1997.

[3] Maxwell L Bileschi, David Belanger, Drew H Bryant, Theo Sanderson, Brandon Carter, D Sculley, Alex Bateman, Mark A DePristo, and Lucy J Colwell. Using deep learning to annotate the protein universe. *Nature Biotechnology*, 40(6):932–937, 2022.

[4] Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1):365–370, 2003.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[6] Benjamin Buchfink, Klaus Reuter, and Hajk-Georg Drost. Sensitive protein alignments at tree-of-life scale using diamond. *Nature methods*, 18(4):366–368, 2021.

[7] UniProt Consortium. Uniprot: a hub for protein information. *Nucleic acids research*, 43(D1):D204–D212, 2015.

[8] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[9] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2024.

[10] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

[11] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

[12] 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.

[13] Shengchao Liu, Yutao Zhu, Jiarui Lu, Zhao Xu, Weili Nie, Anthony Gitter, Chaowei Xiao, Jian Tang, Hongyu Guo, and Anima Anandkumar. A text-guided protein design framework. *arXiv preprint arXiv:2302.04611*, 2023.

[14] Erik Nijkamp, Jeffrey A Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[16] Predrag Radivojac, Wyatt T Clark, Tal Ronnen Oron, Alexandra M Schnoes, Tobias Wittkop, Artem Sokolov, Kiley Graim, Christopher Funk, Karin Verspoor, Asa Ben-Hur, et al. A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221–227, 2013.

[17] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

[18] 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.

[19] Jin Su, Zhikai Li, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, Dacheng Ma, Sergey Ovchinnikov, Fajie Yuan, et al. Saprothub: Making protein modeling accessible to all biologists. *bioRxiv*, 2024.

[20] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[21] Michel Van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nature Biotechnology*, 42(2):243–246, 2024.

[22] Chao Wang, Hehe Fan, Ruijie Quan, and Yi Yang. Protchatgpt: Towards understanding proteins with large language models. *arXiv preprint arXiv:2402.09649*, 2024.

[23] Chentong Wang, Yannan Qu, Zhangzhi Peng, Yukai Wang, Hongli Zhu, Dachuan Chen, and Longxing Cao. Proteus: exploring protein structure generation for enhanced designability and efficiency. *bioRxiv*, pages 2024–02, 2024.

[24] Minghao Xu, Xinyu Yuan, Santiago Miret, and Jian Tang. Protst: Multi-modality learning of protein sequences and biomedical texts. In *International Conference on Machine Learning*, pages 38749–38767. PMLR, 2023.

[25] 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.

Methods

Protein-function pair construction

For a given protein, we created protein-function pairs using the descriptive information from almost all subsections in the UniProt database [7]. We categorized the subsection information of a protein into two types: sequence-level and residue-level. Sequence-level information consists of one or multiple sentences that describe the overall characteristics of the protein, such as its function and catalytic activity. Residue-level information contains phrases describing specific information about certain residues in the protein sequence, which cannot be used directly, such as binding sites and active sites. For residue-level subsections, we employ GPT-4 [26] to generate multiple specific template sentences, thus organizing the information into a coherent sentence. For sequence-level subsections, we also utilize GPT-4 to paraphrase the descriptions and generate multiple alternative sentences to enhance the model's robustness to textual input. In the end, each subsection in a given protein results in a text sentence, and these sentences are paired with the protein to form protein-function pairs, which are used for training or evaluation purposes.

Pre-training dataset construction

We first performed a 50% sequence similarity clustering on the human-reviewed Swiss-Prot database [4]. We designated 1000 clusters for validation and another 1000 clusters for testing, using the remaining data as the training set. For each protein, we constructed the protein-function pairs as described above, resulting in a final training set of 14 million protein-function pairs. This high-quality dataset was used to train an initial version of ProTrek. The initial ProTrek model, a 35M version, was pre-trained on 12 NVIDIA 80G A100 GPUs over 100K steps.

Next, we used the initial ProTrek model to score and filter 300 million protein-function pairs from UniRef50 [7]. We retained all protein-function pairs with model scores higher than the average score on Swiss-Prot, resulting in 25 million pairs. These 25 million pairs, combined with the original 14 million pairs, form the final pre-training dataset.

Pre-training loss function

We adopted InfoNCE loss [30] for protein sequence–structure-function contrastive learning. The InfoNCE loss can be detailed as:

$$\mathcal{L}_{InfoNCE} = -log\frac{exp(f(z_i, z_j)/\tau)}{\sum_{k=1}^{N} exp(f(z_i, z_k)/\tau)} \tag{1}$$

Here, z_i and z_j represent the embeddings of any two modalities. $f(z_i, z_j)$ denotes a similarity score between the embeddings. N is the total number of pairs in a batch and τ is a learnable temperature parameter. We calculated the InfoNCE loss for the sequence-structure, sequence-function and structure-function pairs from two directions, resulting in 6 contrastive learning functions.

We additionally added 2 Masked Language Modeling (MLM) [27] loss functions to the ProTrek sequence and structure encoder to maintain model's recognition at the amino acid and 3Di token levels. The training objective is to predict masked tokens by capturing dependencies between masked positions and surrounding context, with the loss function formally described as:

$$\mathcal{L}_{MLM} = \sum_{i \in T} -logP(s_i|S_{\setminus T}) \tag{2}$$

where T is a set of token positions to be masked and $S_{\setminus T}$ represents a protein/structural sequence with tokens at specific positions masked. We formed the final loss function by averaging 2 MLM and 6 InfoNCE loss functions.

Pre-training setting

We initialized the ProTrek sequence encoder from the ESM-2 650M [12] and the text encoder from the PubMedBERT [11], and we randomly initialized the ProTrek structure encoder as there was no available pre-trained model. For the optimization, we used the DeepSpeed strategy [31] and employed the AdamW optimizer [29], setting $\beta_1 = 0.9, \beta_2 = 0.98$ and we utilized L_2 weight decay of 0.01. We gradually increased the learning rate from 0 to 4e-4 over the first 2000 steps decreased it to 4e-5 using the cosine annealing schedule [28]. The overall training phase lasted approximately 100K steps trained on 20 NVIDIA 80G A100 GPUs. We truncated protein and structural sequences to a maximum of 512 tokens, and truncated the test descriptions to a maximum of 100 tokens. Our total batch size consisted of 1280 protein sequence-structure-function pairs. Additionally, we employed mixed precision training to train ProTrek.

Protein-function retrieval benchmark

We utilized 4,000 proteins from the Swiss-Prot test set to construct a benchmark for the protein-function task. To evaluate ProTrek's generalization performance with proteins in other databases, we included 100,000 randomly sampled proteins from UniProt as negative samples in the test set. The textual descriptions of these proteins were added to construct the text collection. For the protein-to-text search task, the model needed to rank the functional descriptions for the proteins in the Swiss-Prot test set. Conversely, for the text-to-protein task, the model had to rank Swiss-Prot proteins that match the query function description from a set of nearly 104,000 negative sample proteins. We selected the top 33 function entries based on the number of protein-text pairs for evaluation. This approach aims to mitigate the bias impact of entries with fewer pairs.

Mean average precision

Mean Average Precision (MAP), a commonly used metric for information retrieval tasks, is the Average Precisions (APs) for each query. For a given query, AP is

calculated by the following formula:

$$AP = \frac{1}{R}\sum_{i=1}^{N} P(i)r(i) \tag{3}$$

where R is the number of relevant results and N denotes the total number of results. For the ith result, $P(i)$ is the precision at the i-th position in the ranked list, and $r(i)$ indicates whether the result is relevant to the query (1 if relevant and 0 if not relevant). The MAP is calculated by averaging the APs from all queries:

$$MAP = \frac{1}{Q}\sum_{q=1}^{Q} AP(q) \tag{4}$$

Deep learning baseline models

For protein-function searches, we compared ProTrek to ProtST-ESM-1b from ProtST [24] and ProtBERT_BFD-512-1e-5-1e-1-text-512-1e-5-1e-1-InfoNCE-0.1-batch-9-gpu-8-epoch-5 from ProteinDT [13]. For the downstream protein tasks, we used the ESM-2 650M version [12] and ProtST-ESM-1b as baselines.

Downstream task fine-tuning

We strictly followed SaprotHub's fine-tuning settings [19], including dataset preprocessing, splitting, fine-tuning hyper-parameters, etc. We fine-tuned all model parameters until convergence and selected the best checkpoints based on their performance on the validation set.

Tool execution commands

We ran each alignment-based tool according to its standard usage example:

Foldseek

We used Foldseek with the **ef4e960ab84fc502665eb7b84573dfff9c2aa89d** version. The command line was executed with default parameters: **foldseek easy-search pdb_dir targetDB aln.m8 tmpFolder**

MMseqs2

We used MMseqs2 with the **edb8223d1ea07385ffe63d4f103af0eb12b2058e** version. The command line was executed with default parameters: **mmseqs easy-search seqs.fasta targetDB alnRes.m8 tmp**

BLASTP

We used BLASTP with the **Protein-Protein BLAST 2.15.0+** version. We ran BLASTP from the command line: **blastp -query seqs.fasta -db db -outfmt 6 -out blastp_result**

DIAMOND

We used DIAMOND with the **v2.1.9.163** version. We ran DIAMOND in very-sensitive mode following the standard case: **diamond blastp -q seqs.fasta -d db -o result.tsv –very-sensitive -k 0**

Data availability

The pre-trained ProTrek weights can be downloaded from https://huggingface.co/westlake-repl/ProTrek_650M_UniRef50. All pre-computed protein sequence, structure and text embeddings for Swiss-Prot database are available at https://huggingface.co/datasets/westlake-repl/faiss_index_ProTrek_650M_UniRef50 (will release protein embeddings for more ldatabase soon). The structural sequences are available at https://github.com/steineggerlab/foldseek. The text descriptions are available at https://www.uniprot.org/

Code availability

ProTrek is open-sourced under the MIT license. The code repository is available at https://github.com/westlake-repl/ProTrek. The ProTrek webserver is located at https://huggingface.co/spaces/westlake-repl/Demo_ProTrek_650M_UniRef50.

References

[26] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[28] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[29] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2018.

[30] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[31] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506, 2020.

Acknowledgements

We express our gratitude to Sergey Ovchinnikov for his valuable suggestions in evaluating ProTrek.

Supplementary information

In progress...