VENUSFACTORY: A Unified Platform for Protein Engineering Data Retrieval and Language Model Fine-Tuning

Yang Tan[1,2,3,*], **Chen Liu**[3,*], **Jingyuan Gao**[1,*], **Banghao Wu**[1], **Mingchen Li**[1,2,3], **Ruilin Wang**[3], **Lingrong Zhang**[1], **Huiqun Yu**[3], **Guisheng Fan**[3], **Liang Hong**[1,2], **Bingxin Zhou**[1,†]

[1] Shanghai Jiao Tong University, China
[2] Shanghai Artificial Intelligence Laboratory, China
[3] East China University of Science and Technology, China

Open-source repository: https://github.com/ai4protein/VenusFactory
Demonstration video: https://www.youtube.com/watch?v=MT6lPH5kgCc

Abstract

Natural language processing (NLP) has significantly influenced scientific domains beyond human language, including protein engineering, where pre-trained protein language models (PLMs) have demonstrated remarkable success. However, interdisciplinary adoption remains limited due to challenges in data collection, task benchmarking, and application. This work presents VENUSFACTORY, a versatile engine that integrates biological data retrieval, standardized task benchmarking, and modular fine-tuning of PLMs. VENUSFACTORY supports both computer science and biology communities with choices of both a command-line execution and a Gradio-based no-code interface, integrating 40+ protein-related datasets and 40+ popular PLMs. All implementations are open-sourced on https://github.com/ai4protein/VenusFactory.

1 Introduction

Discrete tokens provide a natural representation of data across various fields, such as human language, amino acid sequences, and molecular structures (Brown et al., 2020; Guo et al., 2025). The recent success of natural language processing and large language models has introduced novel solutions to fundamental scientific and engineering challenges (Pan, 2023; Zhou et al., 2024a). In enzyme engineering, pre-trained protein language models (PLMs) have been developed to analyze and extract hidden amino acid interactions and evolutionary features from protein sequences (Meier et al., 2021; Rives et al., 2021; Li et al., 2024, 2025; Tan et al., 2024c, 2025; Liu et al., 2025). The growing interest in AI-driven scientific research in protein engineering has led to the development of many open-source PLMs for both the computer science and computational biology communities. For example, ESM2-650M (Lin et al., 2023), arguably the most popular sequence-encoding PLM, has over one million downloads per month from HuggingFace[1]. Meanwhile, by integrating task-specific labeled data and predictive modules, these models facilitate downstream tasks such as sequence generation, catalytic activity enhancement, function prediction, and properties assessment, thereby advancing enzyme production and application (Madani et al., 2023; Zhou et al., 2024b,c; Kang et al., 2025).

Despite the availability of high-impact models and successful applications in certain scenarios, interdisciplinary collaboration between biologists and computer scientists remains limited. Most algorithm development and validation focus on a few specific benchmarks for particular objectives, while many other datasets and engineering challenges lack readily available tools, even when compatible with existing deep learning methodologies. We attribute this gap to three key complexities: (1) **Collection**: While some public databanks provide access to protein sequences, structures, and functions, they often lack efficient bulk download options and standardized formatting, which are essential for computer scientists to train PLMs. (2) **Benchmarking**: AI-driven protein engineering lacks a systematic framework that consolidates benchmarks and baselines. As a result, benchmark datasets from experimental research are underutilized in model development, and state-of-the-art models are rarely integrated into daily research workflows as seamlessly as traditional computational biology tools. (3) **Application**: Beyond the absence of multifunctional integrated systems, existing PLM solutions often require substantial coding expertise, making them less accessible to non-programmers (*e.g.*, biologists) compared to web-based tools.

[*] Equal contribution and this work was done during the internship at Shanghai Artificial Intelligence Laboratory.
[†] Corresponding author (bingxin.zhou@sjtu.edu.cn).

[1] https://huggingface.co/facebook/esm2_t33_650M_UR50D

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 230–241
July 27 - August 1, 2025

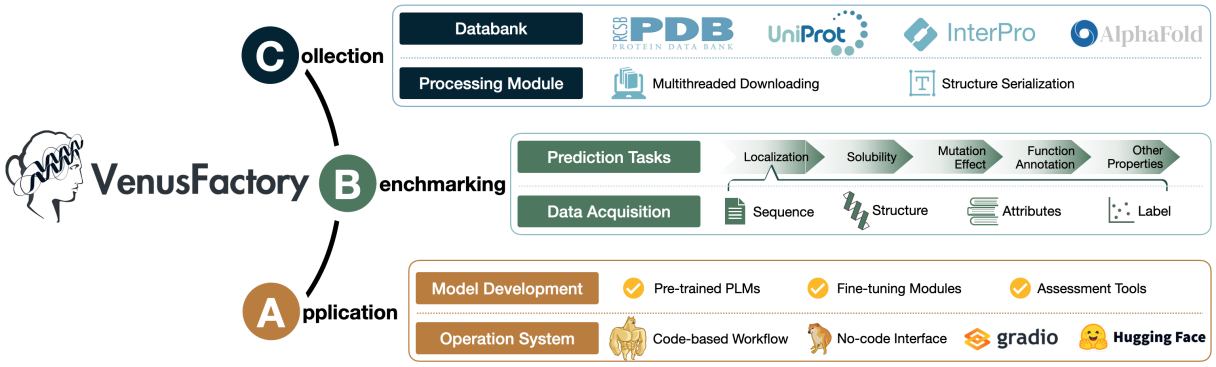

Figure 1: VENUSFACTORY supports high-throughput raw data download, structure sequencing, a wide range of downstream task datasets, and interface or command-line protein language model fine-tuning and reasoning.

To address these challenges, we developed a versatile engine for AI-based protein engineering, namely VENUSFACTORY (Figure 1). It integrates a full suite of tools from data acquisition to model training, evaluation, and application. It is designed for users from computer science and biology, regardless of their expertise level in programming. Specifically, VENUSFACTORY supports **efficient biological data retrieval** with multithreaded downloading and indexing from major biological databases, *e.g.*, `RCSB PDB` (Burley et al., 2019), `UniProt` (Consortium, 2025), `InterPro` (Paysan-Lafosse et al., 2023), and `AlphaFold DB` (Varadi et al., 2022). It also includes implementations for **comprehensive biological prediction tasks and evaluations** covering solubility, localization, function, and mutation prediction, compiled from 40+ protein-related datasets in a unified format. Moreover, VENUSFACTORY provides **effortless PLM implementations** for both pre-trained encoders (*e.g.*, ESM2 (Lin et al., 2023) and PROTTRANS (Elnaggar et al., 2021)) and downstream task fine-tuning (*e.g.*, `LoRA` series (Hu et al., 2022a; Dettmers et al., 2023; Liu et al., 2024), `Freeze` & `Full` fine-tuning, and `SES-Adapter` (Tan et al., 2024a) for protein-related tasks).

To the best of our knowledge, VENUSFACTORY is the most comprehensive engine for AI-driven protein engineering. It integrates extensive biological data resources, essential processing tools, state-of-the-art PLMs, and fine-tuning modules. It supports both Gradio-based web interface (Abid et al., 2019) and command-line execution, enabling researchers from both computer science and biology backgrounds to access and utilize its components effortlessly. Built on PyTorch (Paszke et al., 2019) and released under the CC-BY-NC-ND-4.0 license, VENUSFACTORY ensures broad accessibility and reproducibility, with all datasets and model checkpoints available on Hugging Face.

2 Data Collection

The first `Collection` module enables efficient data retrieval from four major protein databanks. This section outlines its core functionalities and implementation techniques, with additional details provided in Appendix E.

2.1 Databanks

VENUSFACTORY supports data collection from four well-established sources for protein sequences, structures, and functions. (1) `RCSB PDB` contains over $200,000$ experimentally determined atom-level protein 3D structures. (2) `UniProt` provides comprehensive amino acid sequences and functional annotations for over 250 million proteins curated literature and user submission. (3) `InterPro` assigns accession numbers and functional descriptions to $\sim 41,000$ proteins according to their family, domain, and functional site annotations. (4) `AlphaFold DB` hosts AlphaFold2-predicted 3D structure of proteins from UniProt. It enables structure retrieval by UniProt ID.

2.2 Multithreaded Downloading

The `Collection` module facilitates multithreaded data downloading by simulating `HTTP` requests using the `requests`, `fake_useragent`, and `concurrent` libraries. Data from `UniProt` (sequences) and `AlphaFold DB` (sequences and structures) can be accessed by UniProt IDs, *e.g.*, "*A0A0C5B5G6*". `RCSB PDB` is available in multiple formats, including `.cif`, `.pdb`, and `.xml`. All metadata are stored in `.json` format and indexed by the RCSB ID (*e.g.*, "*1A00*"). Queryable metadata fields including "*pubmed_id*" and "*assembly_ids*".

Essential	
aa_seq	Amino acid sequence, *e.g.*, *MASG...*
label	Target label, integer, float, or list, *e.g.*, *0*
Optional	
name	Unique Protein or Uniprot ID, *e.g.*, *P05798*
ss3_seq	3-class of DSSP sequence, *e.g.*, *CHHHH...*
ss8_seq	8-class of DSSP sequence, *e.g.*, *THLEH...*
foldseek_seq	Foldseek structure sequence, *e.g.*, *CVFLV...*
esm3_structure_seq	ESM3 structure sequence, *e.g.*, *[85, 3876, ...]*
detail or other	Auxiliary information or detailed description

Table 1: Benchmark dataset format example.

For `InterPro` family data, downloads can be performed using individual InterPro IDs or by parsing family `.json` files from the website. Retrieved data includes family descriptions (*e.g.*, "*pfam*" and "*go_terms*") as well as detailed protein annotations (*e.g.*, sequence fragments and gene information).

2.3 Structure Serialization

Protein structures are crucial for describing protein characteristics, yet structural information alone is often challenging to directly use as input for models like PLMs. VENUSFACTORY supports conversion tools that encode protein structures into discrete tokens. Three popular serialization methods are considered, including DSSP (Kabsch and Sander, 1983), FOLDSEEK (Van Kempen et al., 2024), and the ESM3 encoder (Hayes et al., 2025). DSSP converts structures into 3-class or 8-class secondary structure representations. FOLDSEEK employs VQ-VAE (van den Oord et al., 2017) to transform continuous structural data into 20-dimensional 3Di tokens. The ESM3 encoder constructs $4,096$-dimensional integer representations for local subgraphs centered on each amino acid.

3 Task Benchmarking

Assessing the predictive accuracy of protein representations extracted by PLMs is crucial for both developing new models and guiding biological applications. VENUSFACTORY integrates over 40 benchmark datasets from the literature and categorizes them into five major bioengineering tasks to help users gain a comprehensive understanding of common tasks and access relevant datasets. To enhance usability, we have standardized the data formats for all datasets (Table 1). We introduce the benchmark datasets for the five classes. Further details are provided in Appendix C.

3.1 Localization

Protein function is closely linked to its cellular compartment or organelle, where specific physiological conditions enable distinct activities. VENUSFACTORY curates and refines protein localization datasets from Almagro Armenteros et al. (2017) and Thumuluri et al. (2022), including (1) **DeepLocBinary**: a binary classification of membrane association, (2) **DeepLocMulti**: a multi-class classification for precise localization, and (3) **DeepLoc2Multi**: a multi-label, multi-class classification for complex localization scenarios. All three benchmarks include sequence data and AlphaFold2-predicted structures, with additional ESMFold-predicted structures available for **DeepLocBinary** and **DeepLocMulti**.

3.2 Solubility

Solubility is a prerequisite for proteins to function *in vitro*. However, many proteins, especially those engineered manually, often face solubility challenges. Therefore, it is crucial to predict the solubility of a protein of interest in terms of reducing experimental costs. VENUSFACTORY includes three binary classification benchmarks – **DeepSol** (Khurana et al., 2018), **DeepSoluE** (Wang and Zou, 2023), and **ProtSolM** (Tan et al., 2024d) – as well as one regression benchmark, **eSol** (Chen et al., 2021). All datasets include protein structures predicted by ESMFold, with **eSol** additionally providing AlphaFold2-predicted structures.

3.3 Annotation

Accurately predicting protein function is essential for understanding enzymatic activity, molecular interactions, and cellular roles in metabolism, signaling, and regulation (Zhou et al., 2024a). VENUSFACTORY includes four multi-class, multi-label prediction benchmarks from Su et al. (2024a): **EC**, which uses Enzyme Commission numbers (Bairoch, 2000) as function annotation labels; and **GO-CC**, **GO-BP**, and **GO-MF**, which employ Gene Ontology annotations (Ashburner et al., 2000). For all four benchmarks, protein structures are generated using AlphaFold2 and ESMFold.

3.4 Mutation

Mutating amino acids is a key approach in protein engineering for modifying protein function and properties, such as enzymatic activity, stability, selectivity, and molecular interactions. VENUSFACTORY includes a total of 19 benchmark datasets

Model	Fine-tuning	Localization			Solubility				Annotation			
		DL2M	DLB	DLM	DS	DSE	PSM	ES	EC	BP	CC	MF
ESM2-650M	Freeze	81.22	90.97	80.63	66.52	54.58	64.63	73.16	84.32	48.36	57.74	63.99
	LoRA	81.74	93.40	83.04	74.41	54.23	64.30	74.15	85.15	48.31	46.09	66.42
	SES-Adapter	80.00	93.50	82.90	75.51	54.23	65.88	72.47	84.80	46.63	52.59	63.38
Ankh-Large	Freeze	79.51	90.34	80.53	64.82	**55.52**	64.40	71.49	85.14	45.90	54.70	61.29
	LoRA	76.39	**93.69**	83.04	74.06	55.19	66.71	**76.16**	75.58	28.68	38.15	48.62
	SES-Adapter	81.11	92.71	82.93	73.16	55.13	66.59	69.12	86.03	47.54	49.64	64.48
ProtBert	Freeze	77.85	87.85	74.54	66.32	53.55	61.79	69.59	70.08	42.04	54.55	52.31
	LoRA	43.25	92.30	78.59	**75.81**	55.32	62.34	66.22	76.41	24.52	31.61	16.09
	SES-Adapter	78.85	92.71	77.57	74.76	54.94	62.34	67.07	76.56	41.47	49.52	54.58
ProtT5-XL-U50	Freeze	82.50	91.78	81.18	69.22	55.13	66.08	73.22	82.57	48.84	**59.07**	64.39
	LoRA	81.94	93.11	84.06	74.86	54.03	65.17	72.77	**87.35**	46.40	56.55	**67.35**
	SES-Adapter	**82.89**	92.71	**85.19**	75.26	54.94	**67.59**	73.11	84.56	**49.49**	56.86	65.11

Table 2: Performance comparison with highlighted best results of each model and **each task**. The detail and evaluation metrics of the dataset can be found in Appendix C.

with numeric labels, making them suitable for regression tasks. Specifically, we incorporate three enzyme solubility benchmarks from Tan et al. (2024b) (**PETA_TEM_Sol**, **PETA_CHS_Sol**, and **PETA_LGK_Sol**), fluorescence intensity and stability benchmark from Rao et al. (2019) (**TAPE_Fluorescence** and **TAPE_Stability**), as well as seven adeno-associated virus fitness benchmarks (**FLIP_AAV**) and five nucleotide-binding protein benchmarks (**FLIP_GB1**) from Dallago et al. (2021) with clearly defined splitting rules, such as one-vs-rest training and random sampling.

3.5 Other Properties

Beyond the commonly explored tasks and open benchmarks, we have curated five additional datasets that characterize other protein properties. One dataset focuses on stability prediction **Thermostability** (Su et al., 2024a). The second **DeepET_Topt** (Li et al., 2022) provides optimal temperature predictions for enzymes. Additionally, we include two binary classification tasks: **MetalIonBinding** (Hu et al., 2022b), which identifies metal ion-protein binding, and **SortingSignal** (Thumuluri et al., 2022), which detects sorting signals involved in protein localization. All datasets incorporate AlphaFold2-predicted structures. Furthermore, **Thermostability**, **DeepET_Topt**, and **SortingSignal** also include structures by ESMFold.

4 Model Application

While many PLMs have been developed, bridging them to biological applications requires applying them to downstream tasks. This involves seamlessly accessing pre-trained PLMs and integrating them with appropriate fine-tuning modules for task-specific training and inference. To facilitate this, VENUSFACTORY provides a dedicated `Application` module with specific architectures and optimization strategies to improve performance across diverse tasks.

4.1 Pre-trained PLMs

VENUSFACTORY supports fine-tuning across two primary categories of over 40 Transformer-based PLMs: Encoder-Only and Encoder-Decoder models. The Encoder-Only category includes both classic and state-of-the-art models, including ESM2 (ranging from 8M to 15B parameters) (Lin et al., 2023), ESM-1B (Rives et al., 2021), ESM-1V (Meier et al., 2021), PROTBERT (Elnaggar et al., 2021), IGBERT (Kenlay et al., 2024), PROSST (Li et al., 2024), PETA (Tan et al., 2024b),40+ and PROPRIME (Jiang et al., 2024). For Encoder-Decoder architectures, VENUSFACTORY incorporates models including the ANKH series (Elnaggar et al., 2023), PROTT5 (Elnaggar et al., 2021), and IGT5 (Kenlay et al., 2024). Further details can be found in Appendix A.

Collate Function When training a PLM, protein sequences are typically truncated based on batch size, similar to operations in NLP. However, proteins are complex systems where subtle token replacements can lead to significant functional and structural changes. Additionally, their intrinsic spatial characteristics introduce long-range dependencies between tokens. To address these factors, VENUSFACTORY supports not only conventional

Model	Fine-tuning	Mutation							Other			
		CHS	LGK	TEM	AAV	GB1	STA	FLU	SIG	MIB	DET	TMO
ESM2-650M	Freeze	26.68	27.74	13.93	70.58	71.48	68.33	45.32	88.72	67.82	67.15	68.85
	LoRA	35.66	30.17	30.37	93.75	93.96	78.16	50.69	90.09	73.38	60.59	**70.80**
	SES-Adapter	-	-	-	-	-	-	-	90.83	68.87	68.22	66.32
Ankh-Large	Freeze	32.33	41.23	20.33	69.23	76.32	67.54	52.50	84.41	75.49	64.31	66.52
	LoRA	37.48	36.27	20.52	93.89	94.60	62.95	68.13	87.63	74.07	64.84	69.68
	SES-Adapter	-	-	-	-	-	-	-	**91.35**	**78.35**	63.71	69.21
ProtBert	Freeze	13.49	20.50	15.51	65.96	67.26	65.35	43.73	84.83	66.77	64.83	65.58
	LoRA	19.22	10.56	14.09	94.05	94.41	75.11	42.85	87.22	68.42	64.82	67.05
	SES-Adapter	-	-	-	-	-	-	-	90.94	67.97	64.84	66.68
ProtT5-XL-U50	Freeze	37.58	38.78	31.10	63.62	75.52	74.50	48.46	88.17	75.79	69.15	69.15
	LoRA	43.84	27.06	34.68	94.09	95.13	83.50	66.00	89.13	76.69	67.42	68.46
	SES-Adapter	-	-	-	-	-	-	-	**91.35**	74.14	**70.70**	69.71

Table 3: Performance comparison with highlighted best results of each model and **each task**. The detail and evaluation metrics of the dataset can be found in Appendix C.

sequence truncation but also a non-truncating approach, which statistically determines an optimal token limit per batch to maintain sequence integrity during training.

Normalization We provide multiple normalization methods to enhance training stability and convergence. Supported options include Min-Max normalization, Z-score standardization, Robust normalization, Log transformation, and Quantile normalization.

4.2 Fine-tuning Modules

For fine-tuning pre-trained PLMs, VENUSFACTORY supports two classic approaches: `freeze` fine-tuning and `full` fine-tuning, along with various `LoRA`-based efficient training methods (Hu et al., 2022a; Dettmers et al., 2023; Liu et al., 2024) and a protein-specific SES-ADAPTER method (Tan et al., 2024a) (see Table 6 for a complete list). Specifically, `freeze` fine-tuning keeps PLM parameters fixed while updating only the readout layers, whereas `full` fine-tuning updates the entire model. `LoRA` and its variants enable parameter-efficient fine-tuning to reduce computational costs, and SES-ADAPTER employs cross-attention between PLM representations and sequence-structure embeddings (*e.g.*, from FOLDSEEK) to enhance protein-specific fine-tuning.

Classification Head VENUSFACTORY supports three classification heads: a two-layer fully connected network with average pooling, dropout, and GeLU activation; a lightweight head (Stärk et al., 2021) that combines 1D convolutional feature extraction with attention-weighted pooling for efficient sequence aggregation; and ATTENTION1D (Tan et al., 2024a) that employs masked 1D convolution-based attention pooling and a non-linear projection layer for multi-class classification.

4.3 Performance Assessment

Loss Function For model training and validation, various loss functions are selected based on the prediction task. `MSELoss` is used for regression tasks, `BCEWithLogitsLoss` is applied to multi-class and multi-label tasks, and `CrossEntropyLoss` is employed for the rest classification tasks.

Evaluation Metrics VENUSFACTORY supports a diverse set of evaluation metrics for robust assessment. For numeric labels, `Spearman's` ρ and `MSE` are used to evaluate ranking consistency and quantify prediction differences from the ground truth. For classification tasks, standard metrics such as `accuracy`, `precision`, `recall`, `F1-score`, `MCC`, and `AUROC` are included. Specifically, multi-label classification is assessed using the `F1-max score`. Further details are in Appendix D.

5 Experiments

We evaluate a range of models across various downstream tasks to demonstrate the practicality of VENUSFACTORY in integrating diverse models, benchmarks, and fine-tuning strategies. Appendix C provides additional information on the selected evaluation datasets, partitioning strategies, and monitored metrics.

5.1 Experimental Setup

All fine-tuning methods follow a standardized setup: Each batch is constrained to a maximum of $12,000$ tokens to accommodate long protein sequences, with gradient accumulation set to 8, effectively yielding a batch size of approximately 200. The AdamW optimizer (Loshchilov et al., 2017) is used with a learning rate of 0.0005. Training runs for a maximum of 100 epochs, with early stopping applied if no improvement is observed for 10 consecutive epochs. To ensure reproducibility, the random seed is set to 3407. For the SES-Adapter method, input structural sequences are derived from FoldSeek and DSSP 8-class representations. All experiments are conducted on a cluster of 20 RTX 3090 GPUs over two months.

5.2 Results

We evaluate different PLMs across multiple tasks using three fine-tuning strategies: `Freeze`, `LoRA` (vanilla), and SES-Adapter (Tables 2-3). SES-Adapter consistently outperforms other methods, particularly in solubility prediction (**DSE**, **PSM**) and mutation effect prediction (**AAV**, **GB1**). `LoRA` demonstrates strong performance in localization tasks and achieves the highest scores for **DLB**, but exhibits less consistency across solubility and annotation tasks. `Freeze` generally yields the lowest performance, especially in annotation tasks (**BP**, **MF**), but remains competitive in EC classification.

From a within-model perspective, ProtT5-XL-U50 achieves the highest overall performance, particularly excelling in annotation and mutation prediction, while Ankh-Large and ESM2-650M perform comparably but show task-dependent variations. In contrast, ProtBert underperforms in mutation prediction and certain annotation tasks, suggesting potential limitations in capturing functional variations. From a within-fine-tuning perspective, SES-Adapter consistently provides the best results across different models, demonstrating its robustness for protein-related tasks. `LoRA` exhibits strong performance in specific tasks, such as localization, but lacks stability across broader benchmarks. The `Freeze` method exhibits the largest performance gap across tasks, indicating that `full` fine-tuning or lightweight adaptation is essential for optimal PLM performance in protein engineering. These results highlight the importance of both model selection and fine-tuning strategies, emphasizing that the optimal configuration should be task-specific to maximize predictive accuracy and generalization.

Feature / Module	ProteusAI	SaProtHub	VenusFactory
$\geq$ 10 Built-in PLMs	✗	✗	✓
$\geq$ 30 Benchmark Datasets	✗	✗	✓
Data Retrieval Module	✗	✗	✓
No-code Web UI	✓	✓	✓
Structure-Sequence Integration	✗	✓	✓
Fine-tuning Method Diversity	✗	✗	✓
Model & Data Sharing	✗	✓	✓

Table 4: Comparison of features in VenusFactory with existing popular systems.

6 Related Work

The use of platforms for LLM fine-tuning and benchmarking has become a widely adopted routine in NLP to accommodate users with diverse domain expertise and programming backgrounds. LlamaFactory (Zheng et al., 2024), Janus (Chen et al., 2024) integrate multiple efficient fine-tuning methods with a no-code interface, while LLaMA-Adapter (Zhang et al., 2024b), FastChat (Zheng et al., 2023), and LMFlow (Diao et al., 2024) enable lightweight adaptation for instruction-following and multi-modal tasks.

In biology, existing systems primarily focus on protein data integration (Szklarczyk et al., 2019; Burley et al., 2019; Paysan-Lafosse et al., 2023; Consortium, 2025) and visualization (Humphrey et al., 1996; DeLano, 2002; Pettersen et al., 2004; Bobrov et al., 2024). For AI-driven protein engineering, only a few platforms offer specialized functionality. ProteusAI (Funk et al., 2024) streamlines the protein engineering pipeline by establishing an iterative cycle from mutant design to experimental feedback. SaProtHub (Su et al., 2024b), built upon SaProt (Su et al., 2024a), provides a Colab-based interface for model training and sharing. As shown in Table 4, VenusFactory is the first platform to support a broader range of PLMs and fine-tuning strategies while also incorporating database scraping and standardized benchmark construction, making it a comprehensive tool for protein-related AI applications.

7 Conclusion and Discussion

This work introduces VenusFactory, a versatile engine for unveiling biological systems, offering the most comprehensive resources to date for AI-driven protein engineering. By integrating data collection, benchmarking, and application modules for both pre-trained PLMs and fine-tuning strategies,

VENUSFACTORY enables researchers in computer science and computational biology to efficiently access open-source datasets and develop models for diverse protein-related tasks. Future iterations will expand its capabilities with generative modeling for *de novo* protein design, improved fine-tuning efficiency through advanced adaptation techniques, and broader protein function prediction tasks. We aim to provide a more accessible and powerful tool for researchers at the intersection of AI and biology, fostering innovation and discovery even with minimal computational expertise.

Limitations

While VENUSFACTORY provides a robust foundation, we acknowledge its current limitations. Presently, its primary focus is on predictive tasks such as classification and regression, with generative modeling and more specialized user-requested tasks (*e.g.*, interaction site prediction) planned for future development. It is also helpful to enhance UI/UX features, such as experiment configuration management and user guidance, particularly for those less familiar with PLM hyperparameters. Furthermore, the platform's scalability on extremely large models or datasets warrants further investigation and optimization. Addressing these points will be central to our future development efforts.

Ethics Statement

VENUSFACTORY aims to foster significantly broader impact by democratizing access to powerful PLMs, enabling researchers to accelerate discovery in beneficial areas like drug design and enzyme engineering. However, we acknowledge the inherent dual-use risks associated with technologies that simplify biological engineering. While not its intent, the platform's accessibility could potentially lower the threshold for misuse, such as in the modification of pathogens. Therefore, we emphasize the critical importance of responsible use. We release VENUSFACTORY as an open-source tool to encourage transparency and community oversight, and we urge all users to strictly adhere to all applicable ethical guidelines and biosecurity protocols in their research.

Acknowledgements

This work was supported by the grants from the National Key Research and Development Program of China (2024YFA0917603), National Science Foundation of China (Grant Number 62302291), and Computational Biology Key Program of Shanghai Science and Technology Commission (23JS1400600).

References

Abubakar Abid, Ali Abdalla, Ali Abid, Dawood Khan, Abdulrahman Alfozan, and James Zou. 2019. Gradio: Hassle-free sharing and testing of ml models in the wild. *arXiv:1906.02569*.

José Juan Almagro Armenteros, Casper Kaae Sønderby, Søren Kaae Sønderby, Henrik Nielsen, and Ole Winther. 2017. DeepLoc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, 33(21):3387–3395.

Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. 2000. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29.

Amos Bairoch. 2000. The ENZYME database in 2000. *Nucleic Acids Research*, 28(1):304–305.

Artem Bobrov, Domantas Saltenis, Zhaoyue Sun, Gabriele Pergola, and Yulan He. 2024. DrugWatch: A comprehensive multi-source data visualisation platform for drug safety information. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 180–189, Bangkok, Thailand. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Stephen K Burley, Helen M Berman, Charmi Bhikadiya, Chunxiao Bi, Li Chen, Luigi Di Costanzo, Cole Christie, Ken Dalenberg, Jose M Duarte, Shuchismita Dutta, et al. 2019. RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Research*, 47(D1):D464–D474.

Jianwen Chen, Shuangjia Zheng, Huiying Zhao, and Yuedong Yang. 2021. Structure-aware protein solubility prediction from sequence through graph convolutional network and predicted contact map. *Journal of cheminformatics*, 13:1–10.

Xiaoyi Chen, Siyuan Tang, Rui Zhu, Shijun Yan, Lei Jin, Zihao Wang, Liya Su, Zhikun Zhang, XiaoFeng Wang, and Haixu Tang. 2024. The Janus interface: How fine-tuning in large language models amplifies the privacy risks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 1285–1299.

UniProt Consortium. 2025. UniProt: the universal protein knowledgebase in 2025. *Nucleic Acids Research*, 53(D1):D609–D617.

Christian Dallago, Jody Mou, Kadina E Johnston, Bruce Wittmann, Nick Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K Yang. 2021. FLIP: Benchmark tasks in fitness landscape inference for proteins. In *Advance in Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Warren L DeLano. 2002. Pymol: An open-source molecular graphics tool. *CCP4 Newsl. Protein Crystallogr*, 40(1):82–92.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Shizhe Diao, Rui Pan, Hanze Dong, KaShun Shum, Jipeng Zhang, Wei Xiong, and Tong Zhang. 2024. LMFlow: An extensible toolkit for finetuning and inference of large foundation models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: System Demonstrations)*, pages 116–127, Mexico City, Mexico. Association for Computational Linguistics.

Ahmed Elnaggar, Hazem Essam, Wafaa Salah-Eldin, Walid Moustafa, Mohamed Elkerdawy, Charlotte Rochereau, and Burkhard Rost. 2023. Ankh: Optimized protein language model unlocks general-purpose modelling. *arXiv:2301.06568*.

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

Jonathan Funk, Laura Machado, Samuel A. Bradley, Marta Napiorkowska, Rodrigo Gallegos-Dextre, Liubov Pashkova, Niklas G. Madsen, Henry Webel, Patrick V. Phaneuf, Timothy P. Jenkins, and Carlos G. Acevedo-Rocha. 2024. Proteusai: An open-source and user-friendly platform for machine learning-guided protein design and engineering. In *bioRxiv*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv:2501.12948*.

Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J. Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q. Tran, Jonathan Deaton, Marius Wiggert, Rohil Badkundri, Irhum Shafkat, Jun Gong, Alexander Derry, Raul S. Molina, Neil Thomas, Yousuf A. Khan, Chetan Mishra, Carolyn Kim, Liam J. Bartie, Matthew Nemeth, Patrick D. Hsu, Tom Sercu, Salvatore Candido, and Alexander Rives. 2025. Simulating 500 million years of evolution with a language model. *Science*, 0(0):eads0018.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022a. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Mingyang Hu, Fajie Yuan, Kevin K Yang, Fusong Ju, Jin Su, Hui Wang, Fei Yang, and Qiuyang Ding. 2022b. Exploring evolution-aware & -free protein language models as protein function predictors. In *Advances in Neural Information Processing Systems*.

William Humphrey, Andrew Dalke, and Klaus Schulten. 1996. VMD: visual molecular dynamics. *Journal of Molecular Graphics*, 14(1):33–38.

Fan Jiang, Mingchen Li, Jiajun Dong, Yuanxi Yu, Xinyu Sun, Banghao Wu, Jin Huang, Liqi Kang, Yufeng Pei, Liang Zhang, et al. 2024. A general temperature-guided language model to design proteins of enhanced stability and activity. *Science Advances*, 10(48):eadr2641.

Wolfgang Kabsch and Christian Sander. 1983. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Research on Biomolecules*, 22(12):2577–2637.

Liqi Kang, Banghao Wu, Bingxin Zhou, Pan Tan, Yun Kang, Yongzhen Yan, Yi Zong, Shuang Li, Zhuo Liu, and Liang Hong. 2025. AI-enabled alkaline-resistant evolution of protein to apply in mass production. *eLife*, 13:RP102788.

Henry Kenlay, Frédéric A Dreyer, Aleksandr Kovaltsuk, Dom Miketa, Douglas Pires, and Charlotte M Deane. 2024. Large scale paired antibody language models. *PLOS Computational Biology*, 20(12):e1012646.

Sameer Khurana, Reda Rawi, Khalid Kunji, Gwo-Yu Chuang, Halima Bensmail, and Raghvendra Mall. 2018. Deepsol: a deep learning framework for sequence-based protein solubility prediction. *Bioinformatics*, 34(15):2605–2613.

Gang Li, Filip Buric, Jan Zrimec, Sandra Viknander, Jens Nielsen, Aleksej Zelezniak, and Martin KM Engqvist. 2022. Learning deep representations of enzyme thermal adaptation. *Protein Science*, 31(12):e4480.

Mingchen Li, Yang Tan, Xinzhu Ma, Bozitao Zhong, Huiqun Yu, Ziyi Zhou, Wanli Ouyang, Bingxin Zhou, Pan Tan, and Liang Hong. 2024. ProSST: Protein language modeling with quantized structure and disentangled attention. In *Advances in Neural Information Processing Systems*.

Song Li, Yang Tan, Song Ke, Liang Hong, and Bingxin Zhou. 2025. Immunogenicity prediction with dual attention enables vaccine target selection. In *The Thirteenth International Conference on Learning Representations*.

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

Chen Liu, Mingchen Li, Yang Tan, Wenrui Gou, Guisheng Fan, and Bingxin Zhou. 2025. Sequence-only prediction of binding affinity changes: A robust and interpretable model for antibody engineering. *arXiv:2505.20301*.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*.

Ilya Loshchilov, Frank Hutter, et al. 2017. Fixing weight decay regularization in adam. *arXiv:1711.05101*, 5.

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

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alex Rives. 2021. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in Neural Information Processing Systems*, 34:29287–29303.

Jie Pan. 2023. Large language model for molecular chemistry. *Nature Computational Science*, 3(1):5–5.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32.

Typhaine Paysan-Lafosse, Matthias Blum, Sara Chuguransky, Tiago Grego, Beatriz Lázaro Pinto, Gustavo A Salazar, Maxwell L Bileschi, Peer Bork, Alan Bridge, Lucy Colwell, et al. 2023. InterPro in 2022. *Nucleic Acids Research*, 51(D1):D418–D427.

Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Gregory S Couch, Daniel M Greenblatt, Elaine C Meng, and Thomas E Ferrin. 2004. UCSF Chimera—a visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25(13):1605–1612.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. 2019. Evaluating protein transfer learning with tape. *Advances in Neural Information Processing Systems*, 32.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. 2021. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118.

Hannes Stärk, Christian Dallago, Michael Heinzinger, and Burkhard Rost. 2021. Light attention predicts protein location from the language of life. *Bioinformatics Advances*, 1(1):vbab035.

Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. 2024a. SaProt: Protein language modeling with structure-aware vocabulary. In *The Twelfth International Conference on Learning Representations*.

Jin Su, Zhikai Li, Chenchen Han, Yuyang Zhou, Yan He, Junjie Shan, Xibin Zhou, Xing Chang, Shiyu Jiang, Dacheng Ma, The OPMC, Martin Steinegger, Sergey Ovchinnikov, and Fajie Yuan. 2024b. SaprotHub: Making protein modeling accessible to all biologists. In *bioRxiv*.

Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. 2019. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Research*, 47(D1):D607–D613.

Yang Tan, Mingchen Li, Bingxin Zhou, Bozitao Zhong, Lirong Zheng, Pan Tan, Ziyi Zhou, Huiqun Yu, Guisheng Fan, and Liang Hong. 2024a. Simple, efficient, and scalable structure-aware adapter boosts protein language models. *Journal of Chemical Information and Modeling*.

Yang Tan, Mingchen Li, Ziyi Zhou, Pan Tan, Huiqun Yu, Guisheng Fan, and Liang Hong. 2024b. PETA: evaluating the impact of protein transfer learning with sub-word tokenization on downstream applications. *Journal of Cheminformatics*, 16(1):92.

Yang Tan, Ruilin Wang, Banghao Wu, Liang Hong, and Bingxin Zhou. 2024c. Retrieval-enhanced mutation mastery: Augmenting zero-shot prediction of protein language model. *arXiv:2410.21127*.

Yang Tan, Jia Zheng, Liang Hong, and Bingxin Zhou. 2024d. ProtSolM: Protein solubility prediction with multi-modal features. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 223–232. IEEE.

Yang Tan, Bingxin Zhou, Lirong Zheng, Guisheng Fan, and Liang Hong. 2025. Semantical and geometrical protein encoding toward enhanced bioactivity and thermostability. *eLife*, 13:RP98033.

Vineet Thumuluri, José Juan Almagro Armenteros, Alexander Rosenberg Johansen, Henrik Nielsen, and Ole Winther. 2022. DeepLoc 2.0: multi-label subcellular localization prediction using protein language models. *Nucleic Acids Research*, 50(W1):W228–W234.

Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. 2017. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, et al. 2022. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Research*, 50(D1):D439–D444.

Chao Wang and Quan Zou. 2023. Prediction of protein solubility based on sequence physicochemical patterns and distributed representation information with deepsolue. *BMC Biology*, 21(1):12.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2024a. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*.

Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024b. LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advance in Neural Information Processing Systems Datasets and Benchmarks Track*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

Bingxin Zhou, Yang Tan, Yutong Hu, Lirong Zheng, Bozitao Zhong, and Liang Hong. 2024a. Protein engineering in the deep learning era. *mLife*, 3(4):477–491.

Bingxin Zhou, Lirong Zheng, Banghao Wu, Yang Tan, Outongyi Lv, Kai Yi, Guisheng Fan, and Liang Hong. 2024b. Protein engineering with lightweight graph denoising neural networks. *Journal of Chemical Information and Modeling*.

Bingxin Zhou, Lirong Zheng, Banghao Wu, Kai Yi, Bozitao Zhong, Yang Tan, Qian Liu, Pietro Liò, and Liang Hong. 2024c. A conditional protein diffusion model generates artificial programmable endonuclease sequences with enhanced activity. *Cell Discovery*, 10(1):95.

Model	# Params.	Num.	Type	Implement
ESM2 (Lin et al., 2023)	8M-15B	6	Encoder	facebook/esm2_t33_650M_UR50D
ESM-1b (Rives et al., 2021)	650M	1	Encoder	facebook/esm1b_t33_650M_UR50S
ESM-1v (Meier et al., 2021)	650M	5	Encoder	facebook/esm1v_t33_650M_UR90S_1
ProtBert-Uniref100 (Elnaggar et al., 2021)	420M	1	Encoder	Rostlab/prot_bert_Uniref100
ProtBert-BFD100 (Elnaggar et al., 2021)	420M	1	Encoder	Rostlab/prot_bert_bfd
IgBert (Kenlay et al., 2024)	420M	1	Encoder	Exscientia/IgBert
IgBert_unpaired (Kenlay et al., 2024)	420M	1	Encoder	Exscientia/IgBert_unpaired
ProtT5-Uniref50 (Elnaggar et al., 2021)	3B/11B	2	Encoder-Decoder	Rostlab/prot_t5_xl_uniref50
ProtT5-BFD100 (Elnaggar et al., 2021)	3B/11B	2	Encoder-Decoder	Rostlab/prot_t5_xl_bfd
Ankh (Elnaggar et al., 2023)	450M/1.2B	2	Encoder-Decoder	ElnaggarLab/ankh-base
ProSST (Li et al., 2024)	110M	7	Encoder	AI4Protein/ProSST-2048
ProPrime (Jiang et al., 2024)	690M	1	Encoder	AI4Protein/Prime_690M
PETA (Tan et al., 2024b)	80M	15	Encoder	AI4Protein/deep_base

Table 5: Detail of PLMs in terms of parameters, architecture, and implementation sources.

Fine-tunning Method	Type
Freeze	Sequence
Full	Sequence
LoRA (Hu et al., 2022a)	Sequence
DoRA (Liu et al., 2024)	Sequence
AdaLoRA (Zhang et al., 2024a)	Sequence
IA3 (Liu et al., 2022)	Sequence
QLoRA (Dettmers et al., 2023)	Sequence
SES-Adapter (Tan et al., 2024a)	Sequence & Structure

Table 6: Supported fine-tuning methods with data modality compatibility.

Model	Fine-tuning	Params. (M)	Ratio (%)
ESM2-650M	Freeze	1.66	0.25
	LoRA	3.67	0.56
	SES-Adapter	14.86	2.23
Ankh-Large	Freeze	2.38	0.21
	LoRA	5.31	0.46
	SES-Adapter	21.71	1.85
ProtBert	Freeze	1.06	0.25
	LoRA	2.53	0.60
	SES-Adapter	9.52	2.22
ProtT5-XL-U50	Freeze	1.05	0.09
	LoRA	4.00	0.33
	SES-Adapter	9.71	0.80

Table 7: The trainable parameters of different models using different fine-tuning methods and their proportion in the total model.

A Models

Table 5 presents an overview of popular PLMs used in computational biology and protein engineering.

B Training Methods

B.1 Supported Methods

Table 6 provides an overview of fine-tuning methods used for PLMs, categorized by their adaptation approach.

B.2 Training Parameters

Table 7 compares the number of trainable parameters and their relative proportion in different PLMs when applying various fine-tuning methods.

C Evaluated Benchmark Datasets

Table 8 summarizes datasets used for training and evaluating PLMs. The columns provide details on training, validation, and test splits, evaluation metrics (*e.g.*, accuracy, F1-score, Spearman's correlation), and implementation sources. Additionally, the mean and standard deviation of AlphaFold2 (AF2) and ESMFold (EF) predicted confidence scores (pLDDT) are reported. For **FLIP_AAV** and **FLIP_GF1**, we only selected the sampled partitioning method for testing.

D Metrics

Table 9 lists the supported evaluation metrics, abbreviations, and corresponding problem types.

E Collection

E.1 Introduction

Collection is designed for automated extraction of protein-related data from InterPro, RCSB PDB, UniProt, and AlphaFold DB. It supports structured metadata, sequence information, and 3D structural data retrieval, streamlining large-scale protein engineering research[2].

[2] https://github.com/AI4Protein/VenusFactory/blob/main/download/README.md

Dataset	AF2_pLDDT	EF_pLDDT	Train	Valid	Test	Metrics	Implement
Localization							
DeepLoc2Multi (DL2M)	$77.46_{(12.51)}$	-	21,948	2,744	2,744	f1_max	AI4Protein/DeepLoc2Multi
DeepLocBinary (DLB)	$79.57_{(12.06)}$	$77.10_{(14.62)}$	5,735	1,009	1,728	accuracy	AI4Protein/DeepLocBinary
DeepLocMulti (DLM)	$77.34_{(12.77)}$	$74.88_{(15.23)}$	9,324	1,658	2,742	accuracy	AI4Protein/DeepLocMulti
Solubility							
DeepSol (DS)	-	$79.59_{13.36}$	62,478	6,942	2,001	accuracy	AI4Protein/DeepSol
DeepSoluE (DSE)	-	$80.68_{(12.79)}$	10,290	1,143	3,100	accuracy	AI4Protein/DeepSoluE
ProtSolM (PSM)	-	$73.80_{(15.51)}$	57,725	3,210	3,208	accuracy	AI4Protein/ProtSolM
eSOL (ES)	$90.79_{(7.07)}$	$83.45_{(10.39)}$	2,481	310	310	Spearman's ρ	AI4Protein/eSOL
Annoation							
EC	$92.78_{(6.42)}$	$85.08_{(8.48)}$	13,090	1,465	1,604	f1_max	AI4Protein/EC
GO_MF (MF)	$91.77_{(6.68)}$	$82.84_{(9.68)}$	22,081	2,432	3,350	f1_max	AI4Protein/GO_MF
GO_BP (BP)	$91.35_{(7.06)}$	$82.00_{(10.65)}$	20,947	2,334	3,350	f1_max	AI4Protein/GO_BP
GO_CC (CC)	$90.07_{(8.05)}$	$79.57_{(11.61)}$	9,552	1,092	3,350	f1_max	AI4Protein/GO_CC
Mutation							
PETA_CHS_Sol (CHS)	-	-	3,872	484	484	Spearman's ρ	AI4Protein/PETA_CHS_Sol
PETA_LGK_Sol (LGK)	-	-	15,308	1,914	1,914	Spearman's ρ	AI4Protein/PETA_LGK_Sol
PETA_TEM_Sol (TEM)	-	-	6,445	808	808	Spearman's ρ	AI4Protein/PETA_TEM_Sol
FLIP_AAV_sampled (AAV)	-	-	66,066	16,517	16,517	Spearman's ρ	AI4Protein/FLIP_AAV_sampled
FLIP_GB1_sampled (GB1)	-	-	6,988	1,745	1,745	Spearman's ρ	AI4Protein/FLIP_GB1_sampled
TAPE_Stablity (STA)	-	-	53,614	2,512	12,851	Spearman's ρ	AI4Protein/TAPE_Stability
TAPE_Fluorescence (FLU)	-	-	21,446	5,362	27,217	Spearman's ρ	AI4Protein/TAPE_Fluorescence
Other							
MetalIonBinding (MIB)	$92.36_{(6.43)}$	$83.66_{(8.73)}$	5,068	662	665	accuracy	AI4Protein/MetalIonBinding
Thermostability (TMO)	$79.02_{(12.26)}$	$74.60_{(13.82)}$	5,054	639	1,336	Spearman's ρ	AI4Protein/Thermostability
DeepET_Topt (DET)	$92.98_{(5.32)}$	$85.18_{(8.74)}$	1,478	185	185	Spearman's ρ	AI4Protein/DeepET_Topt
SortingSignal (SIG)	$81.09_{(11.66)}$	-	1,484	185	186	f1_max	AI4Protein/SortingSignal

Table 8: Overview of the selected datasets for evaluating, including localization, solubility, annotation, mutation effects, and other properties. The table lists dataset sizes, evaluation metrics, and pLDDT from `AlphaFold2` and `ESMFold`, with standard deviations in parentheses.

Short Name	Metrics Name	Problem Type
accuracy	Accuracy	single/multi-label cls
recall	Recall	single/multi-label cls
precision	Precision	single/multi-label cls
f1	F1Score	single/multi-label cls
mcc	MatthewsCorrCoef	single/multi-label cls
auc	AUROC	single/multi-label cls
f1_max	F1ScoreMax	multi-label cls
spearman_corr	SpearmanCorrCoef	regression
mse	MeanSquaredError	regression

Table 9: Supported metrics with abbreviations. "Single-label cls" refers to single-label classification tasks, while "multi-label cls" refers to classification tasks where multiple labels can be assigned to each instance.

E.2 Implementation and Workflow

Implemented in Python, **Collection** leverages `requests` for API interactions and multiprocessing for parallel processing. It supports both single and batch retrieval via `.txt` or `.json` input. The workflow consists of input parsing, data fetching, data processing, and file storage, with structured output in `.fasta`, `.json`, `.pdb`, and `.mmCIF` formats. API requests include error handling with automatic retries to manage rate limits and network failures.

E.3 Data Organization

Output is stored hierarchically, with metadata, sequences, and structures categorized for easy access. For instance, `InterPro` metadata includes domain details (`detail.json`), accession metadata (`meta.json`), and associated `UniProt` IDs (`uids.txt`). UniProt sequences are saved in `.fasta` format, with an option to merge entries, while AlphaFold structures are organized by ID prefix for optimized storage.

E.4 Error Handling and Logging

Collection logs failed downloads in "failed.txt", recording network timeouts, missing IDs, and API errors for debugging and reattempts. Parallel downloading, caching, and adaptive rate limiting enhance retrieval efficiency, reducing redundant API calls and optimizing request frequency.