ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230481

Active Finetuning Protein Language Model: A Budget-Friendly Method for Directed Evolution

Ming Qin[a,b,c], Keyan Ding[a,b,c;*], Bin Wu[a,b], Zhenping Li[d], Haihong Yang[a,b,c], Zeyuan Wang[a,b,c], Hongbin Ye[a,e], Haoran Yu[a,c], Huajun Chen[a,b,c] and Qiang Zhang[a,b,c;]**

[a]Zhejiang University
[b]ZJU-Hangzhou Global Scientific and Technological Innovation Center
[c]AZFT Joint Lab for Knowledge Engine
[d]University of Chinese Academy of Sciences
[e]Zhejiang Lab

Abstract. Directed evolution is a widely-used strategy of protein engineering to improve protein function via mimicking natural mutation and selection. Machine learning-assisted directed evolution (MLDE) approaches aim to learn a fitness predictor, thereby efficiently searching for optimal mutants within the vast combinatorial mutation space. Since annotating mutants is both costly and labor-intensive, how to efficiently sample and utilize informative protein mutants to train the predictor is a critical problem in MLDE. Previous MLDE works just simply utilized pre-trained protein language models (PPLMs) for sampling without tailoring to the specific target protein of interest, which has not fully exploited the potential of PPLMs. In this work, we propose a novel method, the **A**ctively-**F**inetuned **P**rotein language model for **D**irected **E**volution(**AFP-DE**), which leverages PPLMs to actively sample and fine-tune themselves, continuously improving the model's sampling and overall performance through iterations, to achieve efficient directed protein evolution. Extensive experiments have shown the effectiveness of our method in generating optimal mutants with minimal annotation effort, outperforming previous works even with fewer annotated mutants, making it budget-friendly for biological experiments.

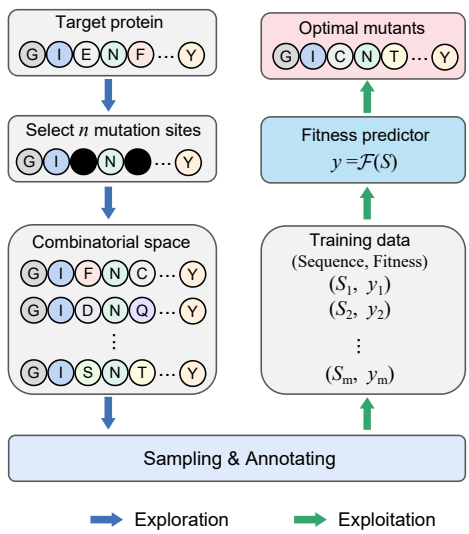

Figure 1: The paradigm of the machine learning-assisted directed evolution method.

1 Introduction

Deep learning has been widely applied to protein-related studies, including 3D structure prediction [8], function annotation [30], amino acid sequence generation, and protein design [1]. Among these studies, one challenging task is to obtain a protein that possesses the desired functions (or fitness in a biological expression), such as its expression level or catalytic activity. Due to the huge space of protein sequences and the costly biochemical experiments, it is infeasible to brute-force search for a protein with the highest fitness. To perform this task, researchers from biological science often rely on the technology of directed evolution (DE), which mimics the natural mutation and selection process [12].

Directed evolution [26, 22, 2] was awarded the Nobel Prize in Chemistry in 2018. As a fundamental approach in protein engineering, DE has been widely used in biological laboratories and the industrial development of protein products, such as enzymes and antibody, and thus attracted great research interest. General DE works by optimizing an existing wild-type protein, via mutating amino acids at some key sites while selecting high-fitness mutants [12]. Intuitively, DE aims to find the global optimum from a combinatorial space of protein mutants. That is, there are twenty different types of amino acid candidates for each mutation site and in total 20^n for the n mutation sites of a protein. This vast combinatorial space with sparse functioning sequences poses a great challenge to the traditional DE methods, which mainly rely on biological experiments. DE is still a formidable task due to the vast combinatorial space and the complex landscape of protein sequence-fitness mapping.

To efficiently explore such a combinatorial space, recently many researchers have tried machine learning-assisted directed evolution (MLDE) methods [32]. These methods usually consist of two stages: exploration and exploitation (see Figure 1). In the exploration stage, one aims to collect some informative samples from the whole combinatorial mutation space for training the sequence-fitness prediction

* Ming Qin and Keyan Ding contributed equally to this work. Email: qinandming@zju.edu.cn
** Corresponding Author. Email: qiang.zhang.cs@zju.edu.cn

model as a cheap surrogate of the expensive oracle; while in the exploitation stage, one aims to predict the global landscape of fitness values, such that mutants with high fitness can be screened. Since biological experiments are expensive and time-consuming, how to select more informative mutant samples for model training is a key problem in MLDE.

Protein language models are typically pre-trained on large-scale, unlabeled protein sequence data. The learned latent semantic knowledge allows for applying PPLMs to protein sampling and downstream property prediction tasks. However, existing MLDE methods have not fully utilized the potential of PPLMs. Specifically, previous works still simply use protein language models for sampling based on masking mechanisms [27] or only take PPLMs as an encoder to provide protein embeddings that assist in sampling[14, 15]. However, directly employing pre-trained PPLMs for mutant sampling is suboptimal, as they are trained on general protein databases and are not tailored to the specific target protein of interest. Therefore, how to efficiently fine-tune PPLMs on the target protein, and then achieve cost-effective and efficient directed protein evolution, is practically important but has not been fully explored.

To fill these gaps, we propose an actively-finetuned protein model for directed evolution (AFP-DE), which leverages PPLMs to actively sample and fine-tune itself, continuously improving the model's predictive performance through iterations, to achieve budget-friendly and efficient directed protein evolution. as shown in Figure 2, which includes the following three stages:

(1) **Exploration.** We employ a PPLM as the sampler to screen the informative mutants. Specifically, given a target protein sequence, we first mask the potential mutation sites and employ the PPLM to predict the probability distribution of masked tokens. We then use the metric of optimal transport distance to actively sample a small batch of informative mutants whose distribution is similar to the predicted distribution of masks at mutation positions and maintain sufficient diversity between each round.

(2) **Exploitation.** We send the screened samples to the oracle (i.e., biological fitness experiments) for annotation, and then use them to train a fitness predictor with a regression loss. The predictor is composed of the PPLM inherited from the previous stage and a multilayer perceptron (MLP), where the former plays as a sequence feature extractor and the latter is a prediction head.

(3) **Refinement.** We select a large number of predicted high-fitness mutants to refine the PPLM sampler with self-supervised learning (i.e., masked token prediction).

It is worth noting that we iteratively conduct the exploration-exploitation-refinement procedure such that the optimized PPLM can be gradually adapted from the general protein domain to the specific protein of interest, steadily improving the fitness prediction performance. We conduct *in silico* evolution experiments on the GB1 [28] and PhoQ [13] protein datasets and a biological wet experiment on the protein CNFRS [33]. The results show that our method outperforms the previous works even with fewer mutant samples.

The contributions of this paper are summarized as follows:

- We propose to actively finetune a pre-trained protein model for direct evolution, which is a budget-friendly method to tailor a PPLM to a specific target protein of interest.
- We propose to use a pre-trained model as the sampler and design an optimal transport distance-based sampling strategy to identify informative mutants that are both representative and diverse.
- We conduct extensive experiments showing that we can find the desired mutant with fewer annotated training data than state-of-the-art baselines.

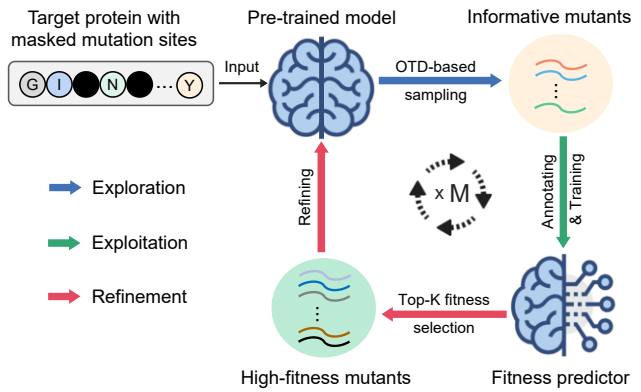

Figure 2: The framework of the proposed AFP-DE method. Starting from a target protein with masked mutation sites, a pre-trained protein language model is utilized to sample informative mutants based on optimal transport distance (OTD). Those sampled mutants are annotated and used to train a fitness predictor. The predicted high-fitness mutants are then used to refine the pre-trained model. We iteratively implement the above procedures M rounds depending on the maximum sampling number.

2 Related Works

Machine Learning-assisted Directed Evolution With the success of machine learning in modeling complex functions in computer vision and natural languages, researchers have tried some machine learning methods to assist the directed evolution of proteins, allowing *in silico* screening of all mutant candidates [32]. As a typical black-box optimization problem, directed evolution has often been tackled in an exploration-exploitation framework, similar to Bayesian optimization. During exploration, Qiu *et al.* [14] developed a hierarchical clustering method to sample informative mutants and further designed multiple evolutionary scores to enhance the initial sampling [15]. Yuan *et al.* [34] developed a Thompson sampling method for sequence optimization, with a simple Bayesian linear model. As for exploitation, the screened samples are annotated and used to train a machine-learning model for fitness prediction. The trained model evaluates the fitness of all mutant candidates and keeps those with high predicted fitness values. Recently, Wittman *et al.* [27] systematically examined the usefulness of three protein-encoding strategies (one-hot, physiochemical, and learned embeddings), and 22 fitness prediction models including XGBoost, 1D convolutional network, and Long Short-Term Memory. They have also shown that the zero-shot prediction of the PPLM-based mask-filling protocol has a relatively high correlation with true fitness.

However, the above methods need to empirically predefine the number of samples, e.g., 384 in CLADE [14]. A small sampling size may be insufficient to train a robust fitness predictor, while a large one can result in excessive costs for annotating mutants. Moreover, all of them have not considered the diversity of selected data, which lead to the sampling strategy inefficiently. To alleviate these problems, we propose to use an active finetuning method with PPLM to sample the informative mutants.

Pre-trained Protein Language Model As pre-trained language models have been proven effective in natural language processing [4], researchers try to extend such models to proteins. Rives *et al.* [20] firstly explored whether the Transformer architecture can be used to deal with proteins and find that the features learned by

PPLMs contribute to the structure prediction performance. To figure out why PPLMs work, Vig *et al.* [25] focused on reconciling attention with known protein properties and found that different layers can capture different structural information. To incorporate co-evolutionary signals from Multiple Sequence Alignment (MSA), Rao *et al.* [18] developed the MSA-Transformer and showed that homologous protein sequences can provide conformational information and promote contact prediction performance. Zhang *et al.* [35] introduced a Co-evolution Transformer that considers the relationship between MSAs and the target protein and mitigates the influence of non-homologous information. Both models mine co-evolutionary information from homologous protein sequences which have similar amino acid sequences and achieve promising performance.

Previous studies have mainly addressed both unsupervised pretraining and supervised finetuning, but little attention has been given to leveraging the available annotation budget for finetuning. Besides, in practical applications of protein engineering, considering the limited budget for annotating mutants, the problem of which mutant protein should be prioritized for annotation is a common issue. To address these gaps, this paper proposes an actively-finetuned protein model for the directed evolution method. By actively sampling with a PPLM and finetuning the PPLM itself during the iterative process, the method achieves more cost-effective and efficient directed protein evolution.

3 Problem Formulation

Machine Learning-assisted Directed Evolution Generally, a protein is made up of a sequence of amino acid tokens, i.e., $S = \{s_1, \ldots, s_N\} \in \mathbb{S}$, where each token is one of the 20 types of amino acids, N is the sequence length and $\mathbb{S}$ is the combinatorial space which refers to all possible mutants at multiple positions. Directed evolution is formulated as a black-box optimization task to identify the optimal amino acid sequence S^* with the highest fitness in the combinatorial space,

$$S^* = \arg\max_{S \in \mathbb{S}} \mathcal{F}(S), \tag{1}$$

where $\mathcal{F} : \mathbb{S} \to \mathbb{R}$ is the sequence-fitness predictor function. Various machine learning models have been developed to approximate this function $\mathcal{F}$, as described in Section 2.

The major technical challenge comes from the fact that the combinatorial space of n ($n < N$) mutational sites for the target protein S has 20^n unique sequences, preventing brute-force searching the optimal mutant. To search for the global optimum, mutations are sequentially queried by an oracle (i.e., biological fitness experiments). The financial costs of biological experiments require an efficient query strategy such that the global optimal sequence can be identified with the least effort. Mathematically, we aim to learn $\mathcal{F}(\cdot)$ with the least annotated data.

Active Finetuning We formally formulate the active finetuning mechanism of a pre-trained model. There is a pre-trained model $\mathcal{G}_\theta : \mathbb{S} \to \mathbb{R}^C$, with the parameters θ, where $\mathbb{S}$ is the input sequence data space and $\mathbb{R}^C$ is the normalized high-dimensional embedding space. There is also a large unlabeled dataset $\mathbb{S}_u = \{S^{(i)}\}_{i \in [N_u]}$, where N_u is the combinatorial mutation space size in the MLDE task. Active finetuning first selects informative data from $\mathbb{S}_u$ through a sampling strategy, leading to an annotated subset $\hat{\mathbb{S}} = \{\hat{S}^{(i)}\}_{i \in [N_c]}$, where N_c is the annotation budget size. Then $\mathcal{G}_\theta$ is finetuned on $\hat{\mathbb{S}}$ in a supervised manner. The above steps can be iteratively executed to update the model parameters θ until optimal performance is achieved.

4 Methodology

4.1 Sampling Informative Mutants

A key problem in MLDE is how to select the most informative mutant samples for annotating and training. Here we present a pretraining-based sampler with optimal transport distance (OTD), which utilizes the mask-filling protocol of PPLM and the OTD of intra-group and inter-group candidates.

Mask-Filling Protocol PPLMs are often trained with a protocol of masked-token prediction [4]. That is, given a protein sequence with the masks at some positions, we can obtain the probability distribution of each masked amino acid. Specifically, we first replace the n expert-chosen mutation sites with `[MASK]` symbols, and the protein sequence can be represented as

$$S' = [s_1, \ldots, \texttt{[MASK]}_j, \ldots, \texttt{[MASK]}_n, \ldots, s_N]. \tag{2}$$

We use a PPLM $\mathcal{G}$ to predict the masked tokens from the amino acid vocabulary $\mathcal{V}$. The probability of $\texttt{[MASK]}_j$ as an amino acid $v \in \mathcal{V}$ is calculated as

$$p(v|\boldsymbol{h}_j) = \frac{\exp(\boldsymbol{v}^\top \boldsymbol{h}_j)}{\sum_{v_t \in \mathcal{V}} \exp(\boldsymbol{v}_t^\top \boldsymbol{h}_j)}, \tag{3}$$

where $\boldsymbol{v}$ is the embedding of the amino acid v and $\boldsymbol{h}_j$ is the contextualized embedding of $\texttt{[MASK]}_j$. For each masked token, we select c amino acids as samples, so that the combination of selected amino acids at multiple positions can be adequately informative. We next introduce how to select the c samples per mask from $\mathcal{V}$ based on Eq. (3). To ensure the selected samples are informative enough, we emphasize two aspects: representativeness and diversity.

Intra-group Representativeness In this paper, we include 20 natural amino acids in our candidate set $\mathcal{V}$. The sampling procedure requires selecting c amino acids as a subset $\mathcal{V}_i$ for each masked mutation site, in which each element is coupled with an embedding $\boldsymbol{h}_j$ and its probability $p(v|\boldsymbol{h}_j)$. The intra-group representativeness is defined as the distance between the origin set $\mathcal{V}$ and the subset $\mathcal{V}_i$, indicating the information coverage of the candidate set $\mathcal{V}$ with respect to the subset $\mathcal{V}_i$. We treat it as a transport problem between two unbalanced sets and solve this by optimizing the following intra-group optimal transport distance (OTD),

$$d_{\text{intra}} = \text{OTD}(\mathcal{V}_i, \mathcal{V}), i \in \{1, 2, ..., \binom{20}{c}\}. \tag{4}$$

Given two sets of amino acids, the optimization problem is defined upon a probability distribution of the original set and the subset denoted as $p(\mathcal{V})$ and $p(\mathcal{V}_i)$, respectively. Here we adopt the entropic regularization-based unbalanced Sinkhorn OTD [21], formulated as

$$\begin{aligned} \text{OTD}(\mathcal{V}_i, \mathcal{V}) = &\min_\gamma \langle \gamma, \mathbf{M} \rangle_F + a \cdot \Omega(\gamma) \\ &+ b \cdot (\text{KL}(\gamma \mathbf{1}, p(\mathcal{V})) + \text{KL}(\gamma^T \mathbf{1}, p(\mathcal{V}_i))) \\ &s.t. \gamma \geq 0, \end{aligned} \tag{5}$$

where γ represents the optimal transport plan. The cost matrix $\mathbf{M}$ stores the pair-wise Euclidean distance of token embeddings. $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product. $\Omega(\gamma) = \sum_{i,j} \gamma_{i,j} \log(\gamma_{i,j})$ is the entropic regularization term. KL is the Kullback-Leibler divergence. a and b are the entropy regularization coefficient and the marginal relaxation coefficient, respectively.

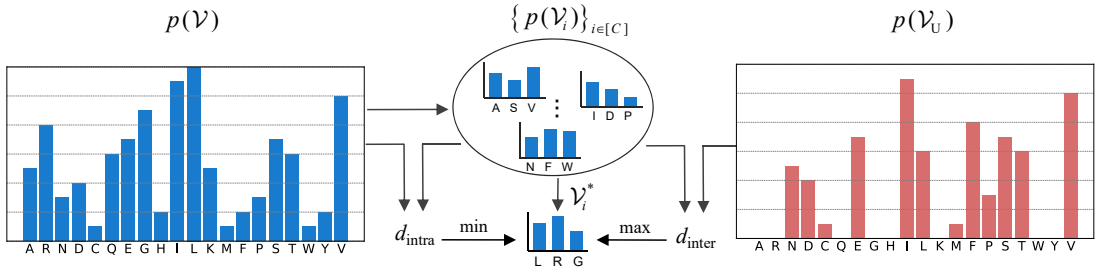

Figure 3: The proposed sampling strategy using the optimal transport distance (OTD) of intra-group and inter-group on a mutation site. $p(\mathcal{V})$ is the probability distribution of 20 amino acid candidates predicted by the PPLM. $p(\mathcal{V}_i)$ is the subset of c amino acids from $p(\mathcal{V})$, which has $C = \binom{20}{c}$ possible combinations in total. $p(\mathcal{V}_U)$ is the sampled amino acids distribution in the previous rounds. By minimizing d_{intra} and maximizing d_{inter} simultaneously, we can find the most informative subset $\mathcal{V}_i^*$ from the entire subsets.

Notably, we employ the optimal transport distance to quantify the difference between two probability distributions of amino acids. While other methods such as KL divergence also seem to work, OTD better accounts for the difference between amino acid types. This is important because different types of amino acids have distinct physical and chemical properties, such as polarity, hydrophobicity, and volume. If two pairs of amino acids exhibit similar probability differences, the pair with closer properties should incur a smaller transportation cost, which is computed by the Euclidean distance between pre-trained token embeddings of different amino acids.

Inter-group Diversity To improve the quality and variance of mutants and boost sample efficiency during iterations, the subset of selected amino acids should be as diverse as possible compared to the samples from previous rounds. Similar to d_{intra}, we compute the inter-group OTD d_{inter} which measures the distance between the current subset $\mathcal{V}_i$ of amino acids and the union of previous subsets $\mathcal{V}_U$.

$$d_{\text{inter}} = \text{OTD}(\mathcal{V}_i, \mathcal{V}_U), \tag{6}$$

where $\text{OTD}(\cdot,\cdot)$ is defined in Eq. (5). $p(\mathcal{V}_U)$ is the probability distribution estimated by the frequency of sampled amino acids accumulated in the previous rounds, obtained by calculating the frequency of each amino acid

$$p(\mathcal{V}_U) = \frac{N(v)}{\sum_{v_t\in\mathcal{V}} N(v_t)}, \tag{7}$$

where $N(v)$ represents the occurrences number of amino acid v in previous rounds.

Informative Samples. With the representativeness and diversity measurements, we aim to minimize d_{intra} to find a group of representative amino acids and meanwhile maximize d_{inter} to find the diverse amino acids. It can be formulated by

$$\mathcal{V}_i^* = \arg\min_{\mathcal{V}_i\subset\mathcal{V}} (d_{\text{intra}} - \lambda \cdot d_{\text{inter}}), \tag{8}$$

where $\mathcal{V}_i^*$ is the found most informative subset, and λ is a trade-off parameter.

Figure 3 shows the proposed OTD-based sampling method on a mutation site. For the n mutation site, we implement the above sampling procedure n times independently and in parallel, and $N_c = c^n$ informative samples (denoted as $\hat{\mathbb{S}} = \{\hat{S}^{(i)}\}_{i\in[N_c]}$) will be screened to train a fitness predictor.

4.2 Training the Fitness Predictor

We rely on the oracle to annotate the fitness of selected informative samples. Then we exploit these annotated data to train a predictor $\mathcal{F}$ with a regression loss. Specifically, the predictor is composed of two modules:

(1) The PPLM $\mathcal{G}$ inherited from the sampler in the exploration stage. The only difference is we take the whole protein embedding $\boldsymbol{h} = \mathcal{G}(S)$, instead of the masked $\boldsymbol{h}_j$ in Eq. (3).

(2) The two-layer MLP network $\mathcal{H}$, where the input is the protein representation $\boldsymbol{h}$ and the output is a fitness value. Since the annotated data is small-scaled, we freeze the parameters $\theta_{\mathcal{G}}$ in the PPLM and optimize the parameters $\theta_{\mathcal{H}}$ in the MLP only, in order to avoid overfitting. In specific, $\theta_{\mathcal{H}}$ is updated by minimizing the following mean absolute error:

$$\ell_{\mathcal{H}} = \frac{1}{N_c}\sum_{S\in\hat{\mathbb{S}}} |\mathcal{H}(\mathcal{G}(S)) - y(S)|, \tag{9}$$

where y is the annotated fitness value of the mutant S.

Algorithm 1 The procedure of AFP-DE

Input: The target protein with masked mutation sites S'
Output: The fitness predictor $\mathcal{F}$

```
Initialize 𝕊̂ = ∅, the maximum rounds M
while m < M do
    Feed S' into 𝒢 to calculate p(v|h_j) by Eq.(3)
    Screen informative samples 𝕊̂_m by Eq. (8)
    Use annotated 𝕊̂ = 𝕊̂ ⋃ 𝕊̂_m to train ℋ by Eq. (9)
    Predict the top-k high-fitness mutants 𝕊̃
    Use 𝕊̃ to finetune 𝒢 by Eq. (10)
end while
return ℱ = ℋ(𝒢(·));
```

4.3 Refining the Pre-trained Model

As we just select extremely limited informative samples in the first round, the predictor may not be trained well and thus calls for more samples for training. Before the exploration step in the next round, we propose to refine the sampler's PPLM, such that it can be adapted from the general protein domain to the target protein. The mutants with high fitness are supposed to be more informative to the PPLM, thus we select the mutants with top-k predicted fitness values (denoted as $\tilde{\mathbb{S}} = \{\tilde{S}^{(i)}\}_{i\in[k]}$) to finetune the PPLM $\mathcal{G}$ through the original masked token prediction. Specifically, the parameter $\theta_{\mathcal{G}}$ is updated by minimizing the following cross-entropy loss:

$$\ell_{\mathcal{G}} = -\frac{1}{k}\sum_{S\in\tilde{\mathbb{S}}}\sum_{v\in\mathcal{V}} q(v)\cdot\log(p(v|\mathcal{G})), \tag{10}$$

Sample size	24			48			384		
Evaluation metric	NDCG	Max	Mean	NDCG	Max	Mean	NDCG	Max	Mean
MLDE	0.664	0.448	0.076	0.735	0.506	0.143	0.852	0.768	0.286
ftMLDE(EVmutation)	0.736	0.569	0.152	0.767	0.602	0.216	0.861	0.942	0.420
ftMLDE(Transformer)	0.736	0.568	0.153	0.768	0.606	0.230	0.831	0.838	0.396
CLADE	0.727	0.557	0.136	0.749	0.447	0.096	0.856	0.724	0.364
CLADE2.0	0.735	0.498	0.125	0.780	0.686	0.157	0.901	0.918	**0.482**
Random-ft	0.712	0.375	0.056	0.744	0.570	0.116	0.837	0.721	0.245
AFP-DE(Ours)	**0.781**	**0.708**	**0.410**	**0.784**	**0.708**	**0.440**	**0.906**	**0.967**	0.457

Table 1: Performance comparison on the GB1 dataset with varying sample sizes. The best is **boldface** and the second best is underline.

Sample size	24			48			384		
Evaluation metric	NDCG	Max	Mean	NDCG	Max	Mean	NDCG	Max	Mean
MLDE	0.636	0.049	0.001	0.692	0.099	0.025	0.718	0.202	0.046
ftMLDE(EVmutation)	0.648	0.165	0.026	0.663	0.065	0.006	0.771	0.432	0.051
ftMLDE(Transformer)	0.685	0.133	0.016	0.678	0.196	0.049	0.727	0.225	0.030
CLADE	0.687	0.161	0.016	0.693	0.192	0.053	0.809	0.216	0.085
CLADE2.0	0.656	0.183	0.028	0.729	0.161	0.026	0.810	0.345	0.106
Random-ft	0.667	0.069	0.005	0.716	0.085	0.028	0.800	0.250	0.057
AFP-DE(Ours)	**0.701**	**0.187**	**0.083**	**0.747**	**0.211**	**0.080**	**0.815**	**0.472**	**0.114**

Table 2: Performance comparison on the PhoQ dataset with varying sample sizes. The best is **boldface** and the second best is underline.

where $q(v)$ is the true probability that the masked token is the amino acid v, and $p(v|\mathcal{G})$ is the corresponding predicted probability.

The refined sampler is then used to screen a batch of informative mutants, followed by annotating and training again. We emphasize that the predictor's PPLM is also finetuned because its parameters are inherited from the sampler, leading to a better representation of the target protein.

Algorithm 1 summarizes the entire procedure of AFP-DE. In short, we conduct the above exploration-exploitation-refinement procedure iteratively until finding a mutant with adequately good fitness values or reaching the maximum sampling rounds.

5 Experiments

This section presents the experimental settings including datasets, evaluation metrics, implementation details, and baselines; analyzes the experiment results and provides an in-depth discussion. Notably, in addition to the *in silico* experiments, we conducted biological wet experiments to further verify the effectiveness of our method.

5.1 Experimental Settings

Datasets We conduct experiments on two publicly available datasets: GB1 [29] and PhoQ [14]. GB1 is the most widely used dataset to test the performance of different MLDE methods. The GB1 refers to the protein G domain B1, and the annotated fitness value represents the binding ability of different GB1 mutants to the antibody IgG-Fc. This dataset describes a four-site combinatorial fitness landscape, with 149,361 experimentally annotated variants out of $20^4 = 160,000$ at four key sites (V39, D40, G41, and V54). There are multiple local optima of the fitness landscape and more than 90% annotated mutants have fitness values below wild-type GB1. Similar to the GB1 dataset, the PhoQ dataset consists of 140,517 annotated data out of $20^4 = 160,000$ at four sites (A284, V285, S288 and T289). The fitness value of PhoQ refers to the phosphatase or kinase activity of different PhoQ mutants. Besides, we exclude the remaining unannotated sequences in the mutant space of both GB1 and PhoQ, which are believed to be meaningless according to biologists [27]. In addition, for both datasets, we normalize their fitness values into the range $[0, 1]$.

Note that we exclude some benchmark datasets like FLIP [3], TAPE [16] and PEER [31] due to the position of the mutation site in the mutant data set is not fixed. Furthermore, these datasets do not cover a large portion of the possible mutants, which means the screened samples are difficult to index to the corresponding fitness value. Therefore, in accordance with [27, 14, 15], this paper chose two datasets GB1 and PhoQ to conduct experiments.

Evaluation Metrics Since picking mutants with the highest predicted fitness (top k) values among candidates is essentially a ranking problem, we use the normalized discounted cumulative gain (NDCG) as one of the evaluation metrics following the previous work [27]. NDCG is commonly used in information retrieval and can measure the correlation between the predicted value and the true value. In the meantime, NDCG puts a high weight on sequences with a high predicted label value and a high true label value, which is consistent with the directed protein evolution mechanism paying more attention to mutants with high fitness values. We also report the mean value of the predicted π-highest ranking mutants and the max value of both the predicted π-highest ranking mutants and training mutants. Following previous works [27, 29, 14, 15], we set $\pi = 96, 32, 56$ when $384, 48, 24$ mutants are used for training, respectively. The "Mean" metric can measure the training performance through different sampling methods, and the "Max" metric is highly related to the goal of directed evolution, that is, to find the mutant with the highest fitness in the whole mutant space.

Implementation Details In the exploration stage, we use the open-source protein pre-training model, ESM-1b [17], as the initial PPLM $\mathcal{G}$. The sub-distribution of $c = 3$ amino acids is selected at $n = 4$ mutation positions, resulting in $N_c = 3^4$ possible combinations. To make a fair comparison with the existing methods [27, 14, 15], we randomly select 24 samples in each round. The default λ in Eq. (8) is 0.5.

In the exploitation stage, we use the Adam optimizer with a learning rate of $1e-5$, and the batch size is set as a multiple of 1 to allow

the model to choose the size of the batch size independently. We empirically set the training epoch to 5, which is enough to guarantee the convergence of loss.

			AFP-DE*	AFP-DE#	AFP-DE
GB1	48	NDCG	0.771	0.782	**0.784**
		Max	0.670	0.708	**0.708**
		Mean	0.391	0.397	**0.440**
	384	NDCG	0.777	0.848	**0.906**
		Max	0.708	0.825	**0.967**
		Mean	0.224	0.410	**0.457**
PhoQ	48	NDCG	0.728	0.724	**0.747**
		Max	0.155	**0.279**	0.211
		Mean	0.073	0.066	**0.080**
	384	NDCG	0.772	0.777	**0.815**
		Max	0.279	0.455	**0.472**
		Mean	0.091	0.082	**0.114**

Table 3: Ablation experiment results of AFP-DE* (without refinement) and AFP-DE# (without inter-group diversity)

In the refinement stage, we take out $k = 8000$ mutation sequence with the largest predicted fitness values and 1000 extra protein sequences randomly selected from the UniRef50 database [24] to refine the PPLM, where the latter plays a regularization role to avoid the model collapse. The details of finetuning PPLM can refer to the training of ESM-1b [17].

Overall, our model is implemented on the PyTorch framework and the FAIRseq library [11]. The code is run on an Ubuntu server with 1 GPU (NVIDIA GeForce 3090Ti).

Baseline To verify the superiority of our proposed method, we compare it with four state-of-the-art baseline methods, the details of which are as follows:

- MLDE [29], which generates training data by random sampling from full combinatorial spaces, and then uses sampling mutants to train a machine learning-based ensemble model to predict the fitness of mutants.
- ftMLDE [27], which combines a variety of sampling, encoding and training methods. We compare our method with two sampling strategies in ftMLDE, EVmutation [7] and MSA-transformer [18].
- CLADE [14], which designs a hierarchical unsupervised clustering sampling method to pick high-fitness mutants, and then trains a fitness predictor with supervised learning.
- CLADE2.0 [15], which is the improved version of CLADE. It uses a scoring function that integrates the profile HMM [5], the MSA-based DeepSequence VAE [19], the EVmutation [7] and a PPLM in the sampling stage.

5.2 Main Results

Performance Comparison To evaluate our proposed approach, we compare AFP-DE with several SOTA baselines on the GB1 and PhoQ datasets. The performance of all methods is listed in Table 1 and Table 2. As shown in the two tables, MLDE performs worst among several methods due to the prevalence of low-fitness sequences in the combinatorial landscapes, and the random draw often produces sequences with very low fitness values, resulting in ineffective training data. The sampling strategies in the ftMLDE and CLADE models alleviate this problem and achieve relatively better results. The proposed AFP-DE outperforms all other methods on both datasets with a sample size of 24. At sample sizes 48 and 384, AFP-DE is also better than other methods except for the Mean metric of CLADE2.0 with a sample size of 384 on the GB1 dataset. It is worth noting that CLADE2.0 uses a scoring function that integrates multiple MSA-based models in the process of sampling. Our method only utilizes an MSA-free PPLM in the sampling stage but still exceeds the performance of CLADE2.0 overall. It means that the proposed AFP-DE can handle proteins even without MSA information, which is of extremely practical significance for directed protein evolution [6]. Random-ft refers to finetune protein language models without active finetuning, using random sampling instead. Comparing Random-ft with AFP-DE, we demonstrate the effectiveness of active finetuning, showing that it leads to superior finetune performance under a limited budget with an equal amount of labeled data. These results highlight the potential of active finetuning for achieving more efficient directed evolution.

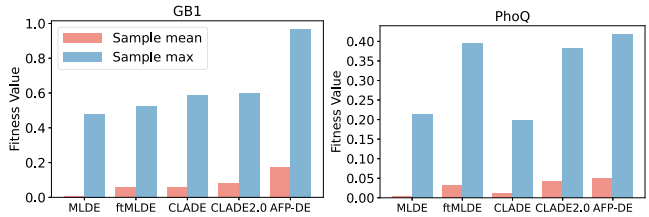

Figure 4: Sampling performance on the GB1 and PhoQ datasets. "Sample mean" and "Sample max" are the average and maximum fitness values of 384 mutants, respectively.

Sampling Result Analysis We observe from the experimental datasets that 92% of mutants have fitness values smaller than 1% of the global maximum in the GB1 and PhoQ datasets. Hence, it is critical to utilize the mutants with greater fitness values to predict the landscape of GB1 and PhoQ. Wittmann *et al.* [27] have found that the enrichment of high-fitness mutants in training data can enhance prediction performance. Therefore, we compare the performance of different sampling strategies by measuring the enrichment of high-fitness mutants. We calculate the mean and maximum fitness value of the 384 samples obtained by the different sampling methods. From Figure 4, we can find that the sampling performance of AFP-DE greatly outperforms other methods on the two datasets, especially on the GB1 dataset where the maximum and the mean value of AFP-DE are 60.6% and 119.9% higher than the SOTA method CLADE2.0, respectively.

Sampling Rounds Analysis Our method screen the informative mutants iteratively based on the refined PPLM. That is, AFP-DE selects 24 samples per round, up to 16 rounds. From Figure 5, we observe that AFP-DE only takes roughly 10 rounds (240 samples) to achieve a better performance than other methods. Therefore, compared to other methods that only select a specified number of mutants at one time, AFP-DE can flexibly choose the sampling size, which can further reduce the cost in practical annotating mutants. Note that we did not compare the mean fitness values obtained by different methods because this metric is unstable due to a high proportion of low-fitness mutants in the datasets.

5.3 Ablation Study

We then conduct ablation studies to investigate the effect of the designed modules, i.e., the inter-group diversity and iterative sampling

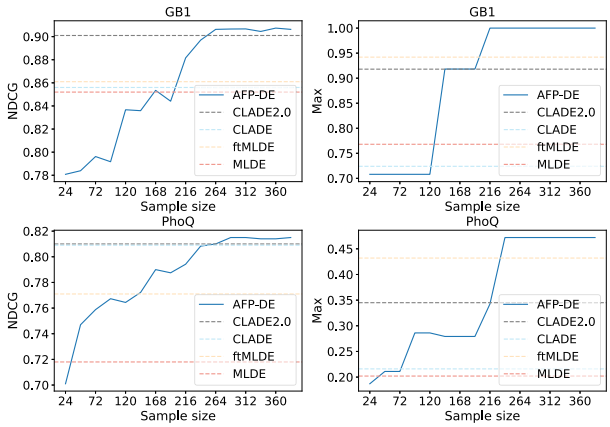

Figure 5: The performance of AFP-DE with an increased number of sampling rounds on the GB1 and PhoQ datasets. The dotted lines denote the baselines' performance. Different from AFP-DE, these baselines are not iteratively refined and we only report the final performance with a sampling size of 384.

with refinement.

Inter-group Diversity The proposed sampling module is composed of intra-group OTD and inter-group OTD, where the latter represents the diversity that is nearly neglected by the existing methods. To verify its importance, we remove the part of inter-group OTD, denoted as AFP-DE$^{\#}$. From the results shown in Table 3, we can find that the performance of AFP-DE$^{\#}$ is lower than the original AFP-DE when the sample size is 48, and this phenomenon becomes significant when the sample size reaches 384, reflecting the importance of inter-group diversity. Note that we do not compare the performance when the sampling size is 24 because there are no labeled mutants to calculate inter-OTD at the first round of sampling.

Refinement To verify the effectiveness of our iterative sampling strategy with refinement, we remove the refinement stage and sample 48 or 384 candidate mutants at one time, which is denoted as AFP-DE*. Table 3 has shown that AFP-DE* is inferior to AFP-DE, especially with a sample size of 384. This verifies the advantage of iterative sampling with refinement.

5.4 *Biological Wet Experiment*

Besides the *in silico* experiment on GB1 and PhoQ datasets, we also conduct a biological wet experiment on the protein CNFRS for directed protein evolution. CNFRS is a variant of the tyrosyl tRNA synthetase from the archaea *Methanococcus jannaschii* (Mj-TyrRS), which can incorporate p-cyanophenylalanine into a protein [33]. Through analyzing the 3D structure of CNFRS and the binding state between CNFRS, our co-authors from the biological background identify four mutation sites that could potentially exert significant influence on the fitness of CNFRS (as shown in Figure 6) and employ the AFP-DE to implement directed evolution. In the initial round of biological wet experiments, our proposed method successfully predicted a mutant protein variant with a fitness value that was 66.4% higher than that of the wild-type CNFRS. In a subsequent round of experiments, we identified a mutated protein variant with a fitness value that was 82.1% higher than that of the wild-type CNFRS. These results demonstrate the efficacy of our method in predicting protein mutations that improve fitness values in biological systems.

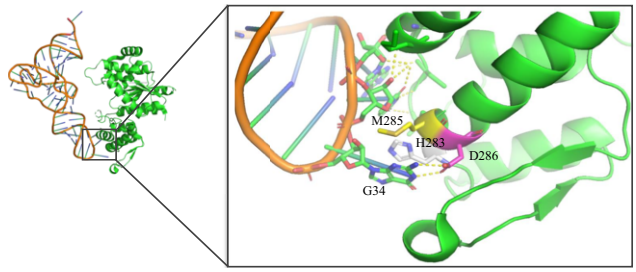

Figure 6: The 3D structure of CNFRS protein and the four selected mutation sites.

5.5 *Discussion*

From the results of *in silico* experiments and biological wet experiments, we can conclude that the advantage of our actively-finetuned protein model lies in two aspects: (1) Flexibility - we can freely choose the number of samples collected in each round of the experiment by controlling the parameter c. This allows our method to be used in laboratories of varying sizes, from small-scale experimental setups to high-throughput facilities. Moreover, human evaluators can dynamically assess the experimental outcomes, as demonstrated in the iterative experiment (Figure 5), where optimal results were achieved with only 240 labeled samples, enabling researchers to avoid unnecessary experimental costs. (2) Ease of use - compared to the CLADE2.0 baseline method that achieves the closest results to ours, we do not require additional input from homologous proteins. This expands the applicability of our method to orphan proteins, making it a more general approach.

Our method treats each mutation position on the protein sequence equally, which may seem to contradict the epistasis [10, 23] effect of protein mutations. That is, the independent consideration of each mutation site would ignore the correlations between them. However, current large-scale protein models can accurately predict the three-dimensional structure of proteins [9], indicating that they have learned the underlying interactions between different positions on the protein sequence to some extent. Hence, our approach can mitigate the epistasis effect. Furthermore, from an ethical perspective, relevant policies should be established to guide the application of such technologies in appropriate domains, while simultaneously mitigating potential societal harm.

6 Conclusion

In this paper, we have presented an actively-finetuned protein model for directed evolution to maximize the benefits of a pre-trained protein model. To efficiently finetune the protein model and explore the combinatorial space of mutants, we designed a novel sampling strategy based on the optimal transport distance to select informative samples and achieve more efficient sampling. Through iterative optimization, the pre-trained protein language model can be gradually adapted from the general protein domain to the specific target protein of interest. Besides, the proposed AFP-DE does not require MSA as input, hence it is a more practical framework that could apply to proteins without MSA information. The *in silico* experiments on the GB1 and PhoQ datasets and biological wet experiments confirm that AFP-DE outperforms state-of-the-art baselines even with fewer annotated mutants.

Acknowledgements

We extend our appreciation to the anonymous reviewers for their diligent efforts and thoughtful feedback. This work is funded by NSFCU19B2027/91846204 and sponsored by CCF-Tencent Open Fund. This work is also supported by the Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

References

[1] Ivan Anishchenko, Samuel J Pellock, Tamuka M Chidyausiku, Theresa A Ramelot, Sergey Ovchinnikov, Jingzhou Hao, Khushboo Bafna, Christoffer Norn, Alex Kang, Asim K Bera, et al., 'De novo protein design by deep network hallucination', *Nature*, **600**(7889), 547–552, (2021).

[2] Frances H Arnold, 'Design by directed evolution', *Accounts of chemical research*, **31**(3), 125–131, (1998).

[3] 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*, 2021–11, (2022).

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 'BERT: Pre-training of deep bidirectional transformers for language understanding', in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, (2019).

[5] Robert D Finn, Jody Clements, and Sean R Eddy, 'Hmmer web server: interactive sequence similarity searching', *Nucleic acids research*, **39**, 29–37, (2011).

[6] Mu Gao, Davi Nakajima An, Jerry M Parks, and Jeffrey Skolnick, 'Af2complex predicts direct physical interactions in multimeric proteins with deep learning', *Nature communications*, **13**(1), 1–13, (2022).

[7] Thomas A Hopf, John B Ingraham, Frank J Poelwijk, Charlotta PI Schärfe, Michael Springer, Chris Sander, and Debora S Marks, 'Mutation effects predicted from sequence co-variation', *Nature biotechnology*, **35**(2), 128–135, (2017).

[8] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al., 'Highly accurate protein structure prediction with alphafold', *Nature*, **596**(7873), 583–589, (2021).

[9] 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).

[10] Charlotte M Miton and Nobuhiko Tokuriki, 'How mutational epistasis impairs predictability in protein evolution and design', *Protein Science*, **25**(7), 1260–1272, (2016).

[11] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli, 'Fairseq: A fast, extensible toolkit for sequence modeling', in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, (2019).

[12] Michael S Packer and David R Liu, 'Methods for the directed evolution of proteins', *Nature Reviews Genetics*, **16**(7), 379–394, (2015).

[13] Anna I Podgornaia and Michael T Laub, 'Pervasive degeneracy and epistasis in a protein-protein interface', *Science*, **347**(6222), 673–677, (2015).

[14] Yuchi Qiu, Jian Hu, and Guo-Wei Wei, 'CLADE: Cluster learning-assisted directed evolution', *Nature Computational Science*, 809–818, (2021).

[15] Yuchi Qiu and Guo-Wei Wei, 'CLADE 2.0: Evolution-driven cluster learning-assisted directed evolution', *Journal of Chemical Information and Modeling*, **62**(19), 4629–4641, (2022).

[16] 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).

[17] Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives, 'Transformer protein language models are unsupervised structure learners', *International Conference on Learning Representations*, (2021).

[18] Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives, 'MSA transformer', in *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8844–8856, (2021).

[19] 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).

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

[21] Thibault Séjourné, Jean Feydy, François-Xavier Vialard, Alain Trouvé, and Gabriel Peyré, 'Sinkhorn divergences for unbalanced optimal transport', *arXiv preprint:1910.12958*, (2019).

[22] George P Smith and Valery A Petrenko, 'Phage display', *Chemical Reviews*, **97**(2), 391–410, (1997).

[23] Tyler N Starr and Joseph W Thornton, 'Epistasis in protein evolution', *Protein science*, **25**(7), 1204–1218, (2016).

[24] Baris E Suzek, Yuqi Wang, Hongzhan Huang, Peter B McGarvey, Cathy H Wu, and UniProt Consortium, 'Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches', *Bioinformatics*, **31**(6), 926–932, (2015).

[25] Jesse Vig, Ali Madani, Lav R. Varshney, Caiming Xiong, richard socher, and Nazneen Rajani, 'BERTology meets biology: Interpreting attention in protein language models', in *International Conference on Learning Representations*, (2021).

[26] Greg Winter, Andrew D Griffiths, Robert E Hawkins, and Hennie R Hoogenboom, 'Making antibodies by phage display technology', *Annual review of immunology*, **12**(1), 433–455, (1994).

[27] Bruce J Wittmann, Yisong Yue, and Frances H Arnold, 'Informed training set design enables efficient machine learning-assisted directed protein evolution', *Cell systems*, **12**(11), 1026–1045, (2021).

[28] Nicholas C Wu, Lei Dai, C Anders Olson, James O Lloyd-Smith, and Ren Sun, 'Adaptation in protein fitness landscapes is facilitated by indirect paths', *Elife*, **5**, e16965, (2016).

[29] Zachary Wu, SB Jennifer Kan, Russell D Lewis, Bruce J Wittmann, and Frances H Arnold, 'Machine learning-assisted directed protein evolution with combinatorial libraries', *Proceedings of the National Academy of Sciences*, **116**(18), 8852–8858, (2019).

[30] Minghao Xu, Hang Wang, Bingbing Ni, Hongyu Guo, and Jian Tang, 'Self-supervised graph-level representation learning with local and global structure', in *International Conference on Machine Learning*, pp. 11548–11558, (2021).

[31] Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Chang Ma, Runcheng Liu, and Jian Tang, 'Peer: A comprehensive and multi-task benchmark for protein sequence understanding', *arXiv preprint:2206.02096*, (2022).

[32] Kevin K Yang, Zachary Wu, and Frances H Arnold, 'Machine-learning-guided directed evolution for protein engineering', *Nature methods*, **16**(8), 687–694, (2019).

[33] Douglas D Young, Travis S Young, Michael Jahnz, Insha Ahmad, Glen Spraggon, and Peter G Schultz, 'An evolved aminoacyl-tRNA synthetase with atypical polysubstrate specificity', *Biochemistry*, **50**(11), 1894–1900, (2011).

[34] Hui Yuan, Chengzhuo Ni, Huazheng Wang, Xuezhou Zhang, Le Cong, Csaba Szepesvári, and Mengdi Wang, 'Bandit theory and thompson sampling-guided directed evolution for sequence optimization', *arXiv preprint:2206.02092*, (2022).

[35] He Zhang, Fusong Ju, Jianwei Zhu, Liang He, Bin Shao, Nanning Zheng, and Tie-Yan Liu, 'Co-evolution transformer for protein contact prediction', in *Advances in Neural Information Processing Systems*, eds., A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, (2021).