Protein Design by Directed Evolution Guided by Large Language Models

Trong Thanh Tran and Truong Son Hy

***Abstract*—Directed evolution, a strategy for protein engineering, optimizes protein properties (i.e., fitness) by a rigorous and resource-intensive process of screening or selecting among a vast range of mutations. By conducting an *in silico* screening of sequence properties, machine learning-guided directed evolution (MLDE) can expedite the optimization process and alleviate the experimental workload. In this work, we propose a general MLDE framework in which we apply recent advancements of Deep Learning in protein representation learning and protein property prediction to accelerate the searching and optimization processes. In particular, we introduce an optimization pipeline that utilizes Large Language Models (LLMs) to pinpoint the mutation hotspots in the sequence and then suggest replacements to improve the overall fitness. Our experiments have shown the superior efficiency and efficacy of our proposed framework in the conditional protein generation, in comparision with traditional searching algorithms, diffusion models, and other generative models. We expect this work will shed a new light on not only protein engineering but also on solving combinatorial problems using data-driven methods. Our implementation is publicly available at https://github.com/HySonLab/Directed_Evolution.**

***Index Terms*—Directed evolution, protein engineering, machine learning, large language models.**

I. Introduction

PROTEIN are essential biomolecules that play a diverse range of critical roles in every living organism. They are involved in virtually every important activity that happens inside living things: digesting food, contracting muscles, moving oxygen throughout the body, attacking foreign viruses, etc. The effective resolutions to biological challenges can also serve as effective resolutions to human challenges, given the extensive utilization of proteins in various domains such as food, chemicals, consumer products, and medicinal applications. However, comprehending the complex structure and function of proteins remains an immensely challenging task, owing to the diversity of protein sequences and the three-dimensional structure they have. Therefore, computational biology and machine learning (ML) methods have emerged as indispensable tools among researchers and engineers. These methods offer the capability to process vast amounts of biological data, extract meaningful patterns, and make predictions that would be impractical through traditional experimental approaches alone [1]. Pioneering efforts like AlphaFold [2] have demonstrated remarkable success in accurately predicting protein tertiary structures from their primary sequences, revolutionizing structural biology. Additionally, ML techniques have been employed for protein fitness prediction [3]–[5], which enables the design of proteins with specific functions or properties. They are also used for forecasting protein-ligand binding affinity [6], [7], a critical aspect of drug discovery. Moreover, large language models have been pretrained on extensive protein sequence databases [8], [9], enabling them to capture intricate sequence-structure-function relationships. These models empower researchers to extract valuable insights from protein sequences, understand their evolution, and predict their behavior in various biological contexts.

The field of protein engineering aims to strategically create or identify proteins that have practical applications in technology, scientific research, or medical contexts. The amino acid sequence of a protein governs the properties associated with its function [10], [11]. The relationship is inverted through the process of protein engineering, with the aim of identifying a specific functional sequence. Nevertheless, the space of possible proteins is too large to search exhaustively naturally, in the laboratory, or computationally. The problem of the identification of ideal sequences is classified as NP-hard, as there currently exists no polynomial-time algorithm for efficiently exploring this particular domain [12]. Functional proteins are scarce in this vast space of sequences, and as the desired level of function increases, the number of sequences having that function decreases exponentially [13], [14]. Consequently, the occurrence of functional sequences is rare and overshadowed by nonfunctional and mediocre sequences. In recent years, generative models have emerged as powerful tools for generating creative and diverse content, ranging from text [15]–[17], images [18], [19] to speech [20] and more. Leveraging the success of generative models in text and image generation, numerous studies have been made to harness these techniques to design novel proteins with tailored properties and functions [21]–[23]. This developing paradigm promises to accelerate wet-lab experiments with data-driven models to find desirable proteins more efficiently.

Directed evolution, a laboratory methodology wherein biological entities possessing desired characteristics are generated through iterative cycles of genetic diversification and library screening or selection, has emerged as a highly valuable and extensively utilized instrument in both fundamental and practical realms of biological research [24]–[31]. This method was conceptualized using the natural process of evolution and the idea of natural selection as its primary sources of inspiration. In nature, organisms with advantageous traits are more likely to survive and reproduce, passing on their benefi-

Corresponding author: Truong Son Hy.

Trong Thanh Tran is with FPT Software AI Center, Hanoi, Vietnam (email: ThanhTVT1@fpt.com)

Truong Son Hy is with Indiana State University, Terre Haute, USA (email: TruongSon.Hy@indstate.edu)

cial characteristics to the next generation. Similarly, directed evolution starts with a protein having some level of the desired function. Subsequently, a series of mutation and screening rounds are undertaken, wherein mutations are introduced (e.g., through site-saturation mutagenesis) to generate a collection of variant proteins. Through this iterative process, the most optimal variant is identified, and the cycle continues until a satisfactory level of improvement is achieved. Fig. 1A demonstrates the workflow of this algorithm.

Machine learning-guided directed evolution (MLDE), as illustrated in Fig. 1B, is a strategy for protein engineering that can be applied to a range of biological systems, reducing the burden of laboratory experiments by performing *in silico* population selection through model-based fitness prediction. These techniques can leverage data from both improved and unimproved sequences in order to accelerate the process of evolution and broaden the range of properties that can be optimized. This is achieved by intelligently selecting new variants for screening, which enables the achievement of higher levels of fitness than what can be accomplished solely through directed evolution [32]–[35]. Employing this novel biology algorithm, we propose a deep learning-based framework for MLDE that optimizes directly on the discrete space of all possible sequences. Our pipeline can be described by Fig. 1C.

Our contributions can be summarized as follows:

- A general framework of Machine learning-guided Directed Evolution for protein design with little to no training or fine-tuning required,
- Our extensive experiments have shown that our framework significantly outperforms other existing methods in conditional protein generation.

II. Related work

A. Directed Evolution

Directed Evolution [24]–[31] is a classical paradigm for protein sequence design that has achieved several successes. Under this framework, ML algorithms play a crucial role in improving the sample efficiency of evolutionary search. [36] utilizes a fast Markov chain Monte Carlo sampler that uses gradients to propose promising mutations. [37] formulates the local search as a proximal optimization problem to derive an objective for searching the steepest improvement. [38] combines hierarchical unsupervised clustering sampling and supervised learning to guide protein engineering. This work is later improved by [39]. [40] provides an approach to target informative sequences without the initial global search. Nevertheless, these methods have not yet incorporated large language models, which have proven efficient in multiple related tasks. Our proposal describes the combination of a protein language model yielding a highly favorable outcome.

B. Protein property prediction

One of the primary objectives of bioinformatics is the prediction of protein function, which can be applied to a vast array of biological issues, including the identification of drug targets and the comprehension of disease mechanisms. Numerous computational approaches have been developed to autonomously forecast protein function, with specific investigations concentrating on site- or domain-specific predictions [41], [42]. Traditional machine learning classifiers, such as support vector machines, random forests, and logistic regression, have been extensively used for protein function prediction [43], [44]. In recent years, deep learning has led to unprecedented improvements in the performance of methods tackling a broad spectrum of problems, including predicting protein function [45]–[50]. These developments facilitate our work as we employ attention, a well-known mechanism in deep learning, to acquire a comprehensive understanding of the protein sequence and anticipate its properties.

C. Protein generation

Significant advancements have been made in the development of methodologies for the generation of functional protein sequences with desired features in recent times. Conventional approaches that utilize multiple sequence alignments of homologous proteins, such as ancestral sequence reconstruction [51], have exhibited efficacy in producing functional proteins, but with certain limitations in their applicability. To access a broader sequence space, different ML techniques have been employed. Some techniques, including reinforcement learning [52], [53], Bayesian optimization [54]–[56], importance sampling [57], and adaptive evolution method [58] have been used and achieved great results in their defined tasks. Modern algorithms such as language models provide a powerful architecture to learn from large sets of amino acid sequences across families for the purpose of generating diverse, realistic proteins [5], [59], [60]. Other generative techniques like variational auto-encoder [4], [61], generative adversarial networks [22], [62], and diffusion [63] also show promising outcomes in numerous protein engineering tasks. Yet, by applying MLDE to generating sequences, we find that this algorithm demonstrates strong performance, especially in comparison to other state-of-the-art (SOTA) methods.

III. Background

A. Notation and Problem Formulation

Through the process of directed evolution, protein fitness is enhanced by emulating natural selection through mutagenesis. From a mathematical standpoint, directed evolution can be conceptualized as an optimization problem for a black-box algorithm, seeking the optimal sequence $x^\star$

$$x^\star = \arg\max_{x \in \mathcal{X}} f(x),$$

where $\mathcal{X}$ is the sequence mutation space, and $f(x)$ is an unknown sequence-to-fitness function for sequence x in $\mathcal{X}$. During the optimization, the algorithm choose the best mutated sequence from an intermediate generated population $\mathbf{x}$ (the size of the population $|\mathbf{x}|$ is n) so that the returned fitness $y^\star = f(x^\star)$ is the best fitness among values $\mathbf{y}$.

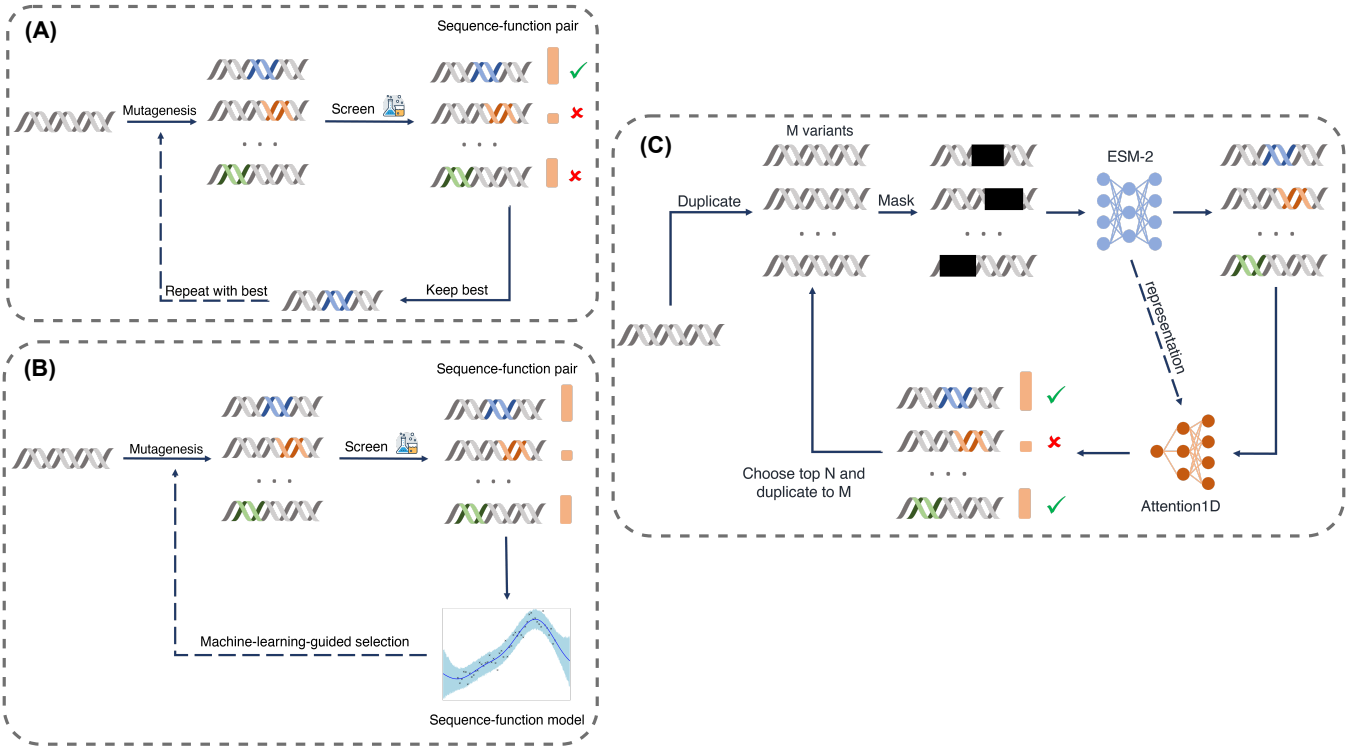


Fig. 1. The full workflow of **(A)** Traditional directed evolution, **(B)** Common MLDE framework, and **(C)** Our proposed MLDE framework. All of these workflows begin by identifying a protein with activity for a target function. Once the starting point is identified, diversity is introduced by mutagenesis, and the resulting variants are screened for function. **(A)** In traditional directed evolution, many variants are screened, and the best variant is then fixed as the parent for the next round of mutagenesis/screening. **(B)** Machine learning methods use the data collected in each round of directed evolution to choose the mutations to test in the next round. A careful choice of mutations to test decreases the screening burden and improves outcomes. **(C)** In our pipeline, using the resulting sequence-function data, a deep learning model is fit that maps protein sequence to protein fitness. This function can be used to predict the fitness of variants not experimentally evaluated. From that results, we can choose many variants as potential parent candidates for the subsequent iteration, taking into consideration the balance between exploitation and exploration.

B. Transformer

Transformer [64] is a formidable force in the modern deep learning stack. Originally, Transformer follows an encoder-decoder scheme and employs a multi-headed self-attention mechanism. The fundamental concept underlying this mechanism is for each element within the sequence to acquire information from other tokens present in the same sequence. Concretely, given a tensor of sequence features $\mathbf{X} \in \mathbb{R}^{n \times d}$, Transformer computes three matrices including query $(\mathbf{Q})$, key $(\mathbf{K})$, and value $(\mathbf{V})$ via three linear transformations $\mathbf{Q} = \mathbf{X}\mathbf{W}_q^\top$, $\mathbf{K} = \mathbf{X}\mathbf{W}_k^\top$, and $\mathbf{V} = \mathbf{X}\mathbf{W}_v^\top$. A self-attention tensor $(\mathbf{H})$ can be computed as follows:

$$\mathbf{H} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_0}}\mathbf{V}\right) \tag{1}$$

where $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$ are learnable parameters in $\mathbb{R}^{d_0 \times d}$, resulting in $\mathbf{H} \in \mathbb{R}^{n \times d_0}$. Moreover, each $\mathbf{H}$ in Eq. 1 denotes an attention head. The output of multiple heads $(\mathbf{H}_1, \ldots, \mathbf{H}_h)$ are then concatenated together to form a final tensor $\mathbf{H} = \text{Concat}(\mathbf{H}_1, \ldots, \mathbf{H}_h)$, where h is the number of attention heads. Finally, the new representation $\mathbf{X}'$ can be computed by feeding $\mathbf{H}$ into a feed-forward neural network (FFN).

C. ESM-2: Protein language model

Inspired by [65], ESM-2 [66] adopts the encoder-only Transformer architecture style with small modifications. The original Transformer uses absolute sinusoidal positional encoding to inform the model about token positions. These positional encodings are added to the input embeddings at the bottom of the encoder stack. Nevertheless, this approach demonstrates limited generalizability beyond the specific context window on which it is trained. The implementation of Rotary Positional Embedding (RoPE) [67] in ESM-2 enables the model to extrapolate beyond its trained context window with no additional computational effort. The ESM-2 model is capable of generating latent representations for individual amino acids inside a protein sequence. This is achieved through pre-training on a vast dataset consisting of millions of protein sequences including billions of amino acids. Consequently, the model is able to produce a concise representation of the full protein sequence, without relying on three-dimensional information.

IV. Method

This section describes the overall pipeline of our framework. Initially, after dividing the sequence population into multiple parts, we apply different masking strategies for each portion to retrieve a partially-masked population. These masked variants are then fed into ESM-2 to predict the tokens being masked. After that, the newly generated population is measured for fitness by a fine-tuned Attention1D. These steps are repeatedly

iterated until they reach a pre-defined iteration number. The full workflow of our method is illustrated in Fig. 1C.

A. Selecting mutation hotspots

We follow the scheme of the text-based masked language model [65], which masks some of the tokens from the input and predicts the original vocabulary id of the masked. In the context of protein generation, we assume that the new prediction tokens can boost the fitness values of the protein sequence. In this study, we conducted two different masking strategies: **(1)** random masking and **(2)** importance masking. In the first scheme, we randomly replace token(s) in the sequence by the ⟨MASK⟩ tokens. For the latter, we follow the soft-masking strategy described in [68]. Concretely, when we split a protein sequence into multiple substrings of length k called *k-mers*, it creates a distribution based on the frequency of each k-mer. Therefore, when masking and recovering the sequences, k-mer with various importance should be treated differently. Following [68], we consider the following aspects to measure and define the importance of k-mers in one sequence:

a) Relevancy: We adopt the TF-IDF weights [69], $w_{\text{TF-IDF}}$, of the k-mer as one way to measure the relevance of k-mer w in one sequence x:

$$w_{\text{TF-IDF}}(w, x) = \frac{f_{w,x}}{\sum_{w' \in \mathbf{x}} f_{w',x}} \log \frac{N}{1 + |\{x \in \mathcal{X} : w \in x\}|},$$

where $f_{w,x}$ is the number of times that k-mer w occurs in sequence $\mathbf{x}$, N is the number of sequences in the corpus, and $\mathcal{X}$ is the set of sentences, and $|\{x \in \mathcal{X} : w \in x\}|$ is the number of sequences where the k-mer w appears. A higher weight for k-mer w in sequence x in TF-IDF means that the k-mer might be more important in the sequence.

b) Entropy: We also consider measuring the amount of information with entropy H [70] in the k-mer w to reflect the importance of that k-mer:

$$H(w) = -p(w) \log(p(w)),$$

where $p(w) = \frac{f_w}{\sum_{j=1}^{V} f_j}$ represents the probability of k-mer w and f is the number of appearances of k-mer in the corpus. A k-mer with lower entropy indicates that it might contain less information and thus be less important compared to the one with higher entropy.

In practice, we combine these two measures (with normalization) to decide the importance I of the k-mer w in one sequence x by:

$$I(w) = \frac{w_{\text{TF-IDF}}(w, x)}{\sum_{w' \in x} w_{\text{TF-IDF}}(w', x)} + \frac{H(w)}{\sum_{w' \in d} H(w')}.$$

Based on the introduced importance I of the k-mers in a sequence, we decide to replace k-mer(s) with the lowest importance value(s) by ⟨MASK⟩ token(s). By doing this, models could replace the least important k-mers with better ones for better generation quality.

Algorithm 1 Machine learning-guided directed evolution

Input: Wild-type x^{wt}, fitness y^{wt}, population n, beam m, iterations N, random ratio α, k-mer size k

Output: population sequence $\mathbf{x}_{\text{top}}$, population fitness $\mathbf{y}_{\text{top}}$

$\mathbf{x}_{\text{prev}} \leftarrow$ Duplicate x^{wt} for n times
$\mathbf{y}_{\text{prev}} \leftarrow$ Duplicate y^{wt} for n times
for $i = 1, \ldots, N$ **do**
  $\mathbf{x}_{\text{cur}} \leftarrow$ Duplicate sequences in $\mathbf{X}_{\text{prev}}$ for m times
  $L \leftarrow \text{len}(\mathbf{x}_{\text{cur}})$
  // Masking
  $\mathbf{x}_{\text{cur}} \leftarrow \textsc{Shuffle}(\mathbf{x}_{\text{cur}})$
  $\mathbf{x}_1 \leftarrow \textsc{RandomMask}(\mathbf{x}_{\text{cur}}[: \text{int}(\alpha L)], k)$
  $\mathbf{x}_2 \leftarrow \textsc{ImportanceMask}(\mathbf{x}_{\text{cur}}[\text{int}(\alpha L) :], k)$
  $\mathbf{x}_{\text{mask}} \leftarrow \textsc{Concat}(\mathbf{x}_1, \mathbf{x}_2)$
  // Predicting mutation and extracting features
  $\mathbf{X}_{\text{repr}}, \mathbf{R}_{\text{feat}} \leftarrow \text{ESM2}(\mathbf{x}_{\text{mask}})$
  // Replacing ⟨MASK⟩ tokens with predicted ones
  $\mathbf{x}_{\text{mut}} \leftarrow \textsc{Argmax}(\textsc{Softmax}(\mathbf{X}_{\text{repr}}))$
  // Predicting fitness
  $\mathbf{y}_{\text{mut}} \leftarrow \textsc{Attention1D}(\mathbf{R}_{\text{feat}})$
  // Choosing best variants
  $\mathbf{x}_{\text{comb}} \leftarrow \textsc{Concat}(\mathbf{x}_{\text{mut}}, \mathbf{x}_{\text{prev}})$
  $\mathbf{y}_{\text{comb}} \leftarrow \textsc{Concat}(\mathbf{y}_{\text{mut}}, \mathbf{y}_{\text{prev}})$
  $\mathbf{x}_{\text{top}}, \mathbf{y}_{\text{top}} \leftarrow \textsc{TopK}(\mathbf{x}_{\text{comb}}, \mathbf{y}_{\text{comb}}, n)$
  $\mathbf{x}_{\text{prev}} \leftarrow \mathbf{x}_{\text{top}}$
  $\mathbf{y}_{\text{prev}} \leftarrow \mathbf{y}_{\text{top}}$
end for

B. Machine learning-guided improvement

After the generation of a population of masked sequences, these sequences are subsequently inputted into the ESM-2 model. Following this, the neural network generates predictions for the masked tokens and produces a latent representation of these variations. In the final stage, we evaluate the sequences that have been newly proposed by measuring their fitness values. For conditional protein generation, a simple deep learning model is developed to establish a mapping between the protein representation and its corresponding fitness. For the sake of comparison, following [37] and [57], we employ the feature set derived from ESM-2 and utilize a fine-tuned Attention1D decoder to predict the fitness values. Given the input representation $\mathbf{X}' \in \mathbb{R}^{B \times L \times d_0}$, the architecture of this modules is as follows:

$$\begin{aligned}
\mathbf{X}_2 &= \text{ReLU}(\text{Linear}_1(X')) && \in \mathbb{R}^{B \times L \times d_{\text{hid}}} \\
\mathbf{X}_3 &= \text{ReLU}(\text{Linear}_2(X_2)) && \in \mathbb{R}^{B \times L \times d_{\text{hid}}} \\
\mathbf{X}_4 &= \text{Attention1D}(\mathbf{X}_3) && \in \mathbb{R}^{B \times d_{\text{hid}}} \\
\mathbf{X}_5 &= \text{ReLU}(\text{Linear}_3(X_4)) && \in \mathbb{R}^{B \times d_{\text{hid}}} \\
\mathbf{X}'' &= \text{ReLU}(\text{Linear}_4(X_5)) && \in \mathbb{R}^{B \times 1}
\end{aligned}$$

where B is the number of sequences, L is the input length, and d_{hid} is the hidden dimension of the module. This model is subsequently trained to comprehend the fitness landscape of the specified protein sequence. The values are thereafter arranged in descending order to generate the most optimal

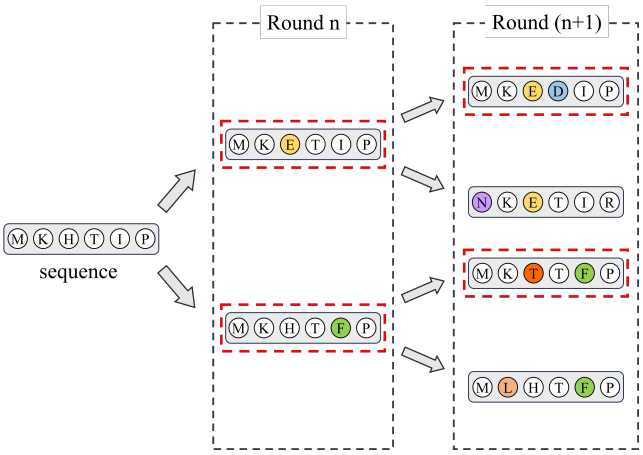


Fig. 2. Illustration of beam search adaptation in directed evolution. In this figure, the population number is 2, and the beam size is 2, which makes the *effective* population size 4 (i.e., sequences in the third generation before dropping inferior ones). The sequences covered in red boxes are chosen to move to the next iteration.

sequences that will proceed to the subsequent iteration of the pipeline.

In summary, our pipeline can be described by Algorithm 1. It is important to note that while selecting the best variants in each iteration, we combine the present "effective" population with the population from the preceding iteration. This procedure guarantees that in the event of mutations resulting in a reduction in fitness values, the pipeline will opt for the variations from the preceding round rather than the inferior ones.

V. Experiments

In this section, we conduct extensive experiments to validate the effectiveness of our proposed pipeline on protein sequence design task.

A. Implementation Details

Starting with a population of wild-type protein sequences, we split them into two separate groups. Each group is then subjected to a specific masking strategy, resulting in a population of masked sequences. These sequences are then fed to the pretrained 35-million-parameter version of the ESM-2 model, which generates the representation of the sequences and introduces mutations to sequences by replacing the $\langle$MASK$\rangle$ with the amino acid that the model confidently predicted. This representation is then fed into the oracle described in Section IV-B to predict the fitness value of sequences. Subsequently, the population is arranged in a descending order to select the optimal n variations according to their respective fitness values. These variants are the prospective candidates that will undergo the same pipeline in the subsequent iteration. Following several rounds, we would observe enhanced protein sequences exhibiting improved fitness ratings.

In our settings, we set the population size to 128 following [53]. Inspired by the beam search algorithm, which has proven its efficiency in the field of natural language processing [71], [72], the model proposes 4 separate mutated sequences (i.e., beam size of 4), instead of 1 mutation in traditional directed evolution, to each variant in the population, making the *effective* population size to 512. This adaptation is illustrated in Figure 2. We configure the random ratio to 0.6 and the k-mer size to 1 for the remaining hyperparameters. It is important to acknowledge that the outcomes generated from these configurations are not guaranteed to be optimal. The examination of the influence of each hyper-parameters is conducted in Section V-F.

To train the Attention1D model, we use a mini-batch size of 256 and a learning rate of 10^{-4}. We apply the Adam algorithm [73] as the optimizer. We randomly split each dataset into training and validation sets with a ratio of 9:1. We run all the experiments five times and report the average scores with their standard deviation.

B. Datasets

Following [37] and [57], we evaluate our method on the following five protein engineering benchmark datasets:

1) Green Fluorescent Proteins (avGFP): Derived from Aequorea victoria, Green Fluorescent proteins (GFPs) are capable of manifesting vivid green fluorescence upon exposure to light within the blue to ultraviolet spectrum. These proteins are commonly employed as biosensors for detecting gene expressions and protein locations. We collect data following [74]. Our objective is to design sequences with enhanced capabilities as gene delivery vectors, quantified by AAV viabilities. The magnitude of the search space encompasses 20^{28} possibilities.

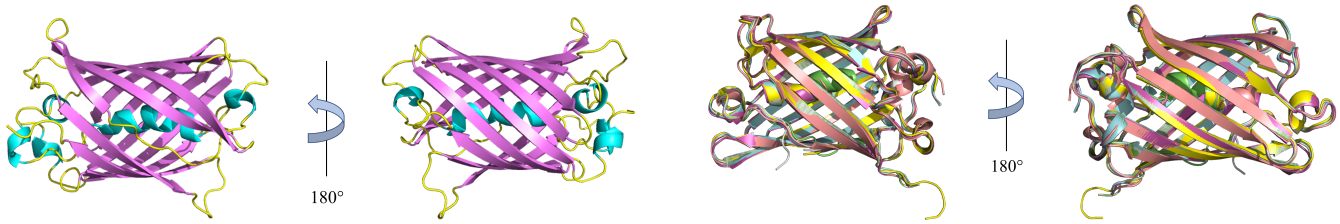
2) Adeno-associated Viruses (AAV): The engineer of a 28-amino acid segment (position 561–588) within the VP1 protein, situated in the capsid of the Adeno-associated virus, has garnered significant interest in the realm of machine learning-guided design. The data is obtained from [75]. We aim to design more capable sequences as gene delivery vectors measured by AAV viabilities. The size of the search space is 20^{28}.

3) Aliphatic Amide Hydrolase (AMIE): The enzyme encoded by amiE, known as Amidase, holds industrial relevance and is derived from Pseudomonas aeruginosa. The data was collected from the study conducted by [76], resulting in two datasets representing distinct fitness landscapes, each associated with different substrates. Our objective is to optimize amidase sequences that result in enhanced enzyme activities, defining a search space comprising 20^{341} sequences.

4) TEM-1 β-Lactamase (TEM): The investigation of the TEM-1 β-Lactamase protein's resistance to penicillin antibiotics in E. coli is a subject of extensive scrutiny, aiming to comprehend mutational impacts and the associated fitness landscape. The data is collected from [77]. The optimization objective involves suggesting sequences with elevated thermodynamic stability compared to the wild-type TEM-1 within a search space of magnitude 20^{286}.

TABLE I
MAXIMUM FITNESS SCORES (MFS) OF ALL METHODS AND AVERAGE FITNESS SCORES (AFS) OF OUR METHOD ON FIVE DATASETS. HIGHER VALUES INDICATE BETTER FUNCTIONAL PROPERTIES IN THE DATASET. RESULTS WITH '†' ARE TAKEN FROM [57].

Models	avGFP	AAV	TEM	E4B	UBE2I	Average
CMA-ES†	4.492	-3.417	0.375	-0.768	2.461	0.629
FBGAN†	1.251	-4.227	0.006	0.369	0.208	-0.479
DbAS†	3.548	4.327	0.003	-1.286	3.088	1.936
CbAS†	3.550	4.336	0.106	-1.000	3.263	2.051
PEX†	3.764	3.265	0.121	5.019	-0.474	2.339
GFlowNet-AL†	5.062	1.205	1.552	3.155	3.576	2.910
IsEM-Pro†	6.185	4.813	1.850	5.737	**4.536**	4.624
MLDE	**11.796 ± 0.676**	**6.585 ± 0.367**	**6.731 ± 0.120**	**10.311 ± 0.519**	2.978 ± 0.099	**7.680 ± 0.356**
MLDE (AFS)	11.670 ± 0.671	6.171 ± 0.374	6.675 ± 0.126	10.176 ± 0.485	2.874 ± 0.098	7.513 ± 0.351
MLDE-random	10.800 ± 0.501	6.231 ± 0.570	6.635 ± 0.150	10.161 ± 0.718	2.744 ± 0.013	7.314 ± 0.390



(a) 3D visualization of sequence with highest fitness.

(b) Superposition of 5 fluorescent proteins in top-5 templates.

Fig. 3. 3D visualization of a designed sequence of green fluorescent protein, validating that our method is highly likely to design a real fluorescent protein.

TABLE II
MFS OF THREE METHODS ON THE AMIE DATASET. HIGHER VALUES INDICATE BETTER FUNCTIONAL PROPERTIES IN THE DATASET. RESULTS WITH '◊' ARE TAKEN FROM [57] AND ASSUMED TO BE AVERAGE SCORES.

	PEX◊	IsEM-Pro◊	MLDE
Acetamide	-	-	0.041 ± 0.001
Isobutyramide	-	-	0.972 ± 0.020
Propionamide	-	-	-0.094 ± 0.002
Average	-0.474	0.062	**0.306 ± 0.008**

5) Ubiquitination Factor Ube4b (E4B): The ubiquitination factor Ube4b plays a pivotal role in cellular waste degradation through interactions with ubiquitin and other proteins. We collect data following [78]. Our emphasis is on the enhancement of E4B enzyme activity within the specified landscape, which encompasses a search space of 20^{102}.

6) SUMO E2 conjugase (UBE2I): The utilization of variants for the functional mapping of human genomes holds considerable significance in both scientific research and clinical treatment. We gather data following [79]. The search space encompasses a size of 20^{159} within a segment of the wild-type sequence.

C. Baseline Algorithms

We assess the performance of our method in comparison to several representative baselines: (1) **CMA-ES** [80] is a famous evolutionary search algorithm. (2) **FBGAN** [62] a novel feedback-loop architecture incorporating a generative model GAN. (3) **DbAS** [81] is a probabilistic modeling framework employing an adaptive sampling algorithm. (4) **CbAS** [82] is an improvement over DbAS that conditions on desired properties. (5) **PEX** [37] is a model-guided sequence design algorithm using proximal exploration. (6) **GFlowNet-AL** [53] applies GFlowNet to design biological sequences. (7) **IsEM-Pro** [57] is a latent generative model employing importance-weighted EM for learning. To comprehensively assess the impact of the importance masking strategy, which serves as a means to enhance performance in our pipeline, we conducted an ablation study using only the random masking strategy for sequence masking, denoted as **DE-random**.

D. Evaluation Metrics

We use two metrics to evaluate our pipeline for designing sequences: (1) **MFS**: Maximum fitness score, and (2) **AFS**: Average fitness score. All of these metrics are applied to the last population of candidates.

E. Main Results

As described in Table I, our approach demonstrates superior fitness scores in 4 out of 5 protein families, surpassing the performance of the previous leading method, IsEM-Pro, by an average of 60.2%. The AFS ratings demonstrate that our technique regularly produces sequences of good quality, with only a slight lag compared to the MFS. These results demonstrate that our method exhibits comparable performance to generative models and even outperforms them in the generation of novel protein sequences.

Table II compares the performance of the two most recent baselines: **PEX** and **IsEM-Pro**, with our method. As the results reported in [57] were not explicitly categorized, we assume that they represent an average across all groups. It

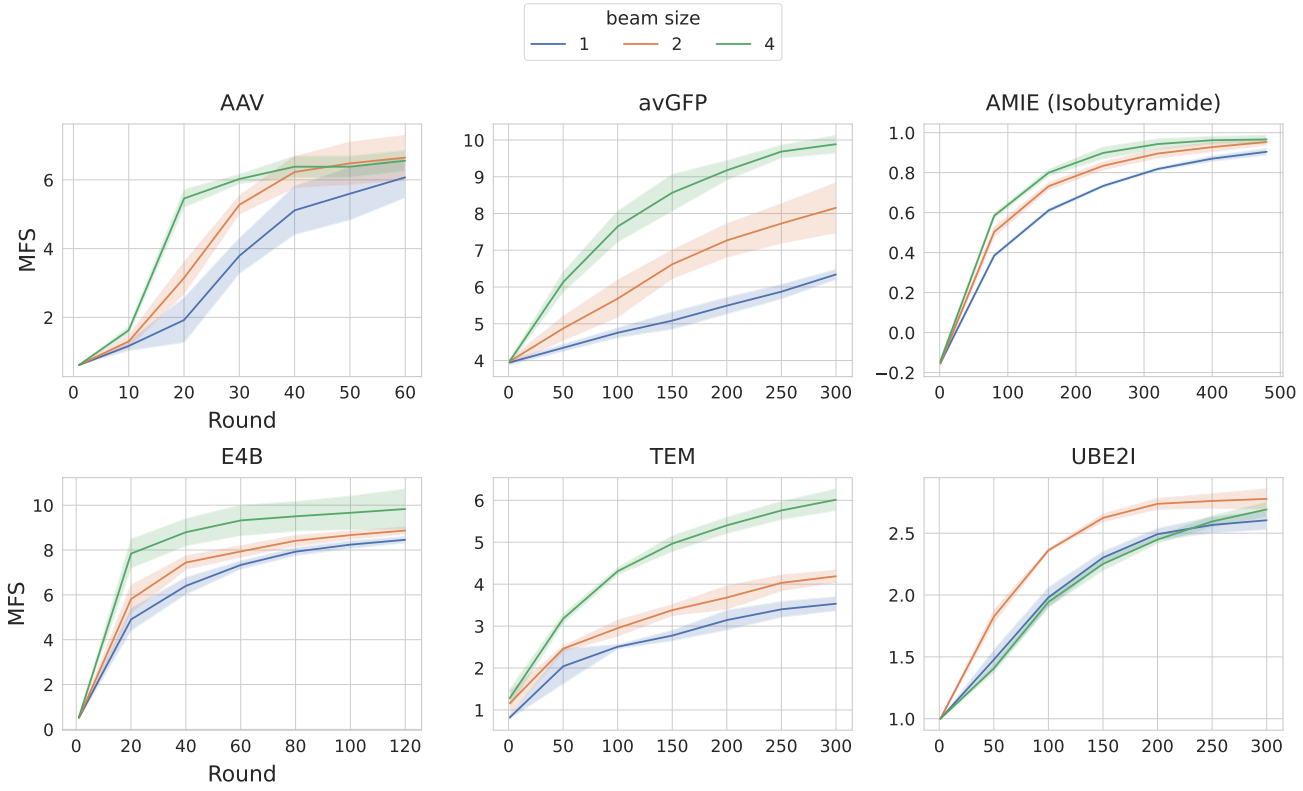


Fig. 4. Effects of different beam sizes (k-mer size is fixed to 1) on five examined tasks. The x-axis is the iteration, and the y-axis is the fitness score. Isobutyramide is selected as the representative example for the AMIE dataset.

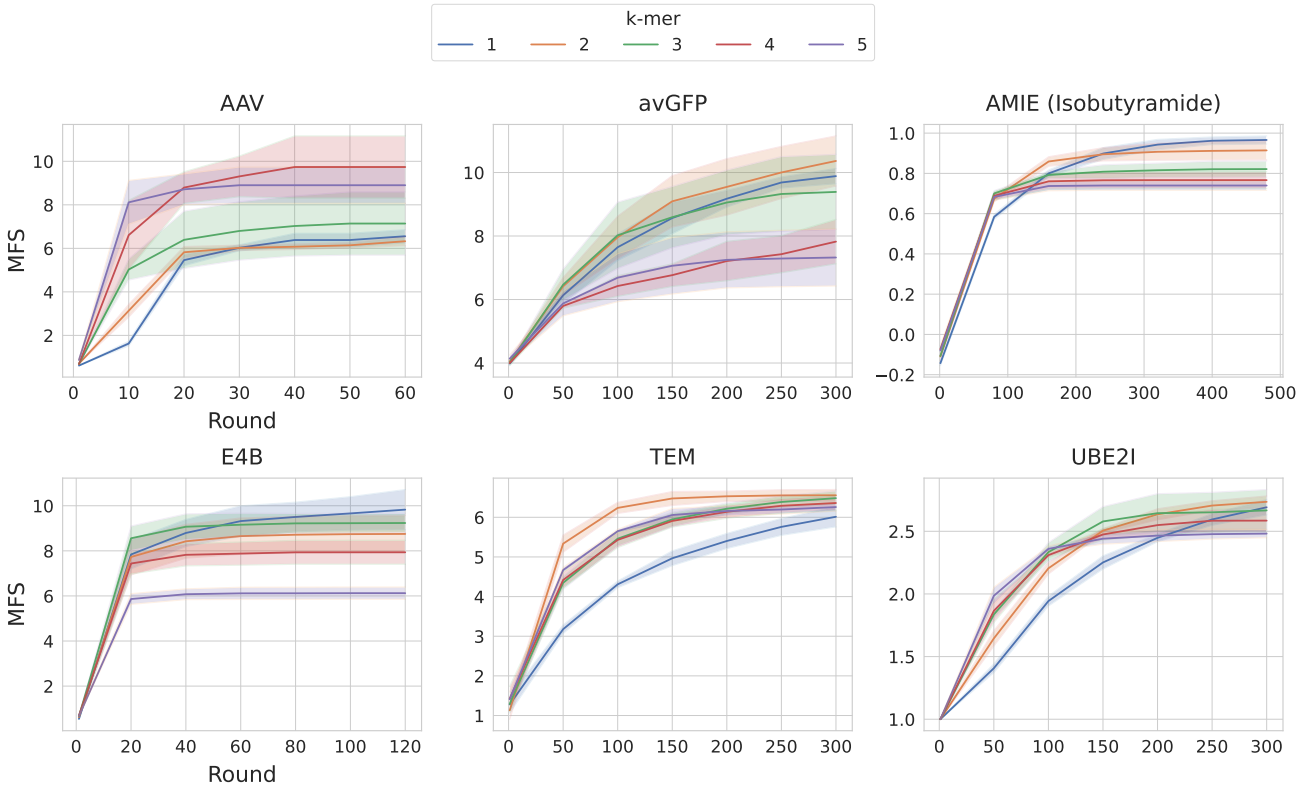


Fig. 5. Effects of different k-mer sizes to mask (beam size is fixed to 4) on five examined tasks. The x-axis is the iteration, and the y-axis is the fitness score. Isobutyramide is selected as the representative example for the AMIE dataset.

is evident from the table that our method outperforms both baselines in terms of the average score, underscoring the quality of the proposed approach.

F. Ablation Study

1) Effect of masking strategy: The lower section of Table I presents the findings from ablation experiments conducted to evaluate the efficacy of various masking strategies. The implementation of the importance masking strategy results in a marginal improvement in the overall performance of our method across all datasets.

2) Effect of beam size: The impact of varying beam sizes is depicted in Fig. 4. The graphic illustrates that increasing the beam size leads to accelerated convergence of the pipeline and results in sequences with better fitness values throughout each iteration. Consequently, this further illustrates the efficacy of the adaptation of the beam search technique.

3) Effect of k-mer size: The efficiency of selecting the k-mer size for sequence masking is depicted in Fig. 5. While the impact of k-mer selection on overall performance is minimal in certain datasets, such as UBE2I, this decision can significantly enhance performance in other benchmark scenarios. As an illustration, the MFS score exhibits a 1.6-fold increase when the k-mer size is increased from 2 to 4. Therefore, it is evident that the impact of the datasets varies and requires careful adjustment in order to achieve optimal outcomes.

VI. Analysis

A. Case study

To assess the quality of the designed proteins, we conducted a detailed analysis of the generated avGFP, AMIE, E4B, TEM, and UBE2I sequences with the highest fitness using the PhyRe2 tool [83]. Taking avGFP as an example, Figure 3a illustrates that the generated protein sequence can fold stably. According to the software, the most similar protein is human Jagunal homolog 1 (Human JAGN1) [84]. A total of 230 residues, constituting 96% of the candidate sequence, have been modelled with 100% confidence using this protein as a template. Figure 3b illustrates the superposition of the five templates in the Protein Data Bank that are most similar to our sequence. These templates are all fluorescent proteins and exhibit a high degree of structural consistency across the majority of their regions. Additional details about other protein sequences are provided in Appendix. **??**.

B. Limitations

One drawback of our study is that, despite the positive outcomes of our approach, the protein sequences we created have not been verified through *in-vitro* experimentation, which introduces a level of uncertainty associated with the oracle model. The results presented in our research are obtained by averaging data from five separate runs with different seeds, which can accommodate some variance on these metrics. Additionally, all fitness data utilized in our study are derived from web-lab experiments, ensuring that the fitness values within the training datasets are realistic. Although there is presently no perfect metric available, we believe that new methods should be encouraged in the field of computational biology.

VII. Conclusion

This paper introduces a Machine Learning-Driven Evolution (MLDE) framework. Incorporating two distinct masking strategies, namely random masking and importance masking, our pipeline utilizes pretrained models with minimal training, resulting in protein sequences exhibiting higher fitness. Experimental results across six protein sequence design tasks demonstrate that our method surpasses several robust baselines across nearly all metrics. In the future work, we aim to apply our Directed Evolution framework to the task of ligand-binding protein redesign, antibody design, etc. We hope our framework will enable scientists and researchers to efficiently optimize proteins with certain given properties.

References

[1] C. Hsu, H. Nisonoff, C. Fannjiang, and J. Listgarten, "Learning protein fitness models from evolutionary and assay-labeled data," *Nature Biotechnology*, vol. 40, pp. 1114–1122, Jul 2022.

[2] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, pp. 583–589, Aug 2021.

[3] T. A. Hopf, J. B. Ingraham, F. J. Poelwijk, C. P. I. Schärfe, M. Springer, C. Sander, and D. S. Marks, "Mutation effects predicted from sequence co-variation," *Nature Biotechnology*, vol. 35, pp. 128–135, Feb 2017.

[4] A. J. Riesselman, J. B. Ingraham, and D. S. Marks, "Deep generative models of genetic variation capture the effects of mutations," *Nature Methods*, vol. 15, pp. 816–822, Oct 2018.

[5] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, D. Bhowmik, and B. Rost, "ProtTrans: Toward understanding the language of life through self-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, pp. 7112–7127, Oct. 2022.

[6] B. Hie, B. D. Bryson, and B. Berger, "Leveraging uncertainty in machine learning accelerates biological discovery and design," *Cell Systems*, vol. 11, pp. 461–477.e9, Nov. 2020.

[7] H. T. Rube, C. Rastogi, S. Feng, J. F. Kribelbauer, A. Li, B. Becerra, L. A. N. Melo, B. V. Do, X. Li, H. H. Adam, N. H. Shah, R. S. Mann, and H. J. Bussemaker, "Prediction of protein–ligand binding affinity from sequencing data with interpretable machine learning," *Nature Biotechnology*, vol. 40, pp. 1520–1527, Oct 2022.

[8] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, and R. Fergus, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *Proceedings of the National Academy of Sciences*, vol. 118, no. 15, p. e2016239118, 2021.

[9] R. Verkuil, O. Kabeli, Y. Du, B. I. Wicky, L. F. Milles, J. Dauparas, D. Baker, S. Ovchinnikov, T. Sercu, and A. Rives, "Language models generalize beyond natural proteins," *bioRxiv*, pp. 2022–12, 2022.

[10] K. K. Yang, Z. Wu, and F. H. Arnold, "Machine learning in protein engineering," *arXiv preprint arXiv:1811.10775*, 2018.

[11] S. Mazurenko, Z. Prokop, and J. Damborsky, "Machine learning in enzyme engineering," *ACS Catalysis*, vol. 10, no. 2, pp. 1210–1223, 2020.

[12] N. A. Pierce and E. Winfree, "Protein Design is NP-hard," *Protein Engineering, Design and Selection*, vol. 15, pp. 779–782, 10 2002.

[13] J. M. SMITH, "Natural selection and the concept of a protein space," *Nature*, vol. 225, pp. 563–564, Feb. 1970.

[14] H. A. Orr, "The distribution of fitness effects among beneficial mutations in fisher's geometric model of adaptation," *Journal of Theoretical Biology*, vol. 238, no. 2, pp. 279–285, 2006.

[15] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[16] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, *et al.*, "Bloom: A 176b-parameter open-access multilingual language model," *arXiv preprint arXiv:2211.05100*, 2022.

[17] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 6840–6851, Curran Associates, Inc., 2020.

[19] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[20] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar, *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," *arXiv preprint arXiv:2306.15687*, 2023.

[21] Z. Wu, K. K. Yang, M. J. Liszka, A. Lee, A. Batzilla, D. Wernick, D. P. Weiner, and F. H. Arnold, "Signal peptides generated by attention-based neural networks," *ACS Synthetic Biology*, vol. 9, no. 8, pp. 2154–2161, 2020. PMID: 32649182.

[22] D. Repecka, V. Jauniskis, L. Karpus, E. Rembeza, I. Rokaitis, J. Zrimec, S. Poviloniene, A. Laurynenas, S. Viknander, W. Abuajwa, O. Savolainen, R. Meskys, M. K. M. Engqvist, and A. Zelezniak, "Expanding functional protein sequence spaces using generative adversarial networks," *Nature Machine Intelligence*, vol. 3, pp. 324–333, Apr 2021.

[23] A. Hawkins-Hooker, F. Depardieu, S. Baur, G. Couairon, A. Chen, and D. Bikard, "Generating functional protein variants with variational autoencoders," *PLOS Computational Biology*, vol. 17, p. e1008736, Feb. 2021.

[24] F. H. Arnold, "Directed evolution: Creating biocatalysts for the future," *Chemical Engineering Science*, vol. 51, no. 23, pp. 5091–5102, 1996.

[25] O. Kuchner and F. H. Arnold, "Directed evolution of enzyme catalysts," *Trends in biotechnology*, vol. 15, no. 12, pp. 523–530, 1997.

[26] F. H. Arnold, "Design by directed evolution," *Accounts of chemical research*, vol. 31, no. 3, pp. 125–131, 1998.

[27] F. H. Arnold and A. A. Volkov, "Directed evolution of biocatalysts," *Current opinion in chemical biology*, vol. 3, no. 1, pp. 54–59, 1999.

[28] E. T. Farinas, T. Bulter, and F. H. Arnold, "Directed enzyme evolution," *Current opinion in biotechnology*, vol. 12, no. 6, pp. 545–551, 2001.

[29] C. A. Tracewell and F. H. Arnold, "Directed enzyme evolution: climbing fitness peaks one amino acid at a time," *Current opinion in chemical biology*, vol. 13, no. 1, pp. 3–9, 2009.

[30] P. Romero and F. Arnold, "Exploring protein fitness landscapes by directed evolution," *Nature reviews. Molecular cell biology*, vol. 10, pp. 866–76, 12 2009.

[31] F. H. Arnold, "Directed evolution: Bringing new chemistry to life," *Angewandte Chemie International Edition*, vol. 57, no. 16, pp. 4143–4148, 2018.

[32] P. A. Romero, A. Krause, and F. H. Arnold, "Navigating the protein fitness landscape with gaussian processes," *Proceedings of the National Academy of Sciences*, vol. 110, no. 3, pp. E193–E201, 2013.

[33] C. N. Bedbrook, K. K. Yang, A. J. Rice, V. Gradinaru, and F. H. Arnold, "Machine learning to design integral membrane channelrhodopsins for efficient eukaryotic expression and plasma membrane localization," *PLOS Computational Biology*, vol. 13, p. e1005786, Oct. 2017.

[34] C. N. Bedbrook, K. K. Yang, J. E. Robinson, E. D. Mackey, V. Gradinaru, and F. H. Arnold, "Machine learning-guided channelrhodopsin engineering enables minimally invasive optogenetics," *Nature Methods*, vol. 16, pp. 1176–1184, Oct. 2019.

[35] Z. Wu, S. B. J. Kan, R. D. Lewis, B. J. Wittmann, and F. H. Arnold, "Machine learning-assisted directed protein evolution with combinatorial libraries," *Proceedings of the National Academy of Sciences*, vol. 116, no. 18, pp. 8852–8858, 2019.

[36] P. Emami, A. Perreault, J. Law, D. Biagioni, and P. S. John, "Plug & play directed evolution of proteins with gradient-based discrete mcmc," *Machine Learning: Science and Technology*, vol. 4, p. 025014, apr 2023.

[37] Z. Ren, J. Li, F. Ding, Y. Zhou, J. Ma, and J. Peng, "Proximal exploration for model-guided protein sequence design," in *Proceedings of the 39th International Conference on Machine Learning* (K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, eds.), vol. 162 of *Proceedings of Machine Learning Research*, pp. 18520–18536, PMLR, 17–23 Jul 2022.

[38] Y. Qiu, J. Hu, and G.-W. Wei, "Cluster learning-assisted directed evolution," *Nature Computational Science*, vol. 1, pp. 809–818, Dec 2021.

[39] Y. Qiu and G.-W. Wei, "Clade 2.0: Evolution-driven cluster learning-assisted directed evolution," *Journal of Chemical Information and Modeling*, vol. 62, no. 19, pp. 4629–4641, 2022. PMID: 36154171.

[40] B. J. Wittmann, Y. Yue, and F. H. Arnold, "Informed training set design enables efficient machine learning-assisted directed protein evolution," *Cell Systems*, vol. 12, pp. 1026–1045.e7, Nov. 2021.

[41] D. C. E. Koo and R. Bonneau, "Towards region-specific propagation of protein functions," *Bioinformatics*, vol. 35, pp. 1737–1744, 10 2018.

[42] W. Torng and R. B. Altman, "High precision protein functional site detection using 3D convolutional neural networks," *Bioinformatics*, vol. 35, pp. 1503–1512, 09 2018.

[43] Y. Guan, C. L. Myers, D. C. Hess, Z. Barutcuoglu, A. A. Caudy, and O. G. Troyanskaya, "Predicting gene function in a hierarchical context with an ensemble of classifiers," *Genome Biology*, vol. 9, no. Suppl 1, p. S3, 2008.

[44] M. N. Wass, G. Barton, and M. J. E. Sternberg, "CombFunc: predicting protein function using heterogeneous data sources," *Nucleic Acids Research*, vol. 40, pp. W466–W470, 05 2012.

[45] M. Kulmanov, M. A. Khan, and R. Hoehndorf, "DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier," *Bioinformatics*, vol. 34, pp. 660–668, 10 2017.

[46] M. Kulmanov and R. Hoehndorf, "DeepGOPlus: improved protein function prediction from sequence," *Bioinformatics*, vol. 36, pp. 422–429, 07 2019.

[47] V. Gligorijević, P. D. Renfrew, T. Kosciolek, J. K. Leman, D. Berenberg, T. Vatanen, C. Chandler, B. C. Taylor, I. M. Fisk, H. Vlamakis, R. J. Xavier, R. Knight, K. Cho, and R. Bonneau, "Structure-based protein function prediction using graph convolutional networks," *Nature Communications*, vol. 12, May 2021.

[48] Z. Wang, S. A. Combs, R. Brand, M. R. Calvo, P. Xu, G. Price, N. Golovach, E. O. Salawu, C. J. Wise, S. P. Ponnapalli, and P. M. Clark, "Lm-gvp: an extensible sequence and structure informed deep learning framework for protein property prediction," *Scientific Reports*, vol. 12, p. 6832, Apr 2022.

[49] K. Ngo and T. S. Hy, "Target-aware variational auto-encoders for ligand generation with multi-modal protein modeling," in *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, 2023.

[50] N. K. Ngo, T. S. Hy, and R. Kondor, "Multiresolution graph transformers and wavelet positional encoding for learning long-range and hierarchical structures," *The Journal of Chemical Physics*, vol. 159, p. 034109, 07 2023.

[51] Y. Gumulya, J.-M. Baek, S.-J. Wun, R. E. S. Thomson, K. L. Harris, D. J. B. Hunter, J. B. Y. H. Behrendorff, J. Kulig, S. Zheng, X. Wu, B. Wu, J. E. Stok, J. J. D. Voss, G. Schenk, U. Jurva, S. Andersson, E. M. Isin, M. Bodén, L. Guddat, and E. M. J. Gillam, "Engineering highly functional thermostable proteins using ancestral sequence reconstruction," *Nature Catalysis*, vol. 1, pp. 878–888, Oct. 2018.

[52] C. Angermueller, D. Dohan, D. Belanger, R. Deshpande, K. Murphy, and L. Colwell, "Model-based reinforcement learning for biological sequence design," in *International conference on learning representations*, 2019.

[53] M. Jain, E. Bengio, A. Hernandez-Garcia, J. Rector-Brooks, B. F. P. Dossou, C. A. Ekbote, J. Fu, T. Zhang, M. Kilgour, D. Zhang, L. Simine, P. Das, and Y. Bengio, "Biological sequence design with GFlowNets," in *Proceedings of the 39th International Conference on Machine Learning* (K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, eds.), vol. 162 of *Proceedings of Machine Learning Research*, pp. 9786–9801, PMLR, 17–23 Jul 2022.

[54] D. Belanger, S. Vora, Z. Mariet, R. Deshpande, D. Dohan, C. Angermueller, K. Murphy, O. Chapelle, and L. Colwell, "Biological sequences design using batched bayesian optimization," 2019.

[55] H. Moss, D. Leslie, D. Beck, J. González, and P. Rayson, "Boss: Bayesian optimization over string spaces," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 15476–15486, Curran Associates, Inc., 2020.

[56] K. Terayama, M. Sumita, R. Tamura, and K. Tsuda, "Black-box optimization for automated discovery," *Accounts of Chemical Research*, vol. 54, no. 6, pp. 1334–1346, 2021. PMID: 33635621.

[57] Z. Song and L. Li, "Importance weighted expectation-maximization for protein sequence design," in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho,

B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 32349–32364, PMLR, 23–29 Jul 2023.

[58] K. Swersky, Y. Rubanova, D. Dohan, and K. Murphy, "Amortized bayesian optimization over discrete spaces," in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)* (J. Peters and D. Sontag, eds.), vol. 124 of *Proceedings of Machine Learning Research*, pp. 769–778, PMLR, 03–06 Aug 2020.

[59] N. Ferruz, S. Schmidt, and B. Höcker, "Protgpt2 is a deep unsupervised language model for protein design," *Nature Communications*, vol. 13, p. 4348, Jul 2022.

[60] E. Nijkamp, J. Ruffolo, E. N. Weinstein, N. Naik, and A. Madani, "Progen2: exploring the boundaries of protein language models," *arXiv preprint arXiv:2206.13517*, 2022.

[61] X. Ding, Z. Zou, and C. L. Brooks III, "Deciphering protein evolution and fitness landscapes with latent space models," *Nature Communications*, vol. 10, p. 5644, Dec 2019.

[62] A. Gupta and J. Zou, "Feedback gan for dna optimizes protein functions," *Nature Machine Intelligence*, vol. 1, pp. 105–111, Feb 2019.

[63] J. L. Watson, D. Juergens, N. R. Bennett, B. L. Trippe, J. Yim, H. E. Eisenach, W. Ahern, A. J. Borst, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, N. Hanikel, S. J. Pellock, A. Courbet, W. Sheffler, J. Wang, P. Venkatesh, I. Sappington, S. V. Torres, A. Lauko, V. De Bortoli, E. Mathieu, S. Ovchinnikov, R. Barzilay, T. S. Jaakkola, F. DiMaio, M. Baek, and D. Baker, "De novo design of protein structure and function with rfdiffusion," *Nature*, vol. 620, pp. 1089–1100, Aug 2023.

[64] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[65] J. Devlin, M.-W. Chang, K. Lee, and K. 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, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019.

[66] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, and A. Rives, "Evolutionary-scale prediction of atomic-level protein structure with a language model," *Science*, vol. 379, no. 6637, pp. 1123–1130, 2023.

[67] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," 2022.

[68] J. Chen, A. Zhang, M. Li, A. Smola, and D. Yang, "A cheaper and better diffusion language model with soft-masked noise," 2023.

[69] D. Dessì, R. Helaoui, V. Kumar, D. R. Recupero, and D. Riboni, "TF-IDF vs word embeddings for morbidity identification in clinical notes: An initial study," in *Proceedings of the First Workshop on Smart Personal Health Interfaces co-located with 25th International Conference on Intelligent User Interfaces, SmartPhil@IUI 2020, Cagliari, Italy, March 17, 2020* (S. Consoli, D. R. Recupero, and D. Riboni, eds.), vol. 2596 of *CEUR Workshop Proceedings*, pp. 1–12, CEUR-WS.org, 2020.

[70] C. Bentz and D. Alikaniotis, "The word entropy of natural languages," 2016.

[71] A. Graves, "Sequence transduction with recurrent neural networks," 2012.

[72] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, "Audio chord recognition with recurrent neural networks.," in *ISMIR*, pp. 335–340, Curitiba, 2013.

[73] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[74] K. S. Sarkisyan, D. A. Bolotin, M. V. Meer, D. R. Usmanova, A. S. Mishin, G. V. Sharonov, D. N. Ivankov, N. G. Bozhanova, M. S. Baranov, O. Soylemez, N. S. Bogatyreva, P. K. Vlasov, E. S. Egorov, M. D. Logacheva, A. S. Kondrashov, D. M. Chudakov, E. V. Putintseva, I. Z. Mamedov, D. S. Tawfik, K. A. Lukyanov, and F. A. Kondrashov, "Local fitness landscape of the green fluorescent protein," *Nature*, vol. 533, pp. 397–401, May 2016.

[75] D. H. Bryant, A. Bashir, S. Sinai, N. K. Jain, P. J. Ogden, P. F. Riley, G. M. Church, L. J. Colwell, and E. D. Kelsic, "Deep diversification of an aav capsid protein by machine learning," *Nature Biotechnology*, vol. 39, pp. 691–696, Jun 2021.

[76] E. E. Wrenbeck, L. R. Azouz, and T. A. Whitehead, "Single-mutation fitness landscapes for an enzyme on multiple substrates reveal specificity is globally encoded," *Nature Communications*, vol. 8, p. 15695, Jun 2017.

[77] E. Firnberg, J. W. Labonte, J. J. Gray, and M. Ostermeier, "A Comprehensive, High-Resolution Map of a Gene's Fitness Landscape," *Molecular Biology and Evolution*, vol. 31, pp. 1581–1592, 02 2014.

[78] L. M. Starita, J. N. Pruneda, R. S. Lo, D. M. Fowler, H. J. Kim, J. B. Hiatt, J. Shendure, P. S. Brzovic, S. Fields, and R. E. Klevit, "Activity-enhancing mutations in an e3 ubiquitin ligase identified by high-throughput mutagenesis," *Proceedings of the National Academy of Sciences*, vol. 110, no. 14, pp. E1263–E1272, 2013.

[79] J. Weile, S. Sun, A. G. Cote, J. Knapp, M. Verby, J. C. Mellor, Y. Wu, C. Pons, C. Wong, N. van Lieshout, F. Yang, M. Tasan, G. Tan, S. Yang, D. M. Fowler, R. Nussbaum, J. D. Bloom, M. Vidal, D. E. Hill, P. Aloy, and F. P. Roth, "A framework for exhaustively mapping functional missense variants," *Molecular Systems Biology*, vol. 13, no. 12, p. 957, 2017.

[80] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[81] D. H. Brookes and J. Listgarten, "Design by adaptive sampling," 2020.

[82] D. Brookes, H. Park, and J. Listgarten, "Conditioning by adaptive sampling for robust design," in *Proceedings of the 36th International Conference on Machine Learning* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 773–782, PMLR, 09–15 Jun 2019.

[83] L. A. Kelley, S. Mezulis, C. M. Yates, M. N. Wass, and M. J. E. Sternberg, "The phyre2 web portal for protein modeling, prediction and analysis," *Nature Protocols*, vol. 10, pp. 845–858, Jun 2015.

[84] S. Liu, S. Li, Y. Yang, and W. Li, "Termini restraining of small membrane proteins enables structure determination at near-atomic resolution," *Science Advances*, vol. 6, no. 51, p. eabe3717, 2020.