EvoOpt: an MSA-guided, fully unsupervised sequence optimization pipeline for protein design

Hideki Yamaguchi
The University of Tokyo
National Institute of Advanced Industrial Science and Technology (AIST)
yamaguchi_hideki_20@stu-cbms.k.u-tokyo.ac.jp

Yutaka Saito
National Institute of Advanced Industrial Science and Technology (AIST)
AIST-Waseda University Computational Bio Big-Data Open Innovation Laboratory (CBBD-OIL)
yutaka.saito@aist.go.jp

Abstract

Recent years have seen rapid growth in machine learning algorithms for protein design. Among them, protein sequence optimization methods to maximize molecular functionality can significantly impact many industries. However, as shown in this study, most existing methods are data-hungry: they tend to be low-performant when available training data is scarce (i.e., in a low-N regime), which is often the case in practical protein engineering scenarios. In response, here we examine the extreme case: what if we have no training data? To answer, we propose a fully unsupervised sequence optimization pipeline named EvoOpt that leverages evolutionary information provided by multiple sequence alignments (MSAs) and the generative power of MSA Transformer, a protein language model (PLM) that takes an MSA as input. The extensive evaluation herein demonstrates that EvoOpt outperforms or is on par with the existing supervised methods even in relatively high-N regimes. We also report that the optimization performance with MSA Transformer is almost equivalent to or superior to that with a PLM that takes a single sequence as input, such as ESM-1b or ESM2 of far more model parameters. These results indicate the advantage of using an MSA to guide an algorithm toward promising candidates in the search space, directly exploiting evolutionary information.

1 Introduction

Recently, designing highly functional proteins is gaining much attention as the industries like antibody therapeutics or biomanufacturing proliferate. In particular, directed evolution [1] is a standard experimental approach that improves a natural protein's molecular function, or fitness, by iterative mutation and selection of better candidates. Because wet experiments are often labor-intensive and financially costly, machine learning-based maximization of protein functions, or sequence optimization, is emerging as a promising *in silico* alternative.

Current sequence optimization algorithms are based on supervised learning, even though they take different approaches, such as Bayesian optimization [2], generative models [3, 4], evolutionary strategy [5], reinforcement learning [6], and more. However, the training data provided to an algorithm is not always large enough: for example, measuring enzyme activity typically requires chromatography-based assays whose throughput is about 100 samples under a standard laboratory resource. Thus, we are challenged to build a sequence optimization algorithm that performs well for practical use cases even when the training data is scarce.

Machine Learning for Structural Biology Workshop, NeurIPS 2022.

To this end, here we propose a fully unsupervised method named EvoOpt that we hope serves as a baseline for the problem. EvoOpt is a pipelined method that leverages evolutionary information contained in multiple sequence alignments (MSAs) by exploiting the generative power of MSA Transformer [7], a protein language model (PLM) that takes an MSA as input, and the ability of a PLM to predict a protein's fitness without training data (i.e., zero-shot prediction [8]). The present work is partially motivated by the recent findings on the generative nature of PLMs that they can potentially generate a wild-type like [9] or even far better variants [10] because they can model the fitness landscapes well [11]. Our extensive evaluation of nine supervised methods over three large-scale protein engineering experiments demonstrates that our unsupervised method outperforms or is on par with the supervised methods even when a relatively rich amount of training data is available.

In addition, to clarify the effect of directly setting an evolutionary context with an MSA on generating sequences, we also examined the optimization performances of our pipeline when replacing MSA Transformer with ESM-1b [12] or ESM2 [13] that takes a single sequence instead of an MSA. The key observation here is that the MSA Transformer's optimization performance in the pipeline is equivalent to or superior to the single-sequence PLMs, even though they have up to 150-fold more model parameters. The overall results highlight the effectiveness of using an MSA to provide a sequence generation or optimization algorithm with explicit evolutionary information.

2 Methods

2.1 EvoOpt pipeline

To establish an extremely low-N baseline, we propose a fully unsupervised, three-stage pipeline for protein sequence optimization (Figure 1). In the first stage, we perform a homology search to build an MSA, in which we query an optimization target sequence against a large-scale protein database such as Uniclust [14] via HHblits [15]. To refine the obtained MSA, we filtered out the sequences with more than 30% gaps in the mutated region of the target protein. Furthermore, because too many mutations can lead to high-cost sequence synthesis in lab experiments [16], we selected 128 sequences from the filtered MSA, greedily minimizing the Hamming distances between them starting from the target protein. The preprocessing is also beneficial as it enables faster inference. Then, in the second stage, we generate multiple sequences using MSA Transformer as the generator. Here, we first corrupt the original MSA by masking randomly selected amino acids and then perform maximum likelihood estimation to restore the corrupted tokens based on the inferred logits by MSA Transformer, and again we mask the restored MSA to repeat the cycle for specified times. Through its column-wise attention mechanism, MSA Transformer can learn the evolutionary dependence between inter-sequence amino acids on diverse protein families via masked language modeling [17]. Thus, we expect the output sequence in this stage to have higher protein-likeliness, hopefully leading to better fitness. We note that the output sequences are those restored from the target protein, not from the homologs in the input MSA: the homologs are just used as an "evolutionary context" in our method. Finally, in the third stage, we rank the generated sequences by zero-shot fitness prediction using ESM-1b and output the top-ranked proteins as our proposal. In short, EvoOpt generates evolutionarily preferable sequences without needing supervised learning or even fine-tuning.

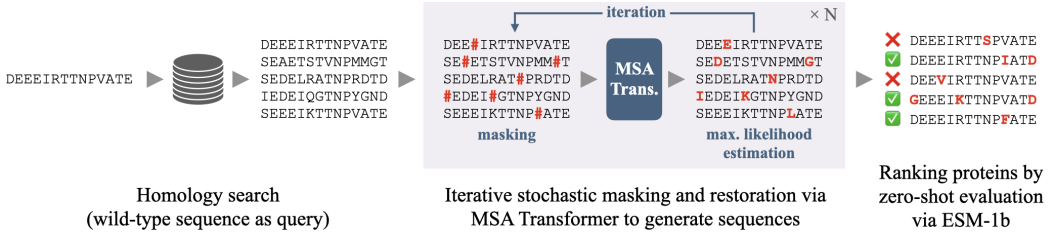

Figure 1: Overall structure of the EvoOpt pipeline.

Table 1: Summary of the protein engineering tasks. "Distance" is the Levenshtein distance between a variant sequence and the corresponding wild-type sequence.

Protein	Wild-type length	#Variants	Avg. distance to wild type	Max. distance to wild type
GFP [20]	237	56,086	4.64	16
AAV [21]	735	284,009	12.3	39
IGPD [22]	220	496,137	6.96	28

2.2 Evaluation setting

For our benchmarking, we sourced the protein engineering data obtained by wet-lab experiments from the well-curated datasets FLIP [18] and ProteinGym [19]. To allow the algorithms to explore board regions in fitness landscapes, we extracted the large-scale experiments, including more than 10,000 samples with more than double mutations. By doing so, we were left with three tasks (Table 1): green fluorescent protein (GFP) [20], adeno-associated virus (AAV) capsid [21], and imidazoleglycerol-phosphate dehydratase (IGPD) [22]. IGPD is suitable for our benchmarking because high-throughput experiments are possible for this enzyme.

To evaluate the fitness for every possible sequence proposed by an optimization algorithm, we trained separate 2-layer MLP predictors on top of the frozen ESM-1b encoder for AAV and IGPD tasks. Note that these predictors are just used as the surrogates for the wet-lab experiments in our benchmarking, not a part of an optimization algorithm: using a trained predictor as such has been done in a previous benchmarking [23], and we followed the setup as well. We split a whole mutational dataset into train/valid/test with the ratio 8:1:1 and only used train/valid sets for training. For the GFP task, we used the pre-trained model provided by the design-bench suite [23]. The prediction accuracies (Spearman's correlation coefficient) of each model on the respective holdout set were 0.84 for GFP, 0.91 for AAV, and 0.80 for IGPD.

We compared EvoOpt with nine supervised methods with diverse approaches (Table 2). We used the model implementations and the evaluation suite provided in design-baselines [25]. All the hyperparameters were the default ones. In this work, we consider the single-round optimization problem: given a training dataset, we train a model (in cases other than EvoOpt) and output a set of sequences. For each task, a single supervised algorithm is given training data with the ground-truth fitness ranging from 50 to 60 percentile in the whole dataset, consistent with the setting in a recent extensive benchmarking [23]. It is worth mentioning that this also emulates a realistic protein engineering situation where we only have rather low-performant molecules and are inaccessible to the higher-fitness variants *a priori*. To examine both low-N and high-N scenarios, we gradually varied the number of training data provided with an algorithm from 32 to 1,024.

Table 2: Summary of the supervised sequence optimization algorithms.

Algorithm	Approach
Autofocused-CbAS [4]	Generative model
CbAS [3]	Generative model
BO-qEI [2]	Bayesian optimization
CMA-ES [5]	Evolutionary strategy
Gradient-ascent [23]	Gradient-based
Gradient-ascent-min-ensemble [23]	Gradient-based
Gradient-ascent-mean-ensemble [23]	Gradient-based
MINS [24]	Inverse mapping
REINFORCE [6]	Reinforcement learning

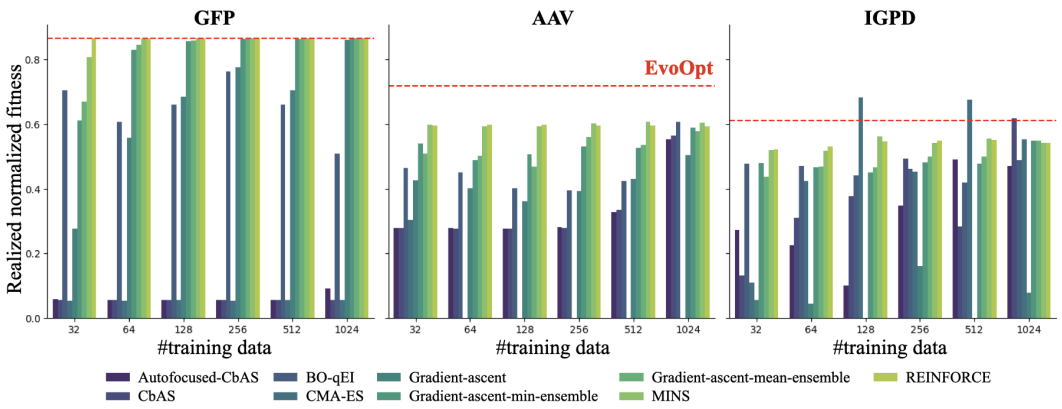

Figure 2: Evaluation results with the varying number of training samples. The red horizontal lines denote EvoOpt's results. The fitness values in each task were normalized to compare the performances across tasks. We report the average normalized fitness values in 16 trials with different random seeds for each setting. CMA-ES results in the AAV task for #training data > 32 are missing because the training did not finish within a computation limit (24 hours). The resultant negative fitness of gradient-ascent method was clipped to 0.0.

3 Results

3.1 Effectiveness of MSA-guided unsupervised sequence optimization

Despite being unsupervised, our method outperformed or was on par with all the other supervised methods (Figure 2). The relatively low performances of the supervised methods suggest that the algorithms searched only the neighbor region around the low-performing training data, which is consistent with the recent finding that the sequence search space can be regulated by the training data compositions [26]. Also, with scarce data, it would be difficult for a supervised algorithm to learn the general properties of proteins required to guess how a generated sequence is of high fitness. In contrast, thanks to the pre-training across diverse families, MSA Transformer could leverage the general knowledge of proteins, leading to the generation of highly probable sequences found in an evolutionarily conditioned area by the input MSA in the search space.

3.2 Comparison between single sequence-input PLMs and MSA Transformer as generators

A recent study [27] revealed that MSA Transformer's representation power is in general similar to ESM-1b as measured in zero-shot fitness prediction accuracy. Besides, a PLM's performance is demonstrated to improve with an increasing number of model parameters [13]. Then, how about the generative power of PLMs? In this respect, we examined the optimization performance of our pipeline when using a single sequence-input PLM as the generator, where we considered ESM-1b and ESM2 instead of MSA Transformer. For this purpose, we did not perform homology search and just input a wild-type sequence to these models. ESM2 is the largest Transformer-based PLM ever with 15B parameters, which is 150 times larger than MSA Transformer.

As summarized in Table 3, we found that the performances of MSA Transformer compared with the other two PLMs were significantly better or comparable in two (GFP and AAV) of three engineering tasks and slightly worse in the IGPD task. To our surprise, ESM2 performed comparably to MSA Transformer in the AAV task but worked poorly in the other two tasks.

4 Discussion

In the present work, we proposed a fully unsupervised, pipelined algorithm for sequence optimization to tackle the data scarcity problem we usually encounter in practical protein engineering. The extensive benchmarking demonstrated the advantage of using an MSA to guide the algorithm toward promising candidates in the search space, directly exploiting evolutionary information. One of the

Table 3: Comparison between MSA Transformer, ESM-1b, and ESM2 as generators in the proposed pipeline. "Performance" indicates the normalized realized fitness values in each task. We report the average ± standard deviations for eight trials with different random seeds. The bold letters indicate the best average performances.

Generator	#Params	Input	Performance		
			GFP	AAV	IGPD
MSA Transformer	100M	MSA	**0.865±0.000**	**0.718±0.000**	0.610±0.019
ESM-1b	650M	Single sequence	0.330±0.279	0.697±0.014	**0.697±0.051**
ESM2	15B	Single sequence	0.445±0.325	0.712±0.008	0.325±0.145

good use cases of our simple but effective method would be to automatically determine the initial amino acid positions to be mutated for directed evolution, which is often empirically done by relying on the biological knowledge of a target protein. We believe EvoOpt would give us far more reasonable guesses than completely random choices, as done in the widely used error-prone PCR method. As shown in Figure 2, EvoOpt could not reach the perfect performance (y=1.0) in any of the three tasks, which means there exist some variants that cannot be realized by relying only on the natural protein information. Thus, it would be interesting to extend the current work to the active learning setting in which we use the retrieved samples to train predictive and generative models.

References

[1] Arnold FH. Directed Evolution: Bringing New Chemistry to Life. Angew Chem Int Ed Engl. 2018 Apr 9;57(16):4143-4148.

[2] Wilson JT, et. al. The reparameterization trick for acquisition functions. CoRR, abs/1712.00424, 2017.

[3] Brookes DH, Park H, Listgarten J. Conditioning by adaptive sampling for robust design. arXiv preprint arXiv:1901.10060, 2019.

[4] Fannjiang C, Listgarten J. Autofocused oracles for model-based design. arXiv preprint arXiv:2006.08052, 2020.

[5] Hansen N. The CMA evolution strategy: A comparing review. In Lozano, J. A., Larranaga, P., Inza, I., and Bengoetxea, E. (eds.), Towards a New Evolutionary Computation - Advances in the Estimation of Distribution Algorithms, volume 192 of Studies in Fuzziness and Soft Computing, pp. 75–102. Springer, 2006.

[6] Williams RJ. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Mach. Learn., 8:229–256, 1992.

[7] Rao RM, et. al. MSA Transformer. Proceedings of the 38th International Conference on Machine Learning. PMLR, 2021, pp. 8844–8856.

[8] Meier J, et. al. Language models enable zero-shot prediction of the effects of mutations on protein function. In Advances in neural information processing systems, pp. 29287-29303, 2021.

[9] Sgarbossa D, Lupo U, Bitbol AF. Generative power of a protein language model trained on multiple sequence alignments. bioRxiv preprint. doi:10.1101/2022.04.14.488405, 2022.

[10] Hie B, et. al. Efficient evolution of human antibodies from general protein language models and sequence information alone. bioRxiv preprint. doi: 10.1101/2022.04.10.487811, 2022.

[11] Hawkins-H A, Jones DT, Paige B. MSA-Conditioned Generative Protein Language Models for Fitness Landscape Modelling and Design. In Machine Learning for Structural Biology Workshop, NeurIPS 2021.

[12] Rives A., et. al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences of the United States of America 118.15 (2021), e2016239118.

[13] Lin Z, et. al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv preprint. doi: 10.1101/2022.07.20.500902, 2022.

[14] Mirdita M, et. al. Uniclust databases of clustered and deeply annotated protein sequences and alignments, Nucleic Acids Res. 2016.

[15] Remmert M., et. al. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nature Methods 9.2 (2012), pp. 173–175.

[16] Fowler D, Stephany J, Fields S. Measuring the activity of protein variants on a large scale using deep mutational scanning. Nat Protoc 9, 2267–2284 (2014).

[17] Devlin J, et. al. 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), pp. 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[18] Dallago C, et. al. FLIP: Benchmark tasks in fitness landscape inference for proteins. In NeurIPS 2021 Track Datasets and Benchmarks Round2.

[19] Notin P, et. al. Tranception: Protein Fitness Prediction with Autoregressive Transformers and Inference-time Retrieval. Proceedings of the 39th International Conference on Machine Learning, PMLR 162:16990-17017, 2022.

[20] Sarkisyan KS, et. al. Local fitness landscape of the green fluorescent protein. Nature 533.7603 (2016), pp. 397–401.

[21] Bryant DH, et. al. Deep diversification of an AAV capsid protein by machine learning. Nat Biotechnol 39, 691–696 (2021).

[22] Pokusaeva VO, et. al. An experimental assay of the interactions of amino acids from orthologous sequences shaping a complex fitness landscape. PLOS Genetics 15.4 (2019), e1008079.

[23] Trabucco B, et. al. Design-Bench: Benchmarks for Data-Driven Offline Model-Based Optimization. Proceedings of the 39th International Conference on Machine Learning, PMLR 162:21658-21676, 2022.

[24] Kumar A, Levine S. Model Inversion Networks for Model-Based Optimization. In Advances in neural information processing systems, pp. 5126-5137, 2020.

[25] Trabucco B, et. al. Conservative Objective Models for Effective Offline Model-Based Optimization. Proceedings of the 38th International Conference on Machine Learning, PMLR 139:10358-10368, 2021.

[26] Saito Y, et. al. Machine-Learning-Guided Library Design Cycle for Directed Evolution of Enzymes: The Effects of Training Data Composition on Sequence Space Exploration. ACS Catal. 2021, 11, 23, 14615–14624.

[27] Hu M, et. al. Exploring evolution-based & -free protein language models as protein function predictors. arXiv preprint arXiv:2206.06583, 2022.