Evolution-Inspired Loss Functions for Protein Representation Learning

Anonymous authors
Paper under double-blind review

Abstract

AI-based frameworks for protein engineering use self-supervised learning (SSL) to obtain representations for downstream mutation effect predictions. The most common training objective for these methods is wildtype accuracy: given a sequence or structure where a wildtype residue has been masked, predict the missing amino acid. Wildtype accuracy, however, does not align with the primary goal of protein engineering, which is to suggest a *mutation* rather than to identify what already appears in nature. Here we present Evolutionary Ranking (EvoRank), a training objective that incorporates evolutionary information derived from multiple sequence alignments (MSAs) to learn more diverse protein representations. EvoRank corresponds to ranking amino-acid likelihoods in the probability distribution induced by an MSA. This objective forces models to learn the underlying evolutionary dynamics of a protein. Across a variety of phenotypes and datasets, we demonstrate that EvoRank leads to dramatic improvements in zero-shot performance and can compete with models fine-tuned on experimental data.

1 Introduction

The success of AlphaFold (Jumper et al., 2021) has inspired a new era of deep-learning frameworks for protein design and engineering. Large protein language models (e.g., ESM (Rives et al., 2019a; Meier et al., 2021a)), structure generative models (e.g., RFDiffusion (Watson et al., 2023), NeuralPLexer (Qiao et al., 2023)) and structure-based self-supervised models (Sumida et al., 2024; Diaz et al., 2023; Lu et al., 2022) can accelerate the development of biotechnology with applications in identifying disease-causing variants (Braunisch et al., 2021; Kouba et al., 2023; Scherer et al., 2021) and enzyme engineering for biomanufacturing. Due to the prohibitive cost of generating experimental data, self-supervised learning (SSL) has become the primary technique used by the community to generate protein representations (*e.g.*, Riesselman et al., 2018a; Rives et al., 2019a; Meier et al., 2021a; Dauparas et al., 2022; Bepler & Berger, 2019; d'Oelsnitz et al., 2023; Notin et al., 2022; Hsu et al., 2022). These methods rely on masking followed by predicting the wildtype (WT) amino acids in extant proteins as the SSL training objective. For example, given as input a protein sequence and masked residue, models can be trained to predict what amino acid has been masked. The loss in WT-mask SSL is typically defined to be the cross entropy between a model's prediction and the one-hot encoding of the masked wildtype amino acid(s). This *wildtype accuracy* metric, also known as recovery ratio, is then reported as a proxy for the quality of the learned representations.

For machine learning-guided protein engineering (MLPE), practitioners desire models that suggest mutations to a protein *away* from wildtype, as opposed to models that merely predict wildtype. To address this disparity, several approaches have been proposed. Structure-based methods often adjust the temperature of the logits (Ingraham et al., 2019; Dauparas et al., 2022; Sumida et al., 2024) to bias away from wildtype. Sequence-based methods require large protein databases and incorporate MSAs as additional inputs (Rao et al., 2021a; Notin et al., 2022).

A more serious and often overlooked issue, however, is that improved wildtype accuracy may *not* correlate with downstream mutation effect performance. We sharply illustrate this phenomenon in Table 1 where we train a structure-based model to increasing levels of wildtype accuracy and show that its downstream performance on thermodynamic stability prediction begins to *decrease* beyond a wildtype accuracy threshold.

Additionally, current frameworks using either sequence or structure modalities can achieve greater than 90% wildtype accuracy (*e.g.*, Rives et al., 2019a; Meier et al., 2021a; Lin et al., 2023; Diaz et al., 2023), forcing the practitioner to make ad-hoc decisions about the optimal choice of wildtype accuracy for downstream applications. Developing a self-supervised learning objective that acts as an effective proxy for mutation effect prediction remains a critical open problem.

Loss	Metrics	MutComputeXGT						ESM2	ProteinMPNN
WT-mask	WT Acc	17%	29%	43%	68%	79%	92%	94%	48%
	Pearson	0.14	0.21	0.30	0.34	0.30	0.24	0.25	0.31
EvoRank	EvoRank	0.24	0.21	0.17	0.15	0.13	0.12	-	-
	Pearson	0.30	0.36	0.45	0.48	0.51	0.50	0.25	0.31

Table 1: We train the MutComputeXGT architecture with SSL for different iterations and evaluate the WT accuracy and zero-shot folding free energy change Pearson correlation on held out validation FireProtDB (Stourac et al., 2021). Note that improvements in the EvoRank objective are more consistent with Pearson correlation.

Our main contribution is a new self-supervised training objective, EvoRank, that incorporates evolutionary information from multiple sequence alignments (MSAs) in order to address the limitations of WT-mask SSL. To emulate the mutation setting, EvoRank uses a ranking objective to force a model to learn fine-grained information about the MSA-induced distribution of amino acids at a particular location. We show that after initializing a model's wildtype predictions with an approximate MSA distribution, EvoRank results in dramatic empirical improvements for zero-shot performance across a variety of commonly studied benchmarks. Additionally, since MSAs are incorporated into the loss, they are only needed during training and not inference time, in contrast to models that require an MSA as an additional input Notin et al. (2022). Further, empirical improvements on the EvoRank loss are correlated with improvements in downstream mutation effect prediction (see Table 1), leading to a reliable benchmark for protein representation learning.

2 Methods

This section introduces the main method. We start with introducing the widetype (WT) based mask prediction for self-supervised representation on proteins. We then propose our two novel techniques: 1) a *MSA-based soft label* to introduce evolution information into the learning; and 2) a *EvoRank* loss that allows us to extra evolution information more efficiently and robustly with a learning-to-rank idea.

Self-Supervised Learning via WT-mask prediction We are given a protein set $\mathcal{P} = \{P\}$, where the representation $P = (\mathcal{A}, \mathcal{V})$ of each protein consists of both its amino acid sequence $\mathcal{A}$ and atoms information $\mathcal{V}$. The sequence $\mathcal{A} = (a_j, \cdots, a_m)$ contains m amino acids, where a_j is the one-hot representation of the 20 amino acid types. The $\mathcal{V} = \{v_j\}_{j=1}^{n}$ represents all the atoms contained in the protein, where v_j contains the information of the j-th atom, including its 3D coordinates, atom type, partial charge and solvent accessible surface.

WT-Mask Prediction In the WT-mask prediction task (Torng & Altman, 2017), we mask an amino acid a_j, and learn a neural network to predict a_j back based on the microenvironment surrounding a_j. The network can then provide useful representation of the protein for downstream tasks. Specifically, Denote by $\mathrm{C}_\alpha(a_j)$ be the α-carbon atom of amino acid a_j, and $\mathrm{Atom}(a_j)$ all the atoms contained in amino acid a_j. We take the microenvironment of a_j to be the atoms within $20\mathring{A}$ distance with $\mathrm{C}_\alpha(a_j)$, excluding all atoms in $\mathrm{Atom}(a_j)$, that is,

$$\mathcal{V}_j^{\mathrm{mask}} = \{v\colon v \in \mathcal{V} \setminus \mathrm{Atom}(a_j), \quad \mathrm{Dist}(C_\alpha^j, v) \leq 20\mathring{A}\},$$

We train a neural network $y = f(x)$, that takes a micro-environment $x = \mathcal{V}_j^{\mathrm{mask}}$ as input and output the logits on the 20 amino acid types. We want to train to model to make $f(\mathcal{V}_j^{\mathrm{mask}}) \approx a_j$:

$$\min_f \sum_{P \in \mathcal{P}} \sum_j \mathrm{D}(a_j,\ \mathrm{Softmax}(f(\mathcal{V}_j^{\mathrm{mask}}))),$$

where D denotes the loss function. A typical choice is the KL divergence, *a.k.a.*, cross entropy loss.

Evolution Information via MSA-based Soft Labels As described in the introduction, we desire a self-supervised learning procedure that (1) discourages low-entropy distributions skewed towards wildtype and (2) incorporates meaningful evolutionary and biochemistry from the input protein structure. Since Multiple sequence alignment (MSA) provides a powerful tool for capturing evolutionary relations between sequences, we propose to incoporate MSA information into the self-supervised learning with an *MSA soft-label loss* (equation 2), where the wildtype one-hot encoded label is replaced with a distribution from a protein's MSA.

Formally, instead of training network f to predict the one-hot vector of the wildtype amino acid, we predict the following soft label based on the following pdf derived from the MSA of the protein:

$$p_j^{\mathrm{MSA}}(\ell) \propto \sum_{P' \in \mathrm{MSA}(P)} \delta(\ell = \mathrm{Amino}(P', j)), \tag{1}$$

where ℓ is one of the 20 amino acids, δ is the delta function, $\mathrm{MSA}(P)$ denotes the set of sequences that are best aligned with P via multiple sequence alignment on UniRef50 (Consortium, 2015). and $\mathrm{Amino}(P', j)$ denotes the amino acid type of protein P' at location j. We refer to this distribution as the empirical amino acid distribution.

We define the MSA soft-label training loss as follows:

$$\min_f \sum_{P \in \mathcal{P}} \sum_j \mathrm{D}(p_j^{\mathrm{MSA}},\ \mathrm{Softmax}(f(\mathcal{V}_j^{\mathrm{mask}}))). \tag{2}$$

Although KL divergence has been the canonical choice, it is known to suffer from mode collapse. We experimented with taking $\mathrm{D}(\cdot;\cdot)$ within a richer family of α-divergences. By applying different α values, we can adjust the sensitivity to multimodal distributions present in MSAs and find a better trade-off between over/under estimates of the top ranked amino acid (which is often wildtype). When we apply reverse KL divergence or $\alpha = 0.5$ divergence (Table 2), we observe marginally improved rank order but overall lower coefficients for the top-5 amino acids. This suggests the need for designing better loss functions.

Divergence	Label	Top-5	Top-10	20
KL Div	WT	0.54	0.38	0.28
KL Div	MSA	0.60	0.52	0.34
Reverse KL Div	MSA	0.54	0.56	0.40
Alpha Div ($\alpha = 0.5$)	MSA	0.57	0.53	0.40

Table 2: Spearman correlation coefficient for amino acids at the same local chemical environment in the test dataset for the mask prediction task. Here, 'Top-5' indicates the amino acids with the top-5 probability score based on the empirical amino acid distribution.

EvoRank: A New Rank-based Learning Objective To further improve the performance of the self-supervised model, we reformulate the training task to correspond more directly to mutation prediction and train with a ranking loss. Rather than predicting the wildtype amino acid type a_j or soft label p_j^{MSA}, we set up a model to take as input a pair of "positive" and "negative" amino acid types a^+ and a^-, and output their relative likelihood in the empirical amino acid distribution. More precisely, we define a rank label of a_j w.r.t. (a^+, a^-) as the following

$$r_i(a^+, a^-) = \frac{p_j^{\mathrm{MSA}}(a^+)}{p_j^{\mathrm{MSA}}(a^+) + p_j^{\mathrm{MSA}}(a^-)} - \frac{1}{2}, \tag{3}$$

where $p_j^{\mathrm{MSA}}(a)$ denotes the probability assigned on a according to p_j^{MSA}, and $\frac{1}{2}$ to ensure neutral predictions are made when $p_j^{\mathrm{MSA}}(a^+) = p_j^{\mathrm{MSA}}(a^-)$. The rank label represents the relative likelihood between with respect to two amino acids to be evolutionarily observed at a particular microenvironment, as demonstrated in Figure 1.

We train a model $f(\mathcal{V}_j^{\mathrm{mask}},\ a^+, a^-)$ to predict the rank label $r_j(a^+, a^-)$ via the following loss:

$$\min_f \sum_{a^+, a^-} \sum_{P \in \mathcal{P}} \sum_j \mathrm{D}(r_j(a^+, a^-),\ f(\mathcal{V}_j^{\mathrm{mask}}, a^+, a^-)), \tag{4}$$

where the a^+, a^- are summed on all the amino acid types and $\mathrm{D}(x, y) = ||x - y||$. We refer to the loss in equation 4 as the EvoRank loss or EvoRank training objective.

In practice, we first initialize the parameters by training using the MSA soft-label loss (equation equation 2) and then apply the EvoRank loss to further improve performance. Similar ideas are used in the recommendation system literature (*e.g.*, Cao et al., 2007; Aggarwal et al., 2016; Liu et al., 2009), where parameters are initialized from a model trained with a standard prediction loss and then trained further using a ranking loss.

-	T2837		S669		S-Sym		Myolobin		FireProtDB		Gβ1		T2837 Reverse	
# Mutations	2837		669		342		134		1764		935		2837	
Metric	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$	$\rho\uparrow$	AUC $\uparrow$
RaSP* (Blaabjerg et al., 2023)	0.58	0.61	0.39	0.69	0.64	0.73	**0.68**	0.75	0.56	0.71	**0.72**	0.66	0.23	0.59
ThermoMPNN* (Dieckhaus et al., 2023)	0.55	0.78	0.39	0.68	0.66	0.82	0.58	0.77	0.57	0.75	0.65	0.78	0.43	0.71
Prostata-IFML (Diaz et al., 2023)	0.53	0.75	0.49	0.76	0.55	0.75	0.54	0.67	-	-	0.66	**0.82**	0.52	0.75
Stability Oracle (Diaz et al., 2023)	**0.59**	**0.81**	**0.52**	**0.74**	**0.72**	**0.87**	**0.68**	**0.81**	**0.61**	**0.79**	**0.71**	**0.82**	**0.59**	**0.81**
ESM2* (Lin et al., 2023)	0.28	0.60	0.04	0.50	0.26	0.56	0.15	0.57	0.25	0.57	0.25	0.63	0.28	0.60
ProteinMPNN* (Dauparas et al., 2022)	0.36	0.70	0.25	0.59	0.32	0.64	0.35	0.66	0.31	0.70	0.35	0.67	0.36	0.70
MutComputeXGT (Diaz et al., 2023)	0.34	0.68	0.27	0.57	0.38	0.72	0.37	0.72	0.30	0.69	0.34	0.66	0.34	0.68
MutComputeXGT w/ MSA soft-label (Ours)	0.37	0.70	0.30	0.59	0.48	0.75	0.45	0.75	0.36	0.71	0.41	0.69	0.37	0.70
MutRank (Ours)	**0.51**	**0.78**	**0.40**	**0.67**	**0.62**	**0.84**	**0.68**	**0.84**	**0.51**	**0.77**	**0.62**	**0.77**	**0.51**	**0.78**
SSL Improvement $\uparrow$	42%	11%	48%	14%	63%	17%	83%	17%	65%	10%	77%	15%	42%	11%
Supervised Fine-Tuning Gap $\downarrow$	14%	4%	23%	9%	14%	3%	0%	-4%	16%	3%	13%	3%	14%	4%

Table 3: Zero-shot results of multiple methods on multiple thermodynamic stability ($\Delta\Delta G$) datasets. ρ equals the Pearson correlation coefficient and AUC is the area under the receiver operating characteristic. The first block reports the performance of frameworks fine-tuned using experimental $\Delta\Delta G$ datasets. The second block reports the performance of self-supervised models common in the literature. The third block reports the performance of two models trained in this work. The first is trained only using the MSA soft-label loss and the second is MutRank, trained with both the MSA soft-label loss and the EvoRank loss (see Section B.2). 'SSL Improvement' compares MutRank with respect to the best zero-shot model in the second block. 'Supervised Fine-Tuning Gap' compares MutRank with respect to the best supervised $\Delta\Delta G$ model in the first block. * denotes that we compute the metrics using the official checkpoint.

Dataset	Phenotype	# Mut	MutComputeXGT			MutRank			ESM2			Stability Oracle		
			Pearson	Spearman	AUC	Pearson	Spearman	AUC	Pearson	Spearman	AUC	Pearson	Spearman	AUC
levoglucosan kinase	ΔSolubility	7195	0.26	0.30	0.61	0.29	**0.34**	**0.64**	0.27	0.32	0.62	**0.32**	**0.34**	0.63
TEM1-β-Lactamase	ΔSolubility	4345	0.16	0.21	0.60	**0.22**	**0.26**	**0.64**	0.08	0.18	0.61	0.10	0.16	0.60
AcrIIA4	Acitivity	1653	0.36	0.34	0.65	**0.59**	**0.53**	**0.75**	0.06	0.06	0.56	0.48	0.40	0.69
Amidase	Activity	6227	0.38	0.39	0.66	**0.64**	**0.64**	**0.83**	0.56	0.56	0.78	0.48	0.46	0.75
Deiminase	Activity	5689	0.26	0.26	0.63	**0.41**	**0.42**	**0.73**	0.38	0.39	0.70	0.24	0.24	0.63
SKEMPI-V2	Protein-Protein $\Delta\Delta G_{\text{bind}}$	4102	0.28	0.26	0.62	**0.42**	**0.42**	**0.69**	0.23	0.19	0.57	0.39	0.39	0.67
S487	Protein-Protein $\Delta\Delta G_{\text{bind}}$	487	0.24	0.25	0.58	**0.38**	**0.38**	0.67	0.01	0.01	0.48	**0.38**	**0.38**	**0.70**
PlatinumDB	Protein-Ligand $\Delta\Delta G_{\text{bind}}$	925	0.05	0.01	0.48	**0.28**	**0.28**	**0.64**	0.03	0.06	0.51	0.26	0.26	**0.64**
ABBind	Antibody-Antigen $\Delta\Delta G_{\text{bind}}$	309	0.36	0.42	0.73	**0.41**	**0.46**	**0.74**	-0.07	-0.05	0.60	0.38	0.42	0.72

Table 4: We show that MutRank improves zero-shot performance for solubility and binding free energy phenotypes. In comparison with both sequence and structure-based models trained using wildtype accuracy, training a structure-based model with EvoRank leads to greatly improved zero-shot performance. Stability Oracle is initialized with MutComputeXGT and fine-tune for ΔΔG.

3 Experimental Results

We retrained a SOTA structure model (Diaz et al., 2023) using both the MSA soft-label loss and the EvoRank loss as described in Section B.2. We name the MutComputXGT structure model trained with EvoRank loss as MutRank. We refer to the resulting model as MutRank.

Zero-shot thermodynamic stability evaluations Table 3 reports the zero-shot Pearson correlation coefficient (ρ) and area under the ROC curve (AUC) performance of various machine learning frameworks across multiple ΔΔG datasets: T2837 (Diaz et al., 2023), S-Sym (Li et al., 2020), S669 (Pancotti et al., 2022), FireProtDB (Stourac et al., 2021), Gβ1 (Nisthal et al., 2019), and Myoglobin (Li et al., 2020). Our results validate the impact prioritizing rank order during self-supervised training has on zero-shot $\Delta\Delta G$ predictions. First, our results on the MSA-based soft labels with α divergence already outperforms literature self-supervised baselines for both Pearson correlation and AUC. Then, by reformulating the training objective with EvoRank we improve over the previous best literature zero-shot model by a significant margin–on average we improve the Pearson correlation and AUC across the six datasets by $\sim$64% and $\sim$14%, respectively. Direct comparison with its WT-masked predecessor, MutComputeXGT, MutRank results in a 66% and 16% improvement in Pearson correlation and AUC, respectively. Notably, compared to the well-known self-supervised methods ESM2 and ProteinMPNN, MutRank achieves on average a Pearson correlation improvements of $\sim$288% and $\sim$72% across the six $\Delta\Delta G$ datasets, respectively. These results demonstrate the effectiveness of MutRank representations for $\Delta\Delta G$.

Next, we compared to the structure-based frameworks RaSP (Blaabjerg et al., 2023) and ThermoMPNN Dieckhaus et al. (2023)) and the sequence-based framework Prostata-IFML (Diaz et al., 2023). Although these frameworks are explicitly fine-tuned on large scale cDNA $\Delta\Delta G$ dataset, our zero-shot results are competitive. Compared to the SOTA-supervised framework, Stability Oracle, our zero-shot Pearson correlation and AUC are only $\sim$13% and $\sim$3% lower on average across the six datasets. Overall, our results demonstrate how the EvoRank loss significantly narrows the gap between supervised fine-tuned framework and zero-shot representation for $\Delta\Delta G$.

Zero-shot evaluation on multiple phenotypes To further characterize the generalization capability of MutRank representations, we evaluate performance on binding free energy change datasets and four DMS datasets: two for solubility and two for activity (Table 4). Unlike folding stability, which has seen significant increases in available public data (Tsuboyama et al., 2023), binding free energy change datasets are scarce, filled with mutation type and label biases, and suffer from noisy labels. These challenges makes developing supervised frameworks challenging for these phenotypes and underlines the importance of zero-shot self-supervised models. For the binding free energy datasets, we use the protein-protein interface binding $\Delta\Delta G$ datasets SKEMPIv2 Jankauskaitė et al. (2019), AB-Bind (Sirin et al., 2016), S487 (Geng et al., 2019) and the protein-ligand interface binding $\Delta\Delta G$ dataset PlatinumDB (Pires et al., 2015). For the solubility and activity datasets, we used Deep Mutational Scanning (DMS) datasets, which leverage a high throughput screen or next-generation sequencing as a proxy for function. For solubility, we use the DMS datasets for for levoglucosan kinase (uniprot id:B3VI55) and TEM1-β-Lactamase (uniprot id: P62593) from Klesmith et al. (2017). For activity evaluation, we use the DMS datasets for the aliphatic hydrolase (uniprot id: P11436), the Anti-CRISPR protein AcrIIA4 (uniprot id: A0A247D711), and Porphobilinogen deaminase (uniprot id: P08397). We compare against two WT-mask SSL frameworks, MutComputeXGT and ESM2, and one supervised fine-tune framework, Stability Oracle. Comparison between just the literature methods on the binding $\Delta\Delta G$ datasets demonstrate that ESM2 did the worst and Stability Oracle did the best across all metrics (Pearson and Spearman correlation and AUC). These results are expected since binding free energy (interactions between proteins) is fundamentally related to folding free energy (interactions within a protein). ESM2 is unable to see the binding partner (protein or ligand) and must rely purely on the single sequence representation.

Remarkably, MutRank outperforms MutcomputeXGT across all datasets for all metrics. This demonstrates that the EvoRank loss improve zero-shot generalization across all phenotypes compared to its WT-masked predecessor. Additionally, MutRank outperforms ESM2 on all datasets for all metrics even though it is a much smaller model trained on only ~23K proteins compared to UniRef50. Surprisingly, MutRank's zero-shot performance surpasses or ties Stability Oracle performance on nearly all metrics for binding $\Delta\Delta G$ datasets (except S487 AUC). Furthermore, it significantly outperforms Stability Oracle on the TEM1-β-Lactamase solubility dataset and the three activity datasets. Stability Oracle performance on the TEM1-β-Lactamase dataset is lower than it's pretrained representation, MutComputeXGT. This finding highlighting the superior phenotype generalization of EvoRank loss and demonstrating how supervised fine-tuning can improve the performance on one phenotype at the expense of others. Finally, we highlight MutRank's substantial improvement on the protein-ligand interface binding $\Delta\Delta G$ dataset, PlatinumDB: compared to MutComputeXGT: MutRank improves the Pearson correlation and AUC from 0.05 and 0.48 (indicating a random classifier) to 0.28 and 0.64. We conclude that for the activity, solubility and binding free energy phenotypes, MutRank representations significantly improves the zero-shot generalization over the WT-mask representations of MutComputeXGT. However, additional evaluations are needed to better understand its generalization across phenotypes for diverse proteins.

4 Conclusion

We propose EvoRank training objective aimed at improving the protein representations obtained from self-supervised learning for zero-shot mutation effect prediction tasks. EvoRank reformulates the learning task to better emulate a mutation by replacing the 20-class classification head with a regression head trained to learn the ranking of amino acids within the MSA distribution at a particular position. To evaluate EvoRank, we trained a structure-based graph transformer with the EvoRank loss and observe performance improvements in all downstream single point mutation effect prediction tasks compared to the WT-mask predecessor. When compared to the most renown sequence-based (ESM2) and structure-based (ProteinMPNN) frameworks, EvoRank demonstrates superior zero-shot performance across all evaluated benchmarks. From our results, we conclude that the EvoRank training objective produces protein representation with an enriched understanding of the complex mutational landscape of proteins.

References

Charu C Aggarwal et al. *Recommender systems*, volume 1. Springer, 2016.

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

Pierre Barrat-Charlaix, Matteo Figliuzzi, and Martin Weigt. Improving landscape inference by integrating heterogeneous data in the inverse ising problem. *Scientific Reports*, 6(1):37812, 2016.

Tristan Bepler and Bonnie Berger. Learning protein sequence embeddings using information from structure. In *International Conference on Learning Representations*, 2019.

Surojit Biswas, Grigory Khimulya, Ethan C Alley, Kevin M Esvelt, and George M Church. Low-n protein engineering with data-efficient deep learning. *Nature methods*, 18(4):389–396, 2021.

Lasse M Blaabjerg, Maher M Kassem, Lydia L Good, Nicolas Jonsson, Matteo Cagiada, Kristoffer E Johansson, Wouter Boomsma, Amelie Stein, and Kresten Lindorff-Larsen. Rapid protein stability prediction using deep learning representations. *eLife*, 12:e82593, 2023.

Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. Proteinbert: A universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110, 2022.

Matthias Christoph Braunisch, Korbinian Maria Riedhammer, Pierre-Maurice Herr, Sarah Draut, Roman Günthner, Matias Wagner, Marc Weidenbusch, Adrian Lungu, Bader Alhaddad, Lutz Renders, et al. Identification of disease-causing variants by comprehensive genetic testing with exome sequencing in adults with suspicion of hereditary fsgs. *European Journal of Human Genetics*, 29(2):262–270, 2021.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pp. 129–136, 2007.

Can Chen, Jingbo Zhou, Fan Wang, Xue Liu, and Dejing Dou. Structure-aware protein self-supervised learning. *Bioinformatics*, 39(4):btad189, 2023a.

Tianlong Chen, Chengyue Gong, Daniel Jesus Diaz, Xuxi Chen, Jordan Tyler Wells, qiang liu, Zhangyang Wang, Andrew Ellington, Alex Dimakis, and Adam Klivans. Hotprotein: A novel framework for protein thermostability prediction and editing. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=YDJRFWBMNby.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023.

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

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

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pp. 7480–7512. PMLR, 2023.

Daniel J Diaz, Chengyue Gong, Jeffrey Ouyang-Zhang, James M Loy, Jordan Wells, David Yang, Andrew D Ellington, Alex Dimakis, and Adam R Klivans. Stability oracle: A structure-based graph-transformer for identifying stabilizing mutations. *bioRxiv*, pp. 2023–05, 2023.

Henry Dieckhaus, Michael Brocidiacono, Nicholas Randolph, and Brian Kuhlman. Transfer learning to leverage larger datasets for improved prediction of protein stability changes. *bioRxiv*, 2023.

Simon d'Oelsnitz, Daniel J Diaz, Daniel J Acosta, Mason W Schechter, Matthew B Minus, James R Howard, Hannah Do, James Loy, Hal Alper, and Andrew D Ellington. Synthetic microbial sensing and biosynthesis of amaryllidaceae alkaloids. *bioRxiv*, pp. 2023–04, 2023.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rihawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing. *arXiv preprint arXiv:2007.06225*, 2020.

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

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with alphafold-multimer. *biorxiv*, pp. 2021–10, 2021.

Cunliang Geng, Anna Vangone, Gert E Folkers, Li C Xue, and Alexandre MJJ Bonvin. isee: Interface structure, evolution, and energy-based machine learning predictor of binding affinity changes upon mutations. *Proteins: Structure, Function, and Bioinformatics*, 87(2):110–119, 2019.

Vladimir Gligorijević, P Douglas Renfrew, Tomasz Kosciolek, Julia Koehler Leman, Daniel Berenberg, Tommi Vatanen, Chris Chandler, Bryn C Taylor, Ian M Fisk, Hera Vlamakis, et al. Structure-based protein function prediction using graph convolutional networks. *Nature communications*, 12(1):3168, 2021.

Chengyue Gong, Adam Klivans, Jordan Wells, James Loy, Alex Dimakis, Daniel Diaz, et al. Binding oracle: Fine-tuning from stability to binding free energy. In *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, 2023.

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.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. *bioRxiv*, 2022. doi: 10.1101/2022.04.10.487779. URL https://www.biorxiv.org/content/early/2022/04/10/2022.04.10.487779.

John Ingraham, Vikas Garg, Regina Barzilay, and Tommi Jaakkola. Generative Models for Graph-Based Protein Design. In H Wallach, H Larochelle, A Beygelzimer, F d'Alché-Buc, E Fox, and R Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/f3a4ff4839c56a5f460c88cce3666a2b-Paper.pdf.

Justina Jankauskaitė, Brian Jiménez-García, Justas Dapkūnas, Juan Fernández-Recio, and Iain H Moal. Skempi 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics*, 35(3):462–469, 2019.

David T Jones. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of molecular biology*, 292(2):195–202, 1999.

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.

Dhananjay Kimothi, Akshay Soni, Pravesh Biyani, and James M Hogan. Distributed representations for biological sequence analysis. *arXiv preprint arXiv:1608.05949*, 2016.

Justin R Klesmith, John-Paul Bacik, Emily E Wrenbeck, Ryszard Michalczyk, and Timothy A Whitehead. Trade-offs between enzyme fitness and solubility illuminated by deep mutational scanning. *Proceedings of the National Academy of Sciences*, 114(9):2265–2270, 2017.

Petr Kouba, Pavel Kohout, Faraneh Haddadi, Anton Bushuiev, Raman Samusevich, Jiri Sedlar, Jiri Damborsky, Tomas Pluskal, Josef Sivic, and Stanislav Mazurenko. Machine learning-guided protein engineering. *ACS catalysis*, 13(21):13863–13895, 2023.

Bian Li, Yucheng T Yang, John A Capra, and Mark B Gerstein. Predicting changes in protein thermodynamic stability upon point mutation with deep 3d convolutional neural networks. *PLoS computational biology*, 16(11):e1008291, 2020.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574.

Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Amy X Lu, Haoran Zhang, Marzyeh Ghassemi, and Alan Moses. Self-supervised contrastive learning of protein representations by mutual information maximization. *BioRxiv*, 2020.

Hongyuan Lu, Daniel J Diaz, Natalie J Czarnecki, Congzhi Zhu, Wantae Kim, Raghav Shroff, Daniel J Acosta, Bradley R Alexander, Hannah O Cole, Yan Zhang, et al. Machine learning-aided engineering of hydrolases for pet depolymerization. *Nature*, 604(7907):662–667, 2022.

Roland Lüthy, Ioannis Xenarios, and Philipp Bucher. Improving the sensitivity of the sequence profile method. *Protein Science*, 3(1):139–146, 1994.

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

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

Kisung Moon, Hyeon-Jin Im, and Sunyoung Kwon. 3d graph contrastive learning for molecular property prediction. *Bioinformatics*, 39(6):btad371, 2023.

Alex Nisthal, Connie Y Wang, Marie L Ary, and Stephen L Mayo. Protein stability engineering insights revealed by domain-wide comprehensive mutagenesis. *Proceedings of the National Academy of Sciences*, 116(33):16367–16377, 2019.

Pascal Notin, Mafalda Dias, Jonathan Frazer, Javier Marchena Hurtado, Aidan N Gomez, Debora Marks, and Yarin Gal. Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. In *International Conference on Machine Learning*, pp. 16990–17017. PMLR, 2022.

Inyup Paik, Phuoc HT Ngo, Raghav Shroff, Daniel J Diaz, Andre C Maranhao, David JF Walker, Sanchita Bhadra, and Andrew D Ellington. Improved bst dna polymerase variants derived via a machine learning approach. *Biochemistry*, 62(2):410–418, 2021.

Corrado Pancotti, Silvia Benevenuta, Giovanni Birolo, Virginia Alberini, Valeria Repetto, Tiziana Sanavia, Emidio Capriotti, and Piero Fariselli. Predicting protein stability changes upon single-point mutation: a thorough comparison of the available tools on a new dataset. *Briefings in Bioinformatics*, 23(2):bbab555, 2022.

Douglas EV Pires, Tom L Blundell, and David B Ascher. Platinum: a database of experimentally measured effects of mutations on structurally defined protein–ligand complexes. *Nucleic acids research*, 43(D1):D387–D391, 2015.

Zhuoran Qiao, Weili Nie, Arash Vahdat, Thomas F Miller III, and Animashree Anandkumar. State-specific protein-ligand complex structure prediction with a multi-scale deep generative model. *Preprint at arXiv https://doi. org/10.48550/arXiv*, 2209, 2023.

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International Conference on Machine Learning*, pp. 8844–8856. PMLR, 2021a.

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International Conference on Machine Learning*, pp. 8844–8856. PMLR, 2021b.

Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature methods*, 9(2):173–175, 2012.

Adam Riesselman, Jung-Eun Shin, Aaron Kollasch, Conor McMahon, Elana Simon, Chris Sander, Aashish Manglik, Andrew Kruse, and Debora Marks. Accelerating protein design using autoregressive generative models. *BioRxiv*, pp. 757252, 2019.

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, 2018a.

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, 2018b.

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. *PNAS*, 2019a. doi: 10.1101/622803. URL https://www.biorxiv.org/content/10.1101/622803v4.

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. *bioRxiv*, 2019b. doi: 10.1101/622803. URL https://www.biorxiv.org/content/10.1101/622803v4.

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

Marc Scherer, Sarel J Fleishman, Patrik R Jones, Thomas Dandekar, and Elena Bencurova. Computational enzyme engineering pipelines for optimized production of renewable chemicals. *Frontiers in bioengineering and biotechnology*, 9:673005, 2021.

Raghav Shroff, Austin W Cole, Daniel J Diaz, Barrett R Morrow, Isaac Donnell, Ankur Annapareddy, Jimmy Gollihar, Andrew D Ellington, and Ross Thyer. Discovery of novel gain-of-function mutations guided by structure-based deep learning. *ACS synthetic biology*, 9(11):2927–2935, 2020.

Richard W Shuai, Jeffrey A Ruffolo, and Jeffrey J Gray. Generative language modeling for antibody design. *bioRxiv*, 2021.

Sarah Sirin, James R Apgar, Eric M Bennett, and Amy E Keating. Ab-bind: antibody binding mutational database for computational affinity predictions. *Protein Science*, 25(2):393–409, 2016.

Tobias Stadelmann, Daniel Heid, Michael Jendrusch, Jan Mathony, Stéphane Rosset, Bruno E Correia, and Dominik Niopek. A deep mutational scanning platform to characterize the fitness landscape of anti-crispr proteins. *bioRxiv*, pp. 2021–08, 2021.

Jan Stourac, Juraj Dubrava, Milos Musil, Jana Horackova, Jiri Damborsky, Stanislav Mazurenko, and David Bednar. Fireprotdb: database of manually curated protein stability data. *Nucleic acids research*, 49(D1):D319–D324, 2021.

Kiera H Sumida, Reyes Núñez-Franco, Indrek Kalvet, Samuel J Pellock, Basile I M Wicky, Lukas F Milles, Justas Dauparas, Jue Wang, Yakov Kipnis, Noel Jameson, Alex Kang, Joshmyn De La Cruz, Banumathi Sankaran, Asim K Bera, Gonzalo Jiménez-Osés, and David Baker. Improving Protein Expression, Stability, and Function with ProteinMPNN. *Journal of the American Chemical Society*, 146(3):2054–2061, jan 2024. ISSN 0002-7863. doi: 10.1021/jacs.3c10941. URL https://doi.org/10.1021/jacs.3c10941.

Julie D Thompson, Desmond G Higgins, and Toby J Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673–4680, 1994.

Julie D Thompson, Toby J Gibson, Frédéric Plewniak, François Jeanmougin, and Desmond G Higgins. The clustal_x windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic acids research*, 25(24):4876–4882, 1997.

Wen Torng and Russ B Altman. 3d deep convolutional neural networks for amino acid environment similarity analysis. *BMC bioinformatics*, 18:1–23, 2017.

Raphael JL Townshend, Martin Vögele, Patricia Suriana, Alexander Derry, Alexander Powers, Yianni Laloudakis, Sidhika Balachandar, Bowen Jing, Brandon Anderson, Stephan Eismann, et al. Atom3d: Tasks on molecules in three dimensions. *arXiv preprint arXiv:2012.04035*, 2020.

Kotaro Tsuboyama, Justas Dauparas, Jonathan Chen, Elodie Laine, Yasser Mohseni Behbahani, Jonathan J Weinstein, Niall M Mangan, Sergey Ovchinnikov, and Gabriel J Rocklin. Mega-scale experimental analysis of protein folding stability in biology and design. *Nature*, 2023. doi: 10.1038/s41586-023-06328-6. URL https://doi.org/10.1038/s41586-023-06328-6.

Jesse Vig, Ali Madani, Lav R Varshney, Caiming Xiong, Nazneen Rajani, et al. Bertology meets biology: Interpreting attention in protein language models. In *International Conference on Learning Representations*, 2020.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Emily E Wrenbeck, Laura R Azouz, and Timothy A Whitehead. Single-mutation fitness landscapes for an enzyme on multiple substrates reveal specificity is globally encoded. *Nature communications*, 8(1):15695, 2017.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

Zuobai Zhang, Minghao Xu, Vijil Chenthamarakshan, Aurélie Lozano, Payel Das, and Jian Tang. Enhancing protein language models with structure-based encoder and pre-training. *arXiv preprint arXiv:2303.06275*, 2023.

Zaixiang Zheng, Yifan Deng, Dongyu Xue, Yi Zhou, Fei Ye, and Quanquan Gu. Structure-informed language models are protein designers. *bioRxiv*, pp. 2023–02, 2023.

A Related Works

Multiple Sequence Alignments (MSAs) A multiple sequence alignment (MSA) is an established tool used to identify the evolutionary relationship between genes and can be generated for DNA, RNA, and protein sequences. For a particular protein, an MSA represents the genetic variation observed in extant homologous sequences present in a database, such as UniProt (Consortium, 2015), and capture evolutionary and structural constraints for a particular protein family (Thompson et al., 1994; 1997). This makes MSAs a rich source of biological information for computational biologist and recently for training machine learning models. For example, Alphafold2 demonstrates that the information within a protein's MSA is sufficient to predict its 3D structure with near experimental accuracy. Additionally, AlphaFold-Multimer demonstrates that using paired-MSA information improves protein-protein interaction predictions, resulting in significant improvements for predicting of protein complexes Evans et al. (2021).

Sequence-based machine learning frameworks have used MSA information to predict mutational effects and protein fitness. Representative methods, *i.e.*, EVmutation (Hopf et al., 2017), DeepSequence (Riesselman et al., 2018b), MSA Transformer (Rao et al., 2021b), use MSA information to model the evolutionary sequence density with potts models, variational auto-encoders, and transformer, respectively. Biswas et al. (2021); Rives et al. (2021); Barrat-Charlaix et al. (2016) consider a semi-supervised manner which adopts a joint training on MSAs and labeled data for the prediction of protein's fitness.In this paper, instead of using MSA information to construct model inputs or for reconstruction, we incorporate MSA information into the training loss in order to learn protein representations with improved understanding of the mutational landscape. In practice, we achieve this by formulating the training loss to prioritize learning the rank order of the position specific amino acid distribution. Additionally, this paradigm shift on the application of MSA information has the benefit of only requiring MSA information at train time and not at inference time.

Protein Language and Structure Models. Protein representation learning borrows various insights from self-supervision research in the natural language processing community Liu et al. (2019); Yang et al. (2019). The main goal of protein representation learning is to extract biological and functional knowledge of proteins from large unlabeled data to enable zero-shot generalization and/or rapid adaptation to various protein-related tasks. To learn amino acid-level representations from sequence, the community has used methods such as auto-encoding Shuai et al. (2021), auto-regressive Rives et al. (2019b); Meier et al. (2021b); Elnaggar et al. (2020); Riesselman et al. (2019), skip-gram language model Kimothi et al. (2016), mask prediction Vig et al. (2020); Brandes et al. (2022) or amino acid contrastive learning objectives Lu et al. (2020), similarity metric learning Bepler & Berger (2019); Alley et al. (2019), *etc*. The most renown protein language models (pLMs) are the evolutionary-scale models (ESMs) Rives et al. (2019a); Meier et al. (2021a) with ESM2 being the most recent and underpins ESMFold, a sequence-based structure prediction framework (Lin et al., 2023).

For protein structures, 3DCNNs (Townshend et al., 2020; Shroff et al., 2020), GNNs (Townshend et al., 2020; Dauparas et al., 2022), and graph-transformers (Diaz et al., 2023) architectures have been developed to learn residue-level representations using the local chemical environment (microenvironment) or the protein backbone as input. These frameworks primarily use masking to obtain their representations but other pre-training task, such as structure contrastive learning Moon et al. (2023), distance/angle prediction Chen et al. (2023a), and denoising (Watson et al., 2023) have been proposed. Several structure-based frameworks have experimentally designed proteins. The microenvironment framework MutCompute Shroff et al. (2020); d'Oelsnitz et al. (2023) has demonstrated the ability to guide the engineering of several functionally diverse enzymes (Lu et al., 2022; Paik et al., 2021; d'Oelsnitz et al., 2023). Inverse Folding frameworks, such as ESM-IF (Hsu et al., 2022) and ProteinMPNN (Dauparas et al., 2022), use the protein backbone to conditionally design novel sequences for de novo binder design (Watson et al., 2023) and enzyme engineering (Sumida et al., 2024). More works (Chen et al., 2023b; Gligorijević et al., 2021; Zheng et al., 2023; Zhang et al., 2023) focus on the effective knowledge integration between sequence and structure data. Due to the prohibitive cost of training a pLM and the added complexity of decoding an entire protein sequence during inverse folding, we focus on initially validating our EvoRank loss using the microenvironment modality.

B Experiment Setting

B.1 Datasets

For the self-supervised training, we use the same procedure as MutComputeX (d'Oelsnitz et al., 2023). Briefly, this dataset consists of a 90:10 split of 2,569,256 micro-environments sampled from

22,759 protein sequences clustered at 50% sequence similarity and having a structure resolution of at least 3Å from the RCSB (November 2021). Our test data for the folding free energy changes and binding free energy changes are proposed in Diaz et al. (2023); Gong et al. (2023) and we refer the readers to these works for details. These datasets are curated from literature datasets and incorporate additional policies (*e.g.*, below 30% sequence similarity between training and test sets) for better quality.

For mutation effect prediction tasks, we use the experimental structure files from RCSB and AlphaFold structures if the protein lacks an experimental structure. Due to the prohibitive cost of generating experimental data, no phenotype has sufficient experimental data to properly benchmark ML frameworks and evaluate generalization. Thus, we explore datasets for several phenotypes. To date, the most characterized mutation effect phenotype is thermodynamic stability of folding ($\Delta\Delta G$) with several established datasets reserved for evaluation of computational tools: S-Sym, S669, T2837, Gβ1, Myoglobin, and P53. Recently, a cDNA-display protelysis technique enabled the multiplex characterization of single domain mini-proteins to provide the first exhaustive, systematically generated training set for machine learning (Tsuboyama et al., 2023). However, this dataset used proteolytic stability as proxy for thermodynamic stability and the technique does not generalize to full-length functional proteins. For evaluating against the binding free energy changes of point mutations, we used SKEMPIv2 (Jankauskaitė et al., 2019) and AB-Bind (Sirin et al., 2016) for protein-protein interface and PlatinumDB for protein-ligand interface Pires et al. (2015). For the activity, we used an anti-CRISPR protein (A0A247D711) (Stadelmann et al., 2021) and an amidase (Wrenbeck et al., 2017). These datasets are curated from the literature, thus, different techniques–with different biases–were used for data collection. Thus, we filtered mutational data for the techniques that provide high quality measurements: SPR, ITC, FL, IASP, SFFL. To evaluate a non-thermodynamic phenotype, we evaluate against the solubility change deep mutational scanning (DMS) datasets of levoglucosan kinase and TEM1-β-lactamase (Klesmith et al., 2017). To obtain these solubility change measurements, a yeast surface display readout was used not of their wildtype sequences but rather for a chimeric variants with a N-terminus Aga2p domain and a C-terminus epitope tag. Thus, solubility change results should be interpreted with caution since the input sequence and structure used to generate predictions are for the native proteins and not chimeras.

B.2 Training

We train the self-supervised model with AdamW optimizer, with 512 batch size, 5×10^{-5} learning rate, 10^{-5} weight decay. We first train the mask prediction model with MSA soft label loss in equation equation 1 for 100K iterations, and then train with the *EvoRank* defined in equation equation 4, for an additional 100K iterations. Training the model typically requires approximately two day GPU days using an A100. We generate MSAs with JackHMMer Remmert et al. (2012) against UniRef90, using the default configuration of AlphaFold2. For the supervised fine-tuning, we train with AdamW optimizer and backbone learning rate 10^{-5} and regression head learning rate 5×10^{-5}. We tune it with 500 iterations on the curated cDNA dataset generated by Diaz et al. (2023).

Evaluation Metrics and Baselines We assess the model's performance using a comprehensive set of evaluation metrics encompassing both regression and classification aspects. The regression metrics include Spearman correlation coefficient, Pearson correlation coefficient, and Root Mean Squared Error (RMSE). For classification evaluation, we employ AUROC (Area Under the Receiver Operating Characteristic curve). This dual approach ensures a thorough and nuanced evaluation of the model's capabilities across different dimensions of prediction tasks. To comparison with results in the literature, we report the Spearman correlation on different DMS datasets. To establish baselines, we incorporate a range of self-supervised and supervised methods. As a representative self-supervised method, we employ the extensively used ESM2 models. The default baseline is set with the 650M-parameter ESM2 model, and we provide results for other scales of ESM2 models and alternative protein language models. We first evaluate different model performance first on different $\Delta\Delta$G datasets, since these datasets have high-quality labels. Then, we further compare models on more phenotype datasets, to examine whether our model can generalize to different settings.

Model Architecture The microenvironment-based model used here is based on previous work by Diaz et al. (2023). Briefly, the model uses a graph transformer backbone to process an input microenvironment, in which $\mathcal{V}_i^{\text{mask}}$ for amino acid a_i is the input and each atom in this set is represented by its 3D coordinates, atom type, partial charge and solvent accessible surface area. After transforming the atomic representations into a continuous latent space using embedding layers, we process the hidden representations for each atom with graph transformer blocks, where the attention bias is based on the atom-wise Euclidean distance. We refer the readers to (Diaz et al., 2023) for more details on the graph transformer backbone architecture.

The regression head accepts two amino acid embedding vectors and the hidden representation of the microenvironment as input. As shown in Figure 1, we use Siamese network architecture to contextualize each amino acid type to the masked microenvironment, and a MLP to decode a ranking prediction between the two contextualized amino acid embeddings. We refer the readers to (Diaz et al., 2023) for more details on the regression head architecture.

C Additional Experiment Results

# Proteins	# Mut (K)	Fine-Tune MutComputeXGT w/ WT-Mask				Fine-Tuned MutComputeXGT w/ EvoRank			
		Pearson	Spearman	AUC	RMSE	Pearson	Spearman	AUC	RMSE
10	11K	0.50	0.52	0.73	1.92	0.58	0.60	0.78	1.73
50	54K	0.55	0.58	0.77	1.78	0.59	0.61	0.80	1.66
116	117K	0.59	0.62	0.81	1.64	0.61	0.63	0.81	1.62

Table 5: The performance of fine-tuned models on the T2837 dataset trained varying training dataset size. The learning rate and number of iterations are tuned for each SSL pretraining task in order to maximize performance. We fine-tune the model on subsets of the cDNA dataset Diaz et al. (2023) and test the model performance on T2837. '#Mut' denotes number of mutations in the training data.

Impact on supervised fine-tuning One of the most important applications of representation learning is to enable transfer learning to domains with limited labeled datasets. Thus, to evaluate the impact of the `MutRank` representations against the WT-mask representations, we conduct a comparative analysis on supervised fine-tuning for thermodynamic stability using the Stability Oracle framework. Table 5 provides a comprehensive comparison between fine-tuned WT-mask representations (Stability Oracle) and fine-tuned `MutRank` representations. To achieve optimal performance, WT-mask representations and `MutRank` representations are fine-tuned with 3000 (same as Stability Oracle) and 500 iterations, respectively. The evaluation metrics include Pearson correlation, Spearman correlation, AUC, and RMSE on the T2837 folding free energy ($\Delta\Delta G$) phenotype. Our results demonstrate that both models reach approximately the same performance on T2837 from training on the cDNA dataset, with EvoRank loss pretraining having a marginal improvement. Interestingly, EvoRank loss impact is most apparent when there is significantly less fine-tuning data available. When fine-tuned with ∼9% of the proteins (10 proteins and 11K mutations) in the cDNA dataset, EvoRank loss pretraining outperforms WT-mask pretraining by 16%, 15%, 7% for Pearson and Spearman correlation and AUC, respectively, and required 6× fewer training iterations. Furthermore, EvoRank loss pretrained model's Pearson and Spearman correlation and AUC metrics are only 2%, 3%, and 4% lower than Stability Oracle, respectively. While the corresponding WT-mask

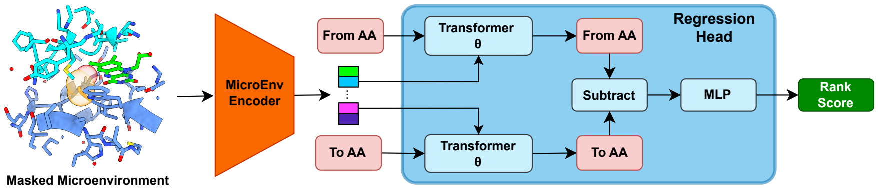

Figure 1: The MutRank architecture, where the rank score is optimized by equation 3. In the regression head, the hidden representation of the microenvironment is used to contextualize the "from" and "to" amino acid embeddings using a Siamese network. The rank hidden representation is generated by subtracting the contextualized amino acid embeddings, which is then decoded into the rank score.

pretrained model's Pearson and Spearman correlation and AUC metrics are 15%, 16%, and 10% lower than Stability Oracle, respectively. These gaps are less drastic when 43% of the proteins (50 proteins and 54K mutations) are used for supervised fine-tuning since the EvoRank loss pretrained model has nearly reached the ceiling of the cDNA dataset. Thus, we conclude that the supervised fine-tuning of the `MutRank` representations can significantly improve the generalization capacity of smaller training sets and simultaneously accelerate training time.

Loss	T2837	S487
$\frac{p_j^{\mathrm{MSA}}(a^+)}{p_j^{\mathrm{MSA}}(a^+)+p_j^{\mathrm{MSA}}(a^-)} - 0.5$	0.51	0.38
$\mathrm{CLMAP}\{\log\{p_j^{\mathrm{MSA}}(a^+)/p_j^{\mathrm{MSA}}(a^-)\}, \pm 5\}$	0.52	0.38
$\mathrm{CLMAP}\{[p_j^{\mathrm{MSA}}(a^+)/p_j^{\mathrm{MSA}}(a^-)]^2, \pm 5\}$	0.50	0.37

Table 6: We demonstrate the model Pearson correlation coefficient with different rank score loss. The first block shows the loss as the default setting. The second block displays the loss with other formulations.

Dataset	Phenotype	EvoRank w/ Classification Head				EvoRank w/ Joint Heads				EvoRank w/ Regression Head			
		Pearson	Spearman	AUC	RMSE	Pearson	Spearman	AUC	RMSE	Pearson	Spearman	AUC	RMSE
T2837	$\Delta\Delta$G	0.47	0.49	0.76	1.78	**0.51**	**0.53**	0.77	1.76	**0.51**	**0.53**	**0.78**	**1.70**
levoglucosan kinase	Δ Solubility	0.28	0.34	0.62	1.40	**0.30**	**0.34**	**0.65**	**1.37**	0.29	**0.34**	0.64	1.39
S487	Protein-Protein $\Delta\Delta G_{\text{bind}}$	0.36	0..35	0.65	1.35	0.37	0.37	**0.67**	1.36	**0.38**	**0.38**	**0.67**	**1.26**
platinumDB	Protein-Ligand $\Delta\Delta G_{\text{bind}}$	0.25	0.24	0.61	1.58	0.27	0.27	**0.65**	1.58	**0.28**	**0.28**	0.64	**1.53**
ABBind	Antibody-Antigen$\Delta\Delta G_{\text{bind}}$	0.39	0.45	0.72	1.48	**0.41**	**0.46**	0.72	1.57	**0.41**	**0.46**	**0.74**	**1.42**

Table 7: We illustrate that `MutRank` without additional regression head can still generate good results on the test sets. The numbers reported are averaged over three trials.

Head architecture ablations In our approach, to train with the EvoRank loss, we replace the classification head with a regression head. This head contextualize the embedding vectors for the two amino acids with the hidden representation for a particular microenvironment in order to compute a residue specific rank score. Alternatively, we can use the EvoRank loss with the original classification head by calculating the rank score from the logits. In this ablation study, shown in Table 7, we observe that introducing the additional regression head generally results in a modest performance improvement ranging from 1% to 4% across 5 datasets. More importantly, these results demonstrate the superior zero-shot generalization of the EvoRank representations over the WT-mask baseline regardless of the head architecture.

Exploring different loss formulation Training with EvoRank loss is a two-stage procedure. Initially, we train the backbone using MSA-based soft labels with the α-divergence loss and subsequently fine-tune with the EvoRank loss. 1) We evaluate the impact of jointly training with α-divergence loss and EvoRank (Table 7, middle column). Our results indicate that the linear combination of the α-divergence and EvoRank losses with 0.4 and 0.6 coefficients, respectively, provides the best performance. However, these results match our previous performance. 2) We then evaluate different ways to compute the rank score for a residue from the MSA distribution, and benchmark on the T2837 and S487 datasets. As demonstrated in Table 6, all rank score formulations converge to similar performance on T2837 and S487. Thus, the exact formulation for computing the rank score has an insignificant impact on performance and further demonstrates the robustness of the EvoRank loss.

Dataset	#Mut	`MutRank`-2M	`MutRank`-8M	`MutRank`-24M	`MutRank`-48M
T2837	2837	0.48	0.51	0.51	0.51
levoglucosan kinase	9011	0.27	0.29	0.29	0.28
Gβ1	935	0.58	0.62	0.62	0.62
S487	487	0.36	0.38	0.40	0.40
PlatinumDB	925	0.25	0.28	0.28	0.26

Table 8: We demonstrate the model Pearson correlation coefficient with different model sizes. All the results are averaged over three trials.

Method	Pearson	Spearman	AUROC	RMSE
cDNA MSA	0.15	0.13	0.62	2.17
ESM2	0.37	0.37	0.65	5.48
MutComputeXGT	0.38	0.38	0.64	1.89
MutRank	0.45	0.46	0.71	1.09

Table 9: We demonstrate that our method get better generalization compared to naive MSA on the cDNA117K dataset.

Model size ablations The machine learning community has empirically demonstrated the benefits of increasing model size (Dehghani et al., 2023; Chowdhery et al., 2023). This too has been demonstrated by protein language models (Elnaggar et al., 2021; Rives et al., 2019a; Lin et al., 2023). However, to the best of our knowledge no study has explored the impact of model size for protein structure-based machine learning frameworks. We conducted a comprehensive analysis ranging the parameters from ∼2M to ∼48M. The results, presented in Table 8, demonstrate marginal to no improvements from scaling the model parameters. For example, the smallest model (∼2M) exhibit diminished performance compared to the largest (∼48M) model but the average performance improvement across 4 datasets is only ∼6%. But the same analysis between the (∼8M) and (∼48M) models results in an average performance decrease of 1.25%. Further experiments, such as scaling the dataset beyond ∼20K proteins, are required to confirm if structure-based ML frameworks trained with EvoRank loss will benefit from model scaling. All experiments reported in this work are from the 8M parameter model.

Generalizing beyond the MSA distribution While our model is trained with MSA information, the MSA information itself can also directly serve as a predictor for mutation effects. In the literature, MSAs are often used to create a sequence profile (Lüthy et al., 1994) or position-specific scoring matrix (PSSM) (Jones, 1999), which can be used to predict the impact of a mutation by assessing the deviation from the expected amino acid at a specific position.

We evaluate if EvoRank representations outperforms these naive MSA baselines using the large cDNA dataset (∼ 117K mutations from 116 single domain proteins) provided in Diaz et al. (2023). For these 116 proteins, the average and std of their MSA depth is 3.9K±0.6K sequences. To calculate naive predictions from a protein's MSA, we use the log-odds of the empirical amino acid distribution at a position (Figure **??**): $\log(p_{\text{to}}/p_{\text{from}})$. Furthermore, we provide MutComputeXGT and ESM2 as a baselines for comparison. As demonstrated in Table 9, our method not only outperforms ESM2 but also significantly improves upon the naive MSA predictions derived from the cDNA MSAs: for Pearson correlation, our method achieves 0.45, surpassing MSA's 0.15, ESM2's 0.37, and MutComputeXGT's 0.38. These results demonstrate that the MutRank representations capture residue specific variability beyond what is present in a protein's MSA.