LIGHTWEIGHT EQUIVARIANT GRAPH REPRESENTATION LEARNING FOR PROTEIN ENGINEERING

Anonymous authors
Paper under double-blind review

ABSTRACT

This work tackles the issue of directed evolution in computational protein design that makes accurate predictions of the function of a protein mutant. We design a lightweight pre-training graph neural network model for multi-task protein representation learning from its 3D structure. Rather than reconstructing and optimizing the protein structure, the trained model recovers the amino acid types and key properties of the central residues from a given noisy three-dimensional local environment. On the prediction task for the higher-order mutants, where many amino acid sites of the protein are mutated, the proposed training strategy achieves remarkably higher performance by 20% improvement at the cost of requiring less than 1% of computational resources that are required by popular transformer-based state-of-the-art deep learning models for protein design.

1 INTRODUCTION

Mutation is a biological process where the amino acid (AA) type of one or multiple sites of a specific protein is changed. While the wild-type proteins' functions do not always meet the demand of bio-engineering, it is vital to manually optimize the functionality, namely fitness, with favorable mutations so that they are applicable in designing antibodies (Wu et al., 2019; Pinheiro et al., 2021; Shan et al., 2022) or enzymes (Sato & Ishida, 2019; Wittmann et al., 2021).

A protein usually constitutes hundreds to thousands of AAs, where each residue belongs to one of twenty AA types. To optimize a protein's functional fitness, a greedy search in the local sequence is usually conducted, where AA sites were mutated to proper AA types to render a protein mutant with the highest gain-of-function (Rocklin et al., 2017). Such a process is called directed evolution. Normally, multiple AA sites ($\sim$5-10) of the protein need to be mutated to obtain a mutant with great fitness (Arnold, 1998; Cui et al., 2021), namely deep mutants. It requires enormous experimental cost, as the total number of potential combinations of mutations for deep mutants is astronomical.

Since it is impossible to conduct systematic experimental tests on all possible deep mutants, *in silico* examination of protein variants' fitness becomes highly desirable. A handful of deep learning methods have been developed to accelerate the discovery of advantageous mutants. For instance, Sato & Ishida (2019) applied 3DCNN to identify a new polymerase with advantageous single-site mutation and enhanced the speed of degrading PET by 7-8 times at 50°. Luo et al. (2021) proposed ECNET to use evolutionary context to predict functional fitness for protein engineering, guide the engineering of TEM-1 β-lactamase and identify variants with improved ampicillin resistance. Thean et al. (2022) combined deep neural networks and SVM to predict variants' activities in multi-site-saturated mutagenesis libraries from a small sample of variants and identifed Cas9 nuclease variants that possess higher editing activity of derived base editors in human cells.

Due to the scarcity of labeled protein data, researchers often pre-train an encoder for unsupervised learning with protein sequences or structures, and use the learned protein representations to train downstream tasks, such as *de novo* protein design (Hsu et al., 2022), mutation effect prediction (Ingraham et al., 2019; Jing et al., 2020; Meier et al., 2021; Notin et al., 2022), and higher-level structure prediction (Elnaggar et al., 2021). In the context of fitness prediction of mutation effect, existing methods usually transform the problem to mini-*de novo* design, i.e., infer a specific AA type from its micro-environment, including the neighborhood AA types. Current state-of-the-art protein sequence-based methods rely heavily on multiple sequence alignment (MSA) (Riesselman et al., 2018; Frazer et al., 2021; Rao et al., 2021) and protein language models (Elnaggar et al.,

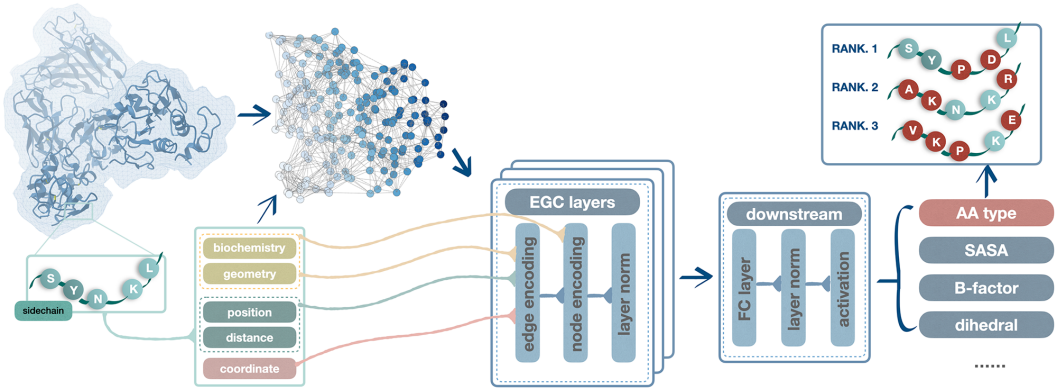

Figure 1: An illustration of the proposed learning scheme. The model is pre-trained with a set of protein graphs that are featured by (perturbed) node and edge attributes, as well as the nodes' 3D positions with a multi-task learning strategy. The structural representations are encoded with a stack of EGC layers to extract rotation invariant and translation equivariant representations for each node on individual graphs. In downstream tasks, fully-connected layers are employed to learn different labels, where the AA type prediction is used for suggesting top-ranked mutations.

2021; Rives et al., 2021; Nijkamp et al., 2022; Brandes et al., 2022). While MSA helps capture important evolutionary properties of the protein family, it nevertheless multiplies the requirements of computing resources. The latter protein language models derived from natural language processing (NLP) encode sequence semantics and often need hundreds of GPU cards to train on billions of protein sequences. Meanwhile, an auto-regressive inference process is usually required along the entire protein sequence to score a mutation on a single site, which further slows down the inference speed (Sato & Ishida, 2019; Meier et al., 2021; Liu et al., 2022; Hsu et al., 2022; Notin et al., 2022). More importantly, when predicting the fitness of the higher-order mutants, most of these models made a crude assumption that the multiple-site mutation effect is a linear summation of the effect of each individual mutation, which is incorrect in most cases (Lehner, 2011; Breen et al., 2012). The ignored epistatic effects between different sites is potentially a key factor hindering the acquisition of favorable high-order mutants in directed evolution (Sarkisyan et al., 2016; Rollins et al., 2019).

Mutation of AA sites also occurs in nature, where an AA site might be mutated to any of the other 19 AA types in a random manner. It is suggested by natural selection that only the mutation that exhibits the best fitness and fits the environment survives. Altering AA types of a protein in nature can be viewed as adding corruptions to the node features of the protein graph, and denoising the graph makes a remedy to search for mutants with the best fitness. We hereby model the protein mutation effect prediction as a denoising problem with equivariant graph neural networks (Satorras et al., 2021). Such a scheme predicts the fitness of mutation effect on a protein with an arbitrary number of mutant sites and provides efficient recommendations to discover favorable mutants. We treat each protein as a graph with AAs being graph nodes. The 3D spatial structure of the folded proteins is encoded with the graph geometry. The first-level information, such as AA types, spatial coordinates of Cα, and C-N angles between neighboring AAs, are embedded in node features. We call the generated representation a *protein graph*, which provides an elegant structural description of the protein.

Compared to existing zero-shot state-of-the-art methods for mutation effect prediction like ESM-1v (Meier et al., 2021) and ESM-IF1 (Hsu et al., 2022), the designed lightweight equivariant graph neural network (LGN) stands out in three perspectives.

First, our method improves generalization ability through the multi-task learning strategy and biological prior knowledge (the proposed model is fed with protein tertiary structure directly). The former trains data-driven black-box encoders to describe a given protein's microenvironment, and the latter inserts domain knowledge towards practically meaningful representations.

Secondly, the model is efficient in both the training and inference phases. The spatial graph inputs portray the topological properties of proteins, which circumvents data augmentation that is typically

required by sequence or figure representations. Equivariant message passing layers of the model, alternatively, provides a feature distillation unit with translation invariance and rotation equivariance (Atz et al., 2021), which can precisely capture the micro-frame property in the protein graph geometry. Meanwhile, the downstream task allows outputting the probability of all the amino acid residues at a time, which is hundreds of times faster than classic auto-regressive methods.

Thirdly, the typical approach to calculating the higher-order mutation effect sum up log-odd-ratio scores of the corresponding individual single-site mutants Meier et al. (2021); Hsu et al. (2022). However, the linear summation is unsubstantiated, as the epistatic effect is known to be present that mutations of individual sites are not independent (Lehner, 2011; Breen et al., 2012). Instead, our scoring matrix does not need any of the independent-mutation assumptions, and the output can take any number of mutation sites at a time.

2 Zero-shot Learning for Protein Recovery

2.1 Graph Representation of Protein Structure

For a given protein, we create a k-nearest neighbor (kNN) graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to describe its 3D structure and molecular properties. Here each node $v_i \in \mathcal{V}$ represents an amino acid residue with SE(3)-invariant node attributes $\boldsymbol{X} \in \mathbb{R}^{34}$ constituting biochemical properties and geometric properties of amino acids. The former includes 20-dimensional attributes of one-hot encoded amino acid types ($\boldsymbol{X}_{\text{aa}}$), two scalars for each residue, i.e., solvent-accessible surface area (SASA) and the standardized crystallographic B-factor, and 5 normalized surface-aware node features. The geometric properties include the direction position ($\boldsymbol{X}_{\text{pos}}$) of each residue by 3D coordinates of its α-carbon and the relative position of the amino acid in the protein chain ($\boldsymbol{X}_{\text{agl}}$) by the dihedral angles $\{\sin, \cos\} \circ \{\phi, \psi\}$ computed from the backbone atom positions. For a specific v_i of the ith amino acid in the protein sequence, the dihedral angles are measured from $C\alpha_{i-1}, N_i, C\alpha_i, N_{i+1}$.

To build edge connections, we first define a symmetric adjacency matrix $\boldsymbol{A}$ with the kNN-graph, i.e., each node is connected to up to k other nodes in the graph that has the smallest Euclidean distance over other nodes, and the distance is smaller than a certain cutoff, e.g., $30\mathring{A}$. Consequently, if v_i and v_j are connected to each other, we have $e_{ij} \in \mathcal{E}$ and $\boldsymbol{A}_{ij} = \boldsymbol{A}_{ji} \neq 0$. The edge attributes $\boldsymbol{E} \in \mathbb{R}^{93}$ are featured on the connected edges, including 15 inter-atomic distances, 12 local N-C positions, and 66-dimensional position encoding that records the relative position in the protein sequence.

2.2 Pre-training with Prior Domain Knowledge for Better Protein Fitness

The wild-type proteins suffer from random perturbations or mutations that not every AA site has the best AA type (Liu et al., 2022). To this end, we pre-train our model with a multitask learning strategy, which removes the natural corruptions and predicts key protein properties to help encode the microenvironment of the stabilized proteins of interest. The following section explains the perturbation and training strategies in accordance.

AA Type Denoising For a node of amino type $\boldsymbol{x}_{\text{aa}}$, we refine its type $\tilde{\boldsymbol{x}}_{\text{aa}}$ by adding a Bernoulli noise, i.e.,

$$\pi(\tilde{\boldsymbol{x}}_{\text{aa}}|\boldsymbol{x}_{\text{aa}}) = p\delta(\tilde{\boldsymbol{x}}_{\text{aa}} - \boldsymbol{x}_{\text{aa}}) + (1-p)\mathcal{M}(n, \pi_1, \pi_2, ..., \pi_n), \tag{1}$$

where the confidence level p is a tunable parameter that controls the proportion of residues that are 'noise-free'. The probability for the residue to become a particular type depends on the distribution of the 20 types $\mathcal{M}(n, \pi_1, \pi_2, ..., \pi_n)$, which involves prior knowledge in molecular biology. This paper defines the distribution by the observed probability density of amino acid types in wild-type proteins [1]. See Appendix E to better understand the influence of different confidence levels.

Geometric Properties Denoising For the continuous-valued features, such as 3D coordinates and dihedral angles, an i.i.d Gaussian noise is introduced, learning to remove which corresponds to approximating the data-generating force field of molecules (Zaidi et al., 2022). To be specific,

$$\tilde{\boldsymbol{x}}_{\text{pos}} = \boldsymbol{x}_{\text{pos}} + \sigma\epsilon, \quad \text{where } \epsilon \sim \mathcal{N}(0, I_3). \tag{2}$$

The noise effect is determined by σ, which is tunable to fit the scale of the noiseless raw feature.

[1] Retrieved from AlphaFold (Varadi et al., 2022) at https://alphafold.ebi.ac.uk/

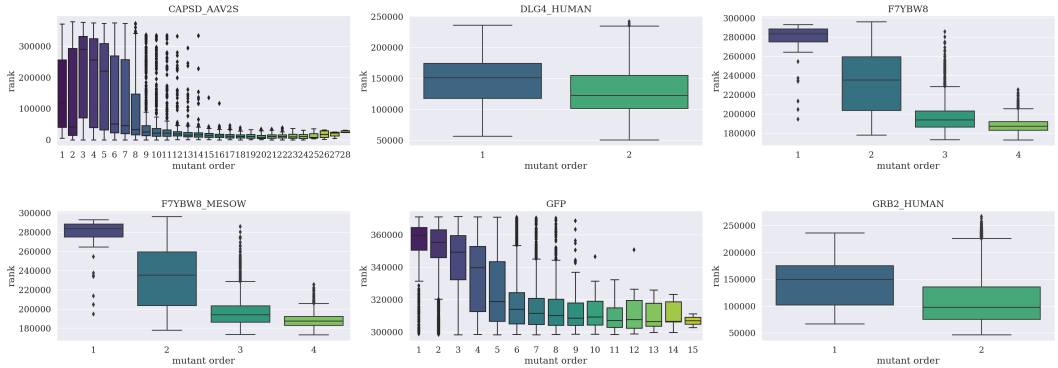

Figure 2: Rank distribution on higher-order mutants. Generally, mutant on more sites would result in a higher score, i.e., a smaller rank value.

Bio-chemistry Properties Recovery Aside from the perturbed residues type and geometric properties to denoise, other auxiliary tasks are introduced to better retrain the hidden microenvironment representation. Specifically, SASA are known to strongly influence AA type preferences, and B-factors are associated with the conformations and mobility of the neighboring AA. We thus introduces inductive biases to the model by predicting these two properties in the downstream tasks.

Label Smoothing with Amino Acid Substitution Matrices Protein sequence alignments provide important insights for understanding gene and protein functions. The similarity measurement of an alignment of protein sequence reflects the favors of all possible exchanges of one amino acid with another. We employ BLOSUM (Henikoff & Henikoff, 1992), a substitution matrix , which accounts for the relative substitution frequencies and chemical similarity of amino acids, derived from the statistics for every conserved regions of protein families in BLOCKS database. As AA sites are more likely to be mutated to the AA type within the block of high similarity scores in the BLOSUM table, we hereby modify our loss function that a mutation to an AA type with a higher similarity score accumulates a smaller penalty than to the one with lower similarity score.

2.3 Protein Structure Representation with Equivariant GNNs

As the initial proteins are structured in the 3-dimensional space, it is vital for the model to predict the same binding complex no matter how the input proteins are positioned and oriented. Instead of practicing expensive data augmentation strategies, we follow Satorras et al. (2021) and construct SE(3)-equivariant and invariant neural layers for graph embedding. At the lth layer, an Equivariant Graph Convolution (EGC) inputs a set of n hidden node properties embedding $\boldsymbol{H}^l = \{\boldsymbol{h}_1^l, \ldots, \boldsymbol{h}_n^l\}$ as well as the node coordinate embeddings $\boldsymbol{X}_{\text{pos}}^l = \{\boldsymbol{x}_1^l, \ldots, \boldsymbol{x}_n^l\}$ for a graph of n nodes. The attributed edges are denoted as $\boldsymbol{E} = \{\ldots, \boldsymbol{e}_{ij}, \ldots\}$. The target of an EGC layer is to output a transformation on the node feature embedding $\boldsymbol{H}^{l+1}$ and coordinate embedding $\boldsymbol{X}_{\text{pos}}^{l+1}$. Concisely: $\boldsymbol{H}_{\text{pos}}^{l+1}, \boldsymbol{X}^{l+1} = \text{EGC}\left[\boldsymbol{H}^l, \boldsymbol{X}_{\text{pos}}^l, \boldsymbol{E}\right]$. To achieve this, an EGC layer defines

$$
\begin{aligned}
\boldsymbol{m}_{ij} &= \phi_e\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \left\|\mathbf{x}_i^l - \mathbf{x}_j^l\right\|^2, \boldsymbol{e}_{ij}\right) \\
\boldsymbol{x}_i^{l+1} &= \mathbf{x}_i^l + \frac{1}{n}\sum_{j \neq i}\left(\mathbf{x}_i^l - \mathbf{x}_j^l\right)\phi_x\left(\mathbf{m}_{ij}\right) \\
\boldsymbol{m}_i &= \sum_{j \neq i} \mathbf{m}_{ij} \\
\boldsymbol{h}_i^{l+1} &= \phi_h\left(\mathbf{h}_i^l, \boldsymbol{m}_i\right),
\end{aligned}
\tag{3}
$$

where ϕ_e, ϕ_h are respectively the edge and node propagation operations, such as multi-layer perceptrons (MLPs). The ϕ_x is an additional operation that projects the vector embedding $\boldsymbol{m}_{ij}$ to a scalar value. The EGC layer preserves equivariance to rotations and translations on the set of 3D

node coordinates $\boldsymbol{X}_{\text{pos}}$, while simultaneously performing equivariance to permutations on the set of nodes $\mathcal{V}$ in the same fashion as GNNs.

2.4 Model Overview

Our model is depicted in Figure 1. We take a set of protein graphs with node and edge attributes, and each node's 3D coordinates as the input of the pre-training model. A stack of EGC layers is trained to extract rotation equivariant and translation invariant representations for each node on individual graphs. The hidden representation is then sent to fully-connected layers for downstream tasks, which contain multiple tasks such as AA type classification, SASA and B-factor prediction, and 3D coordinates denoising. The total loss for the multitask learning task is given by

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{aa}} + \lambda_1 \mathcal{L}_{\text{sasa}} + \lambda_2 \mathcal{L}_{\text{b-fac}} + \lambda_3 \mathcal{L}_{\text{pos}} + \lambda_4 \mathcal{L}_{\text{agl}}, \tag{4}$$

where $\lambda_i, i = 1, \ldots, 4$ are tunable hyper-parameters to balance different losses on auxiliary regression tasks. These losses are measured by mean squared error (MSE) loss. For AA type classification, we measure its loss $\mathcal{L}_{\text{aa}}$ by cross-entropy with label smoothing technique (Szegedy et al., 2016). The classification loss on an arbitrary node i reads

$$\mathcal{L}_{\text{aa}} = (1-\varepsilon)\Big[-\sum_{y=1}^{20} p(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i) \log q_\theta(\hat{y}_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)\Big] + \varepsilon\Big[-\sum_{y=1}^{20} u(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i) \log q_\theta(\hat{y}_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)\Big],$$

where $p(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)$ denotes the ground-truth distribution and $q_\theta(\hat{y}_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)$ is the distribution of predicted labels following a softmax function. In order to improve the generalization and respect the prior biological knowledge, we modify the ground truth label distribution $p(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)$ from the hard one-hot encoding to $(1-\varepsilon)p(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i) + \varepsilon u(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)$ when the predicted $\hat{y}_{\text{aa}} = y_{\text{aa}}$ and $\varepsilon u(y_{\text{aa}}|\boldsymbol{X}_i, \boldsymbol{E}_i)$ otherwise with some tolerance factor ε. In particular, we define the distribution of $u(y|x_i)$ by the BLOSUM substitution matrix.

3 Results

3.1 Experimental Setup

Our LGN is pre-trained on **CATH v4.3.0** (Orengo et al., 1997) with artificial noise to predict AA type, 3D coordinates, dihedral angles, and chemical properties (SASA and B-factor). The hidden embeddings of amino acids are learned by an equivariant GNN. The evaluation is conducted with deep mutational scanning (DMS) datasets for zero-shot fitness of mutation prediction tasks. The model performance is compared against popular state-of-the-art language models and structure-enhanced models.

Baseline Models We compare with a diverse of state-of-the-art models on the fitness of mutation effects prediction. In particualr, MSA Transformer Rao et al. (2021) is a language model with aligned protein sequences of interest; ESM-1v (Meier et al., 2021) make zero-shot mutation predictions with masked language modeling; and ESM-IF1 (Hsu et al., 2022) predicts protein sequence with GVP, a graph representation learning methods for vector and scalar features of protein graphs. The model is pre-trained on a mixture of **CATH** and **AlphaFold2** datasets, and the sequence embedding is enhanced with transformers (Vaswani et al., 2017).

Lightweight Equivariant Graph Neural Networks (LGN) To train our LGN framework, we first generate protein graphs for the sequences in **CATH**. See Appendix A for a detailed introduction on the generation, and Appendix C for summaries in the generated dataset. For the total number of $31,848$ protein graphs of 150 nodes on average, we randomly pick 500 graphs for validation and leave the remaining for model fitting. During the learning phase, we assign random perturbations to AA types and other features we mentioned earlier in Section 2. During the validation step, the noises are fixed for stable and comparable measurements. The specific influence on different choices on the hyper-parameters (e.g., the noise level) will be discussed later in this section and Appendix D-E. The main architecture constitutes a stack of 6 EGC layers following 1 fully-connected layer to make predictions on the different learning tasks. On each node, the output is a vector representation consisting of 20 probabilities of the masked amino acid, 1 predicted SASA, 1 B-factors, and 4

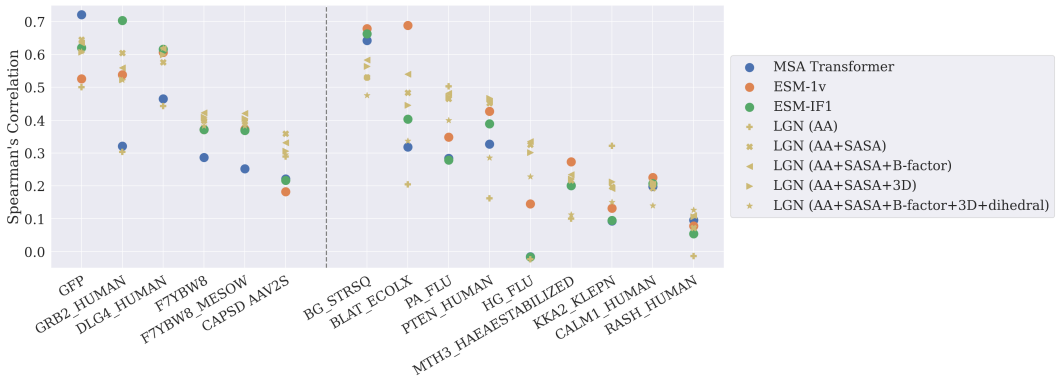

Figure 3: Per task performance on the fitness of deep mutant prediction with zero-shot models. Each point indicates Spearman's correlation coefficients on the corresponding protein dataset (denoted on the x-axis). The left 6 proteins to contain higher-order mutations, and the right 9 proteins only record single-site mutate effects.

dihedral values (when applicable). The 3D-coordinates are derived directly from EGC outputs. The loss function by Equation 4 guides the backward propagation with ADAM (Kingma & Ba, 2015) optimizer. The model is trained with 300 epochs with the initial rate set to 0.001 and weight decay to 0.01. The learning rate is further dampened to 0.0001 after 150 epochs.

Evaluation All the models are evaluated with deep mutational scanning (DMS) assays that assess a diverse set of 15 proteins, where 9 of them only contains single-site mutation scores, and 6 of them have both single-site and higher-order mutant records (see Appendix B). The protein structures are folded from the provided sequence information with ALPHAFOLD (Jumper et al., 2021), following the exact same pre-processing steps as in **CATH** for generating protein graphs. The only difference is that we do not append artificial noises onto the test proteins, as we assume they are already noisy. We then send the unmutated test proteins to the pre-trained LGN model and use the log-odd-ratio in Equation 5 of the predicted probabilities of AA types for suggesting the deep mutant ranks. The final performance on deep mutant prediction is evaluated on Spearman's correlation coefficient between the predicted and experimental scores on all the mutation combinations in the deep scanning dataset.

3.2 FITNESS OF DEEP MUTANTS PREDICTION

The first experiment evaluates the fitness of proteins' mutation effects prediction with zero-shot models, i.e., the fitness scores are inferred directly from a pre-trained model without fine-tuning on a task-specific model. We visualize the overall performance comparison on protein-wise Spearman correlation coefficients in Figure 3. Our LGN outperforms baseline methods in deep mutant tasks and achieves at least comparable results in single-site mutant tests. Overall, our model achieves 0.5037 weighted average correlations on deep mutant effect prediction by predicting AA types and SASA, and 0.4819 on AA types, B-factor and SASA. In contrast, MSA TRANSFORMER only achieves 0.3756 Spearman's correlation. The scores for ESM-IF1 and ESM-1V are 0.4270 and 0.3902, respectively. For the single-site mutations, we also achieved at-least comparable performance to the baseline methods at a significantly less computational cost.

3.3 PROTEIN RECOVERY

In this experiment, we investigate the auxiliary learning tasks of the pre-trained model, including the learning performance in predicting AA types, SASA, and B-factor. In specific, we visualize the confusion matrix of the predicted AA types with respect to the ground-truth AA types to see evaluate the model's capability to recover from noisy sequences to the original sequence, i.e., if the large values are accumulated to the diagonal of the confusion matrix. For the SASA and B-factor predictions, we examine the R^2 of the predicted and the ground-truth values on **CATH**. The results are visualized in Figure 4 with denoised AA type, as well as the predicted SASA and B-factor as the output tasks. The AA type prediction achieves high accuracy with the majority of

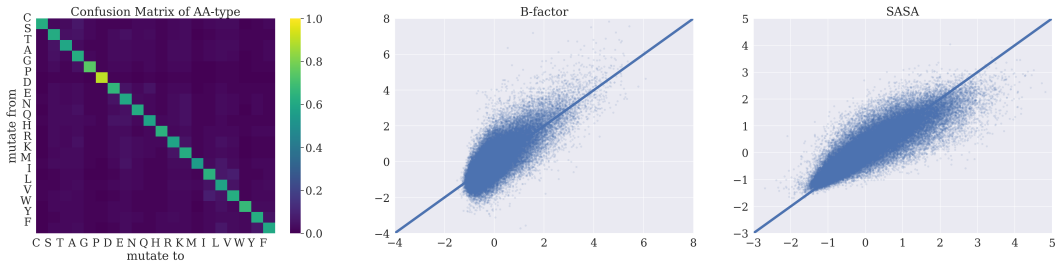

Figure 4: Denoising performance of the pre-trained model on the perturbed proteins. The three plots (from left to right) display the confusion matrix of predicted AA types, and linear regression on the predicted SASA and B-factor, respectively.

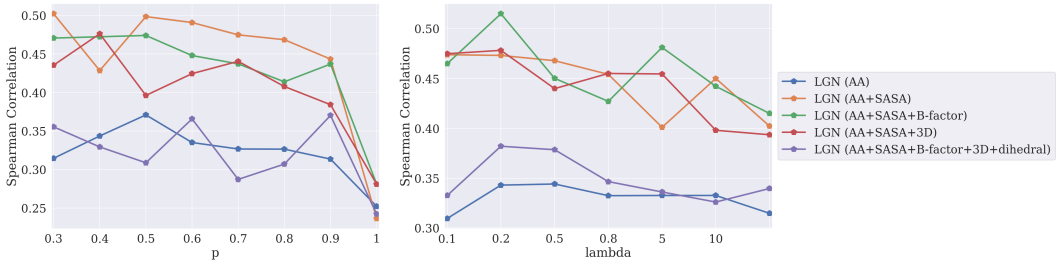

Figure 5: Average Performance on different Choices of ps (left) and λs (right) with different variants of auxiliary tasks.

predictions accumulated on the diagonal line. For the two regression tasks, we fit the true value and the predicted value with linear regression. The estimated coefficients are 1.008 and 0.989 for B-factor and SASA, respectively. The p-value for both coefficients is 0.000. In addition, Pearson's correlation coefficients for the two variants are respectively 0.884 and 0.791.

3.4 Selection Criteria on the Additional Hyper-parameters

As LGN introduces two additional hyper-parameters in training the model, this section examines the influence of selecting different ps and λs. Figure 5 visualizes a selection of model performance on the Spearman's correlation with variant ps and λs with wild-type noise distribution.

While the choice of p can be determined by prior knowledge regarding the quality of wild-type proteins, here we treat p as a data-driven hyper-parameter to be optimized during the model training. We exclude extremely small ps to avoid drastic perturbation rates and search for the optimal $p \in \{0.3, 0.4, \ldots, 0.9, 1\}$. The different choices on ps are validated with different learning tasks on the left side of Figure 5 for higher-order mutants. In general, a moderate p between 0.3 and 0.6 best suits the majority selection of learning modules and noise distribution. Based on the overall performance, we suggest $p = 0.6$ as the default value of the confidence level (See Appendix E for more results on different types of perturbation noise and proteins).

We also investigate a wide range of the choices of λs. For simplicity, we let $\lambda_1 = \lambda_2 = \lambda_3 \in \{0.05, 0.1, 0.2, 0.5, 0.8, 5, 10\}$ and fix $\lambda_4 = 0.5$. All the results are conducted under the recommended $p = 0.6$ with wild-type noise. We report the average performance on deep mutants in the right plot of Figure 5, which demonstrates a relatively flat and steady trend with a mild peak at $\lambda = 0.2, 0.5$. We supplement additional experimental results in Table 7 of Appendix E with different model setups.

3.5 Inference Speed

Our LGN consumes significantly fewer computational resources in training and inference. This section reports a direct comparison of the model scale, inference time, and prediction performance. The results are visualized in Figure 6.

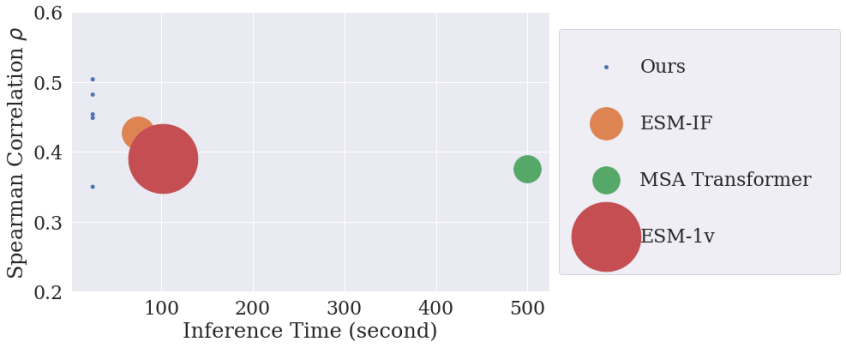

Figure 6: Comparison of Inference Efficiency. The area of the ball indicates the number of network parameters of a model. Our model (in blue) can achieve SOTA performance (y-axis) with minimum inference time (x-axis) and 1% number of parameters of the ESM.

In Table 1, we compare the model size and the required resources with the baseline methods: MSA TRANSFORMER (Rao et al., 2021), ESM-1V (Meier et al., 2021), and ESM-IF1 (Hsu et al., 2022). To be specific, the attributes for training the model are provided by the authors, and the inference speed and memory are tested by us. As each protein requires independent inference progress, we hereby take **GFP** as an example protein sample. The protein constitutes 236 amino acid residues, and it has over $50,000$ mutant records. (see Table 2 for more details). Note that:

1. the $2,687$ input token length only refers to the maximum protein length we used during training. In fact, the model itself can process large protein graphs containing over tens of thousands of amino acids.
2. the training speed and required resources for MSA TRANSFORMER are retrieved from Meier et al. (2021). The original work by Rao et al. (2021) only reports that they used $32\times$V100 GPUs for training, without revealing the training time.

Table 1: Characteristics of different models used in the experiment.

model	MSA TRANSFORMER	ESM-1V	ESM-IF1	LGN(ours)
input	sequence	sequence	sequence+structure	structure
training dataset	**Uniref50** (2018-03)	**Uniref90**	**CATH+AF2** (2020-03)	**CATH** v4.3.0
training size	45M	98M	12M	0.03M
max. input token	$1,024$	$1,024$	$1,024$	$2,687$[1]
# parameters	100M	650M	142M	1.5M
# layers	12	-	20	6
# head	12	-	8	-
# hid. dim.	-	-	$512-2,048$	512
speed (training day)	13**	6	653	0.17
resource (train)	$128\times$V100[2]	$64\times$V100	$32\times$V100	$1\times$3090
inference speed (second) (**GFP**)	>500	74.84	101.99	25.28

4 RELATED WORK

Protein Sequence and Structure Representation Due to the enormous experimental cost of measuring protein structures, the number of known protein sequences is thousands of times larger than protein structures (Eswar et al., 2008; Hsu et al., 2022). Meanwhile, the protein sequences representation is highly similar to human language, which naturally promotes the fast development of natural language processing (NLP), especially transformer-based methods for encoding protein sequences (Coin et al., 2003; Meier et al., 2021; Ofer et al., 2021). However, the geometry of proteins also

suggests higher-level structures and topological relationships that are vital to protein functionality. Structure prediction of proteins always attracts great attention in the field (Chi & Liberles, 2016; Jumper et al., 2021; Baek et al., 2021; Varadi et al., 2022). The breakthrough progress in protein folding also enriches structured proteins. For instance, Hsu et al. (2022) and Ma et al. (2022) mixed experimentally-tested and ALPHAFOLD-predicted for model training, which greatly eases the data shortage problem and achieves significant performance gain.

Structural encoding for Protein Graphs According to the laws of physics, the atomic dynamics do not changed no matter how a protein is translated or rotated from one place to another (Han et al., 2022). Therefore, the inductive bias of symmetry should be incorporated into the design of protein structure-based models. To this end, research work has been proposed to respect the spatial relationship of amino acids (Torng & Altman, 2017; Sato & Ishida, 2019). Such CNN-based methods aggregate the local structure of each residue and integrate estimated local qualities into the whole protein properties. However, these methods neglect geometric equivariance, which can usually be captured by equivariant graph neural networks (Ganea et al., 2021; Stärk et al., 2022).

Protein Representation As existing protein language models require high computational costs and are difficult to train, finding an effective feature representation of protein data is important for downstream tasks (Thompson et al., 2012). Contrastive learning and self-prediction Elnaggar et al. (2021), Zhang et al. (2022), andHsu et al. (2022) used self-supervised pre-training methods provide ways to extract good representation for reducing computational resources. Despite only applying classical representations learning methods on protein, some researchers designed sophisticated methods. Li et al. (2022) proposed $\vec{W}$-GNN variants efficiently interact scalar-vector features. Besides, Somnath et al. (2021) introduced a multi-scale model HOLOPROT connecting surface to structure and sequence.

Mutation Effect Prediction Multiple sequence alignment (MSA) is an essential ingredient for many of the existing state-of-the-art methods to predict the effect of single amino acid substitutions such as DEEPSEQUENCE (Riesselman et al., 2018), ALPHAFOLD (Jumper et al., 2021) and MSA TRANSFORMER (Rao et al., 2021). The MSA for a protein sequence or domain captures meaningful information on evolutionary information of the protein within its family at the cost of bringing severe limitations–not all proteins are alignable, such as CDRs of antibody variable domains (Shin et al., 2021), and not all the alignments are deep enough to train models sufficiently large to learn the complex interactions between residues. To deal with this issue, ESM-1V (Meier et al., 2021) trains a zero-shot model on a large set of unaligned sequences to secure a scalable and bias-free training procedure, and TRANCEPTION (Notin et al., 2022) leverages autoregressive pre- dictions and retrieval of homologous sequences at inference.

5 CONCLUSION

Designing directed evolution on proteins especially the deep mutants for functional fitness is of enormous engineering and pharmaceutical importance. However, existing experimental methods are time and economically costing. In this paper, we proposed a lightweight graph neural network model (LGN) for mutant effect prediction on arbitrary numbers of protein mutations by transferring the problem to denoising protein graphs. We assume the proteins in nature are noisy protein graphs and we train our model to recover amino acid types and other important properties (i.e., b-factor, SASA, the spatial position of Cα, and dihedral angles of neighboring N-C atoms) from observed noisy proteins. Translation invariant and rotation equivariant graph convolution layers in our model extract the invariant geometry features of micro frames within and between amino acids and thus grasp rich information for efficiently learning protein function. The model achieves state-of-the-art performance on PDB datasets with significantly fewer computational resources than existing models. Especially in the deep mutant test, our LGN outperforms other zero-shot models by over 20%.

REFERENCES

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

Kenneth Atz, Francesca Grisoni, and Gisbert Schneider. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

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.

Michael S Breen, Carsten Kemena, Peter K Vlasov, Cedric Notredame, and Fyodor A Kondrashov. Epistasis as the primary factor in molecular evolution. *Nature*, 490(7421):535–538, 2012.

Peter B Chi and David A Liberles. Selection on protein structure, interaction, and sequence. *Protein Science*, 25(7):1168–1178, 2016.

Lachlan Coin, Alex Bateman, and Richard Durbin. Enhanced protein domain discovery by using language modeling techniques from speech recognition. *Proceedings of the National Academy of Sciences*, 100(8):4516–4520, 2003.

Yinglu Cui, Yanchun Chen, Xinyue Liu, Saijun Dong, Yu'e Tian, Yuxin Qiao, Ruchira Mitra, Jing Han, Chunli Li, Xu Han, et al. Computational redesign of a petase for plastic biodegradation under ambient condition by the grape strategy. *ACS Catalysis*, 11(3):1340–1350, 2021.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Wang Yu, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. Prottrans: Towards cracking the language of lifes code through self-supervised deep learning and high performance computing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021. doi: 10.1109/TPAMI.2021.3095381.

Narayanan Eswar, David Eramian, Ben Webb, Min-Yi Shen, and Andrej Sali. Protein structure modeling with modeller. In *Structural proteomics*, pp. 145–159. Springer, 2008.

Jonathan Frazer, Pascal Notin, Mafalda Dias, Aidan Gomez, Joseph K Min, Kelly Brock, Yarin Gal, and Debora S Marks. Disease variant prediction with deep generative models of evolutionary data. *Nature*, 599(7883):91–95, 2021.

Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi S Jaakkola, and Andreas Krause. Independent se (3)-equivariant models for end-to-end rigid protein docking. In *International Conference on Learning Representations*, 2021.

Jiaqi Han, Yu Rong, Tingyang Xu, and Wenbing Huang. Geometrically equivariant graph neural networks: A survey. *arXiv preprint arXiv:2202.07230*, 2022.

Rhys Heffernan, Kuldip Paliwal, James Lyons, Abdollah Dehzangi, Alok Sharma, Jihua Wang, Abdul Sattar, Yuedong Yang, and Yaoqi Zhou. Improving prediction of secondary structure, local backbone angles and solvent accessible surface area of proteins by iterative deep learning. *Scientific reports*, 5(1):1–11, 2015.

Steven Henikoff and Jorja G Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

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. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8946–8970. PMLR, 17–23 Jul 2022.

John Ingraham, Vikas Garg, Regina Barzilay, and Tommi Jaakkola. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2020.

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.

Diederik P Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representation (ICLR)*, 2015.

Ben Lehner. Molecular mechanisms of epistasis within and between genes. *Trends in Genetics*, 27 (8):323–331, 2011.

Jiahan Li, Shitong Luo, Congyue Deng, Chaoran Cheng, Jiaqi Guan, Leonidas Guibas, Jian Peng, and Jianzhu Ma. Directed weight neural networks for protein structure representation learning. *arXiv preprint arXiv:2201.13299*, 2022.

Yufeng Liu, Lu Zhang, Weilun Wang, Min Zhu, Chenchen Wang, Fudong Li, Jiahai Zhang, Houqiang Li, Quan Chen, and Haiyan Liu. Rotamer-free protein sequence design based on deep learning and self-consistency. *Nature Computational Science*, 2022.

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.

Yunan Luo, Guangde Jiang, Tianhao Yu, Yang Liu, Lam Vo, Hantian Ding, Yufeng Su, Wesley Wei Qian, Huimin Zhao, and Jian Peng. Ecnet is an evolutionary context-integrated deep learning framework for protein engineering. *Nature communications*, 12(1):1–14, 2021.

Wenjian Ma, Shugang Zhang, Zhen Li, Mingjian Jiang, Shuang Wang, Weigang Lu, Xiangpeng Bi, Huasen Jiang, Henggui Zhang, and Zhiqiang Wei. Enhancing protein function prediction performance by utilizing alphafold-predicted protein structures. *Journal of Chemical Information and Modeling*, 2022.

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. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29287–29303, 2021.

Erik Nijkamp, Jeffrey Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. Progen2: exploring the boundaries of protein language models. *arXiv preprint arXiv:2206.13517*, 2022.

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 Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16990–17017. PMLR, 17–23 Jul 2022.

Dan Ofer, Nadav Brandes, and Michal Linial. The language of proteins: Nlp, machine learning & protein sequences. *Computational and Structural Biotechnology Journal*, 19:1750–1758, 2021.

CA Orengo, AD Michie, S Jones, DT Jones, MB Swindells, and JM Thornton. Cath – a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1109, 1997. ISSN 0969-2126. doi: https://doi.org/10.1016/S0969-2126(97)00260-8.

Fernanda Pinheiro, Omar Warsi, Dan I Andersson, and Michael Lässig. Metabolic fitness landscapes predict the evolution of antibiotic resistance. *Nature Ecology & Evolution*, 5(5):677–687, 2021.

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, 2021.

Adam J Riesselman, John B Ingraham, and Debora S Marks. Deep generative models of genetic variation capture the effects of mutations. *Nature methods*, 15(10):816–822, 2018.

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.

Gabriel J Rocklin, Tamuka M Chidyausiku, Inna Goreshnik, Alex Ford, Scott Houliston, Alexander Lemak, Lauren Carter, Rashmi Ravichandran, Vikram K Mulligan, Aaron Chevalier, et al. Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science*, 357 (6347):168–175, 2017.

Nathan J Rollins, Kelly P Brock, Frank J Poelwijk, Michael A Stiffler, Nicholas P Gauthier, Chris Sander, and Debora S Marks. Inferring protein 3d structure from deep mutation scans. *Nature genetics*, 51(7):1170–1176, 2019.

Karen S Sarkisyan, Dmitry A Bolotin, Margarita V Meer, Dinara R Usmanova, Alexander S Mishin, George V Sharonov, Dmitry N Ivankov, Nina G Bozhanova, Mikhail S Baranov, Onuralp Soylemez, et al. Local fitness landscape of the green fluorescent protein. *Nature*, 533(7603):397–401, 2016.

Rin Sato and Takashi Ishida. Protein model accuracy estimation based on local structure quality assessment using 3d convolutional neural network. *PloS one*, 14(9):e0221347, 2019.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Sisi Shan, Shitong Luo, Ziqing Yang, Junxian Hong, Yufeng Su, Fan Ding, Lili Fu, Chenyu Li, Peng Chen, Jianzhu Ma, et al. Deep learning guided optimization of human antibody against sars-cov-2 variants with broad neutralization. *Proceedings of the National Academy of Sciences*, 119(11): e2122954119, 2022.

Jung-Eun Shin, Adam J Riesselman, Aaron W Kollasch, Conor McMahon, Elana Simon, Chris Sander, Aashish Manglik, Andrew C Kruse, and Debora S Marks. Protein design and variant prediction using autoregressive generative models. *Nature communications*, 12(1):1–11, 2021.

Vignesh Ram Somnath, Charlotte Bunne, and Andreas Krause. Multi-scale representation learning on proteins. *Advances in Neural Information Processing Systems*, 34:25244–25255, 2021.

Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In *International Conference on Machine Learning*, pp. 20503–20521. PMLR, 2022.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Dawn GL Thean, Hoi Yee Chu, John HC Fong, Becky KC Chan, Peng Zhou, Cynthia Kwok, Yee Man Chan, Silvia YL Mak, Gigi CG Choi, Joshua WK Ho, et al. Machine learning-coupled combinatorial mutagenesis enables resource-efficient engineering of crispr-cas9 genome editor activities. *Nature Communications*, 13(1):1–14, 2022.

Andrea D Thompson, Amanda Dugan, Jason E Gestwicki, and Anna K Mapp. Fine-tuning multiprotein complexes using small molecules. *ACS chemical biology*, 7(8):1311–1320, 2012.

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

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Bruce J Wittmann, Kadina E Johnston, Zachary Wu, and Frances H Arnold. Advances in machine learning for directed evolution. *Current opinion in structural biology*, 69:11–18, 2021.

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

Sheheryar Zaidi, Michael Schaarschmidt, James Martens, Hyunjik Kim, Yee Whye Teh, Alvaro Sanchez-Gonzalez, Peter Battaglia, Razvan Pascanu, and Jonathan Godwin. Pre-training via denoising for molecular property prediction. *arXiv:2206.00133*, 2022.

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022.