Generating and Harnessing Learned Embeddings for Protein Design

Sanaa Mansoor

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2023

Reading Committee:

David Baker, Chair

Frank DiMaio

Phil Bradley

Program Authorized to Offer Degree:

Molecular Engineering

University of Washington

Abstract

Generating and Harnessing Learned Embeddings for Protein Design

Sanaa Mansoor

Chair of the Supervisory Committee:
Dr. David Baker
Biochemistry

The structure and function of proteins are encoded by their amino acid sequences. The field of protein design aims to uncover the fundamental connection between protein sequence, structure, and function to design novel proteins with important applications in fields such as medicine, biotechnology, and materials science. The complex relationship between protein sequence, structure, and function makes protein design a challenging task. In recent years, learned embeddings have emerged as a powerful tool to help deconvolute this relationship. Learned embeddings can convert high-dimensional protein data, such as protein sequences and structures, into small vectors of biologically relevant information. By capturing all the essential features of a protein in a compact form, embeddings enable the use of machine learning techniques for protein design. My PhD research has focused on generating meaningful learned embeddings of proteins and then harnessing them for various downstream predictions. For studying protein ensembles

and protein structure refinement, I developed embeddings through training generative models on two-dimensional structural data, followed by three-dimensional structural modeling. By incorporating sequence information, a joint representation of protein sequence and structure was developed for predicting the effects of single mutations on protein thermal stability. Finally, following the development and success of an accurate structure prediction model, RoseTTAFold, the embeddings learned from this model were used for “zero-shot” or unsupervised prediction of the effect of point mutations on protein stability and function. These successes demonstrate the importance of using learned protein embeddings for protein design and highlight the need for further research in this area to facilitate the creation of novel proteins with desired properties.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to express my sincerest gratitude to everyone who has supported me during my PhD. First, my supervisor, David Baker, for his invaluable guidance, insightful feedback, and tremendous support throughout this journey. I would like to thank Minkyung Baek for being a constant support, and an exceptional mentor to me for my entire PhD. She has set an amazing example for me, and I thank her for her guidance in every step of this journey. This work would not have been possible without her.

One of the highlights of my PhD was doing an internship with Dr. Eric Horvitz at Microsoft Research, which proved to be such an enriching experience for me. Eric was an incredible mentor and gave me constructive feedback at every step of the project. I would also like to give a massive thanks to my other mentors in the lab, specifically, Hahnbeom Park and Doug Tischer. Thank you for fielding all my questions over the years and for teaching me invaluable skills in machine learning and biochemistry. I would also like to thank the administrative staff at the IPD, especially Luki Goldschmidt for keeping the digs up and running for the entire lab to (mis)use.

I would also like to thank my colleagues in the lab for contributing to valuable discussions around my projects in lab: Justas Dauparas, Ivan Anishchenko, David Juergens, Jue Wang, Gyu-Rie Lee and Sergey Ovchinikov among others. You have all been inspirational scientists and role models for me to follow and I thank you for all your advice over the years.

I would like to thank my family for their continued support over the years and for urging me to take long walks every day. Thank you for also reminding me of how far I have come when

I lose sight of my progress. I would like to thank my brother, Hamid Mansoor, for his endless support and for encouraging me to take the CS101 course back in college which sparked my interest and eventually led me to pursue a career in this field.

Finally, I am so grateful to have met such warm and incredible people at the IPD who have become life-long friends. Adam Chazin-Gray for his infinite support, comfort, and humor. Areeb Shaukat and Meerit Said for their everlasting guidance and for being incredible friends-turned-family. I would also like to thank Sidney Lisanza, Ian Humphreys, Sam Pellock, Chad Miller and Basile Wicky for being tremendous friends along the way. Being around you all has always felt like sunshine in rainy Seattle.

Chapter 1. INTRODUCTION

Proteins are the molecular machines that perform the most critical functions in living organisms. They consist of a sequence of amino acids that spontaneously fold into unique three-dimensional structures to carry out biochemical functions. Understanding this sequence-structure-function relationship is integral in designing new proteins that carry out important and prespecified functions. Advancements in computational protein design have led to remarkable developments in new drug therapies [1], [2], biosensors [3], and small molecule binder proteins [4]. Until recently, most computational methods for understanding and analyzing protein function and dynamics have used either a first principles-based approach involving structural simulations or sequence modeling approaches which identify existing co-evolutionary pressures on proteins.

Rosetta [5], a physics-based protein design software, is guided by an all-atom heuristic energy function that estimates the free energy of a given protein conformation. The energy function is made up of hydrogen bonding, ionic interactions, pairwise inter-atomic terms describing the Van der Waals interactions, as well as solvation and statistical terms. The Rosetta modeling suite still has considerable success in designing proteins since the energy function continues to be refined and reparametrized over the years. Rosetta draws on our foundational understanding of the physics behind protein design and folding. However, it and similar first-principles-based approaches are computationally expensive and require expert domain knowledge to be set up properly.

Another example of a physics-based approach to study protein structure, function, and dynamics is Molecular Dynamics (MD) simulation. MD software, such as GROMACS [6], have been used to simulate the natural motions of proteins and other biomolecules in an all-atom setting. These simulations can capture multiple conformational states that a protein can adopt. Capturing these different conformations is valuable in understanding the functional mechanism of a protein and performing fine-grained structure prediction. This approach, like Rosetta, also requires extensive computational time and expert domain knowledge to set up and interpret.

Statistical sequence modeling has long been used to study protein structure and function. Through this method, conserved regions and motifs of protein sequences are identified which implies conserved function. Sequence based algorithms have used *k*-mer counts, calculated amino acid composition, and predicted secondary structure [7]. Previous work has also been done in using the covariation between amino acids at pairs of positions in the sequence (co-evolution) to predict the three-dimensional protein structure [8]. However, these methods fail to make use of the growing databases of sequence, structure, and functional information deposited online.

Deep-learning methods that are used to predict three-dimensional protein structures from multiple sequence alignments (MSAs) have gained a lot of attention recently [9], [10] because of their near experimental-level accuracy. The use of these models can be extended and adapted for protein design and protein function prediction through activation maximization and unsupervised techniques [11], [12]. AlphaFold and RoseTTAFold are trained on MSAs that have many protein

sequences that are similar enough to be aligned and diverse enough to contain distant coevolutionary information. These models learn on the MSA and structural template information to form an embedding or representation of proteins that is used for the final three-dimensional structure prediction output.

Following the success of large-scale models in the field of Natural Language Processing (NLP) [13]–[15], active research is ongoing in developing deep-learning models for protein representation learning. Raw protein data is transformed into vector representations with the assumption that the functional information is encoded in the input features. The main objective of representation learning is to preserve the semantic similarity between the data points as a function of distance in the vector embedding space. High dimensional data such as protein distance maps or protein structural domains can be converted to low-dimensional representations using methods such as Principal Component Analysis (PCA) [16] or t-SNE [17] . This learned low-dimensional representation can then be exploited to navigate search and sampling of specific features desired in the output of the model. Efficiently limiting the search space of protein data is a hard problem due to the immense size of the conformational states of a single protein. Recent efforts have been made into employing deep learning-based methods for this problem.

To learn a meaningful and low-dimensional representation of the three-dimensional structure of a protein, it can be represented as two-dimensional pairwise distance map between all backbone atoms. This distance map was used to train a Generative Adversarial Network (GAN) to generate fixed-length full-atom protein backbones through a learned embedding [18]. Three-dimensional coordinates have been used as input to a Graph Convolutional Network

(GCN) to generate an embedding for use in protein function prediction [19]. However, the relatively small number of structurally validated proteins, as compared to the expansive set of sequences available, motivates the focus to date on protein sequences for generating useful embeddings and representations for downstream tasks such as structure and function prediction.

Sequence embeddings created via semi-supervised training have demonstrated strong performance over a broad range of biologically relevant downstream tasks [20], [21]. Through the semi-supervised training objective, these models capture long range dependencies between unrelated families of proteins. Early contextualized sequence embedding models included ELMO which uses representations from the hidden states of bi-directional LSTMs [22]. This model was then applied to supervised-training tasks such as prediction of subcellular localization or structure prediction [23]. More recently, semi-supervised models trained on the huge amounts of sequence data available have achieved state-of-the-art performance on a wide variety of benchmark datasets such as protein contact map prediction and function prediction [24], [25]. This early work on protein language models demonstrated the power and potential that these methods would have for protein design.

These language models also learn very informative embeddings that have been used to better capture coevolutionary information and can link sequence to function through transfer learning [13], [15], [24]–[26]. These models continue to increase in accuracy with more compute time and data. This motivates the need to add strong biological priors to aid in making these models more data-efficient, possibly through addition of structural features.

Overall, this introduction outlines and motivates the need for better and more interpretable learned protein embeddings for downstream prediction tasks. Here, I will present my efforts to generate, and harness learned protein embeddings using generative models, semi-supervised and unsupervised approaches. First, I will describe my work using a two-dimensional structure-based generative model to study and explore protein structure refinement and K-Ras ensemble generation. I also present a semi-supervised approach to building a joint embedding on both protein sequence and structure that I used for supervised protein thermal stability prediction. Finally, I used an already existing model, RoseTTAFold, for "zero-shot" or unsupervised mutation effect prediction on protein stability and function.

Chapter 2. HARNESSING JOINT REPRESENTATIONS OF SEQUENCE AND STRUCTURE FOR SUPERVISED THERMAL STABILITY PREDICTION

This section contains content previously published as: Mansoor, S., Baek, M., Madan, U., & Horvitz, E. (2021). Toward More General Embeddings for Protein Design: Harnessing Joint Representations of Sequence and Structure. *BioRxiv*, 2021.09.01.458592. DOI: https://doi.org/10.1101/2021.09.01.458592

2.1 ABSTRACT

Protein embeddings learned from aligned sequences have been leveraged in a wide array of tasks in protein understanding and engineering. The sequence embeddings are generated through semi supervised training on millions of sequences with deep neural models defined with hundreds of millions of parameters, and they continue to increase in performance on target tasks with increasing complexity. For this project, we chose to use a more data-efficient approach to encode protein information through joint training on protein sequence and structure in a semi-supervised manner. We show that the method can encode both types of information to form a rich embedding space which can be used for downstream prediction tasks. We show that the incorporation of rich structural information into the context under consideration boosts the performance of the model by predicting the effects of single mutations. We attribute increases in accuracy to the value of leveraging proximity within the enriched representation to identify sequentially and spatially close residues that would be affected by the mutation, using experimentally validated or predicted structures.

2.2 INTRODUCTION

Proteins consist of a linear chain of amino acids that fold to form a three-dimensional structure to carry out vital processes all living organisms. The sequence to structure to function relationship is integral in understanding how to design proteins for specific functions. Most methods that seek to understand the complex sequence to structure to function relationship take either a first principles approach with structural simulations [5] or leverage sequence embeddings trained through an adaptation of semi-supervised machine learning methods used to construct large-scale neural models developed for natural language processing (NLP) tasks [13], [15], [24], [27].

Sequence embeddings created via semi-supervised training have demonstrated strong performance over a broad range of biologically relevant downstream tasks, particularly in the realm of protein engineering [20], [21]. Through the semi-supervised training objective, these models capture long-range dependencies between unrelated families of proteins. Protein structure is more informative for predicting function than sequence [28]. However, because of the cost and time needed to experimentally validate protein structures, there is a relatively small database of them publicly available. In contrast, as the cost of sequencing continues to decrease, the amount of sequence data generated grows and thus motivating most machine learning research to be focused on using this growing data for protein function prediction tasks. Language models centered on semi-supervised training on sequences continue to grow and become increasingly accurate with larger architectures, more compute time, and more data. In this project, I explored an alternate approach, promising greater data-efficiency via the explicit

introduction of structural information. This can be viewed as providing models with strong biological priors.

In this chapter, I will present on a deep learning framework that explicitly encodes structural information into pre-trained sequence embeddings to form a more informative and generalizable embedding of proteins. Use of the approach led to an improvement in masked sequence and structure recovery. Additionally, this project showed that the joint training with sequence and structure information improved predictions of the effect of single mutations on thermal stability. With recent considerable improvement in protein structure prediction techniques [9], [10], this multi-task trained model was able to accurately predict protein properties from structural models. getting around the need of experimentally validated structures.

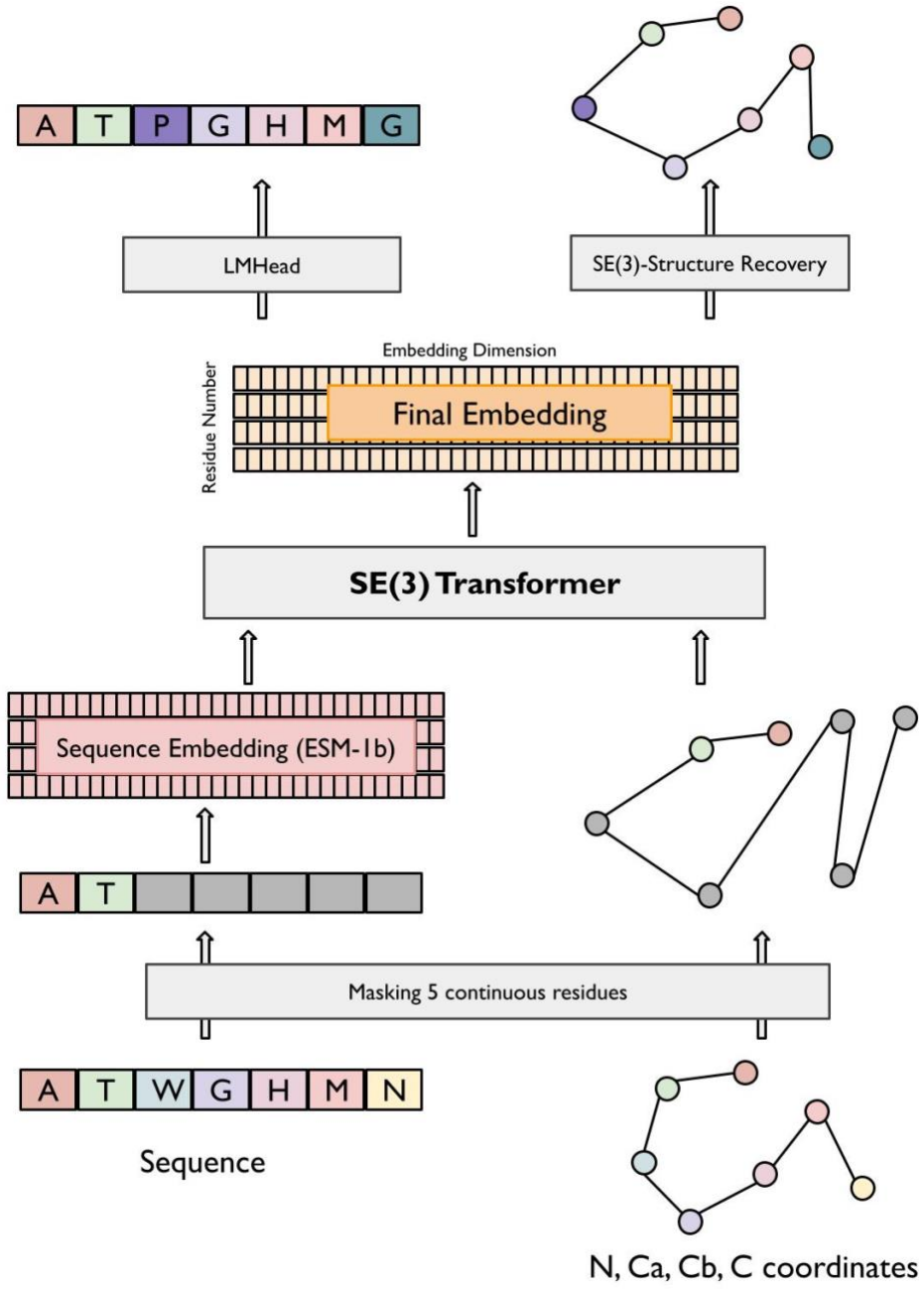


Figure 2-1. Model architecture for generating joint embeddings. Same sequence and structure regions are first masked. Both 1D and 2D features are generated from passing the masked sequence through the pre-trained ESM-1b model. Masked structure and sequence representation are passed as input to a SE(3) transformer, which is trained to output a 128 dimensional embedding space. The embedding space is used to predict the masked sequence and structure regions.

2.3 RESULTS

2.3.1 *Masked structure and sequence recovery*

We evaluated the performance of the model for recovering the masked structure on the validation set. The same masking scheme was employed, where continuous regions of 5 residues were masked to make up a total of 15% masked residues. Figure **2-2**.A shows that all the masked validation points were corrected when passed through the model. On average, the initial validation samples were perturbed by 5 angstroms, and post-correction, the samples had an RMSD of about 2 angstroms from their respective native protein. To evaluate whether addition of structure leads to an improvement in sequence recovery, we compared our results with the sequence-only baseline (ESM-1b). The masking for this task was adopted from ESM-1b, where random tokens were chosen to make up a total of 15% tokens masked. Figure **2-2**.B shows the accuracy of the predicted masked amino acid for both the joint embedding model and baseline. Due to the added structural context, the model was able to recover the masked sequence tokens to a higher accuracy than the sequence-only baseline, for most validation points. Average baseline sequence recovery was 0.06 (6%), whereas for the joint embedding model, we were able to achieve an accuracy of 0.13 (13%) on the same test set.

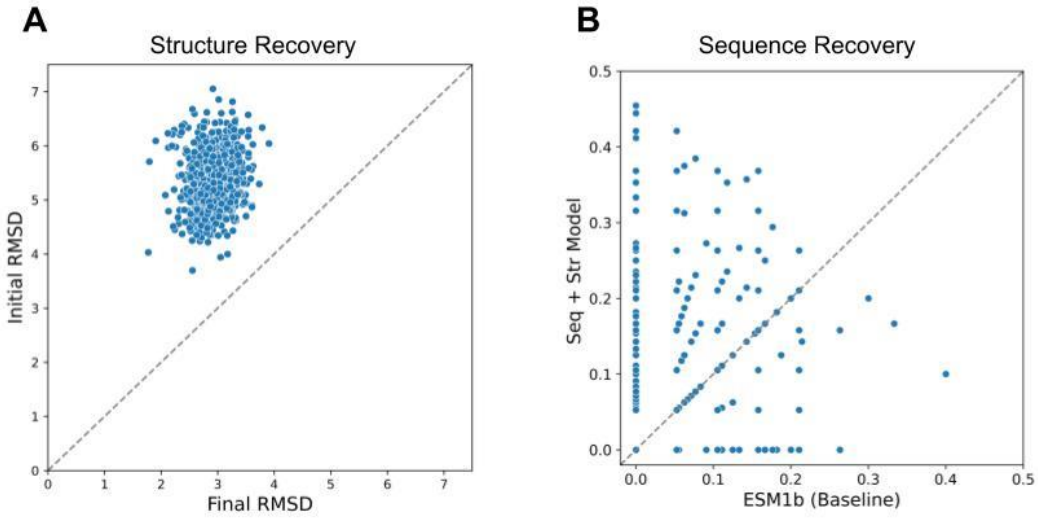


Figure 2-2. Structure and sequence recovery of joint embedding model. Scatterplot of RMSD of masked (perturbed) input structures and RMSD of corresponding corrected structures. (B) Scatterplot comparing sequence recovery of trRosetta validation samples from general embedding model and sequence-only baseline (ESM-1b).

2.3.2 *Predicting the effect of a single mutation on thermal stability*

To predict the effect of a single mutation on thermal stability, we fine-tuned the sequence prediction head (LMHead) for both the sequence-only baseline and the joint embedding model. Since the architecture of the sequence prediction head is the same, this serves as a head-to-head comparison of the value of the enriched information captured in the embedding space and how well equipped it is to predict the effect of a single mutation. The train and test set consisted of mutants from non-overlapping wild-type proteins. Figure **2-3**.A shows the correlation between predicted and measured ΔΔG values for the same test set. The general embedding model has a higher Spearman rank-order correlation coefficient (spearmanr) of its predictions of ΔΔG to the ground truth than the sequence-only baseline, providing evidence that the explicit addition of structure to the embedding space leads to a more informative protein encoding for the prediction of single mutations.

2.3.3 *Evaluation of mutant versus wild-type embedding space*

To pursue insights about the operation of the model in predicting the effect of a single mutation, we compared the difference in the embedding space of the wild-type to the mutant protein for both the sequence-only baseline and the joint embedding model (Figure **2-3**.B and Figure S2-1). When plotted on the same scale, we can see that the structure-aware embedding shows a higher degree of difference between the wild-type and mutant embedding spaces for PDB1FXA. The general embedding model also shows sequentially distant residues being influenced by the mutation. Inspection of the 3D structure of the protein revealed that these distant sites are structurally close. These finding point to the joint embedding being genuinely structure-aware, making it more sensitive to mutation effects than the sequence-only model.

2.3.4 *Single mutation effect predicting using predicted structural models as input*

Computational methods for predicting the effect of mutations can provide great value for protein design. We evaluated the performance of the model on ΔΔG prediction using structural models, rather than experimentally validated structures. We used RoseTTAFold [10] for predicting models of the ProTherm test set for this experiment. On average, the predicted model structures differed about 1.2 angstroms from their respective native structures (Figure S2-2). Figure **2-3**.A (right plot) shows the correlation between predicted and ground truth ΔΔG for predicted models for the test set. This points to the generalizability of the model to withstand perturbations to the input structure and that the model does not require experimentally determined structures to provide high accuracy predictions of the effect of single mutations. The sufficiency of using predicted structures can significantly raise the efficiency of protein design,

with uses, for example, in ranking the efficacy of function of candidates and better triaging them for expensive wet-lab experiments.

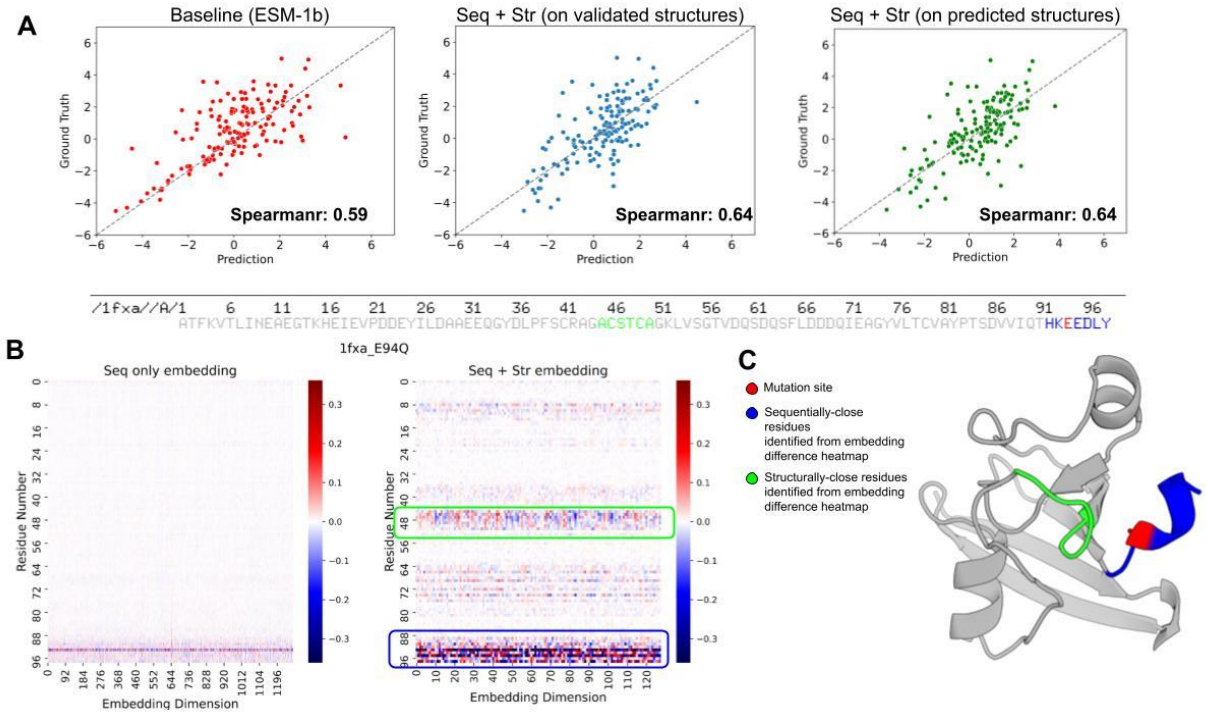


Figure 2-3. Accuracy of prediction of ΔΔG of single mutants and analysis of PDB 1FXA, with single mutation. (A) Scatterplots showing correlation of predicted ΔΔG to experimental values (ground truth) using sequence-only (ESM-1b), joint sequence and structure embedding model using experimentally validated structures and joint sequence and structure embedding model using predicted structural models from RoseTTAFold. (B) Heatmaps showing the difference in mutant and wild-type embeddings generated for single mutation on residue position 94 for PDB 1FXA, using sequence-only and joint sequence and structure embedding models, with regions having the highest difference in embedding values highlighted in blue and green. (C) 3-D structure of PDB 1FXA with the mutation site colored in red. The most-affected regions identified from the embedding space difference between wild-type and mutant are highlighted using the same color schemes.

2.4 DISCUSSION

The exponential increase in protein sequence databases has led to major developments in developing generative and predictive models for protein engineering using deep learning. Validated protein structures, on the other hand, constitute a small fraction of all known biological sequences. Effectively combining structure and sequence information promises to be beneficial for downstream prediction problems where data is limited. With increasing progress in language modeling for proteins, we have seen that these models continue to benefit from more data and computation power. However, the explicit joining of structure and sequence information in embedding models can provide a more data-efficient approach to encoding proteins that can reduce required quantities of training samples and compute power.

The methodology we have taken to develop a more general embedding for proteins combines both structural and evolutionary information in a semi-supervised manner using a SE(3)-equivariant Transformer. Since the model was trained with a multi-task loss on both the masked sequence and structure prediction from the embedding space, equal weight was given to recovering the sequence and structure. Both the masked sequence tokens and masked structure coordinates were recovered accurately from the 128 dimensional embedding space.

The structure-aware embedding trained was able to capture complex relationships between amino-acids in both sequential and structural space. Our model was more sensitive to single-point mutations in the wild-type protein than the sequence-only baseline, suggesting the value of building richer encoding that contains both protein sequence and structure information. We were able to predict the effect that a point mutation would have on thermal stability of a protein to a

higher accuracy than the sequence-only baseline, which provides evidence of the value of adding structural information. Also, when evaluating the difference in the embedding space of the mutant versus the wild-type protein, the general embedding model was able to identify sequentially-far but spatially-close residues that would be affected by the point mutation. Finally, since determining protein structures is expensive and time-consuming, we evaluated the single-mutant effect prediction accuracy of our model using structure models predicted by RoseTTAFold. The mutant effect prediction accuracy remained high even though the structural models were approximate. This points to the generalizability of the method developed and its viable use case for protein design challenges using design models.

We see multiple directions for improvement of the joint use of sequence and structure information for both masked token recovery and downstream prediction tasks. Through multiple iterations with either shared or independent weights, incremental corrections to the structure coordinates would help with masked recovery. Increasing the model size (currently 16 million parameters) through increasing the size of the embedding space would enable the model to store more information from both sequence and structure. In another direction, a promising path to achieving more accurate structure recovery is to revise the approach to predicting corrections for each atom. We predicted corrections for atoms (Ca, C, N, Cb) independently. In an alternate approach, we can treat atom orientation and displacement as interdependent, where all atom corrections are relative to the Ca correction. We see numerous prediction tasks as benefiting from the construction and leveraging of the joint embeddings. Overall, this method points to the viability of encoding sequence and structure to form richer and more informative embeddings of proteins.

2.5 METHODS

2.5.1 *Input sequence embedding and structure information*

We used the trRosetta2 [29] training and validation datasets for our model. The masked sequence input was fed into the ESM-1b model [24] , loaded with pre-trained weights. By doing so, we are enabling the generation of already learned sequence representations to be used as input for our model. We used the sequence embedding from the last layer (1D feature) along with the attention maps from each of the 33 layers (2D features) from ESM-1b. For structure, we only considered N, Ca, C, and Cb atoms (virtual Cb atoms for GLY). Proteins longer than 128 residues were cropped to fit the memory of a single NVIDIA V100 (16GB).

2.5.2 *Training objective and details*

We employed a semi-supervised training task, where we masked out random continuous regions of 5 residues to make up a 15% total mask of the input protein. Same regions of the sequence and structure were masked. For masking, we used a special mask token for sequence, and perturbed the structure coordinates by at least 5 angstroms, to make the masked structure sufficiently different from native, with additional noise from a standard normal distribution. The sequence tokens were recovered by minimizing cross entropy loss between the model's predictions of the masked amino acid and the true amino acid. The structural loss was defined as the batch average of the mean squared error over the model's predictions of the masked atom coordinates and the true coordinates over the 4 types of atoms considered. The total loss was an addition of a sequence loss and a structural loss weighted equally. The model was optimized using Adam ($\beta 1 = 0.9$, $\beta 2 = 0.999$) with learning rate $1e{-}3$, with an effective batch size of 128.

2.5.3 *Model architecture*

We used a SE(3) equivariant-transformer [30] , a graph neural network, for processing the masked sequence and structure features (Figure **2-1**). The 1D (sequence embedding) and 2D (attention maps) ESM-1b features were linearly projected down to 32 dimensions for noise reduction. For constructing the graph, each node represented a residue of the masked structure where we considered a neighborhood of the 16 closest neighbors based on perturbed Ca distances. Node features included the ESM-1b sequence embedding for each residue and the displacement vector of each atom to Ca (NCa, CCa, CbCa). Edge features included the ESM-1b 2D attention map and the inter-residue orientations [31] (Figure S2-3). The edge connecting residues 1 and 2 would contain the attention map values for the two residues and the inter-residue orientations represented by 3 dihedral (ψ, $\theta12$, $\theta21$) and 2 planar angles ($\varphi12$, $\varphi21$). This graph was fed into a 5-layered SE(3)-equivariant transformer, the output of which was a 128 dimensional hidden embedding of the input features. The embedding space was fed separately into a masked sequence prediction head (LMHead) and another SE(3)-equivariant transformer for masked coordinate prediction (SE3-Structure Recovery or SE3-SR) (Figure **2-1**). The architecture of the masked sequence prediction head is adopted from the LMHead of ESM-1b model [24], a 2 dense layered network with layer normalization. For the SE3-SR top model, we used a 3-layered SE(3)-transformer where the node features included the jointly trained embedding from the first SE(3)-transformer (128D), along with the displacement vector of each atom to Ca. The edge features included the inter-residue orientations only.

2.5.4 *Fine-tuning for single mutant effect prediction*

Following semi-supervised training of the model, we fine-tuned our model for the task of predicting the effect of single mutations on thermal stability (ΔΔG values). A subset of the ProTherm dataset [32] was used which consisted of 1042 mutants from 126 wild-type proteins. Training and test sets consisted of non-overlapping wild-types and their mutants. The training and test set were split 80:20. The fine-tuning task was adopted from the published ESM-1b model, where the mutant position was masked, and the predicted value is the difference between the log probabilities of the mutant amino acid and the wild-type amino acid. Only the sequence prediction top model (LMHead) from the embedding space was re-trained for both the joint embedding model and the baseline (ESM-1b). As mentioned previously, the sequence prediction head for both the joint embedding model and the sequence-only baseline has the same architecture.

2.6 SUPPLEMENTARY FIGURES

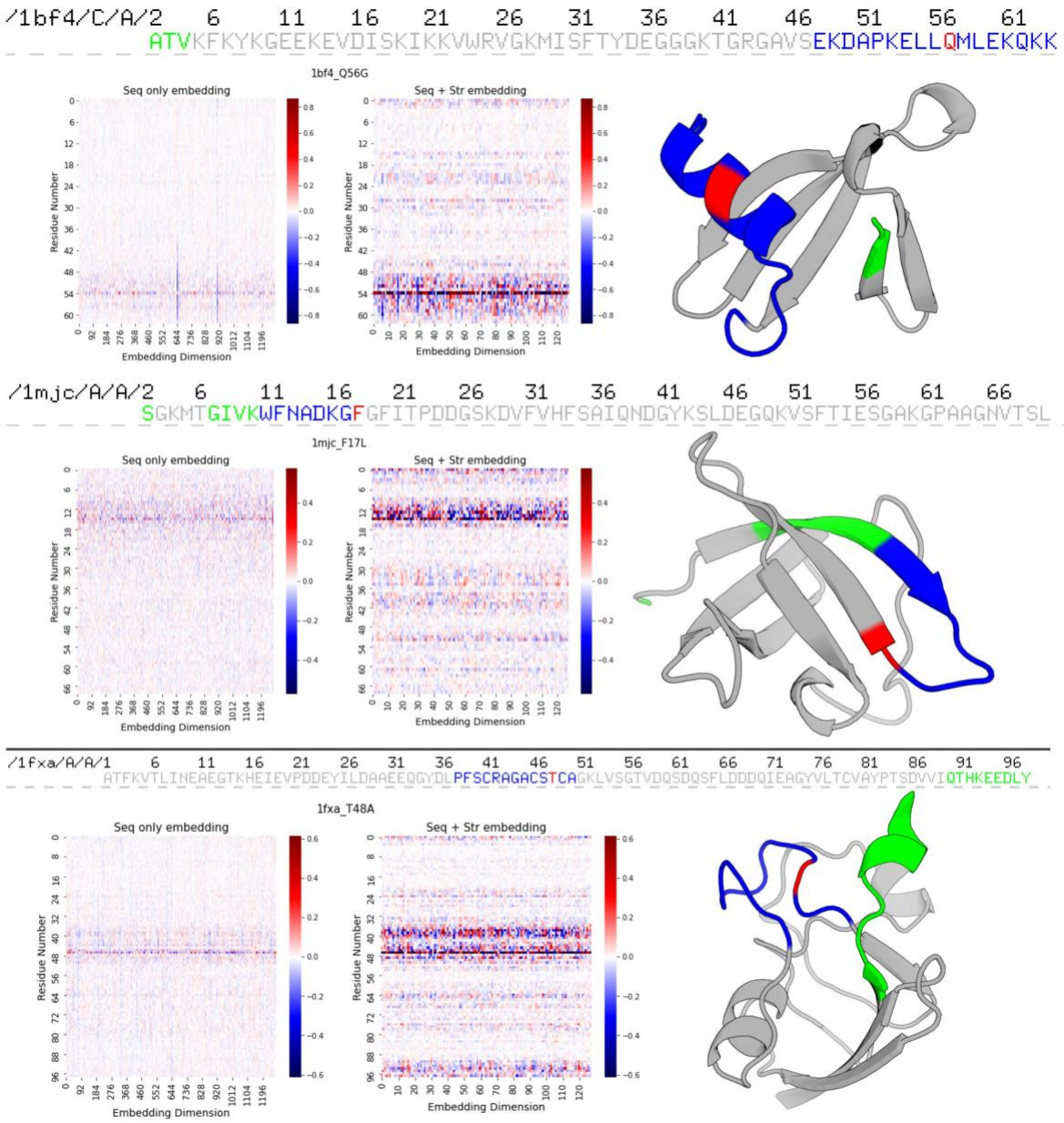


Figure S2-1. Additional single mutation analysis of PDBs 5AZU and 1PGA. Heatmaps showing the difference in mutant and wild-type embeddings generated for single mutation using sequence-only and joint sequence and structure embedding models, with regions having the highest difference in embedding values highlighted in blue and green. 3-D structures of considered proteins with the mutation site colored in red. The most-affected regions identified from the embedding space difference between wild-type and mutant are highlighted using the same color schemes.

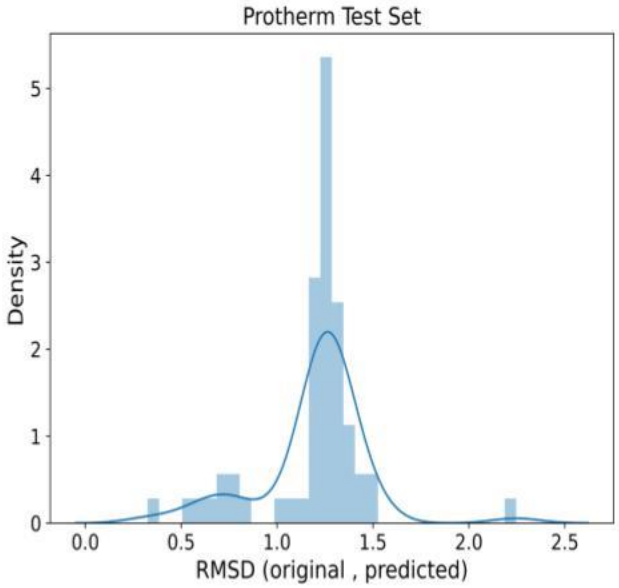


Figure S2-2. RMSD of RoseTTAFold predictions on Protherm test set. Distribution of RMSD of experimentally validated structures to their respective RoseTTAFold structure prediction models.

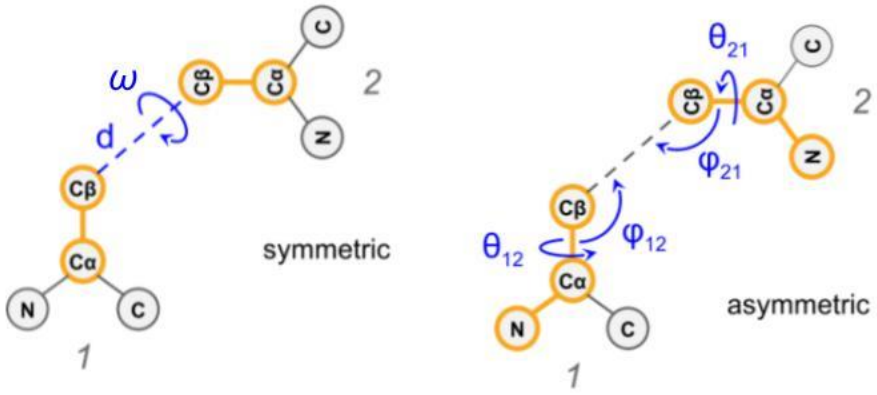


Figure S2-3. Inter-residue orientations used on edges for graph construction. Schematic depiction of inter-residue orientations (3 dihedral and 2 planar angles) used on edges connecting two residues in constructing the graph for input to SE(3) transformer (from Yang et al. (2020) [31]).

Chapter 3. ZERO-SHOT MUTATION EFFECT PREDICTION ON PROTEIN STABILITY AND FUNCTION USING ROSETTAFOLD

This section contains content previously published as: Mansoor, S., Baek, M., Juergens, D., Watson, J. L., & Baker, D. (2022). Accurate Mutation Effect Prediction using RoseTTAFold. *BioRxiv*, 2022.11.04.515218. DOI: https://doi.org/10.1101/2022.11.04.515218

3.1 ABSTRACT

Predicting the effects of mutations on protein function and stability is an outstanding challenge. Here, we assess the performance of a variant of RoseTTAFold jointly trained for sequence and structure recovery, RF_{joint}, for mutation effect prediction. Without any further training, we achieve state-of-the-art accuracy in predicting mutation effects for a set of diverse protein families using RF_{joint}. Thus, although the architecture of RF_{joint} was developed to address the protein design problem of scaffolding functional motifs, RF_{joint} acquired an understanding of the mutational landscapes of proteins during model training that is equivalent to that of recently developed large protein language models. The ability to simultaneously reason over protein structure and sequence could enable even more precise mutation effect predictions following supervised training on the task.

3.2 INTRODUCTION

Accurate prediction of single point mutation effects using sequence information alone would help relate observed sequence polymorphisms to human disease [33], [34] and contribute to the design of proteins with higher functional activities. Deep learning methods have recently shown considerable promise for mutation effect prediction. DeepSequence [20] a probabilistic

model for sequence families, obtained high accuracy in mutation effect prediction using latent variables for capturing higher-order interactions between residues in proteins through training on multiple sequence alignments (MSAs) for the target protein of interest. Large protein language models trained on MSAs (MSA Transformer) [35] or single sequences [21] also perform well at mutation effect prediction using an unsupervised or zero-shot approach. These language models have the advantage over DeepSequence of not requiring specific training on the protein family of interest.

RoseTTAFold was originally developed for protein structure prediction [10] and a recently developed version, RoseTTAFold Joint (RF_{joint}) was further trained to solve 'inpainting' problems in which substantial portions of both sequence and structure are rebuilt to design novel scaffolds around protein functional motifs [11]. RF_{joint} was trained on a masked MSA token recovery task for sequence prediction. To assess RF_{joint}'s understanding of protein mutational landscapes, we set out to investigate whether it could predict experimental mutational data from published deep mutational scanning (DMS) sets [36] with no further training (i.e., using a "zero-shot" approach). We compared the performance of RoseTTAFold Joint on this task to that of MSA Transformer; both are MSA based methods requiring no further training. Overall, we saw an increase in the average ranking correlation of the prediction effects of single mutations on the set of diverse proteins evaluated using RF_{joint} over MSA Transformer.

3.3 RESULTS

RF_{joint} was evaluated on a set of 38 deep mutational scans curated by Riesselman *et al.* [20]. Each of the mutational scans recorded a different protein function with varying measurements. Each dataset was treated as a separate prediction task, and each variant was scored individually. For each target protein, we generated MSAs using iterative sequence search against the UniClust30 database as described in Baek *et al.* [10] and used it for both RF_{joint} and MSA Transformer predictions. For RF_{joint}, the variants were scored by masking out the mutation site in the query sequence in the MSA, and the MSA token recovery head was used to predict the distribution over the masked position. The predicted effect of the mutation was calculated as the log odds ratio of the mutant amino acid and the wild-type amino acid (Figure **3-1**). The performance on each dataset was assessed based on the spearman correlation of the predictions to the observed experimental values.

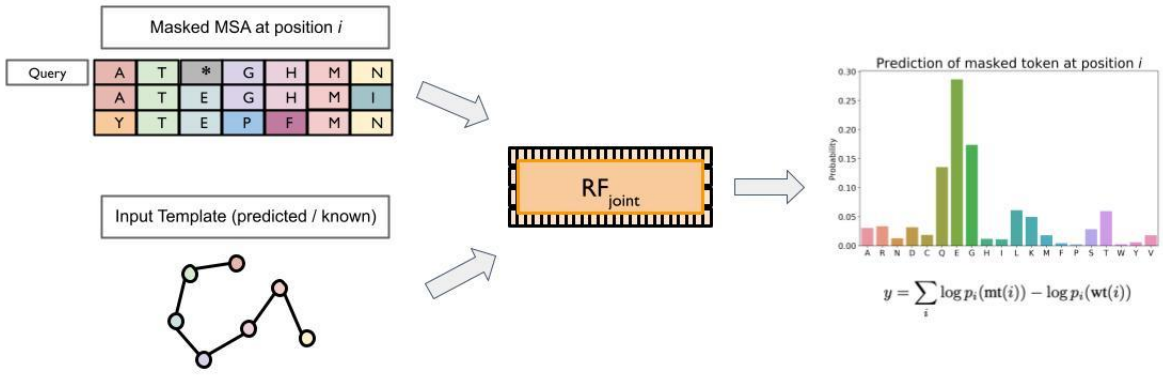


Figure 3-1. Overall pipeline for zero-shot prediction of single mutation effect using RF_{joint}. A MSA is generated and masked at the mutation position in the query sequence, and structural templates are fed into pre-trained RF_{joint}. Using the masked token prediction head, the predicted probability distribution of the 20 amino acids over the mutation site is used to calculate the effect of a mutation as the log odds ratio of the wild-type and mutation amino acid.

We found that RF_{joint} predicts mutational effects considerably better than a baseline calculated as the log odds ratio of the frequency of the mutant amino acid and of the wild-type amino acid in the MSA (Figure **3-2**). RF_{joint} also slightly outperformed MSA Transformer (Figure **3-2**). RF_{joint} has the advantage in principle over the purely sequence based models of also being able to utilize structural template information, but we did not observe a significant improvement with incorporation of template structure information (Figure S3-1; this may be in part because RoseTTAFold generates 3D models from MSA with reasonable accuracy). We also found little dependency of prediction accuracy on MSA depth (Figure S3-2).

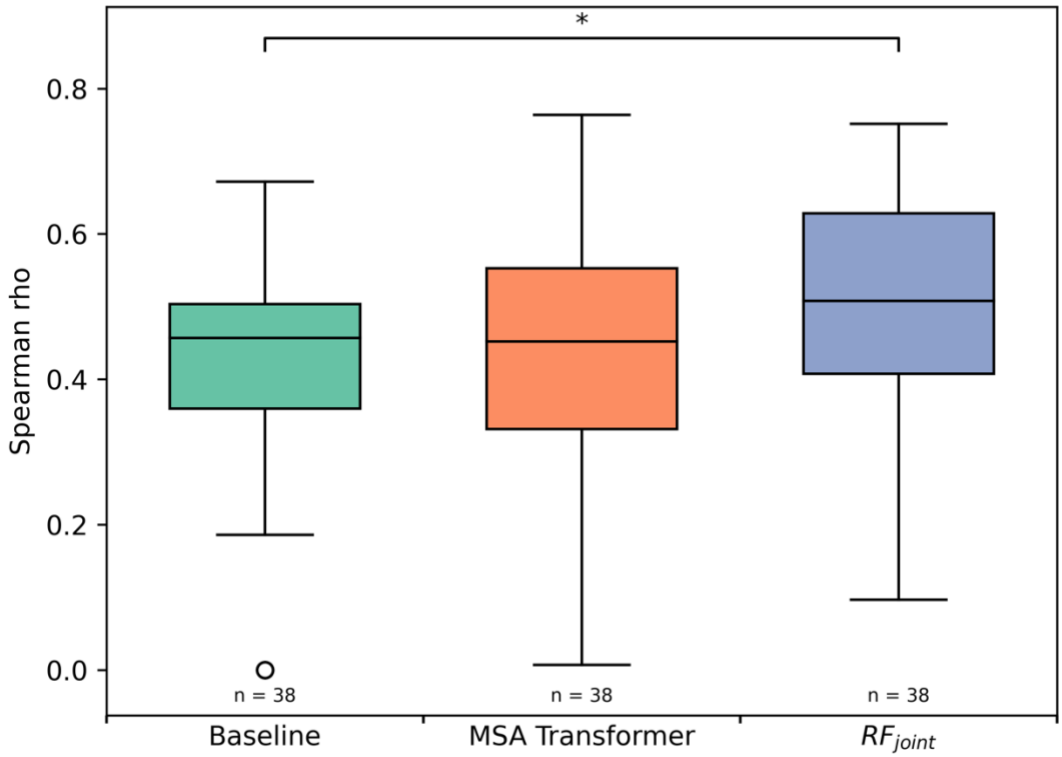


Figure 3-2. Boxplots of spearman rho correlations on deep mutation scanning datasets. Baseline refers to the non-ML MSA baseline. RF_{joint} refers to the model trained on a joint sequence and structure recovery task [11]. Box plots show the median (center line), interquartile range (hinges), and 1.5 times the interquartile range (whiskers); outliers are plotted as individual

points. The average spearman rho correlation is 0.426 for the baseline, 0.430 for MSA Transformer and 0.497 for RF_{joint}.

3.4 DISCUSSION

We find that the RoseTTAFold network, developed originally for structure prediction and then extended to protein design, is also able to predict the effect of single mutations with quite high accuracy. Just as large protein language models, like the MSA Transformer, provide general models of protein sequence, RoseTTAFold Joint may be viewed as a general joint model of protein sequence and structure. With further directed training, it should be possible to further improve performance by better utilizing protein structural information, which can be readily input into RoseTTAFold Joint but not into pure sequence-based models, and by fine-tuning specifically for the mutational effect prediction task. More generally, our results demonstrate that RoseTTAFold Joint has quite a broad understanding of protein mutational landscapes, which will be useful for protein design and other challenges involving inference over both sequence and structure.

3.5 METHODS

3.5.1 *Deep Mutational Scanning (DMS) datasets*

RoseTTAFold was evaluated on a subset of 38 deep mutational scans collected by Riesselman *et al*. [20]. The proteins evaluated perform a wide range of functions and the experimental measures performed are different for each protein. We treat each deep mutational scanning dataset as a separate prediction task. Performance on each task is evaluated by spearman rho

correlations of the calculated (baseline) or predicted (RF_{joint} and MSA Transformer) scores to the experimental values.

3.5.2 *MSA Generation*

The same MSA inputs are used for both RoseTTAFold Joint and MSA Transformer at inference time. The protocol for generating MSAs is adopt, where for each protein, sequences are found by iterative search against UniRef30 [37] and BFD [38] using HHblits [39]. Sequences are then filtered at 90% sequence identity cutoff. The E-value cutoff for sequence search is gradually relaxed (1e-10 to 1e-3) until the generated MSA has at least 2000 sequences with 75% coverage or 5000 sequences with 50% coverage.

3.5.3 *Non-ML Baseline setup*

For establishing the non-ML baseline, we used the input MSA for each protein and calculated the log odds ratio of the frequency of the wild-type amino acid and mutant amino acid for each position (Equation 3-1). All sequences of the input MSA were used in this calculation.

$$y_{baseline} = \log(\ freq_{wt,i} - \ freq_{mt,i}) \quad \text{Equation 3-1}$$

3.5.4 *RF_{joint} inference setup*

We used the published RF_{joint} model [11] in inference mode for the task of single mutation effect prediction. All weights of the model were frozen and no further training was done. Up to 256 sequences were considered from the input MSA of a target protein with an additional 1024 extra sequences passed into the model. All default parameters from RF_{joint} were used and the number

of recycles was set to 1. RoseTTAFold [10] predicted structures for a target protein were used as structural templates for mutation effect prediction. The mutation site of interest was masked in the query sequence of the input MSA and the masked MSA token recovery head was used to predict the probability of all 20 amino acids over that masked position. The predicted effect of a mutation at position *i* was calculated as the log odds ratio of the probability of the wild-type amino acid to the mutant amino acid (Equation 3-2). This scoring is zero-shot i.e. the model requires no further training.

$$y_{RF_{joint}} = \log(\ prob_{wt,i} - \ prob_{mt,i}) \quad \text{Equation 3-2}$$

3.5.5 *MSA Transformer inference setup*

We used the published MSA Transformer [21], [35] loaded with pre-trained weights (annotated as `esm_msa1b_t12_100M_UR50S` on the public ESM github). The default arguments were used, where 400 sequences were randomly sampled from the MSA for inference. We used the masked marginals scoring strategy for scoring mutants from MSA Transformer, which is done by introducing masks at the mutated positions and computing the score for a mutation by considering its probability relative to the wildtype amino acid [21]. This is similar to the setup that we used for predicting the effect of a mutation through RF_{joint} (Equation 3-2).

3.6 SUPPLEMENTARY FIGURES

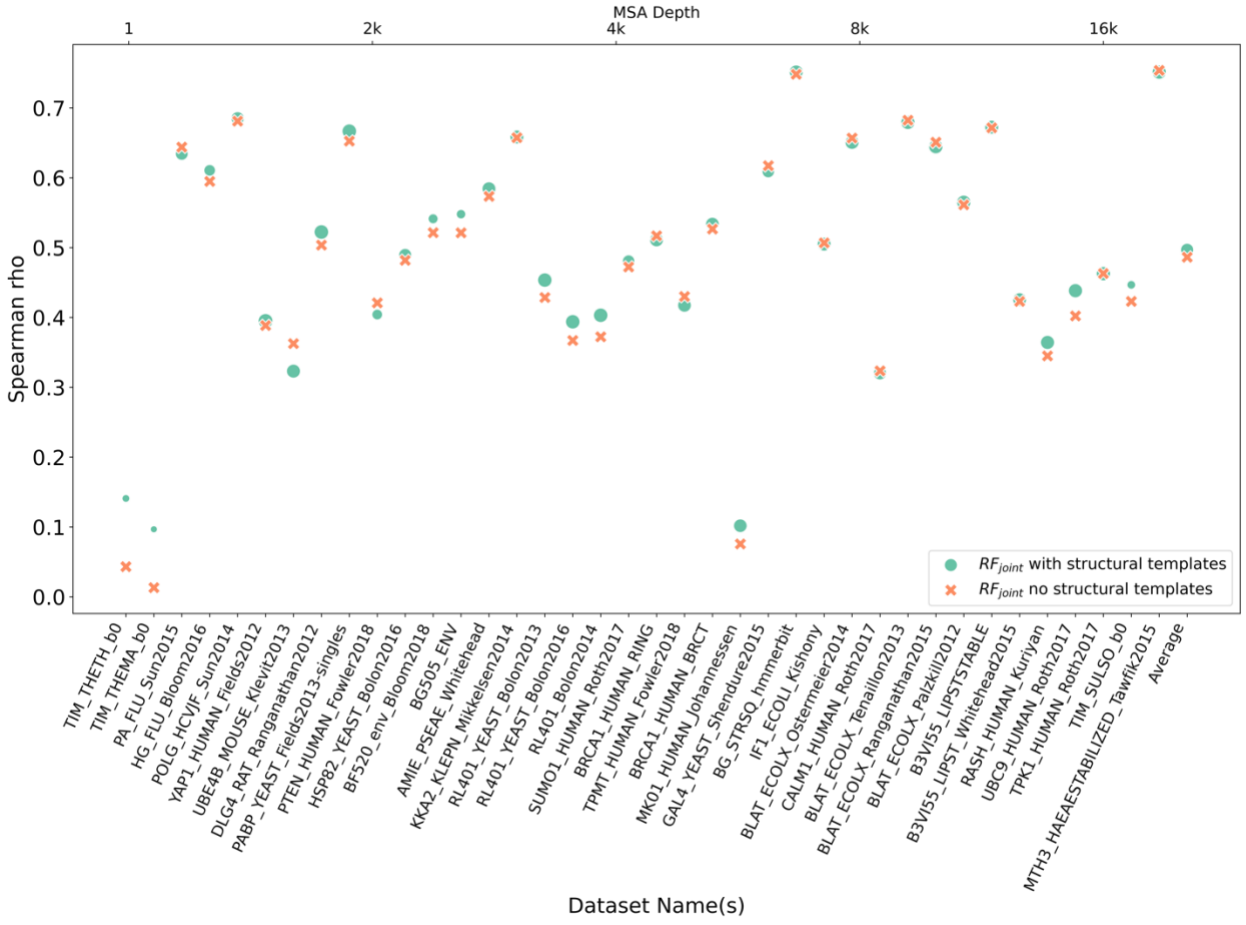


Figure S3-1. Addition of structural templates does not significantly boost performance of RF_{joint}. Comparing the effect of addition of RoseTTAFold-predicted structural templates as input for the task of single mutation effect prediction. Each point corresponds to a different protein and is linearly sized by the pLDDT of the predicted structure template. pLDDT values range between 45 - 99, with most proteins (25 out of 38 proteins) having a pLDDT of above 90. The points are arranged according to increasing MSA depth for RF_{joint} and MSA Transformer, and for reference we provide an approximate depth scale on the top.

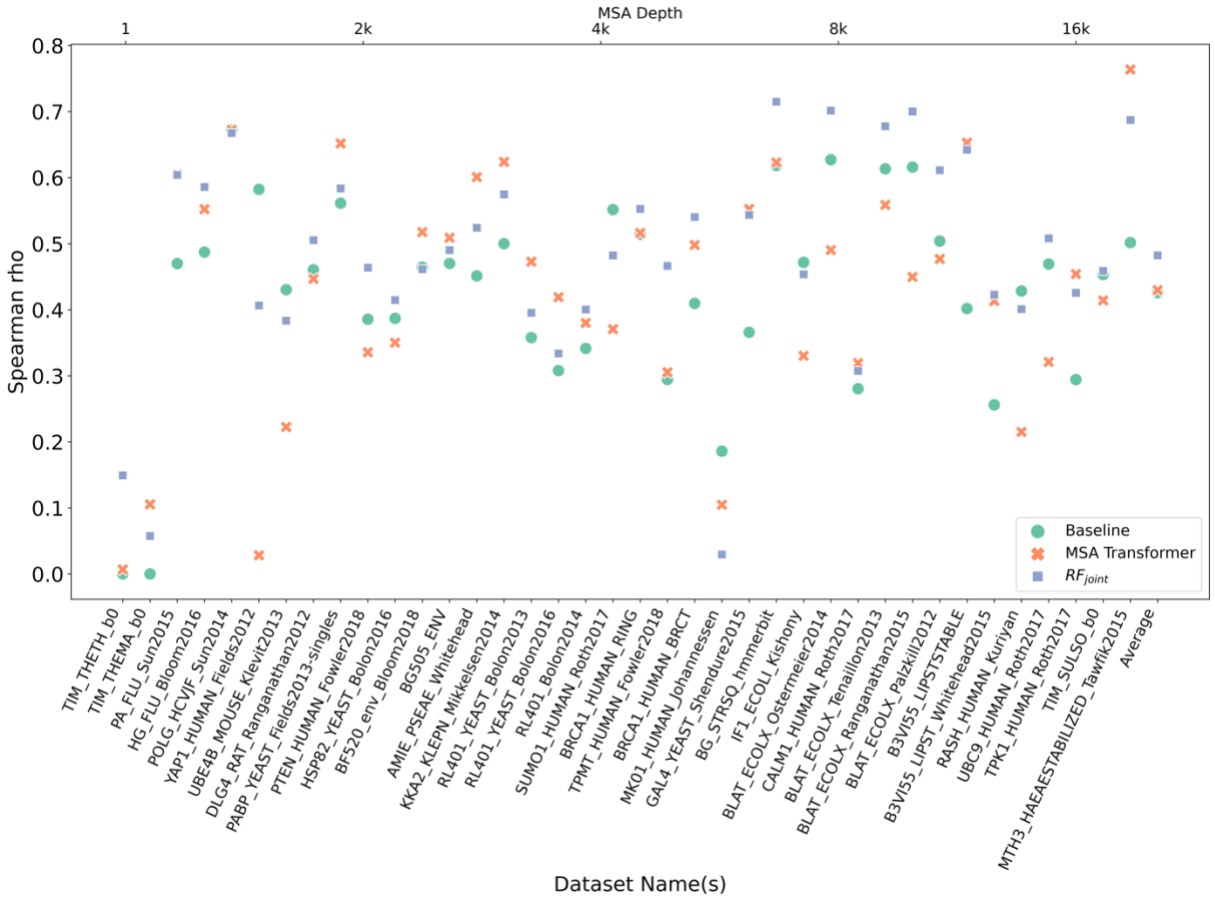


Figure S3-2. Spearman rho correlations for all deep mutational scanning datasets evaluated. Each point corresponds to a different protein. The points are arranged according to increasing MSA depth for RF_{joint} and MSA Transformer, and for reference we provide an approximate depth scale on the top.

Chapter 4. EXPLORATION OF PROTEIN STRUCTURE REFINEMENT IN THE LATENT SPACE OF VARIATIONAL AUTOENCODERS

4.1 ABSTRACT

In this chapter, we explore the use of variational autoencoders (VAE) for reducing the challenge of dimensionality in the protein structure refinement problem. We convert high-dimensional protein structural data to a continuous, low-dimensional representation, and carry out search in this lower dimensional space guided by rapidly computable latent space-based approximations of the Rosetta energy or deep learning-based structure accuracy predictor. We find that predictor guided sampling in the encoded lower dimensional space, followed by decoding to generate full three-dimensional structures, rapidly generates models with higher predictor values, but these models are in general not closer to the actual structure. Success with this approach will likely require rapidly computable score functions with fewer false optima.

4.2 INTRODUCTION

Recent advances in applying deep learning-based methods to predict protein structures directly from multiple sequence alignments (MSAs) have considerably improved our understanding of structure-function relationships. However, even with the considerable increase in model quality, there is still room for improvement to generate models that are accurate enough for critical applications such as drug or enzyme design. The goal of protein structure refinement is to increase the accuracy of protein structure models starting from models which have mostly

correct overall fold but are not atomically correct. As the Critical Assessment of Structure Prediction (CASP) experiments have demonstrated [9], [10], this is a very challenging problem in large part due to the very high dimensionality of protein conformational space: there are more ways to worsen a starting conformation than to improve it and sampling close to the correct structure is very difficult. Successful refinement requires efficient search algorithms that selectively traverse conformational space toward native structures. A previous refinement effort sought to make the sampling problem in refinement more tractable by carrying out search along with the principal components (PCs) of variation in naturally occurring homologs [40], but this linear representation of protein conformational space has a limited ability to represent the actual range of possible variation in a protein family. More recently, deep learning generative models have shown utility for protein modeling tasks including fold recognition [41], de novo design of 64 residue backbones [42], graph-based protein design [43] and to generate nodels of the Ig-fold [44]. Reasoning that variational autoencoders (VAE) should provide a more accurate reduced dimensionality protein representation than PCA, here we explore protein structure refinement by sampling in the latent space of VAEs generated from input sets of models.

4.3 RESULTS

4.3.1 *Input structure reconstruction*

We began by exploring different ways for encoding sets of input structures generated by Rosetta comparative modeling hybridization (Rosetta-CM) [45] calculation in VAE. We found the most effective approach was to generate full-atom distance maps (Ca, C, Cb) from each input structure, which have dimensionality (Nx3) x (Nx3), and to project these down into latent spaces

of dimension 64 (Figure **4-1**). On average, input model distance maps were reconstructed within 1 Å MSD (mean squared deviation), and following three-dimensional (3D) structure generation with Rosetta minimization calculations, reconstructed models were about 0.95 Global Distance Test - total structure (GDT-TS, or GDT in short) [46] to the original structures (Table 1.A).

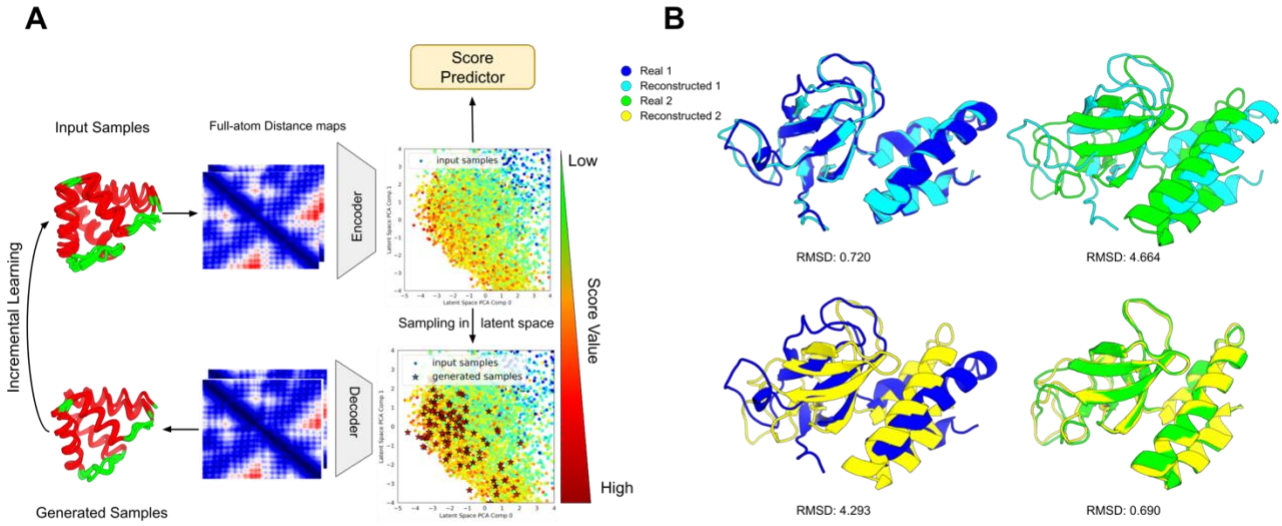


Figure 4-1. Training pipeline and structure reconstruction. (A) The input for the model are 2D full atom distance maps obtained from decoys of a given target. A neural network on top of the latent space is jointly trained to predict the target property and used to sample in the 64 dimensional latent space. The generated distance maps are converted to 3D structures through MDS and Rosetta protocols. These generated structures are then fed back into the VAE incremental training. (B) Comparison of original versus reconstructed structures of two randomly selected input decoys of target 4ld6A to each other.

4.3.2 *Latent space interpolation*

As a first test of sampling in the VAE latent space, we linearly interpolated between two embeddings chosen such that the interpolation passes close to the native structure (Figure **4-2**.A). The structures generated have realistic geometries and maintain all the secondary structures with smooth, concerted changes in the alpha helices and loop conformations (Figure **4-2**.B). As

expected, the GDT values of the decoded and reconstructed 3D structures reach a maximum near the native structure around the midpoint of the interpolation trajectory (Figure **4-2**.C).

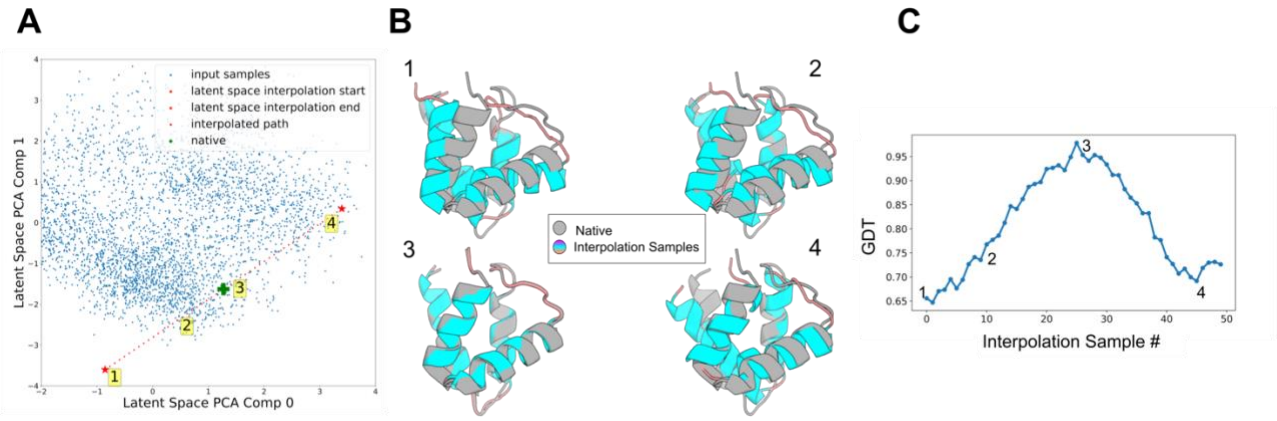


Figure 4-2. Linear interpolation in VAE latent space. (A) PCA plot of latent space of training samples and samples generated by interpolating between two random embeddings in latent space. The interpolation samples are passing through regions very close to native structure embedding. (B) 3D structures of the interpolated structures with the highest GDT structure generated shown as structure 3. Each interpolated structure is superimposed to native structure in gray. (C) Line plot showing GDT values of the interpolation samples generated.

4.3.3 *Use of scoring function for structure generation in latent space*

We next explored scoring functions for guiding search in the latent space. We considered both the Rosetta all-atom energy [47], and a measure of predicted model accuracy based on sequence and backbone coordinates computed by a neural network (CenQ). Both measures operate on 3D structures; for efficient sampling we sought to develop approximations of these metrics that operate in the latent space (3D structure generation requires decoding to generate distance maps, followed by 3D coordinate generation from the distance maps, and in the case of the Rosetta all atom energy, full atom packing calculations). We tested two approaches: first, encoding scores computed on the input structures along with the distance maps during VAE construction, so that each latent space point has an associated score, and second, training a

second neural network to reproduce scores of input structures based on latent space coordinates. For the first approach, the latent space was not organized by the encoded score value, leading to overlap of low and high scoring sample regions. This made sampling close to only high scoring samples difficult. With the second approach, the jointly trained predictor network was able to correlate the latent space with the score values we wanted to optimize, and therefore we could sample in regions that were populated with only high scoring samples.

Score distributions of the original metrics (CenQ score and Rosetta energy) and their latent space based approximations visualized by projection of the latent space along the two largest principal components of variation (Figure **4-3**.A and B and Figure S4-1) were quite similar. The native structure is located within the distribution spanned by the training samples and is close to high scoring training samples for both predicted score functions. Similarly, both score functions track with model accuracy (local distance difference test (lDDT) [48]) over the ensemble of models in the latent space (Figure **4-3** and Figure S4-1.A). For both scoring metrics, the latent space predictors provided reasonable approximations of the values computed from 3D structures (Figure **4-3**.C and Figure S4-1.B) but overall the Rosetta energy was harder to approximate from the low-dimensional latent space likely due to the higher sensitivity to atomic structure detail not fully captured in the latent space (Figure S4-1.C).

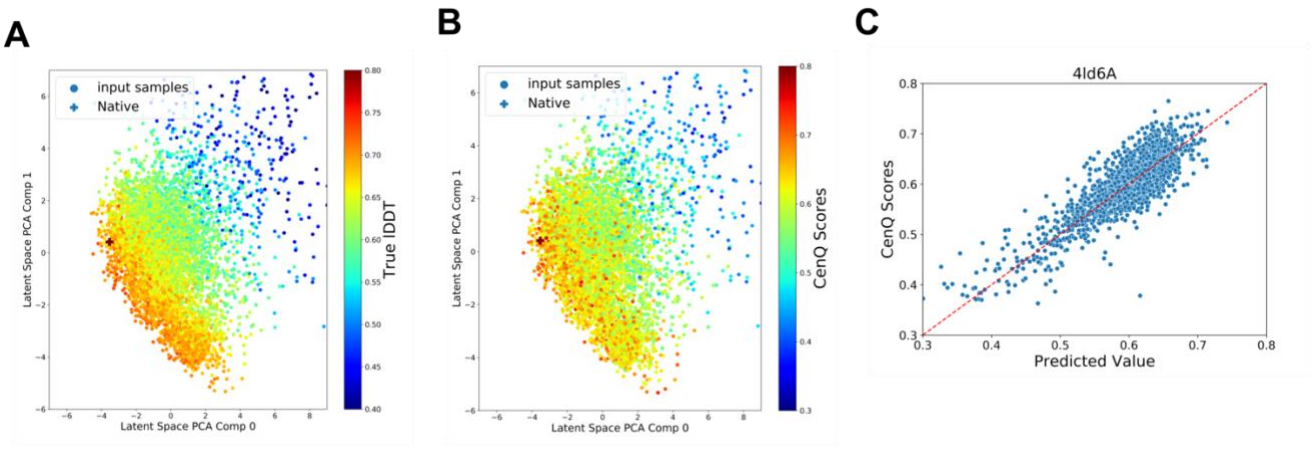


Figure 4-3. Scoring metrics for generating structures in the latent space. (A) and (B) 2D PCA plots showing the latent space representation of the training samples when a scoring metric is predicted from the latent space coordinates for target 4ld6A and its corresponding smoothed values. (C) Scatterplot of predicted values from latent space versus CenQ score.

4.3.4 *Incremental learning using generated samples*

A potential limitation of latent space sampling based on a given set of input structures is that if these differ considerably from the native structure, the latter may not be well represented in the latent space. To overcome this limitation, and to hone in on the high scoring region of the latent space and hopefully sample increasingly accurate structures, we explored an incremental learning approach. In each iteration, new samples were generated from the predicted high-scoring regions of the latent space, the generated samples were appended to the initial training samples and the entire VAE and score predictor model was retrained. The smoothed predicted CenQ scores and Rosetta energy were again used as scoring metrics for sampling the new structures. In this way the latent space itself evolves during the calculation, presumably towards a better representation of conformational space in the vicinity of the native structure.

In incremental learning calculations using the CenQ score predictor with 250 samples per iteration, we were able to readily sample structures with higher scores than in the initial training samples (Figure **4-4**.A and B). However, the clear increase in the highest CenQ score sampled per iteration was not accompanied by an increase in model accuracy (GDT). The correlation of the predicted CenQ scores from the latent space coordinates to the original CenQ scores remains high even after multiple rounds of incremental learning (Figure **4-4**.C), but the correlation between the CenQ scores and the true-lDDT is very low (Figure **4-4**.D), suggesting that this scoring metric is not sufficiently accurate to guide efficient sampling in the latent space.

Incremental learning calculations using the latent space approximation of the Rosetta energy for scoring produced structures with reasonable Rosetta energies and GDT values but were also unable to generate structures that were significantly better than the input training samples (Figure S4-2.A). In this case, the major problem was score estimation in the latent space (Figure S4-2.B). The Rosetta energy latent space predictor was not able to accurately predict the ground truth Rosetta energy (Figure S4-2.C), likely due to overfitting. The correlation of the Rosetta energy to the true-lDDT of the generated samples is higher than that of CenQ scores suggesting that if approximated correctly, it could be used to guide iterative refinement of the target (Figure S4-2.D).

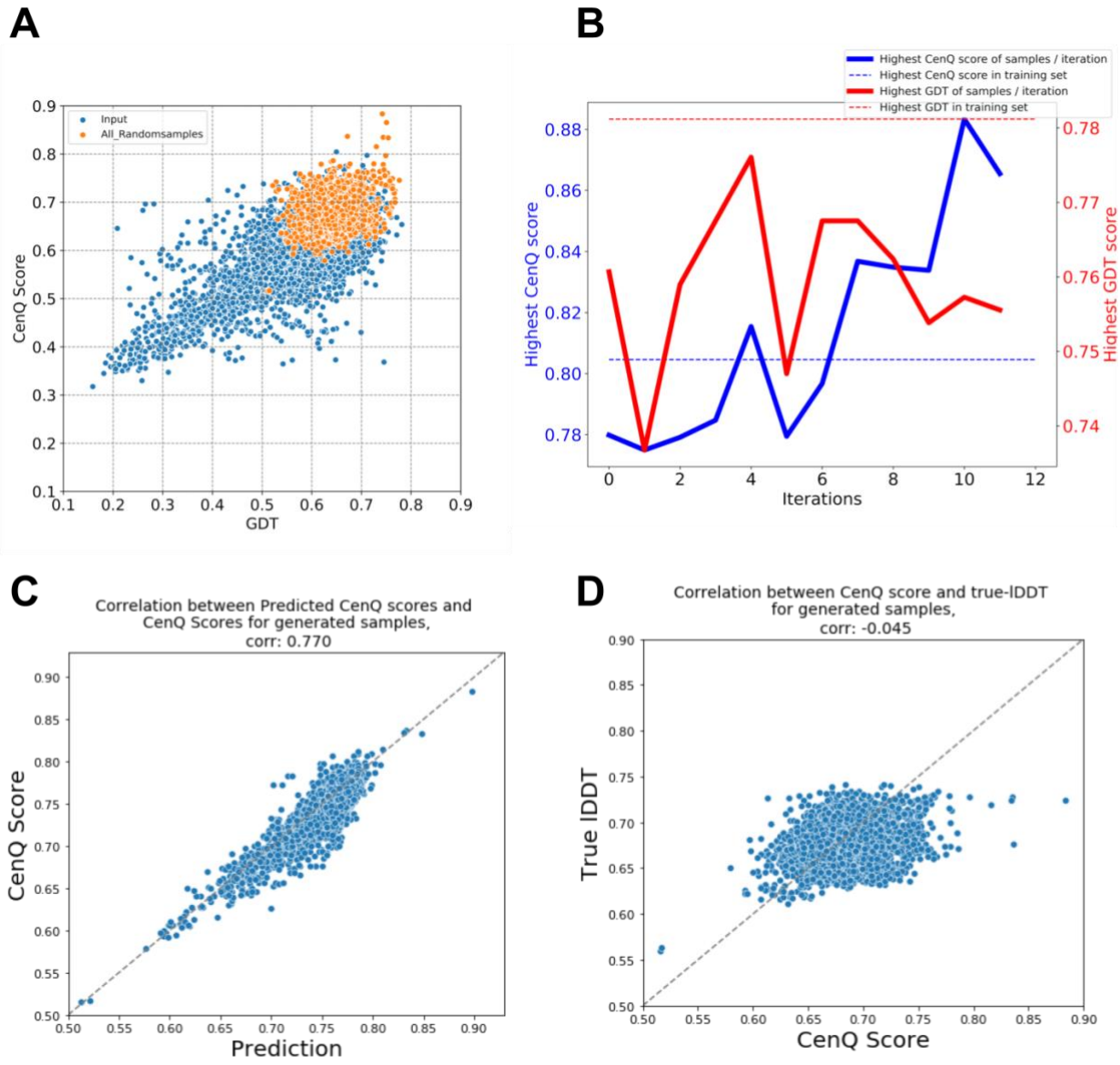


Figure 4-4. Optimization of Smoothed CenQ scores through Incremental Learning for target 4ld6A. (A) Scatterplot of the GDTs and CenQ scores of the generated samples and initial training sample after 12-th iteration. (B) Highest GDT and CenQ score sampled in every iteration. (C) Scatterplot showing correlation of predicted vs original CenQ score of generated samples after 12 incremental learning iterations. (D) Scatterplot showing correlation between original CenQ scores and true-lDDT of the generated samples.

4.4 DISCUSSION

We found that given sets of input 3D structure models for a protein of interest, VAEs operating on distance maps could accurately capture the differences between the models, and sampling in a 64 dimensional latent space followed by 3D structure generation produced physically reasonable models with features smoothly interpolated between the input structures. Samping in regions close to the native structure in the latent space generated very high GDT, native-like structures not present in the training dataset. Thus, in principle, the latent spaces of VAEs provide a reduced dimensionality space for increasing the tractability of the sampling problem in protein structure refinement.

The challenge for our approach is that to guide latent space-based refinement, scoring metrics are needed that robustly identify regions corresponding to higher accuracy models. We found that while both neural network-based model accuracy measures and the Rosetta energy, and latent space based approximations to these, correlated with structure accuracy in starting model populations, direct optimization against the latter latent space based functions failed to generate improved models, although it did generate models with higher predicted scores. This is likely a reflection of a general problem with direct optimization against neural network computed scores: because of the large numbers of parameters, even models fit on large training sets well have many "false" optima that are attractors during direct optimization. Our incremental learning strategy sought to address this problem by adding newly sampled structures to the input models and retraining the VAE and score predictor at each iteration, but even in this case there was a divergence between score optimization and structure improvement. Success with this overall approach will likely require use of scoring functions with fewer parameters that are not as readily

over optimized, but there is a tradeoff with computational cost: for example, overfitting was much less a problem when optimizing against the all atom Rosetta energy (not the network predicted energy) (Figure S4-2), but this was computationally far more expensive due to the necessity for 3D structure generation for each score evaluation. While recent advances [9], [10] make protein structure refinement a less pressing problem for the classic single state native protein structure prediction problem, our latent space sampling approach could be useful for mapping out ensembles of excited states for modeling protein dynamics and flexible backbone protein and small molecule docking (Chapter 5).

4.5 METHODS

4.5.1 *Input training dataset for VAE*

The input dataset is obtained from the refinement protocol implemented by Park *et al*. [49]. This protocol carries out large-scale sampling of the energy landscape through structural variation of the starting model and using an evolutionary algorithm to iteratively guide these generated models towards energy minima. This generates a pool of energetically favorable, and structurally diverse models for each target, termed 'decoys'. We chose 5 different low-resolution refinement targets taken from previous CASP datasets. For each of these targets, there were about 8000 decoys generated by the refinement protocol, and we used a 80: 20 split for the training and validation sets. The input to the encoder of the VAE are the 2D full-atom distance maps (N,Ca,C) for each of the decoys for a given target. The VAE, in its current state, is trained separately for each refinement target.

4.5.2 *Scoring Metrics: Centroid Level Accuracy Metric and Rosetta Energy*

CenQ, an accuracy predictor based on 2D residual convolutional networks which works on coarse-grained representation of the input protein structures, was trained for the purposes of speeding up calculations of backbone quality. It estimates the coarse backbone quality based on the torsion angles, Cb distance maps, inter-residue orientations [31], and sequence information. For each of the decoys for a given target, the CenQ scores were obtained and used as ground truth to train the additional neural network on top of the latent space of the VAE. Rosetta energy was calculated using the ref2015 score function [47] for each of the decoys for a given target and used to train the dense neural network on top of the latent space. For smoothing out the scoring metrics (CenQ scores or Rosetta energy) in the latent space, k-nearest neighbors were used. The smoothed value for each training latent space coordinate was the average of the score of 10 of its closest neighbors. For new samples generated from the latent space, the smoothed value was the average of the predicted score (from latent space predictor) of 10 of its closest neighbors.

4.5.3 *VAE architecture and training*

A Vanilla VAE [50] was used where the model minimizes a reconstruction loss and a KL-divergence loss that ensures that the latent space is isotropic gaussian. The reconstruction loss is made up of two parts; mean-squared-error over all distances, and a "local loss" that places a sigmoidal penalty on the mean-squared-error calculated over distances within 20 Å. This local loss leads to better local structure recovery within the generated structures. We found that decreasing the weight of the KL-divergence by a factor, β, of 1e-3 (or in some cases 1e-2), prevented the latent space from collapsing, while still maintaining a well-regularized isotropic

gaussian latent space. The encoder and decoder were made up of 2D convolutional layers and the latent space was kept at a constant of 64 dimensions for all targets.

A four layered, dense neural network on the latent space, termed Predictor, was jointly trained with the VAE (Figure **4-1**). This network took in the latent embeddings and predicted the CenQ scores or Rosetta energy values. This allowed for correlations between the latent space and a target property which made it possible to sample in regions of high predicted backbone quality. The loss function was minimizing the mean-squared-error between the predicted score values and the ground truth. The overall loss for this model was weighted 1:β:1 for reconstruction loss : KL divergence : predictor loss.

4.5.4 *Sampling in latent space*

We used the Simplex search algorithm [51] for generating new structures in the latent space through optimizing the smoothed predicted CenQ score and Rosetta energy. The starting structures were a subset of the training samples which passed through a 'diversity filter', where the highest k scoring samples are selected and diversified by ensuring that no two samples have distance maps closer than 0.4 Å. The objective function was the smoothed predicted score from the latent space. The simplex trajectory was run on each starting point and then its subsequent point for 5 iterations.

We also tried isotropic Gaussian sampling, where we generated a million samples from a Gaussian distribution and selected the highest predicted latent space coordinates for structure generation. The structures generated through this were not diverse enough and would often occupy the same high scoring region of the latent space. Even by enforcing the diversity filter,

we were unable to generate sufficiently high scoring and diverse samples through this sampling method.

4.5.5 *Structural Modeling*

For converting generated distance maps from the latent space into 3D protein structures, we used a combination of Multi-Dimensional Scaling (MDS) [52] and Rosetta protocols [5]. The generated distance maps are first passed through MDS to obtain a coarse 3D structure. The idealize-mover in Rosetta is used to correct the bond length and angles of the MDS generated structures. Finally, the structures are passed through torsion and cartesian minimization in Rosetta, using the distance constraints generated by the VAE, to obtain a final 3D structure. For the Rosetta energy models, since we used a full-atom scoring function, there was an additional full-atom relaxation step for the generated structures.

4.6 SUPPLEMENTARY FIGURES AND TABLES

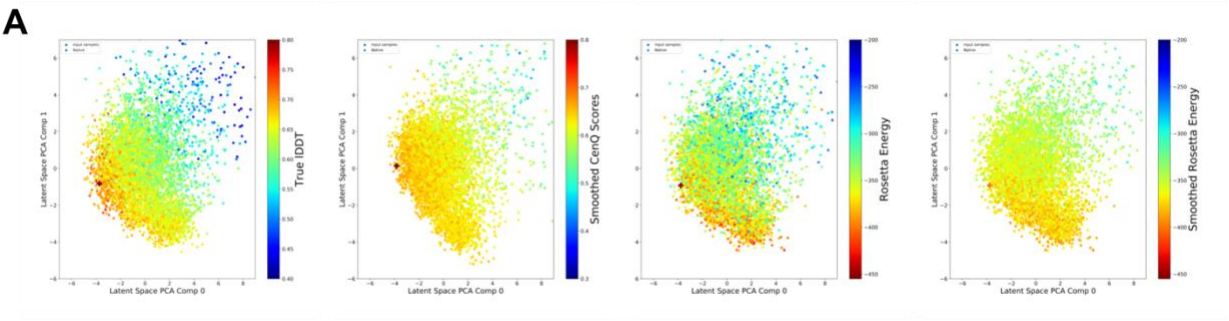
Table 1. Reconstruction Accuracy and scoring metric prediction accuracy for 5 targets.
(A) Table listing distance map reconstruction and structural quality of decoded target input.
(B) Table listing spearman rho correlations of predicted scoring metric (Rosetta energy or CenQ scores) from the latent space.

A

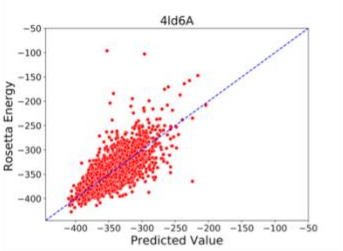
Target	Distance map reconstruction (RMSD Å)	Structure Recovery (GDT)
4ld6A	0.737 ± 0.43	0.93 ± 0.1
TR283	0.800 ± 0.34	0.94 ± 0.06
TR574	0.654 ± 0.31	0.95 ± 0.06
TR280	0.427 ± 0.15	0.983 ± 0.02
TR663	0.280 ± 0.085	0.993 ± 0.02

B

	SpearmanR correlation of predicted scoring metric to scoring metric ground truth	
Target	Rosetta Energy	CenQ scores
4ld6A	0.695	0.853
TR283	0.826	0.946
TR574	0.893	0.933
TR280	0.719	0.909
TR663	0.832	0.660



B



C

	SpearmanR to truelDDT (top 250 validation samples)			
Target	Raw CenQ Scores	Smoothed CenQ scores	Raw Rosetta Energy	Smoothed Rosetta Energy
TR280	0.537	0.705	-0.166	-0.103
TR283	0.0563	0.0803	-0.053	-0.246
TR574	0.598	0.768	-0.173	-0.440
4ld6A	0.203	0.358	-0.107	-0.262
TR663	0.260	0.450	-0.017	-0.199

Figure S4-1. Latent space representation using different scoring metrics and correlations to ground truth scoring metric (true lDDT). (A) 2D PCA plots showing the latent space representation of the training samples when a scoring metric is predicted from the latent space coordinates for target 4ld6A. (B) Scatterplot of predicted Rosetta energy from latent space versus ground truth. (C) Table of Spearman correlations of the raw and smoothed-out CenQ scores and Rosetta energy to the true-lDDT values for the top 250 scoring samples in validation set for 5 targets.

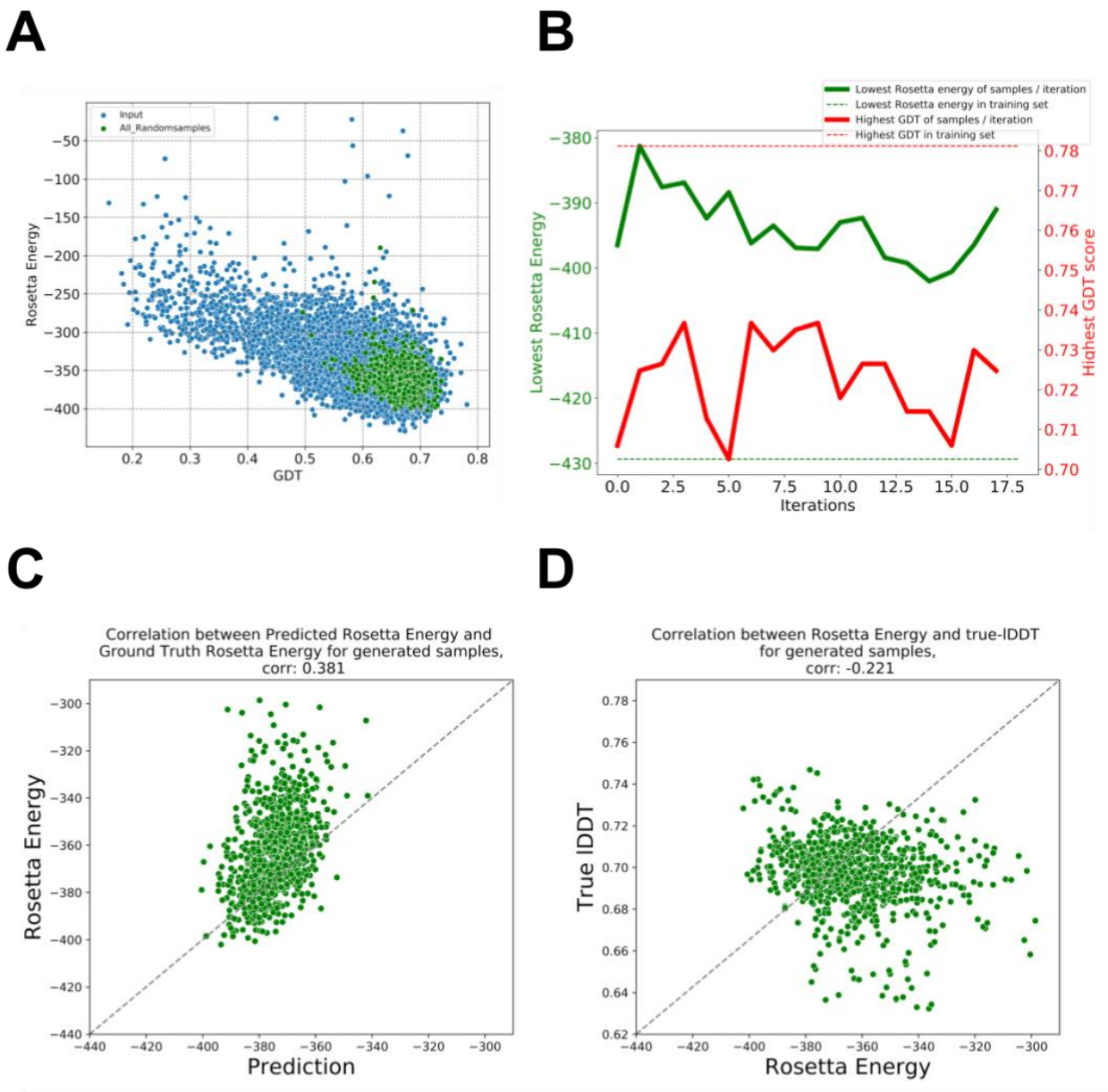


Figure S4-2. Optimization of Smoothed Rosetta energy through Incremental Learning for target 4ld6A. (A) Scatterplot of the GDTs and Rosetta energy of the generated samples and initial training sample after 18 iterations. (B) Highest GDT and Rosetta energy sampled in every iteration. (C) Scatterplot showing correlation of predicted vs original Rosetta energy of generated samples after incremental learning. (D) Scatterplot showing correlation between original Rosetta energy and true-lDDT of the generated samples.

Chapter 5. KRAS ENSEMBLE GENERATION THROUGH SOFT-INTROSPECTIVE VARIATIONAL AUTOENCODERS AND ROSETTAFOLD

5.1 ABSTRACT

Proteins are dynamic molecules that can adopt a variety of conformations. These different conformations can play important roles in protein function, such as in ligand binding, protein-protein interactions, and enzymatic catalysis. Protein ensemble generation is crucial for understanding the relationship between protein structure and function, as it enables the exploration of a broad range of protein conformations that may be relevant to biological activity. Here, we explore the use of soft introspective variational autoencoder (SI-VAE) for the task of protein ensemble generation by converting high-dimensional protein structure data to a low-dimensional representation and carry out search in this space guided by the categorical cross-entropy loss to the AlphaFold predicted structures of different known K-Ras sequences. Generated structural template features from the SI-VAE were fed into pre-trained RoseTTAFold for 3D structural modeling. We find that the CCE guided sampling in the encoded lower dimensional space, followed by decoding to generate full 3D structures, rapidly generates models of high structural quality and can reproduce most held-out K-Ras structures to a sub-angstrom accuracy. Further docking studies show that the sampled structures can recapitulate the cryptic pockets in the structures to allow for small molecule docking.

5.2 INTRODUCTION

As the Critical Assessment of Structure Prediction (CASP) experiments have demonstrated [9], [10], the classic, single state native protein structure prediction problem has made great advancements in recent years, however mapping out ensembles of protein states for modeling protein dynamics remains a challenging task. Modeling flexibility in protein backbones is essential to achieving protein design success since most soluble proteins occupy an ensemble of backbones in equilibrium around a native state in solution. Early work in generating ensembles was done by randomly perturbing the phi/psi angles and saw that the different sequence profiles on the distinct conformations captured the original input template and its homologous structures [53]. Molecular Dynamics (MD) is the most used method for generating ensembles by diversifying the initial structure with results indicating that the ensembles generated for proteins of common folds shared sequence spaces to those of corresponding homologous families [54] . More recent work has gone into modeling more naturally occurring protein motions [55] and kinematic closure (KIC), which is an algorithm for sampling diverse protein loops [56]. However, there is a lot more potential in this field with the advancement of single-structure prediction models. Here we explore the use of such structure prediction models along with sampling in the latent space of generative models from an input set of models for K-Ras.

We use structure-based generative models for the purpose of generating an ensemble of K-Ras. K-Ras has a crucial role in cancer biology and is sometimes referred to as the Holy Grail of drug discovery [57]. K-Ras drug discovery is mostly limited to one mutant - G12C - excluding other oncogenic variants which we still do not fully understand. The native structures of the active and inactive K-Ras proteins lack visible druggable pockets for inhibitors [58]. Therefore, effective

inhibitors have been developed through binding to cryptic sites unseen in the static, native structures. The identification of these druggable, cryptic sites usually involve computationally expensive MD simulations or long experiments involving NMR experiments. Being able to identify novel cryptic pockets of K-Ras can prove to be highly impactful in computer-aided drug design of K-Ras inhibitors.

We chose to use a Soft-Introspective VAE (SI-VAE) as our generative model, which trains a vanilla variational autoencoder adversarially, using the encoder to discriminate between real and generated samples [59]. The additional adversarial training has been shown to significantly improve the resolution of output images, which has traditionally been a major impediment for vanilla VAEs to be used in protein design methods. The compressed latent space representation learned from this generative model has been shown to be descriptive enough to accurately reconstruct the input, and well-regularized such that it can generate new, plausible data occupying the same data distribution as the input. For the task of protein ensemble generation, this generative model can first limit the conformational space, and be able to intelligently sample in it to generate plausible conformations of the same protein sequence.

For encoding 3D structural information into our SI-VAE framework, we chose to use the two-dimensional RoseTTAFold [10] template features. Each input structure is converted to its corresponding pairwise Cb distance map and orientations (phi, psi, omega). We chose to use raw distances and orientation values as input for a more interpretable latent space. The reconstructed template features were then used as input template features for 3D structure generation with RoseTTAFold, with the target MSA provided (Figure **5-1**). Once the SI-VAE was trained, we

used the categorical cross-entropy (CCE) as the guiding metric for sampling in the latent space. The CCE metric was calculated between the Cb distogram of AlphaFold predicted model of the held-out K-Ras conformation and the generated Cb distances (discretized using radial basis function). We carried out gradient optimization of this metric in the latent space to search in areas that scored highly and were predicted to generate high quality structures. We employed a per-target training, where for a specific target, we used the MD snapshots from our set of chosen crystal structures of K-Ras which were at least an angstrom away from the target. This model demonstrates that by jointly learning the latent representation of protein conformations from a diverse set of input structures, we can effectively sample and generate the held-out K-Ras conformation. Accurate protein ensemble generation would radically change what is possible in medicine and biotechnology. The development of novel techniques to sample protein conformational space more efficiently brings us closer to this goal.

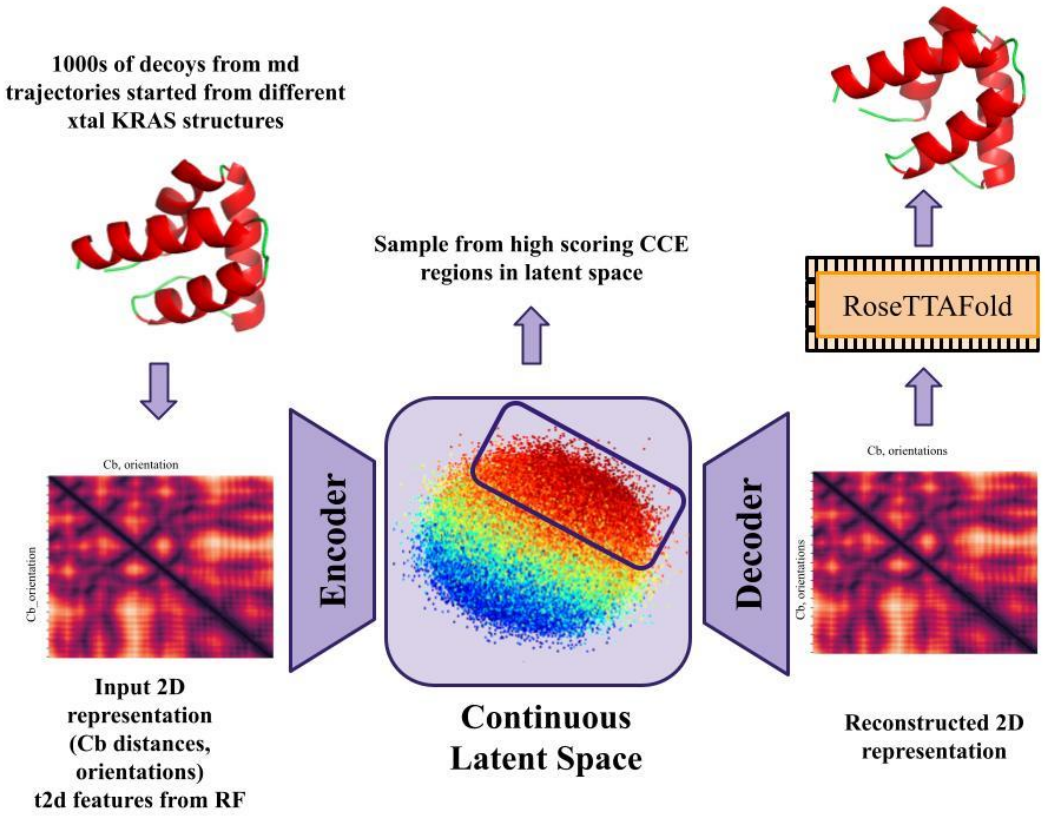


Figure 5-1. Overall pipeline of SI-VAE + RoseTTAFold structural modeling. 3D structures from MD simulations of training crystals that were at least an angstrom away, are converted to 2D template features from RoseTTAFold [10]. The decoded template features are converted to 3D structures through RoseTTAFold [10], provided the target MSA. The trained latent space is used for sampling structures that are optimized in the score metric chosen.

5.3 RESULTS

5.3.1 *Reconstruction accuracy of target K-Ras conformation from SI-VAE and AF2*

To mark our upper threshold of sampling accuracy from the latent space, we first looked at the reconstruction accuracy of each target crystal. We encoded the target template features into the trained SI-VAE and decoded into its reconstruction. The reconstructed template features were passed into RoseTTAFold [10] for structural modeling. The results show that for each target crystal, the SI-VAE was able to reconstruct it to a higher accuracy (lower coordinate

RMSD) than the AF2 model predicted for the same target (Figure **5-2**). It is also important to note that for most of the targets (13/20), the reconstructed target from the SI-VAE was of sub-angstrom accuracy (RMSD < 1A). This was a significant improvement over the AF2 models, where only 2/20 target predictions were of sub-angstrom accuracy.

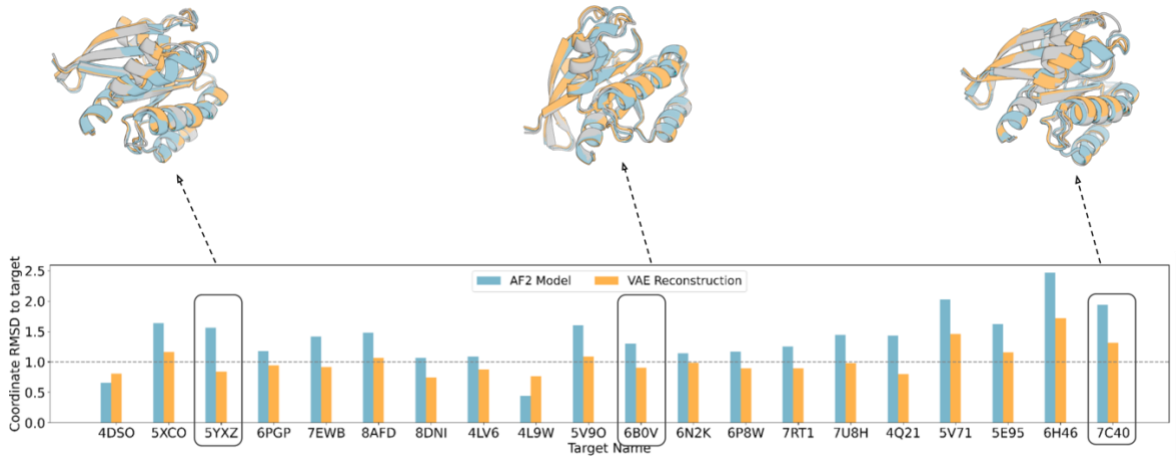


Figure 5-2. Structure reconstruction accuracy of AF2 and SI-VAE. Bar chart plotting the coordinate RMSD of the closest AF2 predicted model and the reconstructed model from decoded template features from SI-VAE used as input for structural modeling using RoseTTAFold. Structural impositions for 3 targets are highlighted on top with the target crystal in gray, the AF2 prediction in blue and the SI-VAE reconstruction in orange.

5.3.2 *Generated samples reconstruction accuracy to target conformation*

The used the trained latent space for sampling the target K-Ras conformation through gradient optimization of the score metric. For each target, we sampled *n* latent space coordinates following a normal gaussian distribution. Each latent space coordinate was then optimized for the CCE metric calculated between the generated Cb distance map and the AF2 predicted Cb distances for the target until convergence. The generated samples were evaluated using coordinate RMSD to the target crystal on either the overall structure recapitulation or cryptic pocket environment

reconstruction, which is defined as residues within 5 angstroms of the ligand binding pocket for each target. The results show that the SI-VAE can reconstruct the overall structure to a higher degree than the closest train crystal, train sample and the closest AF2 model for most targets (Figure **5-3**).

The reconstruction of the cryptic pocket is vital for docking ligands for inhibition. The cryptic pocket environment residue reconstruction shows that the SI-VAE model performs just as well or better than the closest training sample in most cases (MD snapshot) (Figure **5-4**). The structural superimposition shows that the generated samples are not clashing with the superimposed ligand from the target structure, highlighted in orange, and therefore can be docked without hindrance, whereas for the closest train crystal and the closest AF2 model, there are significant clashes present (Figure **5-4**).

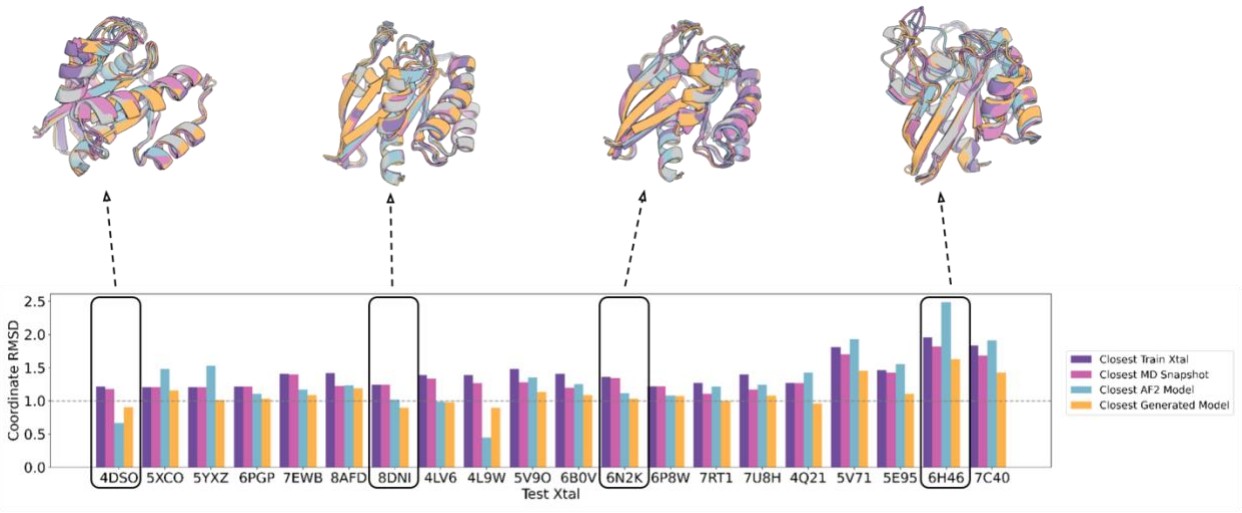


Figure 5-3. K-Ras overall structure reconstruction evaluation. Barchart showing the coordinate reconstruction error of the closest train crystal, the closest training sample, the closest AF2 model and the closest generated sample.

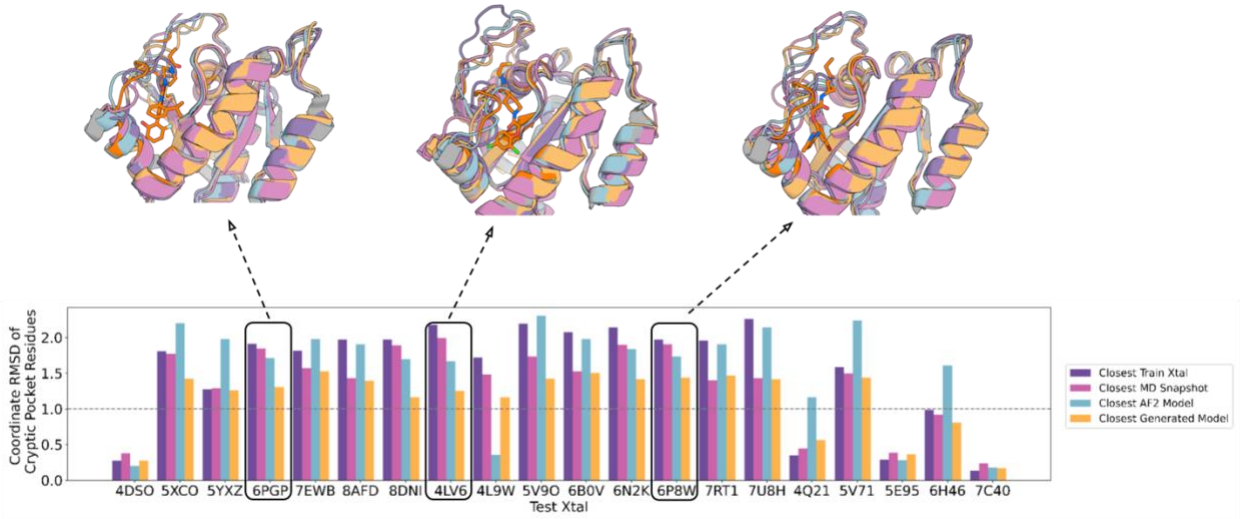


Figure 5-4. K-Ras cryptic pocket reconstruction evaluation. Barchart showing the coordinate reconstruction error of the closest train crystal, the closest training sample, the closest AF2 model and the closest generated sample. The structural impositions shown on top show the ligand inhibitor docked in the target crystal, where the cryptic binding pocket and the ligand are highlighted in orange on the target crystal structure.

5.3.3 *Docking generated samples with ligand inhibitor reveals cryptic pockets*

For evaluating whether the generated samples contained the cryptic, druggable pocket for each target, we took the generated structures from the latent space and docked them with the known ligand inhibitor for that target. For comparison, for each target, we also docked the closest train crystal that the input dataset was generated from, all input MD snapshots that were used as the initial training samples and the 5 AlphaFold models with the known inhibitor as well.

The results show that the generated samples have the lowest ligand RMSDs calculated for the targets evaluated and have cryptic pockets that can be drugged with the known ligand for that target (Figure **5-5**). The orientation of the ligand plays an important role in the inhibition of the target since it forms bonds with the cryptic pocket residues of the target protein that lead to its inhibition. The generated samples, through gradient optimization of the score metric in the latent space, were able to recapitulate the important ligand-protein interactions more so, compared to the closest MD snapshot which is most comparable in terms of the ligand RMSD.

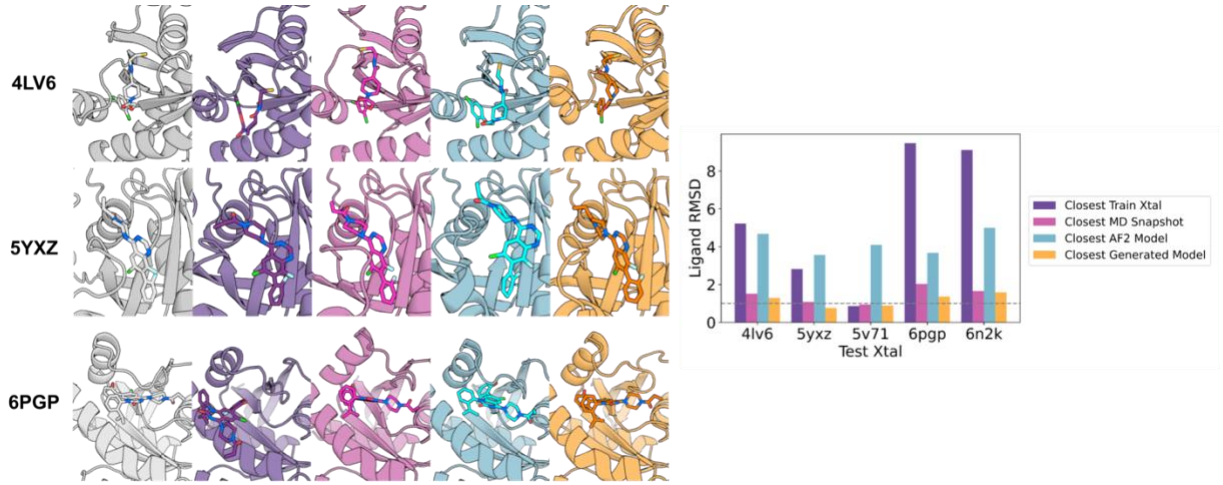


Figure 5-5. Docking small molecule inhibitors. Structures docked with the target ligand inhibitor in a known cryptic pocket in the target protein. The bar chart shows the ligand RMSD is calculated between the docked ligand in each structure to the native, known ligand.

5.4 DISCUSSION

We found that given sets of 3D input structures from MD simulations of known K-Ras structures at least an angstrom away from the K-Ras conformation of interest, through gradient optimization of a score metric in the latent space, we were able to generate samples that were closer to the target conformation than anything in the training set or the AF2 predicted models. Through this pipeline, we also saw that the generated samples have druggable, cryptic pockets present in the target structure through docking of the known ligand inhibitor of the target. This suggests that the generative model we chose to use in our case was able to limit the conformational space and through sampling, be able to generate high quality structures.

The challenge for our approach is that the sampling process is limited by the overall accuracy of the predicted model towards which we are optimizing. In our case, we chose to optimize towards the AF2 predicted model (total 5 models) which had the highest score metric to

a given latent space sample to guide it towards favorable regions. The closest models predicted by AF2 for a given K-Ras structure were about an angstrom away in coordinate space and that points to a limitation of the quality of the generated structures. Further improvements could be made through optimization of the latent space coordinates towards AF2 models, RoseTTAFold predicted models, and towards the ground-truth training crystals embedded into the latent space. Furthermore, the reconstruction accuracy of our generative model could be improved through fine-tuning the trained VAE on a structure-based loss, FAPE. This would allow for 3D plausible structures to be predicted by the 2D decoded template representation. A more extensive search of the hyperparameter space could also yield better results. Overall, the results of this method show the utility of deep generative models towards dynamic modeling through building upon existing highly accurate structure prediction networks.

5.5 METHODS

5.5.1 *Input data setup and incremental learning*

For the input dataset, we began by selecting distinct K-Ras conformations deposited in the PDB that are at least an angstrom away from each other as our ‘training set crystal structures’. In addition to the RMSD cut-off filter, we also selected conformations that had a deposited / known inhibitor. We selected 20 K-Ras structures with these criteria. We ran MD simulations for 25ns starting with each K-Ras crystal structure and selected every *n*th structure to make up about 1000 initial MD snapshots. For each target crystal, the training data consisted of MD snapshots of the training set crystal structures that were at least an angstrom away from it. The final 20 K-Ras conformations that we chose for our project were: 4DSO, 5XCO, 5YXZ, 6PGP, 7EWB, 8AFD, 8DNI, 4LV6, 4L9W, 5V9O, 6B0V, 6N2K, 6P8W, 7RT1, 7U8H, 4Q21, 5V71, 5E95, 6H46 and

7C40. All 3D structures were converted to 2D template features from RoseTTAFold [10] which consists of Cb distances and orientations. We chose to use the raw distance and orientation values for training the model for a more interpretable latent space.

After the first round of training using only MD snapshots as the training data, we then generated 3000 samples from the latent space that were optimized for the score metric and passed the diversity filter (following the protocol laid out in the sampling methods section of this chapter). These 3000 generated structures were then concatenated on the initial MD snapshot training set to form an 'incremental learning' training set of structures for the model. Using this new set, for each target, the training runs were set up again from scratch.

5.5.2 *Soft-Introspective VAE objective and training*

For this project, we chose to use a Soft-Introspective VAE [59] which has been shown to have significantly higher output resolution than the vanilla VAE [50]. The objective function of this model, along with the traditional VAE objective function of reconstruction loss and KL divergence, has adversarial losses incorporated, like GANs [60] but is trained introspectively. In the case of SI-VAEs, the encoder is the implicit 'discriminator' where it is induced to distinguish, through the ELBO (evidence lower bound) [50] values that it assigns to the real and generated samples. The decoder is the 'generator' where it is induced to generate samples to 'fool' the encoder (discriminator). However, unlike GANs, the SI-VAE model does not converge to the data distribution, but to an entropy-regularized version of it [59].

Using default parameters from Daniel *et al.* [59], encoder was trained with the following objective (Equation 5-1):

$$L_{encoder}(x,z) = s\,.\left(\beta_{rec}L_r(x) + \beta_{kl}\,KL\,(x)\right) + \frac{1}{2}\exp\Big(-2s\,.\Big(\beta_{rec}\,L_r\,\big(Dec\,(z)\big) + \beta_{neg}\,KL\big(Dec\,(z)\big)\Big)\Big)$$ Equation 5-1

where $L_r(x)$ = reconstruction loss,

s = 2,

β_{rec} = 10,

β_{kl} = 1e-3,

β_{neg} = latent dimension (256) and

Dec(z) = trained decoder of SI-VAE

The decoder was optimized using the following objective (Equation 5-2):

$$L_{decoder}\,(x,z) = s\,.\beta_{rec}\,L_r\,(x) + s\,.\Big(\beta_{kl}\,KL\,\big(Dec(z)\big) + \gamma_r\,.\beta_{rec}\,L_r\,\big(Dec(z)\big)\Big)$$ Equation 5-2

where L_{rec} = reconstruction loss,

s = *2*,

β_{rec} = 10,

β_{kl} = 1e-3 and

γ_r = 1.0

The reconstruction loss was the mean-squared-error loss over all distances and orientations on the decoded template features from the model. The model was optimized using individual optimizers for the encoder and decoder, both of which were initialized with Adam (β1 = 0.9, β2 = 0.999) with learning rate 1e−3, with an effective batch size of 64. The encoder and decoder

were made up of 3 ResNet blocks with 2D convolutional layers and the latent space was kept at a constant of 256 dimensions for all targets.

5.5.3 *Sampling in latent space through gradient optimization of score metric (CCE)*

To obtain the optimized structures using the trained decoder, we used gradient optimization in the latent space. We first randomly sample *n* numbers from a standard Gaussian distribution with dimension equal to that of the latent space. The initialized latent space coordinates are set to be trainable. Each sample is then decoded into its respective template features and Cb distances are discretized through radial basis function to ensure back propagation. The score metric we chose to optimize is the minimum categorical cross-entropy (CCE) among all 5 AF2 predicted Cb distograms of the target structure and the generated Cb distances. The Adam optimizer modifies the latent space sample to minimize this score metric. This process is repeated until convergence. To ensure that diversity is maintained, the latent space coordinates are restricted to explore only *d* euclidean distance in the latent space from their initial starting coordinates. The overall goal of this exploration technique is to search the latent space to find a better solution near the initial randomly generated coordinates. The final, converged latent space coordinates are decoded into their respective template features and passed into RoseTTAFold, along with the target MSA for structural modeling.

5.5.4 *Docking protocol*

For each docking case, the inhibitor ligand was docked to the receptor model using the protein-ligand docking method Rosetta GALigandDock [61]. The ligand atomic coordinates found in complex crystal structures were extracted and used to prepare for ligand docking. The ligands were protonated, and the AM1-BCC partial charges were calculated using the tools provided by

openbabel, Antechamber in the AMBER suite, and UCSF Chimera [62]. The ligand information was converted to the parameter format that is compatible with the Rosetta generic potential. The initial position of the ligand to initiate docking was determined by superimposing the complex crystal structure to the sampled protein backbone. Protein-ligand docking was performed by allowing the side chains that are within 5A from the ligand to be flexible. The receptor models were optimized in advance using Rosetta FastRelax with high constraints on each backbone. We ran 20 parallel docking runs for each receptor model and ligand pair, and the combined results were analyzed.

BIBLIOGRAPHY

[1] T. A. Whitehead *et al.*, “Optimization of affinity, specificity and function of designed influenza inhibitors using deep sequencing,” *Nat Biotechnol*, vol. 30, no. 6, pp. 543–548, Jun. 2012, doi: 10.1038/nbt.2214.

[2] E. M. Strauch *et al.*, “Computational design of trimeric influenza-neutralizing proteins targeting the hemagglutinin receptor binding site,” *Nat Biotechnol*, vol. 35, no. 7, pp. 667–671, Jul. 2017, doi: 10.1038/nbt.3907.

[3] A. D. Smart, R. A. Pache, N. D. Thomsen, T. Kortemme, G. W. Davis, and J. A. Wells, “Engineering a light-activated caspase-3 for precise ablation of neurons in vivo,” *Proc Natl Acad Sci U S A*, vol. 114, no. 39, pp. E8174–E8183, Sep. 2017, doi: 10.1073/pnas.1705064114.

[4] C. E. Tinberg *et al.*, “Computational design of ligand-binding proteins with high affinity and selectivity,” *Nature*, vol. 501, no. 7466, pp. 212–216, 2013, doi: 10.1038/nature12443.

[5] R. F. Alford *et al.*, “The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design,” *J Chem Theory Comput*, vol. 13, no. 6, pp. 3031–3048, Jun. 2017, doi: 10.1021/acs.jctc.7b00125.

[6] B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl, “GRGMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation,” *J Chem Theory Comput*, vol. 4, no. 3, pp. 435–447, Mar. 2008, doi: 10.1021/ct700301q.

[7] A. Villegas-Morcillo, S. Makrodimitris, R. C. H. J. van Ham, A. M. Gomez, V. Sanchez, and M. J. T. Reinders, “Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function,” *Bioinformatics*, vol. 37, no. 2, pp. 162–170, 2021, doi: 10.1093/bioinformatics/btaa701.

[8] M. Ekeberg, C. Lövkvist, Y. Lan, M. Weigt, and E. Aurell, “Improved contact prediction in proteins: Using pseudolikelihoods to infer Potts models,” Nov. 2012, doi: 10.1103/PhysRevE.87.012707.

[9] J. Jumper *et al.*, “Highly accurate protein structure prediction with AlphaFold.,” *Nature*, 2021, doi: 10.1038/s41586-021-03819-2.

[10] M. Baek *et al.*, “Accurate prediction of protein structures and interactions using a three-track neural network,” *Science (1979)*, vol. 373, pp. 871–876, 2021, Accessed: Feb. 02, 2022. [Online]. Available: https://predictioncenter.org/casp14/

[11] J. Wang *et al.*, “Deep learning methods for designing proteins scaffolding functional sites”, doi: 10.1101/2021.11.10.468128.

[12] S. Mansoor, M. Baek, D. Juergens, J. L. Watson, and D. Baker, “Accurate Mutation Effect Prediction using RoseTTAFold,” *bioRxiv*, p. 2022.11.04.515218, Jan. 2022, doi: 10.1101/2022.11.04.515218.

[13] E. C. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, and G. M. Church, “Unified rational protein engineering with sequence-based deep representation learning,” *Nat Methods*, vol. 16, no. 12, pp. 1315–1322, Dec. 2019, doi: 10.1038/S41592-019-0598-1.

[14] S. Biswas, G. Khimulya, E. C. Alley, K. M. Esvelt, and G. M. Church, “Low-N protein engineering with data-efficient deep learning,” *Nat Methods*, vol. 18, no. 4, pp. 389–396, Apr. 2021, doi: 10.1038/s41592-021-01100-y.

[15] T. Bepler and B. Berger, “Learning protein sequence embeddings using information from structure,” *7th International Conference on Learning Representations, ICLR 2019*, Feb. 2019, Accessed: Feb. 02, 2022. [Online]. Available: https://arxiv.org/abs/1902.08661v2
[16] I. T. Jollife and J. Cadima, “Principal component analysis: A review and recent developments,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065. Royal Society of London, Apr. 13, 2016. doi: 10.1098/rsta.2015.0202.
[17] L. Van Der Maaten and G. Hinton, “Visualizing Data using t-SNE,” 2008.
[18] N. Anand, R. Eguchi, and P. S. Huang, “Fully differentiable full-atom protein backbone generation,” *Deep Generative Models for Highly Structured Data, DGS@ICLR 2019 Workshop*, no. 11, pp. 1–10, 2019.
[19] V. Gligorijević *et al.*, “Structure-based protein function prediction using graph convolutional networks,” *Nat Commun*, vol. 12, no. 1, 2021, doi: 10.1038/s41467-021-23303-9.
[20] A. J. Riesselman, J. B. Ingraham, and D. S. Marks, “Deep generative models of genetic variation capture the effects of mutations,” *Nat Methods*, vol. 15, no. 10, pp. 816–822, 2018, doi: 10.1038/s41592-018-0138-4.
[21] J. Meier, R. Rao, R. Verkuil, J. Liu, T. Sercu, and A. Rives, “Language models enable zero-shot prediction of the effects of mutations on protein function”, doi: 10.1101/2021.07.09.450648.
[22] M. E. Peters *et al.*, “Deep contextualized word representations,” Feb. 2018, [Online]. Available: http://arxiv.org/abs/1802.05365
[23] D. Ofer, N. Brandes, and M. Linial, “The language of proteins: NLP, machine learning & protein sequences,” *Comput Struct Biotechnol J*, vol. 19, pp. 1750–1758, Jan. 2021, doi: 10.1016/J.CSBJ.2021.03.022.
[24] A. Rives *et al.*, “Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences”, doi: 10.1073/pnas.2016239118/-/DCSupplemental.
[25] R. Rao *et al.*, “Evaluating Protein Transfer Learning with TAPE”, doi: 10.1101/676825.
[26] B. Hie, E. D. Zhong, B. Berger, and B. Bryson, “Learning the language of viral evolution and escape,” 2009.
[27] R. Rao, J. Meier, T. Sercu, S. Ovchinnikov, and A. Rives, “TRANSFORMER PROTEIN LANGUAGE MODELS ARE UNSUPERVISED STRUCTURE LEARNERS”, doi: 10.1101/2020.12.15.422761.
[28] C. T. Saunders and D. Baker, “Evaluation of structural and evolutionary contributions to deleterious mutation prediction,” *J Mol Biol*, vol. 322, no. 4, pp. 891–901, 2002, doi: 10.1016/S0022-2836(02)00813-6.
[29] I. Anishchenko *et al.*, “Protein tertiary structure prediction and refinement using deep learning and Rosetta in CASP14,” *Proteins: Structure, Function and Bioinformatics*, vol. 89, no. 12, pp. 1722–1733, Dec. 2021, doi: 10.1002/prot.26194.
[30] F. B. Fuchs, D. E. Worrall, V. Fischer, and M. Welling, “SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks,” no. 3, 2020, [Online]. Available: http://arxiv.org/abs/2006.10503
[31] J. Yang, I. Anishchenko, H. Park, Z. Peng, S. Ovchinnikov, and D. Baker, “Improved protein structure prediction using predicted interresidue orientations,” *Proceedings of the*

National Academy of Sciences, vol. 117, no. 3, pp. 1496–1503, 2020, doi: 10.1073/pnas.1914677117.
[32] G. D. Friedland and T. Kortemme, "Designing ensembles in conformational and sequence space to characterize and engineer proteins," *Curr Opin Struct Biol*, vol. 20, no. 3, pp. 377–384, 2010, doi: 10.1016/j.sbi.2010.02.004.
[33] J. E. Shin *et al.*, "Protein design and variant prediction using autoregressive generative models," *Nat Commun*, vol. 12, no. 1, 2021, doi: 10.1038/s41467-021-22732-w.
[34] T. A. Hopf *et al.*, "Mutation effects predicted from sequence co-variation," *Nat Biotechnol*, vol. 35, no. 2, 2017, doi: 10.1038/nbt.3769.
[35] R. Rao *et al.*, "MSA Transformer," 2021. [Online]. Available: https://github.com/facebookresearch/
[36] L. M. Starita and S. Fields, "Deep mutational scanning: A highly parallel method to measure the effects of mutation on protein function," *Cold Spring Harb Protoc*, vol. 2015, no. 8, pp. 711–714, Aug. 2015, doi: 10.1101/pdb.top077503.
[37] M. Mirdita, L. von Den Driesch, C. Galiez, M. J. Martin, J. Soding, and M. Steinegger, "Uniclust databases of clustered and deeply annotated protein sequences and alignments," *Nucleic Acids Res*, vol. 45, no. D1, pp. D170–D176, Jan. 2017, doi: 10.1093/nar/gkw1081.
[38] M. Steinegger, M. Mirdita, and J. Söding, "Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold," *Nat Methods*, vol. 16, no. 7, pp. 603–606, Jul. 2019, doi: 10.1038/s41592-019-0437-4.
[39] M. Steinegger, M. Meier, M. Mirdita, H. Vöhringer, S. J. Haunsberger, and J. Söding, "HH-suite3 for fast remote homology detection and deep protein annotation," *BMC Bioinformatics*, vol. 20, no. 1, Sep. 2019, doi: 10.1186/s12859-019-3019-7.
[40] B. Qian, A. R. Ortiz, and D. Baker, "Improvement of comparative model accuracy by free-energy optimization along principal components of natural structural variation," *Proceedings of the National Academy of Sciences*, vol. 101, no. 43, pp. 15346–15351, 2004, doi: 10.1073/pnas.0404703101.
[41] R. R. Eguchi and P. S. Huang, "Multi-scale structural analysis of proteins by deep semantic segmentation," *Bioinformatics*, 2020, doi: 10.1093/bioinformatics/btz650.
[42] N. Anand and P. S. Huang, "Generative modeling for protein structures," in *Advances in Neural Information Processing Systems*, 2018.
[43] J. Ingraham, V. K. Garg, R. Barzilay, and T. Jaakkola, "Generative models for graph-based protein design," in *Advances in Neural Information Processing Systems*, 2019.
[44] R. R. Eguchi, N. Anand, C. A. Choe, and P.-S. Huang, "IG-VAE: GENERATIVE MODELING OF IMMUNOGLOBULIN PROTEINS BY DIRECT 3D COORDINATE GENERATION," *bioRxiv*, 2020.
[45] Y. Song *et al.*, "High-resolution comparative modeling with RosettaCM," *Structure*, vol. 21, no. 10, pp. 1735–1742, 2013, doi: 10.1016/j.str.2013.08.005.
[46] A. Zemla, "LGA: A method for finding 3D similarities in protein structures," *Nucleic Acids Res*, vol. 31, no. 13, pp. 3370–3374, 2003, doi: 10.1093/nar/gkg571.
[47] H. Park *et al.*, "Simultaneous Optimization of Biomolecular Energy Functions on Features from Small Molecules and Macromolecules," *J Chem Theory Comput*, vol. 12, no. 12, pp. 6201–6212, 2016, doi: 10.1021/acs.jctc.6b00819.

[48] V. Mariani, M. Biasini, A. Barbato, and T. Schwede, “IDDT: A local superposition-free score for comparing protein structures and models using distance difference tests,” *Bioinformatics*, vol. 29, no. 21, pp. 2722–2728, 2013, doi: 10.1093/bioinformatics/btt473.
[49] H. Park, S. Ovchinnikov, D. E. Kim, F. DiMaio, and D. Baker, “Protein homology model refinement by large-scale energy optimization,” *Proc Natl Acad Sci U S A*, 2018, doi: 10.1073/pnas.1719115115.
[50] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, 2014.
[51] J. A. Nelder and R. Mead, “A Simplex Method for Function Minimization,” *Comput J*, vol. 7, no. 4, pp. 308–313, Jan. 1965, doi: 10.1093/comjnl/7.4.308.
[52] I. Borg and P. Groenen, “Modern Multidimensional Scaling: Theory and Applications,” *J Educ Meas*, 2003, doi: 10.1111/j.1745-3984.2003.tb01108.x.
[53] S. M. Larson, J. L. England, J. R. Desjarlais, and V. S. Pande, “Thoroughly sampling sequence space: Large-scale protein design of structural ensembles,” *Protein Science*, vol. 11, no. 12, pp. 2804–2813, Apr. 2009, doi: 10.1110/ps.0203902.
[54] F. Ding and N. V. Dokholyan, “Emergence of protein fold families through rational design,” *PLoS Comput Biol*, vol. 2, no. 7, 2006, doi: 10.1371/journal.pcbi.0020085.
[55] C. A. Smith and T. Kortemme, “Backrub-Like Backbone Simulation Recapitulates Natural Protein Conformational Variability and Improves Mutant Side-Chain Prediction,” *J Mol Biol*, vol. 380, no. 4, 2008, doi: 10.1016/j.jmb.2008.05.023.
[56] D. J. Mandell, E. A. Coutsias, and T. Kortemme, “Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling,” *Nature Methods*, vol. 6, no. 8. 2009. doi: 10.1038/nmeth0809-551.
[57] R. Spencer-Smith and J. P. O’Bryan, “Direct inhibition of RAS: Quest for the Holy Grail?,” *Seminars in Cancer Biology*, vol. 54. 2019. doi: 10.1016/j.semcancer.2017.12.005.
[58] D. Liu, Y. Mao, X. Gu, Y. Zhou, and D. Long, “Unveiling the ‘invisible’ druggable conformations of GDP-bound inactive Ras”, doi: 10.1073/pnas.2024725118/-/DCSupplemental.
[59] T. Daniel and A. Tamar, “Soft-IntroVAE: Analyzing and Improving the Introspective Variational Autoencoder,” Dec. 2020, [Online]. Available: http://arxiv.org/abs/2012.13253
[60] I. J. Goodfellow *et al.*, “Generative Adversarial Networks,” Jun. 2014, [Online]. Available: http://arxiv.org/abs/1406.2661
[61] H. Park, G. Zhou, M. Baek, D. Baker, and F. Dimaio, “Force Field Optimization Guided by Small Molecule Crystal Lattice Data Enables Consistent Sub-Angstrom Protein-Ligand Docking,” *J Chem Theory Comput*, vol. 17, no. 3, 2021, doi: 10.1021/acs.jctc.0c01184.
[62] E. F. Pettersen *et al.*, “UCSF Chimera - A visualization system for exploratory research and analysis,” *J Comput Chem*, vol. 25, no. 13, 2004, doi: 10.1002/jcc.20084.

VITA

Sanaa Mansoor grew up in Islamabad, Pakistan, and completed her undergraduate degree at Mount Holyoke College, MA in 2018. At Mount Holyoke, she majored in computer science and chemistry while working in the Rotello Lab at UMass Amherst for most of her undergraduate career. In the Rotello Lab, Sanaa was exploring the use of biodegradable nanocomposites for combating multidrug-resistant bacteria. During her junior year of college, she did an internship at Novartis Institute for Biomedical Research (NIBR) in Emeryville, CA. During her internship, she was working on optimizing an improved method for kinase docking and scoring and it was here where she got her first introduction to machine learning. She joined the Molecular Engineering Department at the University of Washington and joined the Baker lab as a graduate research assistant. During her PhD, her work has been focused on using deep learning methods (generative models, supervised and unsupervised training) for different protein design tasks. During 2021, she interned at Microsoft Research, where she worked on generating a joint representation of proteins using both sequence and structure for the task of protein thermal stability. She is excited about the growing use and increasing accuracy of deep learning methods towards applications in protein design.