PRO-LDM: Protein Sequence Generation with Conditional Latent Diffusion Models

Zixuan Jiang[1]|, Sitao Zhang[1,2]|, Rundong Huang[3], Shaoxun Mo[2], Letao Zhu[2], Peiheng Li[4], Ziyi Zhang[1], Xi Chen[5], Yunfei Long[5], Renjing Xu[1]*, Rui Qing[2]*

[1]Function Hub, The Hong Kong University of Science and Technology (Guangzhou), Guangdong, China

[2]State Key Laboratory of Microbial Metabolism, School of Life Sciences and Biotechnology, Shanghai Jiao Tong University, Shanghai, China

[3]Department of Mathematics, School of Computation, Information and Technology, Technical University of Munich, Munich, Germany

[4]Electronic and Information Engineering, School of Electronic Information Engineering, Beihang University, Beijing, China

[5]XtalPi, Inc., New Area Free Trade No. 1 Life Science and Technology Industrial Park Phase 1 No. 2-1, Pudong District, Shanghai, China

*Corresponding authors: Rui Qing, ruiqing.br@sjtu.edu.cn; Renjing Xu, renjingxu@hkust-gz.edu.cn

Abstracts

Protein design with deep-learning based algorithms represents an emerging and highly promising field in molecular biology. Yet it remains a challenging task due to the complexity of protein sequences. Recent developments in deep generative models such as diffusion model have demonstrated impressive performance in protein structure design, yet left much to be desired in creating new sequences. Here we present PRO-LDM: an efficient protein sequence design framework that utilizes diffusion model in the latent space, utilizing the joint autoencoder as the backbone, to capture latent variable distribution and generate meaningful embeddings of protein sequences. PRO-LDM is capable of (1) learning feature representations in natural proteins both at amino-acid level and sequence level; (2) generating new sequences with natural properties and high diversity; and (3) conditionally designing new protein sequences with designated fitness. Our model presents a feasible pathway a generalizable new tool to extract both physical and chemical properties of amino acids, as well as evolutionary information within the protein sequences, for protein sequence design and functional optimization.

Introduction

The field of protein engineering has undergone significant developments in multiple fields including site-directed mutagenesis, rational design(1), directed-evolution(2-4), semi-rational design(5-7), and computational design(8-10). Compared with traditional protein design methods, computational protein design can provide clear modification plans, greatly reduce the workload required to establish and screen mutant libraries, and made remarkable progresses in protein de novo design(11-13), enzyme substrate selectivity(14-16), thermal stability design(17-19), etc.

In recent years, deep neural networks have been introduced into the field of protein engineering(8,11,20-22). Notable efforts of such include sequence optimization models represented by ReLSO(23), and deep generation models represented by VAE(24), ProteinGAN(25),

and ProGen(26,27). Sequence optimization models focus on training with labeled datasets containing fitness, building a well-organized latent space, and optimizing search algorithms to obtain the best-performing variants. However, this type of model is limited to datasets with fitness labels and not applicable to the vast sequence space of proteins. On the other hand, deep generative models have powerful generation capabilities and can design novel functional variants highly similar to natural proteins when trained on large unlabeled sequence datasets. However, this type of model requires huge computational power, involving large datasets and considerable computation time.

Inspired by the recent success of diffusion models(28-31) in generating images(32-34) and text(35,36), as well as diffusion model in latent space(31,32) for improving computing speed and generation capabilities, we here present PRO-LDM (Protein Sequence Generation with Latent Conditional Latent Diffusion Models), a multi-task learning model to accomplish both fitness prediction and sequence generation tasks within a single algorithmic framework. It has significantly lower requirements in computational time and power, storage space, and maintenance costs compared to other single-task models(37,38). We assessed the capacity of PRO-LDM for representation learning by visualizing the latent space, both at the amino acid level and protein level. Subsequently, we conducted training and sequence generation using PRO-LDM, utilizing either labeled and unlabeled data respectively. Furthermore, we validated the generated sequences by assessing their similarity, local and global amino acid relationships, and functionality.

Results

1. The architecture of PRO-LDM

PRO-LDM is based on a jointly trained autoencoder (JT-AE) with Latent Conditional Diffusion (LCD) module, which is tailored to learn fundamental patterns from protein sequences. When using protein sequence dataset with fitness labels, PRO-LDM achieves conditional generation of specific protein sequences and fitness prediction. When the labels are uniformly set to 0, PRO-LDM can also achieve unsupervised learning tasks on unlabeled datasets and generate protein sequences with similar properties to the training sets.

JT-AE

JT-AE, the fundamental structure of ReLSO, is a combination of supervised and unsupervised learning in an autoencoder framework. The encoder consists of a transformer model in the encoder, integrated with a CNN-based decoder and a MLP-based regressor in parallel. The output of encoder is subjected to additional dimensionality reduction through a bottleneck module composed of fully connected layers, resulting in latent variable z. The collection of all z values constitutes the latent space. Latent variable z, along with the labels representing sequence fitness, are simultaneously passed to the diffusion module to simulate the distribution in the latent space, which will be elaborated upon in the following section.

Latent Conditional Diffusion Models (LDM)

The main purpose of LDM is to learn the data distribution in the latent space of sequences and to capture the characteristics of sequences with different fitness. Our LDM employs a UNet(39) as the neural network backbone and utilizes DDPM-type ancestral sampling for the generation process.

During the training phase, given an input sequence x, the encoder f_θ encodes x into a latent representation $z = f_\theta(x)$, and the decoder g_θ reconstructs the sequence from the latent variable, giving $\tilde{x} = g_\theta(z) = g_\theta(f_\theta(x))$. In the latent space, we divide the diffusion process into T steps and add Gaussian noise according to a variance schedule $\beta_1, \dots, \beta_T$:

$$q(z_{1:T}|z_0) \coloneqq \prod_{t=1}^{T} q(z_t|z_{t-1}), q(z_t|z_{t-1}) \coloneqq \mathcal{N}\left(z_{t-1}; \sqrt{1-\beta_t} x_{t-1}, \beta_t I\right) \quad (1)$$

Where $z_0 \sim p(z_0)$. the reverse process can be defined as a Markov chain, starting at $p_\theta(z_T) = \mathcal{N}(0, I)$:

$$p_\theta(z_{0:T}) \coloneqq p(z_T) \prod_{t=1}^{T} p_\theta(z_{t-1}|z_t), p_\theta(z_{t-1}|z_t) \coloneqq \mathcal{N}(z_{t-1}; \mu_\theta(z_t, t), \textstyle\sum_\theta(z_t, t)) \quad (2)$$

β_t is kept constant as a hyperparameter, and $\alpha_t \coloneqq 1 - \beta_t, \bar{\alpha}_t \coloneqq \prod_{s=1}^{T} \alpha_s$, and $\textstyle\sum_\theta(z_t, t) = \sigma_t^2 I$, where $\sigma_t^2 = \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$. Consequently, we can sample $z_{t-1} \sim p_\theta(z_{t-1}|z_t)$ using the following equation:

$$z_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(z_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(z_t, t)\right) + \sigma_t \gamma, \qquad \gamma \sim \mathcal{N}(0, 1) \quad (3)$$

where ϵ_θ is a function approximator designed to predict ϵ from z_t. In the case of conditional sampling, the latent variable z is drawn along with class label c, so that the function approximator is changed into $\epsilon_\theta(z_t, t, c)$. We jointly train an unconditional diffusion model, denoted as $p_\theta(z)$, which is parameterized through a function approximator $\epsilon_\theta(z_t, t)$ together with the conditional model $p_\theta(z|c)$ parameterized through $\epsilon_\theta(z_t, t, c)$, by randomly setting c to the unconditional class identifier $\emptyset$ with a probability p_{uncond} as a hyperparameter:

$$\tilde{\epsilon}_\theta(z_t, t, c) = (1 + w)\epsilon_\theta(z_t, t, c) - w\epsilon_\theta(z_t, t) \quad (4)$$

where w is a hyperparameter that controls the strength of the classifier-free guidance. Consequently, we can train the latent diffusion model using the following objective:

$$L_{LDM} \coloneqq E_{\varepsilon(x), c, \epsilon \sim \mathcal{N}(0,1), t}\left[\|\epsilon - \tilde{\epsilon}_\theta(z_t, t, c)\|_2^2\right] \quad (5)$$

As one of the latest generative models capable of generating high-quality samples while ensuring diverse sample distributions, diffusion models are inherently computationally expensive. To address this issue, our model uses diffusion in the latent space to significantly reduces the dimensionality of the data, compared to other models that use diffusion for sequence generation directly. This architecture greatly accelerates the generation speed and improving model efficiency. Moreover, we circumvent the need for classifier-guided diffusion, which requires an additional pretrained classifier that increases model complexity. Instead, our classifier-free guided diffusion combines conditional and unconditional diffusion for joint training, striking a good balance between model complexity and computational costs, while achieving impressive generation results.

We train our model by minimizing the loss below.

$$L = \|g_\theta(f_\theta(x)) - x\| + \|h_\theta(f_\theta(x)) - y\| + E_{\varepsilon(x), c, \epsilon \sim \mathcal{N}(0,1), t}\left[\|\epsilon - \tilde{\epsilon}_\theta(z_t, t, c)\|_2^2\right] \quad (6)$$

The first and second term gauge the loss of JT-AE, where f_θ represents the encoder, g_θ represents the decoder, h_θ represents the regressor, x is the input sequence and y is the corresponding fitness. The third term measures the loss of the latent diffusion model which have been detailly

described above.

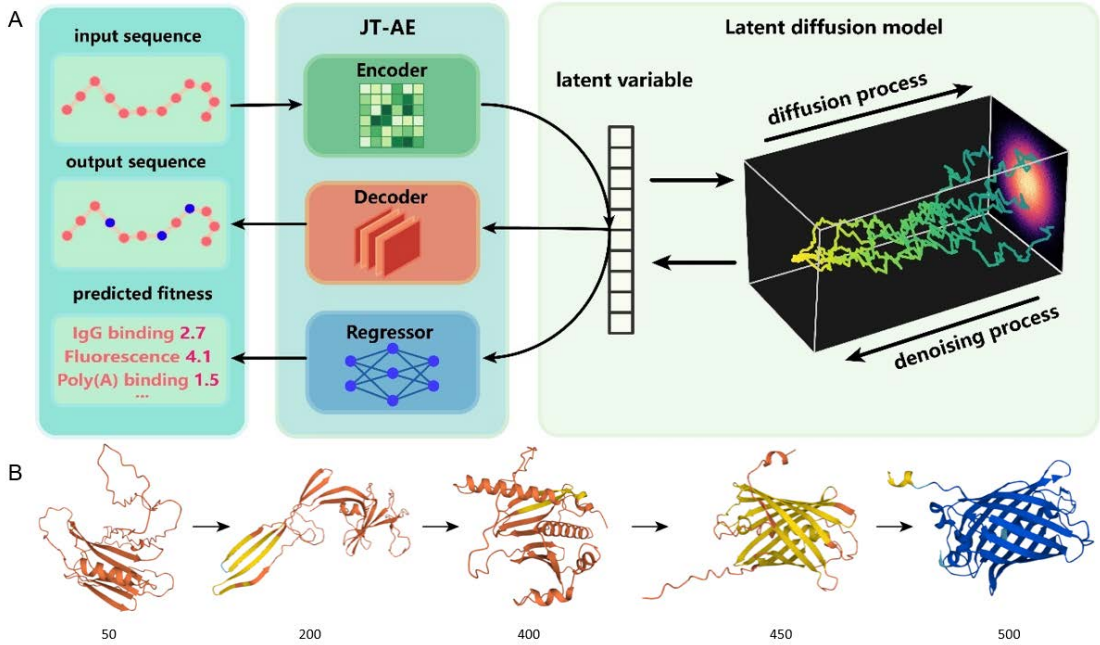

Figure 1 (A) Overview of the PRO-LDM model architecture. In the training stage, input sequences are mapped into the latent space via a transformer-based encoder called JT-AE. A latent diffusion model (LDM) is then applied to capture the distribution of the latent space through the diffusion process. Finally, the latent variable is used to reconstruct the sequence via a CNN-based decoder and simultaneously predict the fitness via an MLP-based regressor. In the sampling stage, the latent variable of the optimized sequence is generated via the denoising process of LDM starting from a simple noise distribution. The output sequence and predicted fitness are obtained using the decoder and regressor, respectively. **(B) 3D-structure iteration of the generated protein during the denoising process**. We select several representative time steps in the denoising process trained on the Gifford dataset and decode the latent variables into sequences. AlphaFold2 is implemented to predict the 3D structures of the generated sequences to visualize the modeling process.

2. PRO-LDM learned meaningful representations of protein sequences

Our model was first trained unconditionally on protein datasets without associated fitness values, to capture inherent property or function in sequences. PRO-LDM generated new variants of proteins by learning information embedded solely in sequences. The model was trained respectively on three datasets, including two datasets on the homologs of bacterial luciferase procured from InterPro (IPR011251), and one dataset on a family of bacterial MDH enzymes (EC1.1.1.37). The luciferase protein sets were divided into two different forms, namely, Luciferase_MSA and Luciferase_RAW, to further scrutinize how multiple sequence alignment (MSA) can affect the learning efficacy and generative performance of this model. MSA was performed for all data in Luciferase_MSA, where a gap character was assigned to all unfilled positions for training purposes. Sequences in Luciferase_RAW were aligned to the N-terminus, and additional padding character was added to the C-terminal end to ensure all sequences the same length.

The propensity and distribution of the 20 amino acids in a given sequence defined the structure and function for proteins. Hence, it was essential for a learning algorithm to capture the intricate

biochemical attributes of amino acids in the latent space. In general, amino acids with similar physicochemical properties or side chain structures tended to exhibit stronger correlations, surpassing the level of correlation observed among dissimilar amino acid types (Figure 2a, Supplementary Table 1). The representation of each amino acid then processed through the principal component analysis (PCA) dimension reduction algorithm and compressed into 2D space for visualizations. Figure 2b shows results for all three datasets, where amino acids more alike were positioned near each other, such as charged acidic and basic amino acids. Amino acids with distinct biochemical properties were spaced further apart, such as nonpolar and polar amino acids. The observation demonstrates that our model was able to extract the characteristics of amino acids solely from their appearances in sequences.

Using the luciferase dataset as an example, we further evaluated the efficiency of our model on learning comprehensive representations of protein sequences. Luciferase proteins could be classified into different subfamilies based on diverse folds, the information of which was extracted from InterPro and only the 9 largest subfamilies were selected. Sequence embedding and family information were then visualized in Figure 2c, where results for both Luciferase_MSA and Luciferase_RAW were shown. Apparent clustering was observed for sequences belonging to the same subfamily for both training sets. The processing on MDH, Luciferase_MSA and Luciferase_RAW training sets clearly shows that our model not only captured characteristics of amino acid embedded in their appearance within sequences, but also grasped overall attributes of sequences as a whole that defined the properties and functions of proteins, which laid the foundation for subsequent designs.

High-throughput molecular biology techniques such as deep mutation scanning, directed-evolution, saturation mutations, and random mutations were valuable tools for studying the intricate relationship between genotype and phenotype. These techniques yielded large amounts of "function-sequence" data used to investigate how genetic changes affect the function of proteins or organisms. In our particular study, 9 mutation datasets were used to train PRO-LDM. These datasets were selected to include sequences with equal or unequal lengths, as well as both indels (insertions/deletions) and substitutions by incorporating diverse types of mutations (Methods; supplementary Table 2). Dimensionality reduction of sequence embeddings was performed by PCA to a 2D space, while fitness values were presented by different color. As shown in Figure 2d, except for B1LPA6 and pab1, latent spaces of other datasets exhibited global organization to fitness, which laid the foundation for directed sequence generation of functional variants. PRO-LDM was also capable of predicting the fitness for protein sequences. It outperformed JT-AE in 4 out of 9 datasets and demonstrated high accuracy in the remaining datasets, albeit with slightly lower values. (Supplementary Table 3).

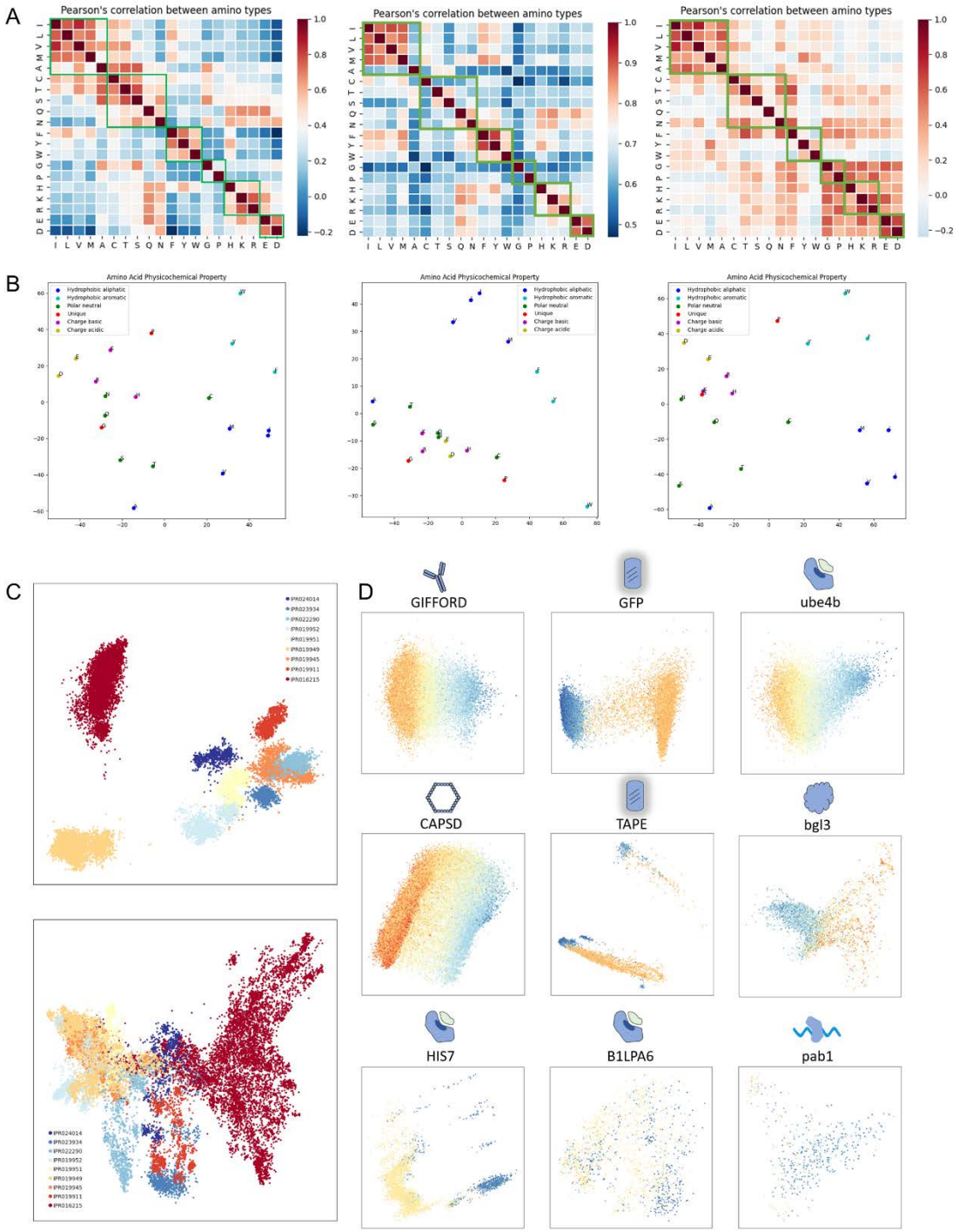

Figure 2 (A) Pearson's correlation between amino acid types. For each output sequence, we generated a 2D matrix representing the likelihood of different amino acids occurring at each residue position. Pearson's correlation was then calculated to measure the relationship between each amino acid pair. Biochemically related amino acids exhibit high correlation. Amino acid pairs within the same green border shared similar characteristics. Dataset from left to right: Luciferase_MSA, Luciferase_RAW, MDH. **(B) PCA of amino-acid embeddings learned by PRO-LDM**. (n = 20 amino acids, dataset from left to right: Luciferase_MSA, Luciferase_RAW, MDH.) **(C) Organization of latent space reflected functional groups**. Visualization illustrates the latent representation of sequences in Luciferase_MSA (top) and

Luciferase_RAW (bottom), projected onto the first two principal components and colored by sub-family annotations derived from InterPro. Only sequences belonging to the 9 largest subfamilies were shown. **(D) PRO-LDM learns smooth representations of protein sequences in labeled datasets**. We displayed the latent embeddings of 9 labeled datasets learned by PRO-LDM. The protein sequence representations were visualized using principal component analysis, and each point was colored according to its corresponding fitness value. From blue to orange color: high fitness to low fitness.

3. PRO-LDM generated new protein variants that maintain intrinsic relationship of natural sequences

The core objective of generation model was to produce data with a similar distribution to the original data. Herein, we first examined the similarity between generated and natural sequences, which was calculated by the proportion of identical residues in both sequence sets. During the training process, 64 sequences were generated every 50 epochs. As the training steps increasing, a concomitant augmentation in similarity was observed (Figure 3a, top). In comparison, the similarity between sequences generated by VAE (Figure 3a, bottom) and the training sequences did not exhibit a significant increase as training progresses, and remained lower than those generated by PRO-LDM.

The Shannon entropy of protein sequences reflected their positional variability, while highly conserved sites were more likely to be associated with the conformation and function of proteins, due to the natural of biological evolution (40,41). They were calculated for each site in the MSAs of generated and training sets. In Figure 3b, it was observed that sequences generated by PRO-LDM exhibited highly similar Shannon entropy to the training set (Supplementary Table 4). In order to investigate whether sequences generated by PRO-LDM maintained the functionality of natural proteins, InterPro was used to predict the functional sites and marked them in the logo figure. Taking MDH as an example, we observed remarkably similar and conserved logo figures between amino acid compositions of functional sites in the training set and the generated set (Figure 3c).

Residues accounting for the functionality of a protein might be spatially adjacent in the folded state, while still being far apart in its primary sequence. Since our model analyzed the sequences from a linear manner, it was important that the algorithm could grasp the relationship of amino acid pairs remote on the primary sequence. To explore such global amino acid relations, we calculated pairwise amino acid frequency distributions for all position pairs across all sequences within MSAs. Subsequently, we determined the correlation of these frequency distributions in both sets. As shown in Figure 3d, the amino acid pairwise relationship between the natural set and PRO-LDM generated set exhibited notable similarity (Supplementary table 5). Furthermore, we randomly selected 10 sequences from MDH generated set and examined whether these sequences have two domains ('Ldh_1_N' and 'Ldh_1_C') mentioned in Pfam(42), each containing more than 100 amino acids. The results showed that 9 of 10 sequences had these two domains and one of them only had 'Ldh_1_C', which demonstrated that PRO-LDM preserved long-distance amino-acid relationships of MDH (Supplementary figure 1).

We then explored whether PRO-LDM could expand the known sequence space and generated more diverse protein sequences, which is a pivotal concern in protein engineering. Across three datasets used above, the diversity of sequences generated by PRO-LDM significantly exceeded the

diversity of sequences generated by VAE and training set at same sequence identity level (Figure 3e). The stability of PRO-LDM generated sequences was equivalent to the training set for MDH and Luciferase_MSA (Figure 3f) and slightly lower than the training set for Luciferase_RAW. Simultaneously, we determined the proportions of different types of amino acids in those datasets (Supplementary Figure 2). The composition of amino acid for generated sequences was similar to the natural ones. Therefore, we demonstrated that PRO-LDM could perform protein sequence design with higher diversity, while maintaining the stability, evolutionary and physicochemical constraints from sequences in the training data at the same time (Figure 3c, Supplementary Figure 2).

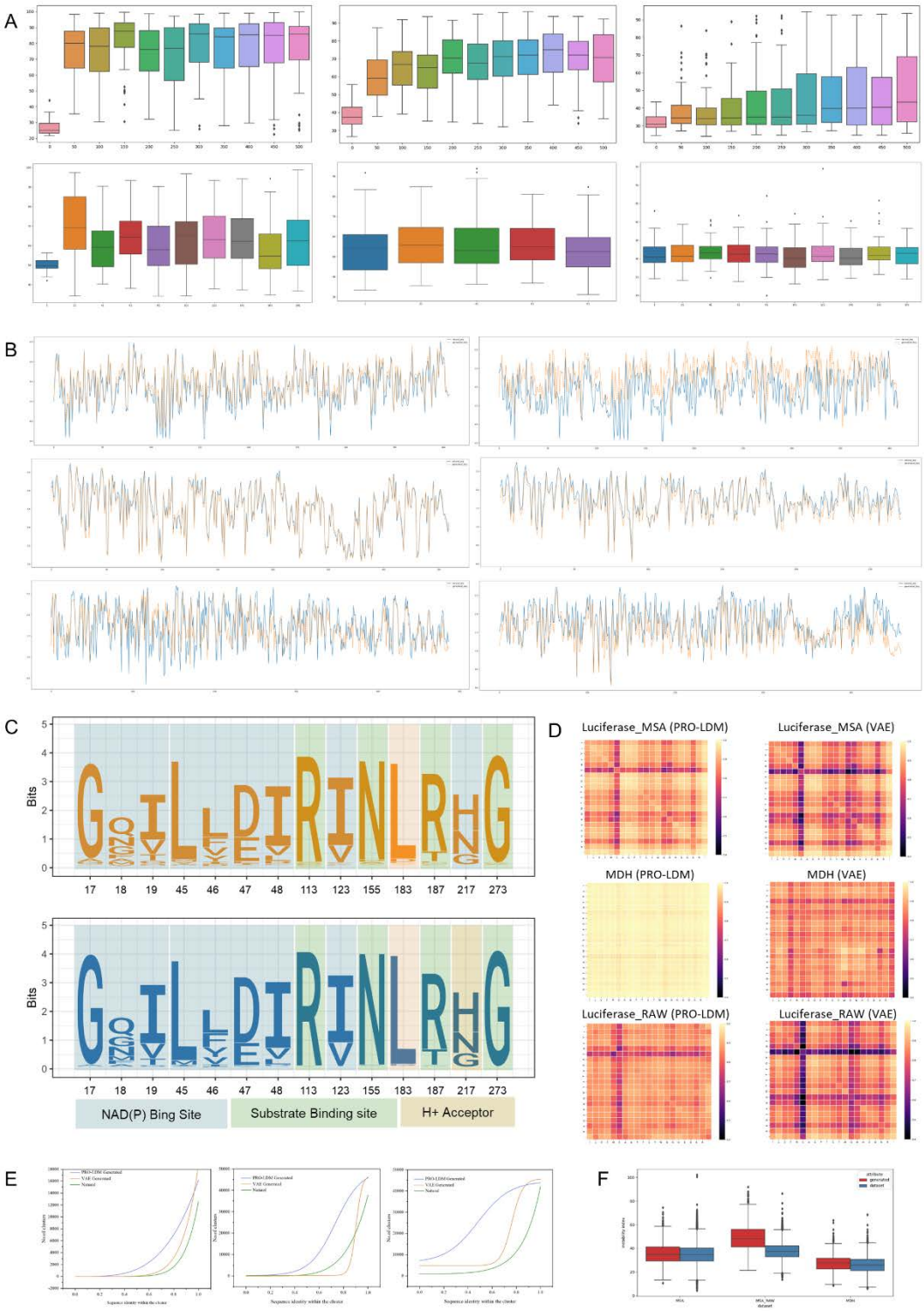

Figure 3 (A) Sequence identity of 64 generated sequences to the nearest natural sequence from training data at different training iterations. (B) PRO-LDM effectively captured the amino-acid distribution in natural sequences. Sequence variability was represented by Shannon entropies for generated and training sequences estimated from MSAs. Low Shannon entropy represents highly conserved and thus functionally relevant sites, whereas high entropy indicated high amino-acid diversity at a given position. **(C) A sequence logo of key conserved positions in the MSA of MDH datasets. (D) Amino-acid pair correlations of generated and natural sequences**. Every point on the

map represented the correlation of amino-acid pair frequencies between the natural sequences of Luciferase_MSA, Luciferase_RAW, MDH and those generated by PRO-LDM and VAEs. High correlation denoted that the same pairwise long-distance amino-acid interactions were found as in natural sequences. **(E) Comparison of sequence diversity of natural and generated sequences for three datasets** (left: MDH; mid: Luciferase_MSA; right: Luciferase_RAW). **(F) Comparison of sequence stability of natural and generated sequences for three datasets** (left: Luciferase_MSA; mid: Luciferase_RAW; right: MDH).

4. PRO-LDM designed novel functional variants

Since PRO-LDM was capable of grasping the functional residue conservation in natural sequences and generative new variants with increased diversity, we further explored whether adding a conditional diffusion module for semi-supervised training on labeled datasets could enable the model to design new protein variants with tailored properties and functions.

Firstly, we recorded the changes in the property value of the sequence during the generation process. As the generation process progresses, with an exception of pab1 dataset, sequence fitness was observed to continuously increases and tend to converge. The observation aligned with the principle of the diffusion model, which commenced with random noises and gradually eliminated them until an image was generated (Figure 4a, Figure 1a, Figure 1b). Given the need for enhanced manipulability of protein characteristics during the design phase, we sought to derive variants with diverse properties by altering the input labels. It was observed that sequences generated under different labels had similar distributions in the latent space to the labeled sequences in the training set (Figure 4b; Supplementary Figure 3), while the correspondingly predicted fitness were also similar (Figure 4c; Supplementary Figure 4).

One critical aspect of protein engineering was to design novel sequences with superior functional performance as compared to native proteins. Using the GFP dataset as an example, we employed XFitness (a zero-shot fitness prediction platform) and Tranception (a transformer-based state-of-the-art fitness prediction model) to predict the fitness of the generated sequences. As shown in supplementary Figure 5, the fitness of 63/64 sequences predicted by XFitness and 64/64 sequences predicted by Tranception was higher than or equal to the wild type protein. Furthermore, we conducted statistical analysis on the mutation sites data of the generated sequences and the training set. The sequences generated by PRO-LDM could achieve multi-site mutations, and the distribution of mutation numbers was similar to that of the training data, and could generate sequences with mutation numbers not present in the training set (Figure 4d). Compared to traditional protein optimization processes that focused on single-site mutations or combinations of several mutation sites, PRO-LDM adopted a *de novo* design approach that break through the boundary of natural protein space exploration to a greater extent, expanded the search space, and improved design efficiency.

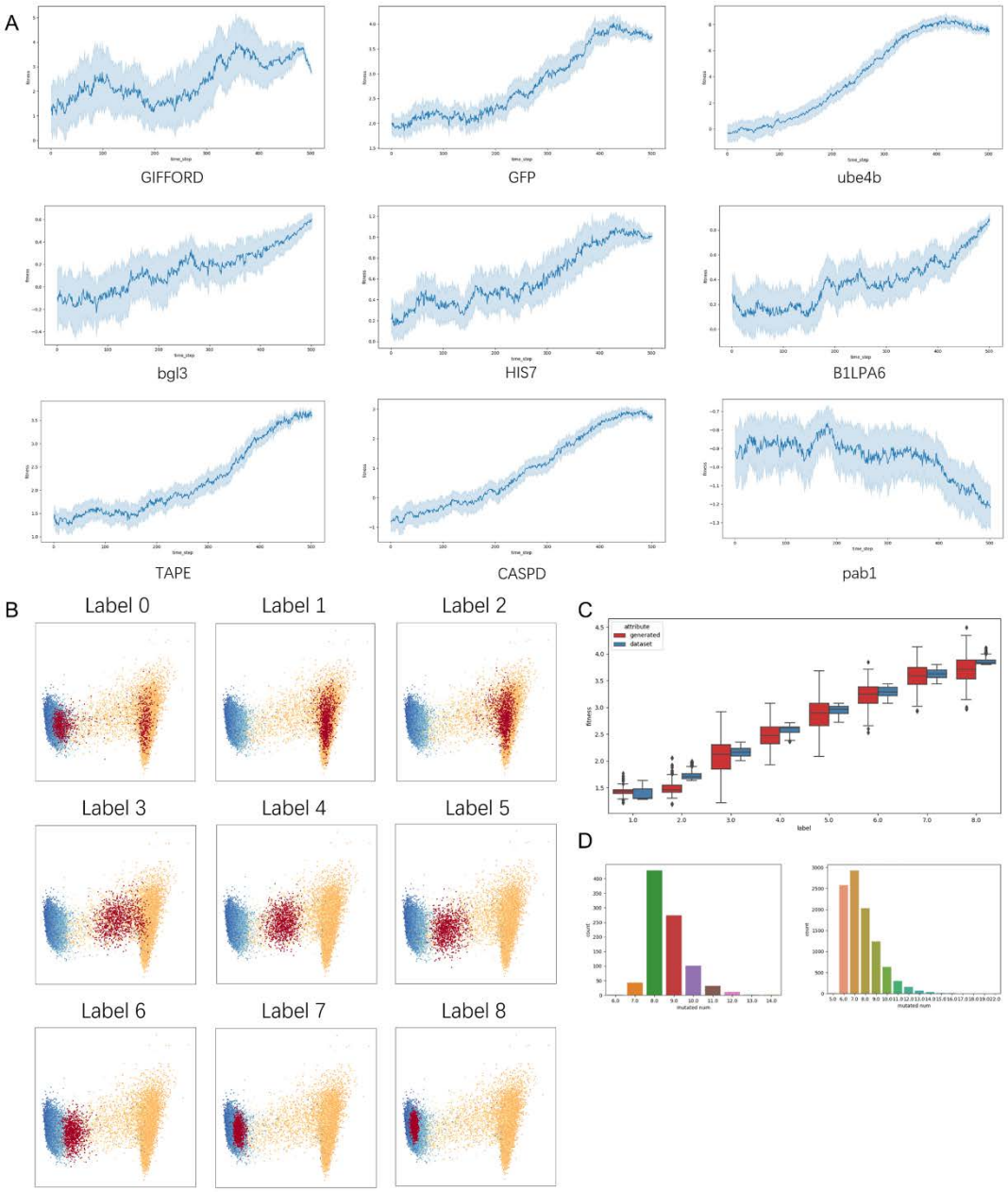

Figure 4 (A) Fitness transformation process of conditional protein sequence generation. To visualize the process of how protein fitness converges into the targeted area, we conditionally generated high-fitness protein sequences in 9 labeled datasets and predict the protein fitness via the pre-trained regressor using the latent variable generated during the denoising process. **(B) Natural sequences of GFP dataset and conditionally generated protein sequences visualized in latent space**. We divided the sequences in GFP dataset into 8 labels according to their fitness and conditionally generated sequences for each label. Generated and natural sequences of each label were mapped into latent space and visualized with PCA. **(C) Fitness distribution of natural and generated sequences for GFP dataset in each label**. We generated 64 sequences of GFP dataset for each label and predicted their fitness using the regressor module in JT-AE. Their fitness distribution was compared to that of natural sequences. **(D) Mutational positions of natural and generated sequences**. We compared 5000 generated sequences and natural sequences in GFP dataset with a reference sequence and determined the number of mutated positions. Left: natural, right: generated.

Discussion

The vast expanse of sequence space poses significant challenges for protein design, particularly as the sequence length increases, leading to exponential growth. Deep learning methodologies offer an opportunity to uncover intrinsic associations within amino acid sequences from existing protein sequence datasets, avoiding the complexities of the vast sequence space. This approach is gaining traction as a prominent avenue for protein design, offering potential for generating a diverse array of functional variants that can be applied in biomedicine and biotechnology. In this context, our proposed PRO-LDM aims to discern commonalities and differences among amino acids in protein sequences, distinguish various functional proteins within the latent space, and systematically generate sequences with increased variability.

Diverse protein datasets exhibit distinct characteristics. Some datasets encompass mutation data, including fitness values from experimental results, while others might lack fitness values but feature retrievable homologous proteins from repositories like Uniprot. This diversity necessitates adaptable approaches for protein design and analysis. PRO-LDM, a latent diffusion model founded on a jointly trained autoencoder, effectively generates protein sequences akin to natural sequences, leveraging both unsupervised learning and the ability to produce functional variants under specific conditions.

Our initial assessment of PRO-LDM focuses on representation learning. Through comparisons of embeddings for amino acids with similar properties and visualizations of sequence representations within fitness landscapes, we demonstrate the model's proficiency in learning fundamental and functional representations from natural sequences. The evaluation of unconditionally generated sequences against natural sequences showcases similarities in terms of sequence identity, stability, local and global amino-acid relationships, and diversity.

In directed sequence generation, PRO-LDM generates protein sequences with specific fitness labels, displaying clustering with corresponding natural sequences in the fitness landscape. The generated sequences align with natural sequences in terms of fitness prediction and also exhibit multisite mutations, underscoring their diversity. Nevertheless, areas for improvement include the fitness prediction module's accuracy. Fitness scores might be influenced by factors like the sensitivity and specificity of high-throughput experiments and the characterization frequency of mutants. Thus, the reliability of fitness scores across datasets can vary, potentially impacting PRO-LDM's understanding of the sequence-function relationship. Further validation through diverse biological experiments is necessary to validate PRO-LDM's design capabilities from multiple perspectives.

The protein sequences generated by PRO-LDM present promising results in terms of rationality, diversity, and functional accuracy. The conditional generative model's potential to optimize proteins with various tags, such as EC numbers or disease types, hints at its application in protein engineering.

Methods

Neural network architecture details.

The backbone of our model is a jointly trained autoencoder (JT-AE) proposed in ReLSO, which comprises an encoder and a decoder. The encoder employs a four-head transformer with six hidden layers, each having a dimension of 200, to capture the interactions between residues in the sequence. A bottleneck module, consisting of a fully connected layer, is applied to compress the embedded information into the latent space. On the other hand, the decoder uses four 1D convolutional layers to reconstruct sequences from latent variables. Rectified Linear Unit (ReLU) activations and batch normalization layers are incorporated between convolutional layers, except for the final layer.

In parallel with the decoder, a multilayer perceptron (MLP)-based regressor, composed of two fully connected layers and featuring a dropout rate of 0.2, is employed to predict fitness. We leverage a classifier-free guided diffusion model between the encoder and decoder, which consists of a four-layer 1D-convolutional U-Net that captures the disturbed latent distribution of each diffusion step. This approach facilitates learning the latent space of the sequences and enables the conditional generation of z.

Network training data and Labeled fitness.

Since conditional diffusion necessitates discrete labels, yet the distribution of sequence fitness is continuous, we propose a method for assigning distinct labels to sequences based on various fitness ranges. Initially, we rounded the fitness values for label assignment. This approach offers the advantage of simplicity; however, it presents several drawbacks. For example, it may result in label scarcity or label abundance in datasets with dense fitness distributions or extensive fitness spans, respectively, subsequently impacting the generation performance significantly.

Through experimentation, we ultimately developed an 8-label division method. The procedure entails preprocessing each dataset and visualizing the relationship between dataset length and fitness distribution. Based on the findings, we establish boundaries for fitness and sequence length, and we uniformly divide the majority of datasets into eight segments, with a few datasets being divided into five segments. For the unsupervised task, the labels are uniformly set to 0.

Network training process and sequence generation.

During the training stage, sequence strings are fed into the encoder. First, the sequence is encoded using an embedding layer with a dimension of 100, followed by the transformer learning the interdependencies between residues. Subsequently, a bottleneck layer compresses the discrete, sparse, high-dimensional sequence information into a 64-dimensional latent variable z. All the z values constitute a continuous, condensed latent space of sequences. At this juncture, the fitness label is introduced into the conditional diffusion model as input to learn the distribution of information in the latent space. Subsequently, z serves as input for both the decoder and regressor, where the fitness label is reintroduced for supervised learning, facilitating sequence reconstruction and fitness prediction tasks.

The model is trained for 500 epochs utilizing the AdamW(43) optimizer with a cosine annealing learning rate starting at 0.00002 on four 32GB V100 graphics processing units and employing a batch size of 512.

Bioinformatic analysis of generated sequences.

***Sub-family information search and visualization*.**

To evaluate the capacity of the sequence-level representation generated by PRO-LDM in the latent space to distinguish the functionality of proteins, we obtained the sub-family accession number for luciferase (Figure 2c) from the InterPro database (https://www.ebi.ac.uk/interpro/). Following this, we utilized PRO-LDM to generate the latent representation for the nine largest sub-family proteins. This representation was subsequently reduced to a 2D space using principal component analysis (PCA) and visualized for further assessment.

***Identity of generated sequences and natural counterparts*.**

In order to assess the similarity between the generated sequences and natural sequences, 64 sequences were generated at 50-epoch intervals for three unlabeled datasets (MDH, luciferase_MSA, luciferase_RAW) throughout the training process. We employed BLAST (32) to conduct sequence alignment on the generated sequences and calculated their identities in comparison to the natural set, which is represented in box plots (Figure 3a).

***Multiple sequence alignments and Shannon entropy*.**

We randomly selected 1,000 training sequences and 1,000 generated sequences, which were then combined and input into Clustal Omega (33) for alignment. In cases where a mismatch occurred at a specific site, the non-matching position was replaced with a dash ("-"), referred to as a gap. Subsequently, the training and generated sequences were separated, and columns exhibiting a 75% gap ratio were removed. Finally, the Shannon entropies of both sets were calculated separately (Figure 3b).

The Shannon entropy was computed using the following equation:

$$SE = -\sum_{i=1}^{20} p\,(x_i) log_{20} p(x_i)$$

where $p(x_i)$ represents the frequency of amino acid i in one column of the multiple sequence alignment (MSA).

***Pairwise Amino-acid frequency distribution*.**

First, we selected 1,000 sequences from both the training set and the generated set. For each protein sequence, we computed pairwise amino acid occurrence frequency matrices of dimensions [seq_length, seq_length]. Next, we reshaped the matrices to [1, seq_length × seq_length] and concatenated all sequences in the training/generated set. As a result, we obtained two groups of metrics with dimensions [seq_num, seq_length × seq_length]. Finally, we employed these two groups of metrics to calculate the Pearson's correlation coefficient.

***Sequence diversity analysis*.** We use MMseqs2 (https://toolkit.tuebingen.mpg.de/tools/mmseqs2) to cluster sequences(44).

***XFitness and Tranception*.** XFitness is a zero-shot protein fitness prediction platform developed by

XtalPi, which has been extensively validated through numerous wet lab experiments conducted within internal Xffinity and XuperHuman platforms. A higher evaluated XFitness score signifies enhanced protein naturalness and developability, leading to an increased likelihood of achieving superior functionality. Tranception is a SOTA transformer-based state-of-the-art fitness prediction model(45).

Reference

1. Hellinga, H.W. (1997) Rational protein design: Combining theory and experiment. *Proceedings of the National Academy of Sciences*, **94**, 10015-10017.
2. Packer, M.S. and Liu, D.R. (2015) Methods for the directed evolution of proteins. *Nature Reviews Genetics*, **16**, 379-394.
3. Romero, P.A. and Arnold, F.H. (2009) Exploring protein fitness landscapes by directed evolution. *Nature Reviews Molecular Cell Biology*, **10**, 866-876.
4. Wang, Y., Xue, P., Cao, M., Yu, T., Lane, S.T. and Zhao, H. (2021) Directed Evolution: Methodologies and Applications. *Chem Rev*, **121**, 12384-12444.
5. Lutz, S. (2010) Beyond directed evolution—semi-rational protein engineering and design. *Current Opinion in Biotechnology*, **21**, 734-743.
6. Chen, Q., Xiao, Y., Shakhnovich, E.I., Zhang, W. and Mu, W. (2020) Semi-rational design and molecular dynamics simulations study of the thermostability enhancement of cellobiose 2-epimerases. *Int J Biol Macromol*, **154**, 1356-1365.
7. Chica, R.A., Doucet, N. and Pelletier, J.N. (2005) Semi-rational approaches to engineering enzyme activity: combining the benefits of directed evolution and rational design. *Current Opinion in Biotechnology*, **16**, 378-384.
8. Wang, J. (2022) Protein sequence design by deep learning. *Nature Computational Science*, **2**, 416-417.
9. Coluzza, I. (2017) Computational protein design: a review. *J Phys Condens Matter*, **29**, 143001.
10. Siegel, J.B., Smith, A.L., Poust, S., Wargacki, A.J., Bar-Even, A., Louw, C., Shen, B.W., Eiben, C.B., Tran, H.M., Noor, E. *et al.* (2015) Computational protein design enables a novel one-carbon assimilation pathway. *Proceedings of the National Academy of Sciences*, **112**, 3704-3709.
11. Anishchenko, I., Pellock, S.J., Chidyausiku, T.M., Ramelot, T.A., Ovchinnikov, S., Hao, J., Bafna, K., Norn, C., Kang, A., Bera, A.K. *et al.* (2021) De novo protein design by deep network hallucination. *Nature*, **600**, 547-552.
12. Gainza, P., Wehrle, S., Van Hall-Beauvais, A., Marchand, A., Scheck, A., Harteveld, Z., Buckley, S., Ni, D., Tan, S., Sverrisson, F. *et al.* (2023) De novo design of protein interactions with learned surface fingerprints. *Nature*, **617**, 176-184.
13. Lee, J.S., Kim, J. and Kim, P.M. (2023) Score-based generative modeling for de novo protein design. *Nature Computational Science*, **3**, 382-392.
14. Li, R., Wijma, H.J., Song, L., Cui, Y., Otzen, M., Tian, Y.e., Du, J., Li, T., Niu, D., Chen, Y. *et al.* (2018) Computational redesign of enzymes for regio- and enantioselective hydroamination. *Nature Chemical Biology*, **14**, 664-670.
15. Wu, L., Qin, L., Nie, Y., Xu, Y. and Zhao, Y.-L. (2022) Computer-aided understanding and engineering of enzymatic selectivity. *Biotechnology Advances*, **54**, 107793.

16. Wijma, H.J., Floor, R.J., Bjelic, S., Marrink, S.J., Baker, D. and Janssen, D.B. (2015) Enantioselective enzymes by computational design and in silico screening. *Angew Chem Int Ed Engl*, **54**, 3726-3730.
17. Huang, P., Chu, S.K.S., Frizzo, H.N., Connolly, M.P., Caster, R.W. and Siegel, J.B. (2020) Evaluating Protein Engineering Thermostability Prediction Tools Using an Independently Generated Dataset. *ACS Omega*, **5**, 6487-6493.
18. Wang, Y., Park, J.-H., Lupala, C.S., Yun, J.-H., Jin, Z., Huang, L., Li, X., Tang, L., Lee, W. and Liu, H. (2019) Computer aided protein engineering to enhance the thermo-stability of CXCR1- T4 lysozyme complex. *Scientific Reports*, **9**, 5317.
19. Yoshida, K., Kawai, S., Fujitani, M., Koikeda, S., Kato, R. and Ema, T. (2021) Enhancement of protein thermostability by three consecutive mutations using loop-walking method and machine learning. *Scientific Reports*, **11**, 11883.
20. Bennett, N.R., Coventry, B., Goreshnik, I., Huang, B., Allen, A., Vafeados, D., Peng, Y.P., Dauparas, J., Baek, M., Stewart, L. *et al.* (2023) Improving de novo protein binder design with deep learning. *Nature Communications*, **14**, 2625.
21. Baek, M. and Baker, D. (2022) Deep learning and protein structure modeling. *Nature Methods*, **19**, 13-14.
22. Wang, J., Cao, H., Zhang, J.Z.H. and Qi, Y. (2018) Computational Protein Design with Deep Learning Neural Networks. *Scientific Reports*, **8**, 6349.
23. Castro, E., Godavarthi, A., Rubinfien, J., Givechian, K., Bhaskar, D. and Krishnaswamy, S. (2022) Transformer-based protein generation with regularized latent space optimization. *Nature Machine Intelligence*, **4**, 840-851.
24. Hawkins-Hooker, A., Depardieu, F., Baur, S., Couairon, G., Chen, A. and Bikard, D. (2020) Generating functional protein variants with variational autoencoders. *PLoS Computational Biology*, **17**.
25. Repecka, D., Jauniskis, V., Karpus, L., Rembeza, E., Rokaitis, I., Zrimec, J., Poviloniene, S., Laurynenas, A., Viknander, S., Abuajwa, W. *et al.* (2021) Expanding functional protein sequence spaces using generative adversarial networks. *Nature Machine Intelligence*, **3**, 324-333.
26. Madani, A., Krause, B., Greene, E.R., Subramanian, S., Mohr, B.P., Holton, J.M., Olmos, J.L., Xiong, C., Sun, Z.Z., Socher, R. *et al.* (2023) Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*.
27. Nijkamp, E., Ruffolo, J., Weinstein, E.N., Naik, N. and Madani, A. (2022) ProGen2: Exploring the Boundaries of Protein Language Models. 10.48550/arXiv.2206.13517.
28. Ho, J., Jain, A. and Abbeel, P. (2020) Denoising Diffusion Probabilistic Models. 10.48550/arXiv.2006.11239.
29. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N. and Ganguli, S. (2015) Deep Unsupervised Learning using Nonequilibrium Thermodynamics. 10.48550/arXiv.1503.03585.
30. Song, Y. and Ermon, S. (2019) Generative Modeling by Estimating Gradients of the Data Distribution. 10.48550/arXiv.1907.05600.
31. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S. and Poole, B. (2020) Score-Based Generative Modeling through Stochastic Differential Equations. 10.48550/arXiv.2011.13456.

32. Rombach, R., Blattmann, A., Lorenz, D., Esser, P. and Ommer, B. (2021) High-Resolution Image Synthesis with Latent Diffusion Models. 10.48550/arXiv.2112.10752.
33. Giannone, G., Nielsen, D. and Winther, O. (2022) Few-Shot Diffusion Models. 10.48550/arXiv.2205.15463.
34. Saharia, C., Chan, W., Chang, H., Lee, C.A., Ho, J., Salimans, T., Fleet, D.J. and Norouzi, M. (2021) Palette: Image-to-Image Diffusion Models. 10.48550/arXiv.2111.05826.
35. Li, Y., Zhou, K., Zhao, W.X. and Wen, J.-R. (2023) Diffusion Models for Non-autoregressive Text Generation: A Survey. 10.48550/arXiv.2303.06574.
36. Li, X.L., Thickstun, J., Gulrajani, I., Liang, P. and Hashimoto, T.B. (2022) Diffusion-LM Improves Controllable Text Generation. 10.48550/arXiv.2205.14217.
37. Vahdat, A., Kreis, K. and Kautz, J. (2021) In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. S. and Vaughan, J. W. (eds.), Vol. 34, pp. 11287--11302.
38. Rombach, R., Blattmann, A., Lorenz, D., Esser, P. and Ommer, B. (2022), *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, pp. 10674-10685.
39. Ronneberger, O., Fischer, P. and Brox, T. (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. 10.48550/arXiv.1505.04597.
40. Strait, B.J. and Dewey, T.G. (1996) The Shannon information entropy of protein sequences. *Biophys J*, **71**, 148-155.
41. Weiss, O., Jiménez-Montaño, M.A. and Herzel, H. (2000) Information content of protein sequences. *J Theor Biol*, **206**, 379-386.
42. Finn, R.D., Bateman, A., Clements, J., Coggill, P., Eberhardt, R.Y., Eddy, S.R., Heger, A., Hetherington, K., Holm, L., Mistry, J. *et al.* (2014) Pfam: the protein families database. *Nucleic Acids Res*, **42**, D222-230.
43. Loshchilov, I. and Hutter, F. (2017) Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
44. Steinegger, M. and Söding, J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, **35**, 1026-1028.
45. Notin, P., Dias, M., Frazer, J., Hurtado, J.M., Gomez, A.N., Marks, D. and Gal, Y. (2022), *International Conference on Machine Learning*. PMLR, pp. 16990-17017.