Secondary Structure-Guided Novel Protein Sequence Generation with Latent Graph Diffusion

Yutong Hu [*1] **Yang Tan** [*2 3 4] **Andi Han** [*5] **Lirong Zheng** [6] **Liang Hong** [1 2 4] **Bingxin Zhou** [1 2]

Abstract

The advent of deep learning has introduced efficient approaches for de novo protein sequence design, significantly improving success rates and reducing development costs compared to computational or experimental methods. However, existing methods face challenges in generating proteins with diverse lengths and shapes while maintaining key structural features. To address these challenges, we introduce CPDiffusion-SS, a latent graph diffusion model that generates protein sequences based on coarse-grained secondary structural information. CPDiffusion-SS offers greater flexibility in producing a variety of novel amino acid sequences while preserving overall structural constraints, thus enhancing the reliability and diversity of generated proteins. Experimental analyses demonstrate the significant superiority of the proposed method in producing diverse and novel sequences, with CPDiffusion-SS surpassing popular baseline methods on open benchmarks across various quantitative measurements. Furthermore, we provide a series of case studies to highlight the biological significance of the generation performance by the proposed method. The source code is publicly available at https://github.com/riacd/CPDiffusion-SS .

1. Introduction

Deep learning-based protein design provides an innovative and effective methodology, which promotes and creates novel or enhanced functionalities and physical properties of proteins varied from peptides to enzymes. Compared with traditional protein design approaches, such as directed evolution and rational design, deep learning-based protein design can significantly lower the human source, time, and financial cost (Chu et al., 2024) and create new proteins that do not exist in nature. Protein sequence is the foundation of protein structure and function, indicating that the sequence design is crucial for designing proteins with desired functions. There has been an increasing amount of work on designing protein sequences with deep generative models and validating the effectiveness of the designed protein products through bio-experiments (Ingraham et al., 2023; Zhou et al., 2023). These new techniques not only offer an opportunity to design novel protein sequences for a protein structure of interest, but also open a new way of designing proteins with significantly enhanced or novel functions for specific biological applications.

The intricate connection between protein sequences and their functions remains largely unknown due to the vast high-dimensional space of protein sequences. Additionally, obtaining accurately labeled data that detail the sequence-function relationship presents a significant challenge. Thus, the sequence-based deep learning models are generated for finding the relationship between sequence and function. To enhance the generative capabilities, some autoregressive generative models have been developed that incorporate homologous wild-type proteins from closely related functional families or engage multiple sequence alignments. Including protein family data could direct the generated proteins to exhibit specified, desirable traits (Truong Jr & Bepler, 2024). Masked language models adopt a different approach by working with fragments of wild-type protein sequences and training the system to complete the remaining parts (Elnaggar et al., 2021; Lin et al., 2023). Even though protein language models have access to a wealth of sequence data to assimilate typical protein sequence patterns and to craft sequences with variable lengths, it remains a complex task to ensure an ample supply of homologous sequences for specific proteins (Rao et al., 2021). A notable shortcoming of these sequence-centric approaches is their tendency to neglect the vital structural features of proteins. These structural elements are critical since they largely dictate protein functionality. Without consideration of these three-dimensional attributes, the models may fail to fully capture

[*]Equal contribution [1]Shanghai Jiao Tong University [2]Shanghai National Center for Applied Mathematics (SJTU center) [3]East China University of Science and Technology [4]Shanghai Artificial Intelligence Laboratory [5]RIKEN AIP [6]University of Michigan Medical School. Correspondence to: Bingxin Zhou <bingxin.zhou@sjtu.edu.cnk>.

Proceedings of the 41st International Conference on Machine Learning AI for Science Workshop, Vienna, Austria. PMLR 235, 2024.

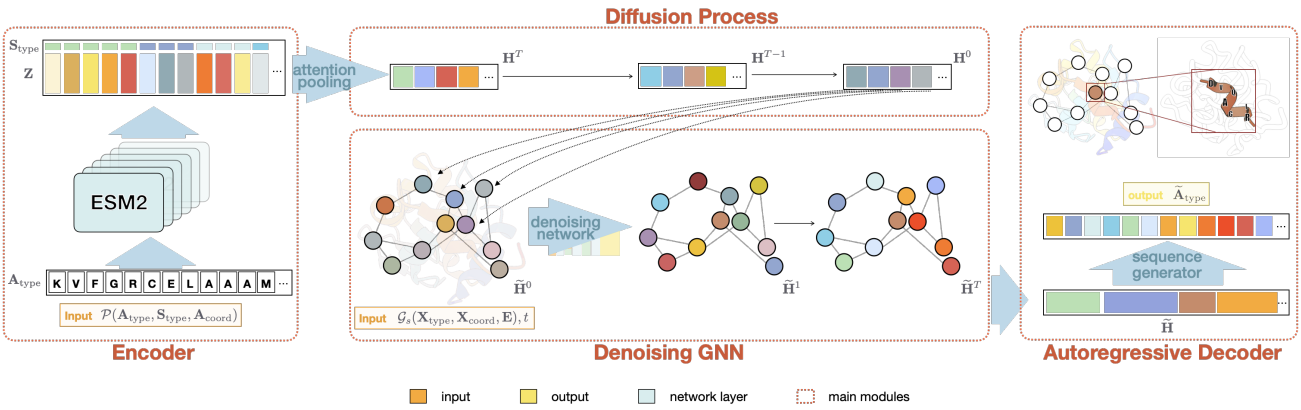

Figure 1. The illustrative figure of CPDIFFUSION-SS. The model embeds AA sequences into a hidden space of secondary structures using the latent graph diffusion model. The generated latent secondary structure representation is then translated into AA sequences of variable lengths by an autoregressive decoder.

the nuances of protein behavior and activity.

Structure-based protein generative models, on the other hand, investigate the conformation of proteins using geometric deep learning models (Satorras et al., 2021). They learn the local interactions of amino acids from the three-dimensional structure of proteins (either from experimental methods (Berman et al., 2000) or folding predictors (Lin et al., 2023)) and suggest amino acid compositions for the given scaffold or backbone (Dauparas et al., 2022; Yi et al., 2024). Structure-based generative models can learn key patterns of protein composition from fewer data and with smaller model sizes. Moreover, for some structure-driven generative objectives, such as thermostability and binding affinity (Zheng et al., 2022), these methods tend to achieve better performance (Tan et al., 2023). However, existing structure-based generative methods strictly require knowledge of the exact primary structures of protein for generation. Further, they cannot generate diverse sequences with flexible lengths or based on coarser-grained information, such as the protein's secondary structure.

This study presents CPDIFFUSION-SS, a deep generative model tailored for protein sequence design guided by coarse structural conditions like secondary structure. Such design is valuable for biologists, allowing tailored proteins with specific structural properties. For instance, adjusting α-helices or β-sheets on a protein's surface can enhance its structural rigidity, potentially increasing thermostability (Zheng et al., 2022). Moreover, optimizing secondary structures can enhance the encapsulation and delivery efficiency of proteins by viral capsids (Yeh et al., 2023). Unlike existing methods, CPDIFFUSION-SS considers protein structure while maintaining flexible amino acid (AA) sequence generation.

We address the challenge of evaluating novel protein sequences generated by deep learning models. Traditionally, quality is assessed based on recovery rate and perplexity compared to wild-type templates (Kucera et al., 2022; Repecka et al., 2021). However, these metrics have limitations, particularly in evaluating *de novo* design. Instead, we established independent benchmarks based on **CATH** and proposed new evaluation metrics for assessing novelty, designability, and diversity of novel protein sequences. The empirical evaluation of CPDIFFUSION-SS includes both quantitative and qualitative analysis. Performance on **CATH 4.3** dataset surpasses baseline methods in generating secondary structure-based AA sequences. Additionally, case studies suggest its potential applications like enhancing protein functionality and reducing size for drug delivery.

2. Related Work

Conditional Protein Sequence Generation Protein sequence generation typically seeks to achieve specific catalytic functions, often necessitating the integration of guiding principles or constraints from either the structural or sequence level to obtain the desired results. At the structural level, a common approach is to utilize a fixed protein backbone, such as the positions of amino acids (AAs) in three-dimensional space, and then output the appropriate AA type of each position, forming an AA sequence that is most likely to fold into the given structure (Hsu et al., 2022). This approach requires models capable of processing geometric structures, for example, using SE(3) equivariant neural networks to learn the geometric relationships between AAs (Satorras et al., 2021). Open benchmarks have validated these methods for their effectiveness in recovering AAs (Dauparas et al., 2022; Yi et al., 2024). Additionally, in some research, to demonstrate their models' effectiveness in generating desired proteins, wet lab experiments are conducted (Zhou et al., 2023). Beyond protein inverse folding,

other conditional protein sequence generation tasks require different inputs, such as protein function (Kucera et al., 2022), protein family (Repecka et al., 2021), and secondary structure (Xie et al., 2023). Although there have been some methods that attempt to incorporate secondary structures for conditional sequence generation, these methods often have limitations, such as being unable to generate sequences of varying AA length (Ni et al., 2023) or secondary structures w fixed order (Ingraham et al., 2023).

Protein Language Model Protein language models (PLMs) have been a hot spot in the field of AI-assisted protein design. PLMs are trained in a self-supervised manner and utilize extensive amino acid (AA) sequences to extract reliable AA representations, which are valuable for various downstream tasks, such as protein folding (Lin et al., 2023) and variant effect prediction (Tan et al., 2023; Truong Jr & Bepler, 2024). There are two prevalent types of PLMs: masked language models and autoregressive models. Masked language models are inspired by BERT (Devlin et al., 2018). These models are trained to predict masked AAs within the context of surrounding unmasked tokens. This approach is exemplified by models like ESM-1b (Rives et al., 2021). To improve the model's understanding of sequence characteristics, additional information including evolutionary data from multiple sequence alignments (MSA) (Rao et al., 2021) or functional annotations (Brandes et al., 2022) can be incorporated. Autoregressive models share an architecture similar to GPT-2 (Radford et al., 2019), generating protein sequences of varying lengths without conditioning (Nijkamp et al., 2023). Some PLMs are based on T5 (Raffel et al., 2020), such as ProtT5 (Elnaggar et al., 2021) and ProstT5 (Heinzinger et al., 2023).

3. CPDIFFUSION-SS: Secondary Structure-Guided Conditional Latent Protein Diffusion

In this section, we introduce the problem formulation to our research questions and propose our solution to it, *i.e.*, CPDIFFUSION-SS. The notations used in this study is summarized in Table 1.

3.1. Problem Formulation

Our study aims to generate AA sequences with secondary structure constraints. Relevant notations can be defined as follows: Let $\mathcal{P}(\boldsymbol{A}_{\text{type}}, \boldsymbol{S}_{\text{type}}, \boldsymbol{A}_{\text{coord}})$ denote an arbitrary protein of n AAs, where the two sequences $\boldsymbol{A}_{\text{type}} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ and $\boldsymbol{S}_{\text{type}} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n)$ represent labels for AA types and secondary structure types, respectively. $\boldsymbol{A}_{\text{coord}}$ is the coordinates of each AA in the three-dimensional Euclidean space.

Table 1. Table of notations.

notation	description
$\mathcal{P}(\boldsymbol{A}_{\text{type}}, \boldsymbol{S}_{\text{type}}, \boldsymbol{A}_{\text{coord}})$	a protein with sequence and structure information
$\mathcal{G}_s(\boldsymbol{X}_{\text{type}}, \boldsymbol{X}_{\text{coord}}, \boldsymbol{E})$	SS-level graph representation of the protein
$\boldsymbol{A}_{\text{type}} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$	AA sequence of a protein with n tokens
$\boldsymbol{S}_{\text{type}} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n)$	SS label for each AA of a protein
$\boldsymbol{A}_{\text{coord}}$	3D coordinates of each AA in a protein
$\boldsymbol{X}_{\text{type}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$	SS sequence of a protein
$\boldsymbol{X}_{\text{coord}}$	3D coordinates of each SS in a protein
$\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n]$	AA-level latent representation
$\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_m]$	SS-level latent representation
$\boldsymbol{H}^t$	latent embeddings at time step t
$\widetilde{\boldsymbol{H}}$	generated SS-level latent representation
$\widetilde{\boldsymbol{A}}$	generated AA sequence
$g_\theta(\cdot)$	conditional generative model
$f_\theta(\cdot)$	denoising neural network

The secondary structures (SS) are organized into a graph that illustrates the relationships between them within a protein, denoted as $\mathcal{G}_s(\boldsymbol{X}_{\text{type}}, \boldsymbol{X}_{\text{coord}}, \boldsymbol{E})$. Here $\boldsymbol{X}_{\text{type}}$ denotes the sequence of SS type, which can be helix (H), sheet (E), or coil (C). $\boldsymbol{X}_{\text{coord}}$ is the coordinates of secondary structures, which is calculated by averaging all AA coordinates within each secondary structure. For instance, in the visualized protein (PDB ID: 1Z25) in Figure 1, the highlighted node represents a helix structure, containing 7 AAs in the wild-type template. Suppose the structure is located at j-th position in $\boldsymbol{X}_{\text{type}}$ and the 7 corresponding AAs are located sequentially starting from the i-th position in $\boldsymbol{A}_{\text{type}}$, then the coordinates are computed as $\boldsymbol{X}_{\text{coord;j}} = \text{mean}(\boldsymbol{A}_{\text{coord;i}}, \ldots, \boldsymbol{A}_{\text{coord;i+6}})$. Then, the SSs are connected to their k nearest neighbor in the Euclidean space, with edge features $\boldsymbol{E}$ encoding the Euclidean distance between each SS pair.

The objective is to train a conditional generative model $g_\theta(\cdot)$ which generates the desired AA sequence $\widetilde{\boldsymbol{A}} = (\tilde{\boldsymbol{a}}_1, \ldots, \tilde{\boldsymbol{a}}_{n'})$, where n' does not necessarily equals n, *i.e.*,

$$\widetilde{\boldsymbol{A}} = g_\theta\left(\mathcal{G}_s(\boldsymbol{X}_{\text{type}}, \boldsymbol{X}_{\text{coord}}, \boldsymbol{E})\right). \tag{1}$$

The major challenge is that only coarse information about the desired structures is provided. Conventional protein language models and inverse folding methods are inadequate: protein language models cannot directly constrain the structure, and inverse folding methods require precise structural inputs, including the exact number and positions of all AAs. To address this, we propose CPDIFFUSION-SS, a secondary structure-guided conditional latent protein diffusion method for approximating $g_\theta(\cdot)$.

3.2. Model Architecture

CPDIFFUSION-SS comprises three components: a sequence encoder, a latent diffusion generator, and an autoregressive decoder. The encoder and decoder form a variational auto-encoder. The sequence encoder embeds amino

acid (AA) sequences into a latent space characterized by secondary structure-level (SS-level) representations, while the decoder translates these SS-level latent representations back to the AA space. Both the encoder and decoder use protein language models for sequence embedding and reconstruction. The central component, a latent graph diffusion model, generates diverse SS-level hidden representations within the latent space conditioned on SS input. Below, we detail the construction of each module.

3.2.1. Encoder-Decoder

Encoder For a protein $\mathcal{P}$ including n AAs and m SSs, the encoder converts the discrete input AA sequence $(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ into a continuous representation sequence $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_m)$ using a protein language model and an attention pooling module. The pre-trained protein language model initially maps the AA sequences of proteins to AA-level vector representations $\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n]$. In this process, we utilize an evolutionary-scale protein language model (Lin et al., 2023) to effectively analyze the structural and functional characteristics of proteins, employing a masked language model training objective (Devlin et al., 2018), *i.e.*,

$$\mathcal{L}_{\text{MLM}} := -\sum_{i \in \boldsymbol{M}} \log \left(\mathbb{P}(\mathbf{a}_i | \mathbf{A}_{\setminus \boldsymbol{M}}) \right),$$

where $\mathbf{A}_{\setminus M}$ represents the masked AA sequence obtained from $\boldsymbol{A}_{\text{type}}$. To obtain secondary structure (SS)-level representations, we utilize an attention pooling module (Yang et al., 2023), which aggregates amino acid (AA)-level representations $\boldsymbol{Z}$ into SS-level representations $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_m]$. Using $\boldsymbol{X}_{\text{type}}$, we rearrange $\boldsymbol{Z}$ into m groups:

$$\boldsymbol{Z} = [\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_m] = [[\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{n_1}], \ldots, [\boldsymbol{z}_{n-n_m+1}, \ldots, \boldsymbol{z}_n]],$$

with n_i being the number of AA in the i-th secondary structure, *i.e.*, $\sum_{i=1}^{m} n_i = n$. For the k-th ($1 \leq k \leq m$) secondary structure, the corresponding latent embedding $\boldsymbol{h}_k$ is summarized from $\boldsymbol{Z}_k$ by

$$\boldsymbol{h}_k = \text{AttnPool}(\boldsymbol{Z}_k) = \text{softmax}(\text{Conv}(\boldsymbol{Z}_k)) \cdot \boldsymbol{Z}_k, \quad (2)$$

where $\text{Conv}(\cdot)$ represents a 1-dimensional convolution along the dimension of the AA sequence and $\cdot$ calculates the weighted average of AA embeddings within the same secondary structure.

Decoder The decoder converts the diffusion-generated SS-level representation (introduced in the following section) $\widetilde{\boldsymbol{H}} = (\tilde{\boldsymbol{h}}_1, \ldots, \tilde{\boldsymbol{h}}_m)$ into $\widetilde{\boldsymbol{A}} = (\tilde{\boldsymbol{a}}_1, \ldots, \tilde{\boldsymbol{a}}_{n'})$. To generate AA sequences of varying lengths, an autoregressive model with multi-layer cross-attention is employed (Vaswani et al., 2017). The learning objective is structured as a sequence

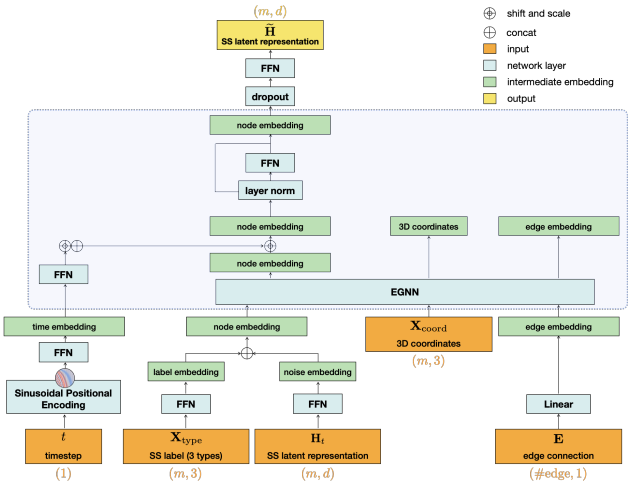

Figure 2. Illustrative architecture of the latent diffusion model.

translation task. For training, the SS-level hidden representation $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_m)$ from the encoder is used. This continuous representation is fed into the decoder as context vectors, guiding the reconstruction of the AA sequence. The decoder's training target is to minimize the KL divergence.

$$\min \sum_{\boldsymbol{a}_i \in \boldsymbol{A}} D_{\text{KL}} \left(\boldsymbol{a}_i || \text{Decoder}(\text{Encoder}(\boldsymbol{A}), \boldsymbol{a}_{<i}) \right). \quad (3)$$

Rotary Position Embedding (RoPE) (Su et al., 2024) is applied for positional encoding of the AA sequences, enhancing the model's ability to effectively capture positional information.

In summary, the encoder-decoder mechanism facilitates the mapping between AA-level protein sequences and SS-level latent space. We utilize an evolutionary model to proficiently perform sequence embedding and train an autoregressive decoder for translating AA sequences of varying lengths. To better align with secondary structure conditions and enrich the diversity of generated outcomes, we incorporate latent graph diffusion to generate SS-level vector representations.

3.2.2. Latent Diffusion

For generating SS-level latent representations, we adhere to the standard pipeline of the denoising diffusion probabilistic model (Ho et al., 2020) within the latent space. For each protein AA sequence, we extract its secondary embeddings from the pre-trained encoder, denoted as $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_m]$. The diffusion model is trained to generate representations of protein sequences that adhere to the secondary structure properties. The architecture for the denoising model is visualized in Figure 2.

Following the denoising diffusion probabilistic model (DDPM), the forward diffusion process gradually adds Gaussian noise to the input embeddings over steps $0 \rightarrow T$. The

objective is to maximize the evidence lower bound, which is equivalent to minimizing the expected reconstruction loss

$$\min_{\theta} \mathbb{E}_{t,\boldsymbol{H}^t} \| f_\theta(\boldsymbol{H}^t, t, \boldsymbol{X}_{\text{coord}}, \boldsymbol{X}_{\text{type}}, \mathcal{G}_s) - \boldsymbol{H}^0 \|$$

where we ignore the weighting constants.

To incorporate conditions based on secondary structure information, we design the denoising neural network $f_\theta(\cdot)$ using equivariant graph neural networks (Satorras et al., 2021). Each protein is represented as an SS-level graph $\mathcal{G}_s(\boldsymbol{X}_{\text{type}}, \boldsymbol{X}_{\text{coord}}, \boldsymbol{E})$, preserving the 3D geometric information of the secondary structures in $\boldsymbol{X}_{\text{coord}} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_m]$. As previously introduced, these coordinates are defined by the average C_α of the corresponding AAs within each secondary structure. In addition to the positions of the secondary structures, their types are encoded using one-hot encoding features $\boldsymbol{X}_{\text{type}}$.

The denoising network $f_\theta(\cdot)$ is conditioned on the 3D positions of protein secondary structures, thus the predicted embeddings should be invariant to orthogonal transformations or translations of the input coordinates. This means that translating, reflecting, or rotating the input should result in equivalent transformations of the output. To achieve this, we use $E(3)$ equivariant graph neural networks (EGNN) (Satorras et al., 2021) as the backbone for $f_\theta(\cdot)$. EGNNs have proven effective for protein representation learning (Tan et al., 2023; Yi et al., 2024; Zhou et al., 2023). At the ℓ-th layer, the hidden representation $\boldsymbol{h}_i^{\ell+1}$ is updated by edge and position updates, followed by node aggregation:

$$\begin{aligned}
\boldsymbol{m}_{ij}^{\ell+1} &= \phi_e(\boldsymbol{h}_i^\ell, \boldsymbol{h}_j^\ell, \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2, \boldsymbol{e}_{ij}) \\
\boldsymbol{x}_i^{\ell+1} &= \boldsymbol{x}_i^\ell + \sum_{j \neq i} (\boldsymbol{x}_i^\ell - \boldsymbol{x}_j^\ell) \phi_x(\boldsymbol{x}_i^\ell) \\
\boldsymbol{h}_i^{\ell+1} &= \phi_h(\boldsymbol{h}_i^\ell, \sum_{j \neq i}, \boldsymbol{m}_{ij}),
\end{aligned}$$

where $\phi_e(\cdot)$ and $\phi_h(\cdot)$ are the edge and node propagation functions, respectively, and $\boldsymbol{e}_{ij}$ represents the edge feature between nodes i and j.

3.3. Model Pipeline

Data Preparation CPDIFFUSION-SS utilizes two types of protein data as model input: the AA-level protein representation $\mathcal{P}(\boldsymbol{A}_{\text{type}}, \boldsymbol{S}_{\text{type}}, \boldsymbol{A}_{\text{coord}})$ and the SS-level graph representation $\mathcal{G}_s(\boldsymbol{X}_{\text{type}}, \boldsymbol{X}_{\text{coord}}, \boldsymbol{E}; \boldsymbol{H})$, as previously discussed in Section 3.1. For $\mathcal{P}$, both $\boldsymbol{A}_{\text{type}}$ and $\boldsymbol{A}_{\text{coord}}$ are directly obtained from structure-informed protein documents, such as PDB. The secondary structure $\boldsymbol{S}_{\text{type}}$ is assigned using DSSP (Touw et al., 2015). Proteins with more than 100 AAs in a single secondary structure are excluded, as they are believed to be problematic or irregularly dominated by loops. The processed AA-level data $\mathcal{P}$ is used solely for training purposes. In contrast, SS-level information $\mathcal{G}_s$ is used for both training and inference. Constructing the associated graph representation requires additional data processing steps. Specifically, the SS-level graph for a protein is defined with each node representing a secondary structure, labeled using one-hot encoding for its class (H, E, or C). Additionally, each secondary structure has a 1280-dimensional hidden representation $\boldsymbol{H}$ from the encoder that describes its AA compositions. During inference, this feature is generated by latent diffusion and represented as $\widetilde{\boldsymbol{H}}$. The three-dimensional coordinate $\boldsymbol{X}_{\text{coord}}$ is defined as the average position of all AAs (determined by the C_α atom) within it. Following the convention for constructing protein graphs, the connections in $\mathcal{G}_s$ are defined using k-nearest neighbor (kNN) graphs, with $k = 3$ based on the fact that secondary structures are less closely related than AAs. The edge matrix $\boldsymbol{E}$ is weighted by the fraction of the Euclidean distance between connected node pairs.

Model Training and Inferencing CPDIFFUSION-SS undergoes a two-stage training process, with separate training phases for the encoder-decoder module and the latent diffusion module. For the encoder-decoder, we utilize the pre-trained ESM2-650M (Lin et al., 2023) and train our Transformer-style decoder to minimize the objective function described in (3). This model is trained on a subset of the **AlphaFoldDB** (Barrio-Hernandez et al., 2023)[1] clustered by FoldSeek (Van Kempen et al., 2024), which includes over 2 million wild-type proteins with ALPHAFOLD2 predictions. In the second stage, we freeze the trained encoder-decoder and train the latent graph diffusion model to reconstruct the latent secondary structure representation $\boldsymbol{H}$. Given that the performance of the latent graph diffusion is closely related to protein structure, we train the model using **CATH4.3** (Sillitoe et al., 2021), which provides over 30,000 protein domain structures with less than 40The inference process begins with the latent diffusion model, which uses the provided secondary structure graph and a randomly generated noise representation $\boldsymbol{H}^T$. It then proceeds through the denoising process using EGNN layers to generate latent representations conditioned on the specified input secondary structure. Subsequently, the sampled $\widetilde{\boldsymbol{H}}$ is fed into the trained decoder to translate each secondary structure representation into explicit AA sequences.

4. Experiments

4.1. Experimental Protocol

Generation Task The models are evaluated through a secondary structure-based protein sequence generation task. We use 50 randomly selected structure templates from

[1]Available at https://alphafold.ebi.ac.uk/

Table 2. Diversity, novelty, and consistency of secondary-structure-guided generation by baseline methods on 50 test protein templates structures from **CATH4.3**. For each measurement, we report the average score with the standard deviation in parentheses. The best performance for each metric is indicated in ***bold***, while the second-best performance is *underlined*.

	Diversity			**Novelty**	**Consistency** (SS3)			**Consistency** (SS3, w/o loop)		
	TM $_{new}$ ↓	RMSD ↑	Seq. ID ↓	TM $_{wt}$↓	ID ↑	ID $_{max}$ ↑	MSE $_{SS\ Composition}$ ↓	ID ↑	ID $_{max}$ ↑	MSE $_{SS\ Composition}$ ↓
VANILLA DECODER	0.27 ± 0.01	3.98 ± 0.22	$\mathbf{6.31 \pm 0.12}$	0.23 ± 0.11	66.28 ± 11.47	76.80 ± 11.64	6.16 ± 3.08	56.82 ± 15.46	70.31 ± 15.30	23.59 ± 13.95
PROSTT5	0.28 ± 0.03	$\mathbf{6.65 \pm 1.01}$	15.78 ± 2.31	$\mathbf{0.12 \pm 0.06}$	74.31 ± 9.98	82.52 ± 12.35	4.19 ± 2.00	66.61 ± 14.00	77.24 ± 15.98	17.32 ± 9.11
ESM2 (1)	$\mathbf{0.26 \pm 0.06}$	3.15 ± 1.11	19.48 ± 6.00	0.29 ± 0.15	45.25 ± 14.67	53.98 ± 17.02	7.31 ± 3.24	35.17 ± 19.75	44.13 ± 21.8	29.26 ± 16.36
ESM2 (0.8)	0.27 ± 0.04	3.44 ± 0.96	13.02 ± 2.52	0.30 ± 0.15	52.41 ± 20.58	58.23 ± 20.75	6.44 ± 3.91	41.50 ± 27.19	50.08 ± 25.99	28.32 ± 18.66
ESM-IF1	0.29 ± 0.01	4.97 ± 0.85	7.46 ± 0.61	0.20 ± 0.09	$\underline{78.53 \pm 10.27}$	$\underline{84.88 \pm 10.13}$	$\underline{3.34 \pm 1.93}$	$\underline{74.44 \pm 11.83}$	$\underline{82.55 \pm 11.92}$	$\underline{14.91 \pm 9.01}$
PROTEINMPNN	0.35 ± 0.19	4.47 ± 1.58	76.50 ± 17.12	0.19 ± 0.14	56.00 ± 22.09	62.56 ± 21.66	6.46 ± 4.16	47.28 ± 26.33	53.73 ± 27.38	17.77 ± 13.32
CPDIFFUSION-SS	0.30 ± 0.02	$\underline{5.69 \pm 0.78}$	$\underline{7.08 \pm 0.36}$	$\underline{0.16 \pm 0.07}$	$\mathbf{81.57 \pm 9.78}$	$\mathbf{86.95 \pm 9.75}$	$\mathbf{1.56 \pm 0.89}$	$\mathbf{77.84 \pm 12.93}$	$\mathbf{84.43 \pm 12.05}$	$\mathbf{6.61 \pm 3.86}$

CATH4.3 for validation. These 50 test templates are excluded from the training set to ensure unbiased evaluation. For each template, 200 sequences are generated and assessed based on their structures predicted by ESMFOLD2. Models are provided with the secondary structure and the minimum essential additional data required for each specific baseline model.

Specifically, for any given template structure, CPDIFFUSION-SS receives an SS-level graph featuring secondary structure labels. Structure-based models (PROTEINMPNN and ESM-IF1) receive AA-level graph representations, where amino acids (AAs) within the same secondary structure are positioned at the group's center. Alternatively, sequence-based methods (ESM2 (Lin et al., 2023) and PROSTT5 (Raffel et al., 2020)) are given a small set of unmasked AA tokens. Both PROSTT5 and ESM2 (1) obtain a randomly selected unmasked AA token in each secondary structure, while ESM2 (0.8) and ESM2 (0.6) are supplied with randomly selected 20% and 40% unmasked AAs from the wild-type protein, respectively. We exclude a comparison with the model described in (Ni et al., 2023) due to the unavailability of the model implementation's checkpoint.

Training Setup For the encoder module, we utilize ESM-650, followed by a convolutional 1D-attention mechanism. The input channel for the convolution operator is set to 1280 (the output dimension of ESM2-650M), with the output channel being 1 and a kernel size of 1. In the latent graph diffusion module, we employ 4 EGNN layers as the denoising layers. The hidden and embedding dimensions are set to 640 and 1280, respectively. We use the `sqrt` noise schedule, with a learning rate of 5×10^{-4} and a weight decay of 10^{-5}. For the decoder, we incorporate 3 Transformer layers, each with 8 attention heads and hidden dimensions of 4960 in the feed-forward network. All implementations are programmed using `PyTorch Geometric` (version 2.4.0) (Fey & Lenssen, 2019) and `PyTorch` (version 2.2). The training is conducted on 8 NVIDIA® Tesla A800 GPUs, each with 80GB HBM2, mounted on an HPC cluster.

4.2. Evaluation Measurements

Diversity assesses the variance of generated amino acid (AA) sequences from the same template structure. We evaluate the diversity of the generated results at both the sequence and structure levels by comparing the pairwise similarity of all generated sequences and reporting the average scores. For sequence-level evaluation, we calculate the AA sequence identity, expressed as a percentage. For structure-level evaluation, we use TM-score and RMSD (Root Mean Square Deviation), both calculated using TM-align (Zhang & Skolnick, 2005). These metrics are crucial as we aim for generated sequences from the same template to exhibit significant differences. Thus, we prefer models that generate sequences with lower average sequence identity, lower average TM-score, and higher average RMSD. In Table 2, these measurements are denoted as *Seq. ID*, *TM $_{new}$*, and *RMSD*, respectively.

Novelty evaluates whether the structures of generated proteins significantly differ from existing wild-type proteins. Maximizing novelty is a common design objective in *de novo* protein design (Watson et al., 2023; Yim et al., 2024). For efficient protein structure comparison, we use Foldseek (Van Kempen et al., 2024) to examine the alignment of the generated protein structures (predicted by ESMFold) with those in the training set. We report the TM-score between the most similar wild-type protein and the generated protein. In this context, a lower TM-score indicates higher novelty, which is desirable. The novelty evaluation is reported under *TM $_{wt}$*. We provide the average TM-scores for all the proteins generated from the test templates.

Consistency evaluates the alignment between the input secondary structure conditions and the predicted secondary structure of generated proteins. This is measured from two perspectives: SS-level sequence identity and structure composition. Similar to AA-level sequence identity, SS-level

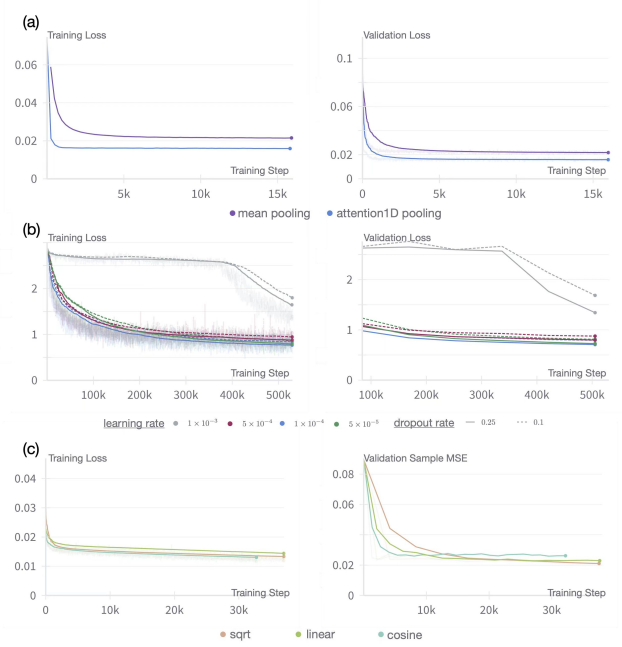

Figure 3. Learning curve with different (a) pooling layers; (b) learning rate and dropout rate. (c) noise schedules in the diffusion model.

sequence identity is computed by aligning sequences and calculating the proportion of matched tokens. The best-aligned sequence is obtained by maximizing the alignment length corresponding to the secondary structure sequence alignment, using a penalty mechanism for mismatches and gaps. This follows the definition of AA sequence identity in BLAST (Altschul et al., 1990) for global sequence comparison, where $\text{identity} = (\text{Matches}/\text{AlignmentLength}) \times 100\%$. The alignment length includes the total number of tokens for matches, gaps, and mismatches. Secondary structure composition calculates the proportions of helices (H), sheets (E), and coils (C) in both the input condition and the generated sequences. It then employs the Mean Squared Error (MSE) measure to quantify their differences. The three introduced metrics are reported in Table 2 as *ID*, *ID* $_{max}$, and *MSE* $_{SS\ Composition}$. Additionally, since sheets and helices are generally considered more important and harder to generate than loops (coils), and their structures are more fixed, we also report the consistency score after removing loops as a reference.

4.3. Results Analysis

The generative performance scores are presented in Table 2, where we assess our proposed CPDIFFUSION-SS against both sequence and structure-based baseline methods across 10 evaluation metrics focusing on the diversity, novelty, and consistency of the generated sequences. In this evaluation, CPDIFFUSION-SS outperforms baseline methods on 9 out of the 10 metrics, except for TM $_{new}$, where language models generally exhibit superior performance compared to structure-aware models. This disparity can be attributed to language models not explicitly integrating structural information, thus allowing for more unrestricted sequence generation. For instance, sequences generated by PROSTT5 for all three examined templates frequently exhibit repetitive patterns of certain amino acids, such as Glycine, Leucine, and Isoleucine. However, these amino acids are commonly found in all proteins for backbone stabilization and lack specificity to individual proteins. Moreover, such sequences are highly improbable to occur naturally, leading to lower sequence identity scores and TM scores.

Additionally, language models tend to produce longer sequences compared to structure-constrained models like ESM-if1 and CPDIFFUSION-SS. This observation is evident in Figure 4, where ProstT5 generates significantly longer sequences compared to both baseline methods and the wild-type templates.

Furthermore, we analyze the learning curve of the trained model and compare it with other hyperparameter configurations, as illustrated in Figure 3. All curves are visualized using `wandb` with moving average smoothing for better clarity. Both training and validation curves rapidly converge to a stable state after a reasonable number of training steps. To justify our choice of hyperparameters and architectures, we compare the learning curve with different pooling methods (average pooling and attention pooling), learning rates, and dropout rates for the encoder-decoder, as well as noise schedules (sqrt, linear, and cosine) for the latent diffusion model.

4.4. Case Study

To demonstrate CPDIFFUSION-SS's efficacy in using secondary structures for protein sequence generation, we conducted experiments generating novel sequences guided by specific secondary structures. We selected protein structures shown in Figure 4(a) as constraints. We then compared the structures of sequences generated by CPDIFFUSION-SS, PROSTT5, ESM-IF1, and PROTEINMPNN under the same conditions. Sequences from CPDIFFUSION-SS fold into plausible protein structures with similar secondary structure compositions to the wild-type template. In contrast, sequences from PROSTT5, ESM-IF1, and PROTEINMPNN deviate significantly from the secondary structural conditions. Notably, PROSTT5 generates sequences much longer than the template, and PROTEINMPNN produces sequences forming only random coils, unlikely to fold into functional proteins.

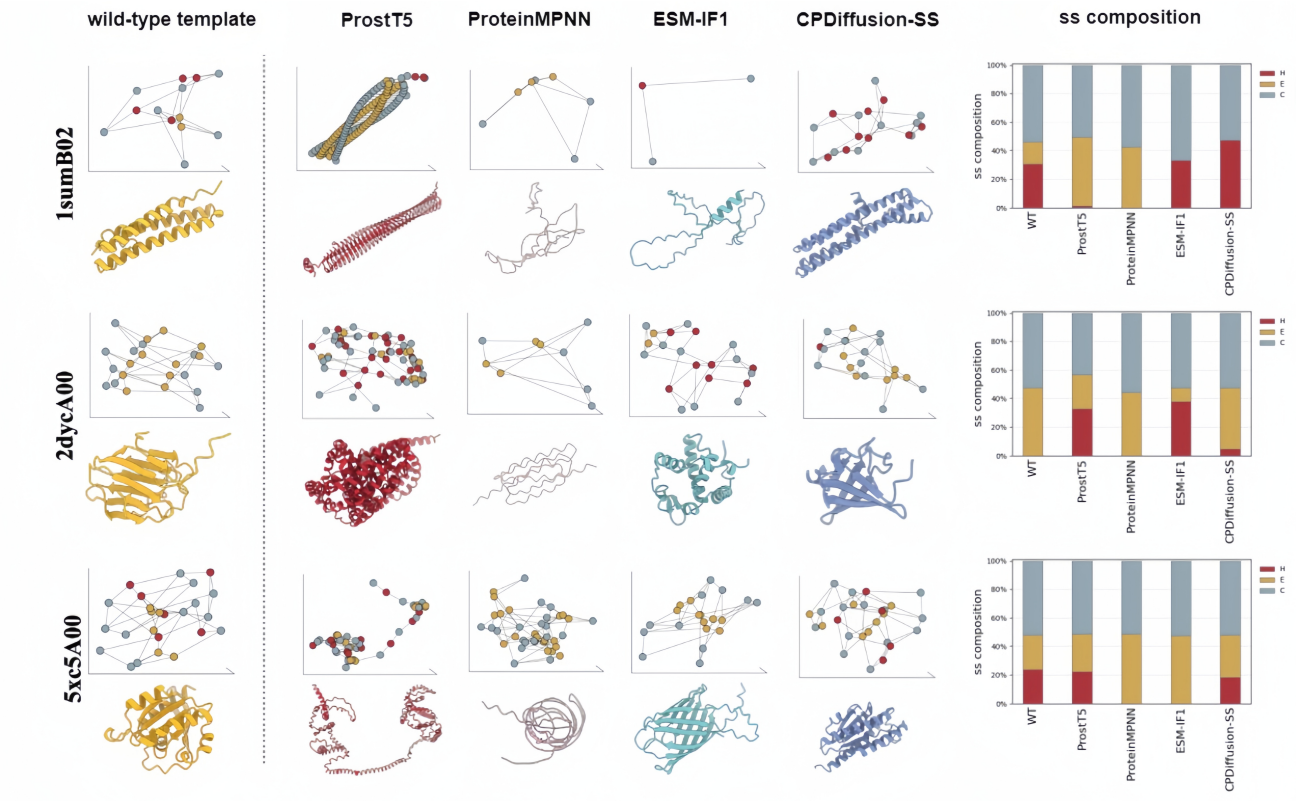

Figure 4. Predicted 3D structures and composition of secondary structures on three cases from the test dataset. Here we use red, yellow, and blue colors to represent helices (H), sheets (E), and coils (C), respectively.

5. Conclusion and Discussion

This study introduces a novel protein generation model guided by secondary structures, crucial elements for protein functionality. Leveraging powerful protein language models and latent graph diffusion models, we develop one of the first deep learning frameworks capable of generating diverse and reliable sequences conditioned on specific secondary structures.

Our experimental findings underscore CPDIFFUSION-SS's ability to generate proteins with target structures while adhering to secondary structure constraints. This capability holds significant implications for protein design and protein-based biotechnology. Structural flexibility, crucial for protein stability and activity, is intricately linked to secondary structure. Proteins often encounter challenges in industrial applications within extreme environments like strong acids, bases, or high temperatures due to structural instability. CPDIFFUSION-SS offers a solution by introducing new helices and sheets on the protein surface, compacting the protein and enhancing its resistance to extreme conditions (Zheng et al., 2022). Additionally, the flexibility of a protein's catalytic pocket profoundly influences its bioactivity (Zheng et al., 2022). By employing CPDIFFUSION-SS to increase loops and turns around catalytic sites, conformational changes can be facilitated during biofunctions, thereby enhancing catalytic activity.

Impact Statement

This paper presents work whose goal is to advance the field of protein de novo design. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

Barrio-Hernandez, I., Yeo, J., Jänes, J., Mirdita, M., Gilchrist, C. L., Wein, T., Varadi, M., Velankar, S., Beltrao, P., and Steinegger, M. Clustering predicted structures at the scale of the known protein universe. *Nature*, 622(7983):637–645, 2023.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat,

T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The protein data bank. *NAR*, 28(1):235–242, 2000.

Brandes, N., Ofer, D., Peleg, Y., Rappoport, N., and Linial, M. ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8): 2102–2110, 02 2022. ISSN 1367-4803.

Chu, A. E., Lu, T., and Huang, P.-S. Sparks of function by de novo protein design. *Nature Biotechnology*, 42(2): 203–215, 2024.

Dauparas, J., Anishchenko, I., Bennett, N., Bai, H., Ragotte, R. J., Milles, L. F., Wicky, B. I., Courbet, A., de Haas, R. J., Bethel, N., et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378 (6615):49–56, 2022.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*, 2018.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Yu, W., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., and Rost, B. Prottrans: Towards cracking the language of lifes code through self-supervised deep learning and high performance computing. *IEEE TPAMI*, pp. 1–1, 2021. doi: 10.1109/TPAMI.2021.3095381.

Fey, M. and Lenssen, J. E. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on RLGM*, 2019.

Heinzinger, M., Weissenow, K., Sanchez, J. G., Henkel, A., Steinegger, M., and Rost, B. Prostt5: Bilingual language model for protein sequence and structure. *bioRxiv*, 2023. doi: 10.1101/2023.07.23.550085.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020.

Hsu, C., Verkuil, R., Liu, J., Lin, Z., Hie, B., Sercu, T., Lerer, A., and Rives, A. Learning inverse folding from millions of predicted structures. In *ICML*, pp. 8946–8970. PMLR, 2022.

Ingraham, J. B., Baranov, M., Costello, Z., Barber, K. W., Wang, W., Ismail, A., Frappier, V., Lord, D. M., Ng-Thow-Hing, C., Van Vlack, E. R., Tie, S., Xue, V., Cowles, S. C., Leung, A., Rodrigues, J. a. V., Morales-Perez, C. L., Ayoub, A. M., Green, R., Puentes, K., Oplinger, F., Panwar, N. V., Obermeyer, F., Root, A. R., Beam, A. L., Poelwijk, F. J., and Grigoryan, G. Illuminating protein space with a programmable generative model. *Nature*, 2023. doi: 10.1038/s41586-023-06728-8.

Kucera, T., Togninalli, M., and Meng-Papaxanthos, L. Conditional generative modeling for de novo protein design with hierarchical functions. *Bioinformatics*, 38(13):3454–3461, 2022.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130, 2023.

Ni, B., Kaplan, D. L., and Buehler, M. J. Generative design of de novo proteins based on secondary-structure constraints using an attention-based diffusion model. *Chem*, 9(7):1828–1849, 2023.

Nijkamp, E., Ruffolo, J. A., Weinstein, E. N., Naik, N., and Madani, A. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140):1–67, 2020.

Rao, R. M., Liu, J., Verkuil, R., Meier, J., Canny, J., Abbeel, P., Sercu, T., and Rives, A. Msa transformer. In *ICML*. PMLR, 2021.

Repecka, D., Jauniskis, V., Karpus, L., Rembeza, E., Rokaitis, I., Zrimec, J., Poviloniene, S., Laurynenas, A., Viknander, S., Abuajwa, W., et al. Expanding functional protein sequence spaces using generative adversarial networks. *Nature Machine Intelligence*, 3(4):324–333, 2021.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

Satorras, V. G., Hoogeboom, E., and Welling, M. E(n) equivariant graph neural networks. In *ICML*, pp. 9323–9332. PMLR, 2021.

Sillitoe, I., Bordin, N., Dawson, N., Waman, V. P., Ashford, P., Scholes, H. M., Pang, C. S., Woodridge, L., Rauer, C., Sen, N., et al. Cath: increased structural coverage of functional space. *NAR*, 49(D1):D266–D273, 2021.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Tan, Y., Zhou, B., Zheng, L., Fan, G., and Hong, L. Semantical and topological protein encoding toward enhanced bioactivity and thermostability. *bioRxiv*, pp. 2023–12, 2023.

Touw, W. G., Baakman, C., Black, J., Te Beek, T. A., Krieger, E., Joosten, R. P., and Vriend, G. A series of pdb-related databanks for everyday needs. *NAR*, 43(D1): D364–D368, 2015.

Truong Jr, T. and Bepler, T. Poet: A generative model of protein families as sequences-of-sequences. *NeurIPS*, 36, 2024.

Van Kempen, M., Kim, S. S., Tumescheit, C., Mirdita, M., Lee, J., Gilchrist, C. L., Söding, J., and Steinegger, M. Fast and accurate protein structure search with foldseek. *Nature Biotechnology*, 42(2):243–246, 2024.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *NeurIPS*, 30, 2017.

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976): 1089–1100, 2023.

Xie, X., Valiente, P. A., and Kim, P. M. Helixgan a deep-learning methodology for conditional de novo design of α-helix structures. *Bioinformatics*, 39(1):btad036, 2023.

Yang, K. K., Zanichelli, N., and Yeh, H. Masked inverse folding with sequence transfer for protein representation learning. *Protein Engineering, Design and Selection*, 36: gzad015, 2023.

Yeh, A. H.-W., Norn, C., Kipnis, Y., Tischer, D., Pellock, S. J., Evans, D., Ma, P., Lee, G. R., Zhang, J. Z., Anishchenko, I., et al. De novo design of luciferases using deep learning. *Nature*, 614(7949):774–780, 2023.

Yi, K., Zhou, B., Shen, Y., Liò, P., and Wang, Y. Graph denoising diffusion for inverse protein folding. *NeurIPS*, 36, 2024.

Yim, J., Campbell, A., Mathieu, E., Foong, A. Y., Gastegger, M., Jiménez-Luna, J., Lewis, S., Satorras, V. G., Veeling, B. S., Noé, F., et al. Improved motif-scaffolding with SE(3) flow matching. *arXiv:2401.04082*, 2024.

Zhang, Y. and Skolnick, J. Tm-align: a protein structure alignment algorithm based on the tm-score. *NAR*, 33(7): 2302–2309, 2005.

Zheng, L., Lu, H., Zan, B., Li, S., Liu, H., Liu, Z., Huang, J., Liu, Y., et al. Loosely-packed dynamical structures with partially-melted surface being the key for thermophilic argonaute proteins achieving high dna-cleavage activity. *NAR*, 50(13):7529–7544, 2022.

Zhou, B., Zheng, L., Wu, B., Yi, K., Zhong, B., Lio, P., and Hong, L. Conditional protein denoising diffusion generates programmable endonucleases. *bioRxiv*, pp. 2023–08, 2023.