Self-supervised Pre-training for Protein Embeddings Using Tertiary Structures

Yuzhi Guo[1], Jiaxiang Wu[2], Hehuan Ma[1], Junzhou Huang[1*],

[1]University of Texas at Arlington, Arlington, TX, 76019, USA
[2]Tencent AI Lab, Shenzhen, 518057, China

Abstract

The protein tertiary structure largely determines its interaction with other molecules. Despite its importance in various structure-related tasks, fully-supervised data are often time-consuming and costly to obtain. Existing pre-training models mostly focus on amino-acid sequences or multiple sequence alignments, while the structural information is not yet exploited. In this paper, we propose a self-supervised pre-training model for learning structure embeddings from protein tertiary structures. Native protein structures are perturbed with random noise, and the pre-training model aims at estimating gradients over perturbed 3D structures. Specifically, we adopt SE(3)-invariant features as the model inputs and reconstruct gradients over 3D coordinates with SE(3)-equivariance preserved. Such paradigm avoids the usage of sophisticated SE(3)-equivariant models, and dramatically improves the computational efficiency of pre-training models. We demonstrate the effectiveness of our pre-training model on two downstream tasks, protein structure quality assessment (QA) and protein-protein interaction (PPI) site prediction. Hierarchical structure embeddings are extracted to enhance corresponding prediction models. Extensive experiments indicate that such structure embeddings consistently improve the prediction accuracy for both downstream tasks.

1 Introduction

The biological functions of a protein, as well as its possible interaction with other molecules, are largely determined by its 3-dimensional structure (Berg et al. 2002). For various protein-related applications, *e.g.* structure-based drug design (SBDD) (Śledź and Caflisch 2018; Batool, Ahmad, and Choi 2019) and protein-protein interaction (PPI) prediction (Sun et al. 2017; Zeng et al. 2020), protein tertiary structures are one of the most critical features. However, it is time-consuming and costly to collect 3D structures for protein-ligand complex and multi-protein complex via experimental structure determination. As a result, the performance of SBDD and PPI models is often restrained by the limited structure data. On the other hand, computational methods for protein structure prediction have attracted increasing attention for many decades. A large number of structure decoys can be generated via various prediction protocols, which raises the question on how to find out the most accurate prediction, *i.e.* protein structure quality assessment (QA) (Olechnovič and Venclovas 2017; Baldassarre et al. 2021). As structure decoys produced by different protocols can be highly diverse and non-*i.i.d.*, it is critical to obtain universal embeddings for protein structures. To conclude, protein structure embeddings are crucial in many protein-related applications, but non-trivial to obtain due to limited data and/or potential bias of data distributions.

Recent advances in natural language processing (NLP) demonstrate that large-scale self-supervised pre-training models can be highly effective in various downstream tasks (Vaswani et al. 2017; Devlin et al. 2019). Similar idea has been adopted to train large-scale language models for proteins, with either amino-acid sequences or multiple sequence alignments (MSAs). In (Rao et al. 2019; Rives et al. 2021), LSTM and Transformer models are trained to predict randomly masked-out amino-acids in FASTA sequences, so as to formulate inter-residue interactions within proteins. Sturmfels *et al.* (Sturmfels et al. 2020) propose to predict profiles derived from multiple sequence alignments, instead of randomly masked amino-acids. In (Rao et al. 2021), Transformer models are trained to predict masked-out position in multiple sequence alignments (rather than FASTA sequences), which better cooperates the co-evolution information embedded in MSAs. All these sequence-based pre-training models have been proved to be effective in learning meaningful embeddings for amino-acid types and providing critical features for secondary structure and contact predictions.

However, such sequence-based pre-training models do not utilize protein tertiary structures, which could be crucial to structure-related downstream tasks mentioned above. Additionally, the computational complexity of large-scale language models are often prohibitively high, and it usually takes weeks or even months to train such models on high-performance GPU clusters (Rao et al. 2021).

To address above issues, we propose a pre-training model for learning structure embeddings from protein tertiary structures. The model is optimized with a self-supervised loss function, which only relies on protein structures and does not require any additional supervision. Specifically, native protein structures are randomly perturbed with Gaus-

*Corresponding author: jzhuang@uta.edu

sian noise, and the model aims at estimating the log probability's gradients over perturbed 3D coordinates. Due to intrinsic symmetries for 3D rotations and translation, the SE(3)-equivariance must be preserved in the gradient estimation. Standard SE(3)-equivariant models often involve complicated and time-consuming computation for spherical harmonics (Thomas et al. 2018; Fuchs et al. 2020) or regular representations (Hutchinson et al. 2020). In contrast, we construct SE(3)-invariant features as the pre-training model's inputs, and then reconstruct gradients over 3D coordinates with SE(3)-equivariance preserved. Such workflow, similar to (Shi et al. 2021), dramatically improves the computational efficiency without sacrificing the SE(3)-equivariance.

We demonstrate the effectiveness of our pre-training model with two downstream tasks: protein structure quality assessment and protein-protein interaction site prediction. Hierarchical structure embeddings (whole-protein, per-residue, and inter-residue) are extracted with the pre-training model, and then fed into corresponding models proposed for each downstream task as enhancement. Extensive experiments indicate that such structure embeddings consistently improve the prediction accuracy of downstream tasks.

The overall contributions of this paper are summarized as:

- We propose the first self-supervised pre-training model for protein tertiary structures, while existing models only utilize amino-acids sequences or multiple sequence alignments.
- Our pre-training model is computationally efficient, and is capable of generating informative structure embeddings at various hierarchical levels.
- We demonstrate that the prediction accuracy of downstream tasks can be consistently improved by cooperating structure embeddings provided by our pre-training model.

2 Related Work

Protein 3D structures dependent tasks

In this paper, we employ two downstream tasks which require protein three-dimensional (3D) structures to evaluate our pre-training model: protein model quality assessment (QA) and protein-protein interaction (PPI) site prediction.

Protein structure QA (estimation of model accuracy) estimates the quality of computational protein models in terms of the divergence from their native structure (Won et al. 2019). It aims at 1) finding the best model in a pool of protein structure prediction models, and 2) refining a model based on its estimated local quality. QA task utilizes two types of evaluation metrics: local score and global score. At the residue level, local score includes Local Distance Difference Test (LDDT) (Mariani et al. 2013) and the Contact Area Difference (CAD) (Olechnovič, Kulberkytė, and Venclovas 2013) scores. At the protein level, global score contains Global Distance Test Total Score (GDT_TS) (Baldassarre et al. 2021), Global Distance Test High Accuracy (GDT_HA) (Zemla 2003), TM-score (Zhang and Skolnick 2004) and the global versions of LDDT and CAD.

Protein–protein interactions refer to the physical contacts between two or more proteins, which are crucial for the function of proteins (De Las Rivas and Fontanillo 2010a; Zeng et al. 2020). The identification of PPI Site is an efficient way to help understand the biological functions of a protein (Li et al. 2018). The PPI Site prediction is a residue level 2-state classification task.

Self-supervised Learning

The self-supervised learning method is well known for its good performance on NLP tasks by using substantial unlabeled data during the training. It does not require explicit human guides, and also brings in flexibility (Vaswani et al. 2017). An effective strategy of self-supervised training is to add certain noise to the data, then train the network to obtain the original data, which is considered as a self-recovery process. For example, masked-token prediction (Devlin et al. 2019) replaces the value of tokens at multiple positions with alternate tokens and allows the network to predict back. Recently, a novel protein sequence self-supervised method called TAPE (Rao et al. 2019) uses this masked-token mechanism to train a pre-training model and achieves good performance on several sequence-based prediction tasks. However, due to the complexity of protein 3D structures, there is no structure-based pre-training method to adapt to the above 3D structure-dependent downstream tasks.

3 Methods

In this section, we describe how protein structures can be represented with SE(3)-invariance preserved, *i.e.* invariant to arbitrary 3D rotations and translations. Afterwards, we present our pre-training framework for protein structures, built upon energy-based models. Finally, we demonstrate how pre-trained models can be utilized in two downstream tasks: protein structure quality assessment (QA) and protein-protein interaction (PPI) site prediction.

SE(3)-invariant Representation of Protein Structures

Protein tertiary structures are largely determined by 3D coordinates of all the amino-acid residues' C_α atoms (Gront, Kmiecik, and Kolinski 2007; Krivov, Shapovalov, and Dunbrack Jr 2009). Therefore, it is often sufficient to represent protein structures with 3D coordinates of C_α atoms. However, such coordinate-based representation depends on the overall configuration (location and orientation) of protein structures. Since rigid-body rotations and translations can be arbitrary and do not affect protein structures, it is required that coordinated-based models must preserve the SE(3)-equivariance to capture such symmetries in the conformation space.

In this paper, we circumvent this SE(3)-equivariance restraint by introducing a SE(3)-invariant representation of protein structures. Specifically, we calculate the Euclidean distance between all the C_α atom pairs, and represent protein structures with the resulting pairwise distance matrix. Since the relative distance remains constant *w.r.t.* any 3D ro-

tations and translations, such SE(3)-invariant representation allows much more flexible choices of subsequent models.

Formally, for a protein with amino-acid sequence of length L, we denote 3D coordinates of all the C_α atom as $\mathbf{X} \in R^{L\times 3}$, where $\mathbf{x}_i$ is the 3D coordinate of i-th residue's C_α atom. The pairwise distance matrix is denoted as $\mathbf{D} \in R^{L\times L}$, where each entry is determined by $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$. Our pre-training model is built upon pairwise distance matrices, thus the model itself does not need to be restrained to preserve the SE(3)-equivariance. Nevertheless, it is worth mentioning that it is feasible to propagate estimated gradients from the pairwise distance matrix to 3D coordinates via the chain rule, which is critical for training energy-based models, as we shall demonstrate later.

Self-supervised Pre-training

In order to extract informative protein and per-residue embeddings, we propose a pre-training model to approximate the data distribution of protein tertiary structures. The intrinsic motivation is that if the underlying data distribution is well approximated, then this pre-training model must have captured the critical information embedded in protein structures, which could be quite beneficial for various downstream tasks.

In (Song and Ermon 2019), Song *et al.* propose to train an energy-based model via denoising score matching (Vincent 2011) for image generation. Original images are perturbed with Gaussian noise of different scales, and the network is trained to estimate the log probability's gradients over perturbed images. Although pairwise distance matrices, as SE(3)-invariant representations of protein structures, can also be viewed as 2D images, it is unreasonable to directly perturb distance matrices with random noise. The key difference lies in that for the image generation task, every randomly perturbed image is valid, so that the perturbed data distribution is still well defined. However, not all $L \times L$ real-valued matrices are valid distance matrices, *i.e.* there may not exist a 3D structure satisfying the randomly perturbed distance matrix.

To tackle this issue, instead of applying random perturbation on distance matrices, we propose to firstly add Gaussian noise on 3D coordinates of all the C_α atoms, and derive the corresponding distance matrix as perturbed inputs. The score network is then trained to estimate gradients over perturbed distance matrices. Both inputs and outputs of the score network are invariant to 3D rotations and translations, so the score network can be instantiated by any convolutional neural networks. Since the random perturbation is performed over 3D coordinates, we only have closed-form ground-truth gradients over 3D coordinates. Therefore, we also need to propagate estimated gradients from distance matrices to 3D coordinates, which is made possible via the chain rule.

Formally, we choose a series of standard deviations for Gaussian noise, $\sigma_1 > \sigma_2 > \cdots > \sigma_K$, where K is the total number of random noise levels. We denote the native protein structure as $\mathbf{X}$, as represented by all the C_α atoms' 3D coordinates, and its perturbed counterpart as $\tilde{\mathbf{X}} \sim p(\tilde{\mathbf{X}}|\mathbf{X}, \sigma_k)$, which is given by:

$$\tilde{\mathbf{X}} := \mathbf{X} + \mathbf{Z},\ \mathbf{Z} \sim \mathcal{N}\left(\mathbf{0}, \sigma_k^2 \mathbf{I}\right) \tag{1}$$

where σ_k is selected as the random noise's standard deviation. The perturbed data distribution's log probability's gradients over perturbed 3D coordinates have a closed-form solution:

$$\nabla_{\tilde{\mathbf{X}}} \log p(\tilde{\mathbf{X}}|\mathbf{X}, \sigma_k) = \frac{\mathbf{X} - \tilde{\mathbf{X}}}{\sigma_k^2} \tag{2}$$

which can be easily derived from the multivariate Gaussian distribution's probability density function.

We denote the pairwise distance matrix corresponding to the perturbed 3D coordinates as $\tilde{\mathbf{D}}$, where $\tilde{d}_{ij} = \|\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j\|_2$. This perturbed distance matrix is then fed into the score network, which consists of multiple residual convolutional blocks. Similar to (Song and Ermon 2019), conditional batch normalization is employed to explicitly let the score network be aware of the random noise's standard deviation for generating the current perturbed input. The detailed network architecture is presented in Section 4. The score network is trained to estimate the log probability's gradients over the elementwise squared perturbed distance matrix:

$$\mathbf{H} := h_{\boldsymbol{\theta}}(\tilde{\mathbf{D}}, s, \sigma_k),\ h_{ij} \approx \nabla_{\tilde{d}_{ij}^2} \log p(\tilde{\mathbf{X}}|\mathbf{X}, \sigma_k) \tag{3}$$

where the amino-acid sequence s is also used as the inputs of the score network. As discussed above, it is non-trivial to derive closed-form ground-truth gradients for the distance matrices. Hence, we apply the chain rule to propagate the estimated gradients to perturbed 3D coordinates:

$$\begin{aligned} \mathbf{G} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_L \end{bmatrix},\ \mathbf{g}_i &= \sum\nolimits_{j=1}^{L} 2\left(h_{ij} + h_{ji}\right)\left(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j\right) \\ &\approx \nabla_{\tilde{\mathbf{x}}_i} \log p(\tilde{\mathbf{X}}|\mathbf{X}, \sigma_k) \end{aligned} \tag{4}$$

where the last term can be explicitly calculated by Eq. (2). For simplicity, we denote the above gradient propagation process as $\mathbf{G} = g(\mathbf{H}, \tilde{\mathbf{X}}) = g(h_{\boldsymbol{\theta}}(\tilde{\mathbf{D}}, s, \sigma_k), \tilde{\mathbf{X}})$.

So far, we have presented the log probability's ground-truth gradients over perturbed 3D coordinates, as well as the score network's estimation. The self-supervised loss function is given by:

$$\begin{aligned} Loss = &\frac{1}{2NK} \sum_{\mathbf{X}\in\mathcal{X}} \sum_{k=1}^{K} \sigma_k^2 \cdot E_{\tilde{\mathbf{X}}\sim\mathcal{N}(\mathbf{X}|\sigma_k^2\mathbf{I})} \\ &\left\| g(h_{\boldsymbol{\theta}}(\tilde{\mathbf{D}}, s, \sigma_k), \tilde{\mathbf{X}}) - \frac{\mathbf{X} - \tilde{\mathbf{X}}}{\sigma_k^2} \right\|_F^2 \end{aligned} \tag{5}$$

where $\mathcal{X}$ is the set of all the native protein structures, and $N = |\mathcal{X}|$ is its cardinality. The above loss function measures the difference between the ground-truth and the estimated gradients for all the K random noise levels. Each level's loss is re-weighted by the corresponding standard deviation σ_k, so that each level approximately has an equal contribution to the overall loss function. By minimizing this loss function, the score network's estimated gradients approximately

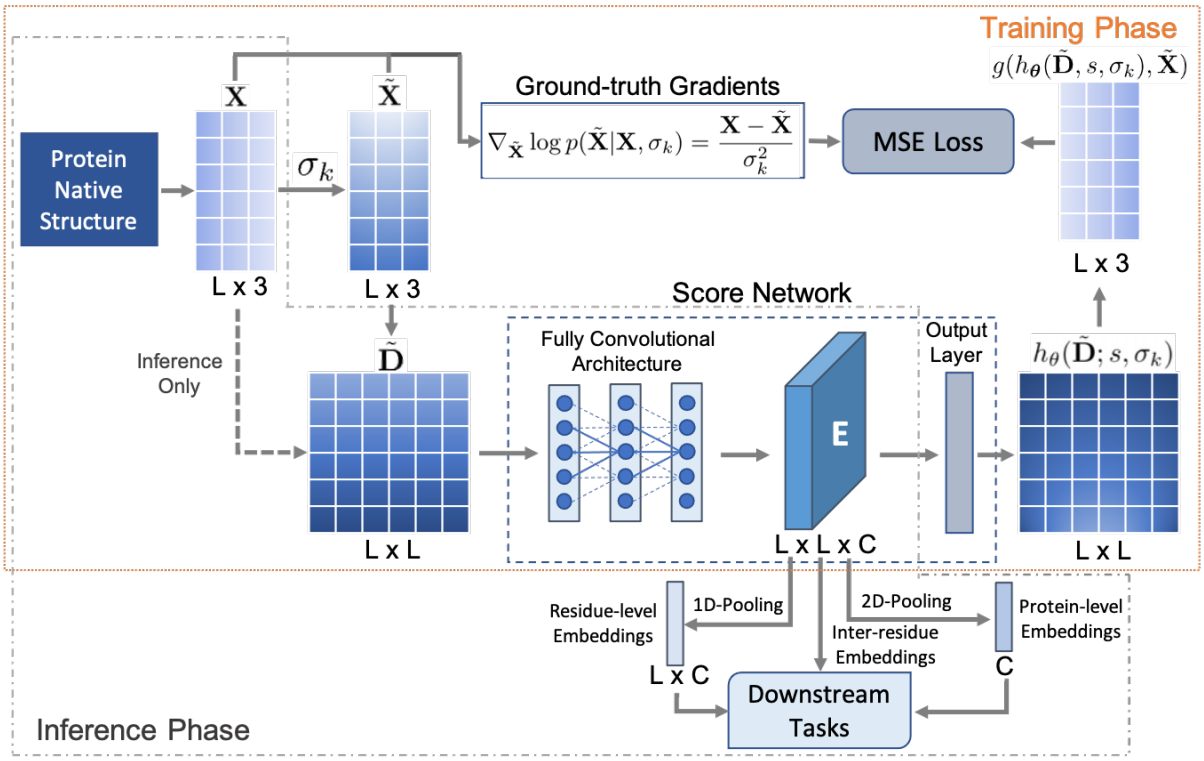

Figure 1: The workflow of the pre-training process. First, we extract C_α atoms' 3D coordinates, which are denoted as X, and perturb it with various levels of random noise to get perturbed 3D coordinates $\tilde{X}$. Then we compute the distance matrix $\tilde{D}$, which is further fed into the score network to predict the corresponding gradients. It is then transformed into the estimated gradients over perturbed 3D coordinates. We calculate the MSE loss between the estimated and ground-truth gradients as the pre-training signals to back-propagate to the score network. For the inference phase, we transfer the 3D coordinates X to distance matrix D without perturbation, and extract the feature matrix E for the downstream tasks.

match ground-truth ones, thus the underlying data distribution of native protein structures is roughly parameterized by the score network. The overall training workflow is illustrated in Figure 1.

Once the pre-training model is sufficiently optimized, we may utilize it to extract structure embeddings for novel protein structures. Recall that the score network adopts the 2D convolutional network as the backbone architecture. For any specific protein structure, we calculate the pairwise distance matrix for all the C_α atoms, and feed it into the pre-training model. The final feature maps (next to estimated gradients) of size $L \times L \times C$ are then extracted, where C is the number of feature map channels. Such feature maps can be viewed as inter-residue structure embeddings, each of dimension C. Furthermore, by applying 1D and 2D global pooling, we obtain C-dimensional per-residue and whole-protein structure embeddings. To wrap up, during the inference phase (as depicted in Figure 1), we can extract whole-protein, per-residue, and inter-residue structure embeddings as additional inputs to downstream tasks.

Pre-training Model for Downstream Tasks

Here, we take two downstream tasks as examples, to demonstrate how structure embeddings produced by the pre-training model can be utilized to boost the prediction accuracy of downstream tasks.

Protein Structure Quality Assessment Due to the randomness in the initialization and optimization process, multiple structure decoys are generated as the candidates for the same amino-acid sequence for most protein structure prediction methods (Yang et al. 2020; Ju et al. 2021). Protein structure quality assessment (QA) aims at identifying the best predicted structure among all the candidates, which is one of the indispensable modules in protein structure prediction. In (Baldassarre et al. 2021), the authors propose GraphQA to formulate the protein structure as a graph, where the nodes are amino-acid residues and the edges are inter-residue interactions. To simultaneously consider the sequential and geometric structure, GraphQA builds the edges for both sequential-adjacent and spatial-neighboring residue pairs. The model consists of multiple message passing operations (Gilmer et al. 2017) to gradually update the node embeddings and predict both local and global lDDT scores (Mariani et al. 2013). Empirical evaluation results indicate that GraphQA achieves similar prediction accuracy to state-of-art-methods for quality assessment, despites the simplicity of the node/edge features being used.

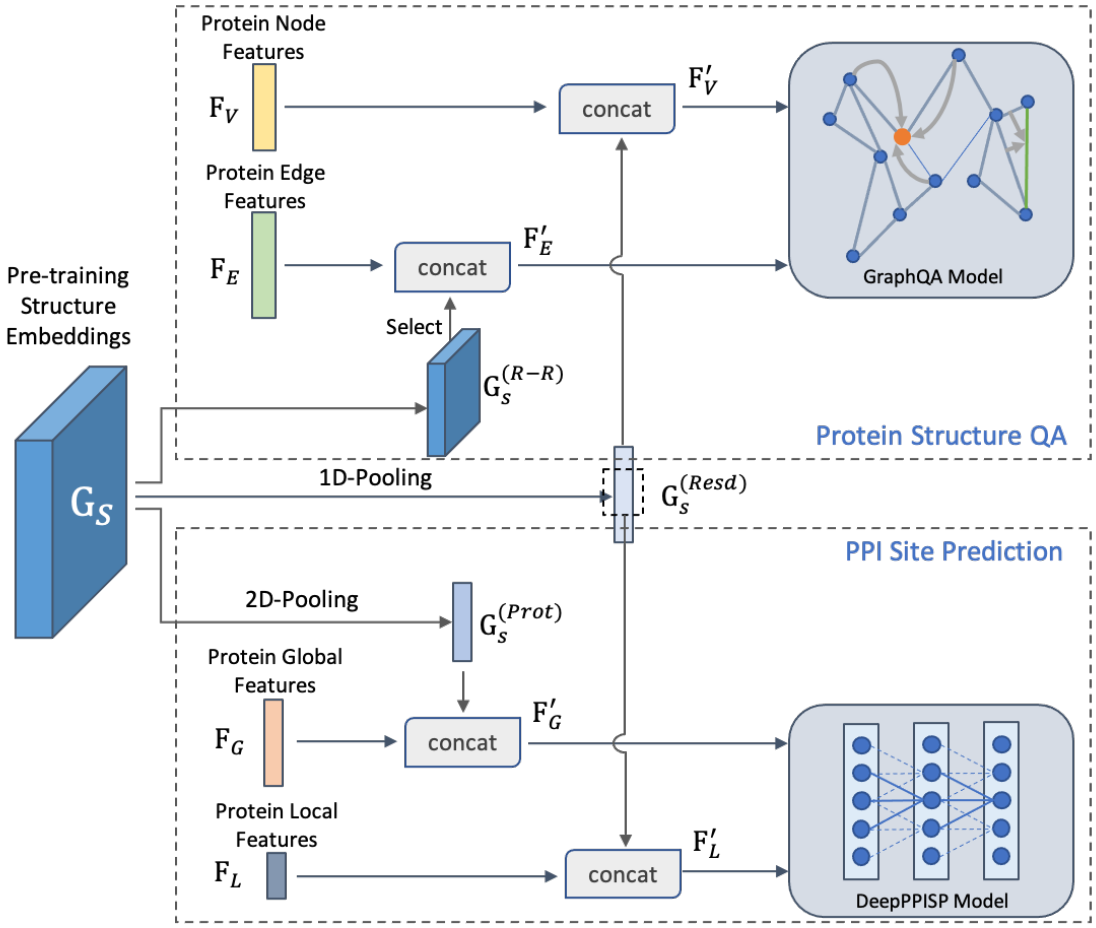

Figure 2: To align with the input feature vectors of the two downstream tasks, we conduct multiple operations on the embeddings generated by our pre-training model: 1) pre-trained edge embeddings is obtained by using the same selecting methods as GraphQA; 2) G_S^{Resd} is computed by 1D average pooling as the pre-trained node feature on QA task; 3) On the basis of G_S^{Resd}, we use the same window clipping operation as the DeepPPISP to obtain the enhanced local feature on i-th residue. 4) We perform 2D average pooling on G_S to get G_S^{Prot} as the pre-trained global feature for PPI Site prediction task.

Here, we employ our pre-training model to extract structure embeddings to further enhance the node and edge features of GraphQA. Specifically, we feed the structure decoy which needs to be assessed into our pre-training model (without random perturbation), and obtain the resulting structure embeddings $\mathbf{G}_S$. Since each spatial location in the feature map corresponds to a pair of residues, we enhance the edge features by selecting feature vectors at the corresponding locations. Similarly, the node features can be enhanced by concatenating the 1D-Pooling results of structure embeddings $\mathbf{G}_S$. The GraphQA model then takes such enhanced node and edge features as input for local and global lDDT prediction. The overall workflow is depicted in the upper part of Figure 2.

Protein-protein Interaction Site Prediction Protein-protein interaction models predict the physical contacts between two or more proteins, which play a vital role in various biological processes (De Las Rivas and Fontanillo 2010b; Li et al. 2019). To better understand how different proteins interact with each other, the first step is to identify which amino-acid residues in each protein are actually involved in the interaction. Formally, we follow (Zeng et al. 2020) to define an amino-acid as a PPI site if its absolute solvent accessibility before and after the protein binding is smaller than 1 $\mathring{A}^2$. Thus, the PPI site prediction task can be viewed as a pre-residue binary classification problem. In (Zeng et al. 2020), the authors propose DeepPPISP as an end-to-end framework, which integrates both local contextual and global sequence features for PPI site prediction. Concretely, local features are extracted from a fixed-size sliding windows centered at each amino-acid residue to capture local patterns, while global features are extracted via an 1-dimensional convolutional network. After that, local and global features are concatenated and used by the subsequent classification sub-network for per-residue classification.

Similarly, our pre-training model can be used as a plug-n-play module to enhance both local and global features used in the DeepPPISP model. For each training sample used in

PPI site prediction, we encode protein structures with our pre-training model to calculate the corresponding structure embeddings. The additional global features are obtained via applying the 2D-Pooling over structure embeddings. As for local features, per-residue structure embeddings can be computed as 1D-Pooling results of full-size structure embeddings. Such embeddings are then grouped by the same sliding window to generate additional local contextual features to describe each amino-acid residue. By concatenating all the original/additional local and global features, the DeepPPISP model can be trained with an enhanced feature set for PPI site prediction.

Summary To wrap up, we have demonstrated how our pre-training model can be utilized to produce structure embeddings at various hierarchical levels. As long as the downstream task relies on structure-based features of proteins, it should always be beneficial to include our structure embeddings to further enhance its feature representation. Potential application scenarios include protein fold classification (Chen et al. 2016) and structure-based drug design (Batool, Ahmad, and Choi 2019).

4 Experiments

Experiments setup

Datasets For the pre-training model, we obtain native protein structures from the RCSB-PDB database (released on 01/05/2021) (Berman et al. 2000), which includes over 170 thousands unlabeled protein tertiary structures. The RCSB-PDB database is somewhat redundant, where identical or highly-similar amino-acid sequences may correspond to multiple protein structures. Therefore, we adopt the official sequence clustering results, BC-30 and BC-100, to filter-out the redundant sequences with at least 30% or 100% sequence identity, respectively. After removing overlap proteins with valid and test data in downstream tasks, the BC-100 dataset contains 73,585 proteins, among which 58,868 are used as the training set, 7,357 as the validation set, and the remaining ones are test set. The BC-30 dataset consists of 29,242 proteins. Within them, 23,394 proteins are used as training set, 2,923 as the validation set, and 2,925 proteins are used for testing.

For the protein QA prediction task, we use the dataset published by GraphQA (Baldassarre et al. 2021). CASP9-CASP12 datasets contain 85k decoys, which are randomly split into a training set (˜270 targets) and a validation set (˜50 targets). CASP13 dataset contains ˜14k decoys (˜72 targets) in the test set.

For the PPI site prediction task, we use the processed data from DeepPPISP (Zeng et al. 2020), *i.e.* Dset_186 of 186 proteins, Dset_72 of 72 proteins (Murakami and Mizuguchi 2010) and PDBset_164 of 164 proteins (Singh et al. 2014). DeepPPISP removes two proteins since they do not have the related protein DSSP files (Kabsch and Sander 1983), which is one of the input features used in the method. DeepPPISP integrates three datasets to a fused dataset to ensure that the training and test set are from an identical distribution. We download the training, validation, and test data list from (Zeng et al. 2020). There are 300 proteins in the training set, 50 proteins for independent validation set, and 70 proteins in the test set.

Input features In addition to the distance matrix described in Section 3, we also encode the protein-specific information as the input features of the score network, which include protein sequence one-hot feature and positional encoding (Vaswani et al. 2017). See Appendix C for the details of features and encodings.

Network architecture and learning hyper-parameters Our score network for pre-training adopt the fully-convolutional neural networks architecture, which consists of 32 residual blocks with dilation convolution. To reduce the computational overhead, we apply the bottleneck mechanism (He et al. 2016) on each residual unit. We also use conditional batch normalization (Song and Ermon 2019) to take random noise's standard deviation level into consideration. The number of hidden layers' channels k is set to 64. We also report the results of different channels number in Appendix E. We use a batch size of 32 for training and validation, and randomly crop the input feature maps with size 32 for data augmentation. The positional encodings' dimension is set to $d_{model} = 24$. We construct random noise's standard deviations for $K = 32$ levels, which ranges from 0.01 to 10.0. When $\sigma_1 = 10.0$, the conformation space can be sufficiently explored, while $\sigma_K = 0.01$ indicates trivial perturbation is introduced to the native structures.

For the optimization, we apply a constant learning rate of 0.0001 and use Adam (Kingma and Ba 2014) as the optimizer for our pre-training model. After training 50 epochs, we select the optimal checkpoint based on the validation loss, and then use it for the upcoming structure embeddings (G_S) generation.

Results

Table 1 shows the comparison performance on protein QA downstream supervised task for CASP13 dataset. Other than evaluating the effectiveness of our method by running experiments with and without our pre-training model, we also compare the performance of protein sequence-based embedding. GraphQA is utilized as the baseline model, and we follow (Rao et al. 2019) to generate TAPE's sequence-based embeddings. Table 1-GDT_TS shows the results of various evaluation metrics for global quality predictions *w.r.t.* GDT_TS. For $RMSE$ and FRL_5, lower is better; for R, R_{target}, and z, higher is better. The results demonstrate that with the embeddings generated by our pre-training model, GraphQA is more capable than all other methods, including using the original features and adding sequence-based embeddings at ranking decoys on their overall quality. Please defer to Appendix C for more details of the QA task's evaluation metrics.

The performance of local quality predictions *w.r.t.* the ground-truth CAD and LDDT scores are also reported in Table 1, higher is better. As observed, our pre-training method further improves the performance at the local level, which indicates the high quality of our embeddings at the local (residue) level, as well as the ability of distinguishing the

Method [1]	GDT_TS					CAD		LDDT	
	RMSE	R	R_{target}	z	FRL_5	ρ	ρ_{decoy}	ρ	ρ_{decoy}
w/o pre-trained embeddings	0.201	0.793	0.751	1.026	0.045	0.637	0.390	0.774	0.510
w/ sequence embeddings	0.158	0.799	0.772	1.101	0.037	0.624	0.387	0.754	0.502
w/ BC-30 embeddings (ours)	0.149	0.818	0.775	1.272	0.035	0.649	0.415	0.782	0.530
w/ BC-100 embeddings (ours)	**0.133**	**0.848**	**0.787**	**1.345**	**0.031**	**0.667**	**0.424**	**0.800**	**0.534**

Table 1: Results on global and local QA prediction task using GraphQA prediction model

Method [1]	ACC	Precision	Recall	F-measure	MCC
w/o pre-trained embeddings	0.589±0.012	0.270±0.006	0.623±0.018	0.377±0.004	0.163±0.006
w/ sequence embeddings	0.592±0.036	0.274±0.009	**0.635±0.058**	0.382±0.003	0.174±0.005
w/ BC-30 embeddings (ours)	0.614±0.016	0.280±0.005	0.604±0.026	0.382±0.001	0.177±0.002
w/ BC-100 embeddings (ours)	**0.621±0.029**	**0.285±0.010**	0.601±0.052	**0.386±0.003**	**0.185±0.004**

Table 2: Results on PPI Site prediction task using DeepPPISP prediction model

correctly predicted parts of the protein chain. In consequence, the embeddings extracted by our pre-training model can make the prediction network capture more complex information and long-range dependencies between residues compared with the original features. Please note that the results of adding sequence embedding on local scores are worse than the baseline. One possible reason is that local QA task is more dependent on inter-residue (edge) information, while TAPE does not contain such information. Moreover, adding a large number of dimensions' node features (768 dimensions of TAPE) makes the original network more difficult to train.

We implement the experiments precisely according to the experimental settings in GraphQA (Baldassarre et al. 2021), including data-splitting, network hyper-parameters, and training strategy. Additional results of the protein QA downstream task are reported in Appendix D.

Table 2 shows the results of DeepPPISP model training with and without the embeddings generated by our pre-training model, and we introduce the TAPE embeddings for comparison as well. Since DeePPPISP does not provide a seed for data loading, we repeat the experiment five times to get the mean and standard deviation to eliminate the randomness and verify the robustness. Although the recall of our method is lower than the performance of baselines, the scores of all other assessment metrics are the highest. It is noteworthy that the PPI Site prediction training problem is imbalanced, thus the downstream task is usually more concentrated on the the performance of MCC and F-measure (Zeng et al. 2016), and DeepPPISP uses F-measure to select the best validation model. More details about the evaluation metrics can be found in Appendix C. Compared with QA task, PPI task has relatively balanced dependence on sequence information and structure information. Thus, it is reasonable that TAPE performs better than the baseline model which only utilizes the original features. Moreover, our structure embeddings is able to achieve better performance by exploring the structure information.

In addition, we conduct experiments with pre-training on a smaller dataset, named the BC-30 filtered dataset, to confirm the effectiveness of proposed method. As shown in Table 1 and 2, although the data involved in pre-training is streamlined, it consistently performs well on downstream tasks. The results indicate that even pre-training on a smaller dataset, our model can still provide high-quality local and global embeddings for downstream tasks. To demonstrate the robustness of our pre-training model, we also run experiments with different hidden sizes of the score network. The corresponding results on downstream tasks are deferred to Appendix E due to the space limitation. The results show that utilizing the pre-trained structure embeddings can still achieve good performance in the downstream tasks when pre-training with smaller datasets.

5 Conclusion

In this work, we propose a self-supervised pre-training model for protein structure. To the best of our knowledge, this is the first attempt to construct and evaluate self-supervised learning on protein 3D structures. In addition, our method can be easily applied to various downstream models. It is empirically demonstrated that our pre-training model can generate high quality structure embeddings for downstream tasks. Recent pre-training strategies mainly focus on the protein sequence dataset since it is easier to obtain and contains huge amount of data. However, even the dataset used for pre-training protein 3D structure is not as large as protein sequence dataset, we argue that the 3D structure contains more information than the sequence. In order to fully utilize the available protein data, our next move is to integrate the 3D structure pre-training strategy with a sequence-based pre-training method to acquire sufficient protein information.

[1]To make a fair comparison, all the settings and data are the same with the original papers when we run baseline, sequence embeddings, and structural embeddings experiments. The evaluation metrics are originally used in GraphQA and DeepPPISP.

Acknowledgments. This work was partially supported by US National Science Foundation IIS-1553687 and Cancer Prevention and Research Institute of Texas (CPRIT) award (RP190107).

References

Baldassarre, F.; Menéndez Hurtado, D.; Elofsson, A.; and Azizpour, H. 2021. GraphQA: protein model quality assessment using graph convolutional networks. *Bioinformatics*, 37(3): 360–366.

Batool, M.; Ahmad, B.; and Choi, S. 2019. A Structure-Based Drug Discovery Paradigm. *International Journal of Molecular Sciences*, 20(11).

Battaglia, P. W.; Hamrick, J. B.; Bapst, V.; Sanchez-Gonzalez, A.; Zambaldi, V.; Malinowski, M.; Tacchetti, A.; Raposo, D.; Santoro, A.; Faulkner, R.; et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.

Berg, J. M.; Tymoczko, J. L.; Stryer, L.; et al. 2002. *Biochemistry*. New York: WH Freeman.

Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; and Bourne, P. E. 2000. The protein data bank. *Nucleic acids research*, 28(1): 235–242.

Chen, D.; Tian, X.; Zhou, B.; and Gao, J. 2016. ProFold: Protein Fold Classification with Additional Structural Features and a Novel Ensemble Classifier. *BioMed Research International*, 2016: 6802832.

De Las Rivas, J.; and Fontanillo, C. 2010a. Protein–protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS Comput Biol*, 6(6): e1000807.

De Las Rivas, J.; and Fontanillo, C. 2010b. Protein-protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS computational biology*, 6(6): e1000807–e1000807.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 1810.04805.

Fuchs, F.; Worrall, D.; Fischer, V.; and Welling, M. 2020. SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks. In *Advances in Neural Information Processing Systems*.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, 1263–1272. PMLR.

Gront, D.; Kmiecik, S.; and Kolinski, A. 2007. Backbone building from quadrilaterals: A fast and accurate algorithm for protein backbone reconstruction from alpha carbon coordinates. *Journal of Computational Chemistry*, 28(9): 1593–1597.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hutchinson, M.; Lan, C. L.; Zaidi, S.; Dupont, E.; Teh, Y. W.; and Kim, H. 2020. LieTransformer: Equivariant Self-attention for Lie Groups. *arXiv Preprint*, 2012.10885.

Ju, F.; Zhu, J.; Shao, B.; Kong, L.; Liu, T.-Y.; Zheng, W.-M.; and Bu, D. 2021. CopulaNet: Learning Residue Co-evolution Directly from Multiple Sequence Alignment for Protein Structure Prediction. *Nature Communications*, 12(1): 2535.

Kabsch, W.; and Sander, C. 1983. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Research on Biomolecules*, 22(12): 2577–2637.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krivov, G. G.; Shapovalov, M. V.; and Dunbrack Jr, R. L. 2009. Improved prediction of protein side-chain conformations with SCWRL4. *Proteins: Structure, Function, and Bioinformatics*, 77(4): 778–795.

Li, M.; Fei, Z.; Zeng, M.; Wu, F.-X.; Li, Y.; Pan, Y.; and Wang, J. 2018. Automated ICD-9 coding via a deep learning approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(4): 1193–1202.

Li, X.; Li, W.; Zeng, M.; Zheng, R.; and Li, M. 2019. Network-based methods for predicting essential genes or proteins: a survey. *Briefings in Bioinformatics*, 21(2): 566–583.

Mariani, V.; Biasini, M.; Barbato, A.; and Schwede, T. 2013. lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21): 2722–2728.

Murakami, Y.; and Mizuguchi, K. 2010. Applying the Naïve Bayes classifier with kernel density estimation to the prediction of protein–protein interaction sites. *Bioinformatics*, 26(15): 1841–1848.

Olechnovič, K.; Kulberkytė, E.; and Venclovas, Č. 2013. CAD-score: a new contact area difference-based function for evaluation of protein structural models. *Proteins: Structure, Function, and Bioinformatics*, 81(1): 149–162.

Olechnovič, K.; and Venclovas, 2017. VoroMQA: Assessment of protein structure quality using interatomic contact areas. *Proteins: Structure, Function, and Bioinformatics*, 85(6): 1131–1145.

Rao, R.; Bhattacharya, N.; Thomas, N.; Duan, Y.; Chen, X.; Canny, J.; Abbeel, P.; and Song, Y. S. 2019. Evaluating protein transfer learning with tape. *Advances in Neural Information Processing Systems*, 32: 9689.

Rao, R.; Liu, J.; Verkuil, R.; Meier, J.; Canny, J. F.; Abbeel, P.; Sercu, T.; and Rives, A. 2021. MSA Transformer. *bioRxiv*.

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

Rost, B.; Sander, C.; and Schneider, R. 1994. Redefining the goals of protein secondary structure prediction. *Journal of molecular biology*, 235(1): 13–26.

Shi, C.; Luo, S.; Xu, M.; and Tang, J. 2021. Learning Gradient Fields for Molecular Conformation Generation. *arXiv Preprint*, 2105.03902.

Singh, G.; Dhole, K.; Pai, P. P.; and Mondal, S. 2014. SPRINGS: prediction of protein-protein interaction sites using artificial neural networks. Technical report, PeerJ PrePrints.

Song, Y.; and Ermon, S. 2019. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*.

Sturmfels, P.; Vig, J.; Madani, A.; and Rajani, N. F. 2020. Profile Prediction: An Alignment-Based Pre-Training Task for Protein Sequence Models. *arXiv preprint arXiv:2012.00195*.

Sun, T.; Zhou, B.; Lai, L.; and Pei, J. 2017. Sequence-based prediction of protein protein interaction using a deep-learning algorithm. *BMC Bioinformatics*, 18(1): 277.

Thomas, N.; Smidt, T.; Kearnes, S.; Yang, L.; Li, L.; Kohlhoff, K.; and Riley, P. 2018. Tensor Field Networks: Rotation- and Translation-Equivariant Neural Networks for 3D Point Clouds. *arXiv Preprint*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*.

Vincent, P. 2011. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7): 1661–1674.

Won, J.; Baek, M.; Monastyrskyy, B.; Kryshtafovych, A.; and Seok, C. 2019. Assessment of protein model structure accuracy estimation in CASP13: Challenges in the era of deep learning. *Proteins: Structure, Function, and Bioinformatics*, 87(12): 1351–1360.

Yang, J.; Anishchenko, I.; Park, H.; Peng, Z.; Ovchinnikov, S.; and Baker, D. 2020. Improved Protein Structure Prediction using Predicted Interresidue Orientations. *Proceedings of the National Academy of Sciences*, 117(3): 1496–1503.

Zemla, A. 2003. LGA: a method for finding 3D similarities in protein structures. *Nucleic acids research*, 31(13): 3370–3374.

Zeng, M.; Zhang, F.; Wu, F.-X.; Li, Y.; Wang, J.; and Li, M. 2020. Protein–protein interaction site prediction through combining local and global features with deep neural networks. *Bioinformatics*, 36(4): 1114–1120.

Zeng, M.; Zou, B.; Wei, F.; Liu, X.; and Wang, L. 2016. Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data. In *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*, 225–228. IEEE.

Zhang, Y.; and Skolnick, J. 2004. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4): 702–710.

Śledź, P.; and Caflisch, A. 2018. Protein structure-based drug design: from docking to molecular dynamics. *Current Opinion in Structural Biology*, 48: 93–102.