Pre-training sequence, structure, and surface features for comprehensive protein representation learning

ICLR2024
Rating: 6-6-6-5

February 29, 2024

Overview

- Introduction to Existing Work
- ProteinINR: Incorporating Surface Features as a Plugin for Protein Pre-training Models
- Experiments
- Conclusion

Protein Structure Pretraining

- Motivation
  - One major reason is the limited number of **annotated proteins**, which makes the training of existing models difficult and prone to overfitting.
  - We have approximately 906,458 protein structure data available, sourced from the AlphaFold Protein Structure Database version 2:
    - Species protein structures(364,520 protein structures)
    - Swiss-Prot(541,938)
- We primarily focus on the pretraining of **protein structures**.

Introduction to Existing Work on Protein Structure Pretraining

- Contrastive Learning and Subgraph Sampling Framework

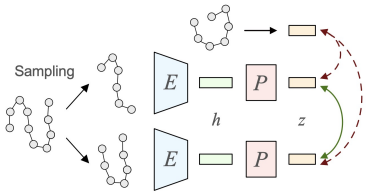
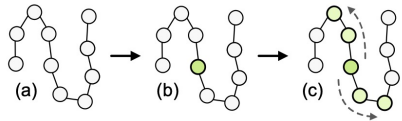


Figure 1: For each protein we sample random substructures which are then encoded into two representations, h and z, using encoders E and P. Then, we minimize the distance between representations z from the same protein and maximize the distance between representations from different proteins.



Sampling method

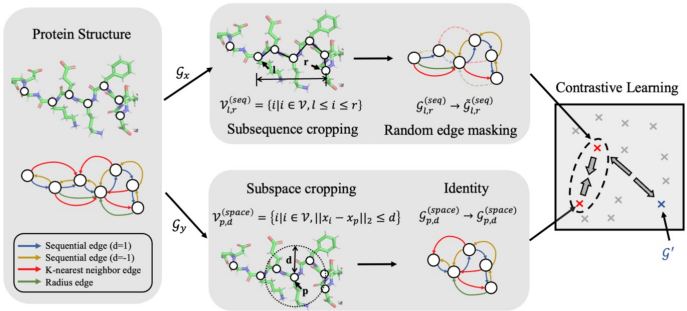
- Contrastive Loss:

$$l_i = -log\frac{exp(s(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k\neq i, k\neq j]} exp(s(z_i, z_k)/\tau)}$$

Contrastive representation learning for 3d protein structures (ICLR2022)

GearNet

- Contrastive Learning and Subgraph Sampling Framework



Protein representation learning by geometric structure pretraining (ICLR2023)

- InfoNCE Loss:

$$\mathcal{L}_{x,y} = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_x, \boldsymbol{z}_y)/\tau)}{\sum_{k=1}^{2B} \mathbb{1}_{[k \neq x]} \exp(\text{sim}(\boldsymbol{z}_y, \boldsymbol{z}_k)/\tau)},$$

GearNet

- Another framework: Self-prediction Methods

Method	Loss function	Sampled items
Residue Type Prediction	$\mathcal{L}_i = \text{CE}(f_{\text{residue}}(\boldsymbol{h}'_i), \boldsymbol{f}_i)$	Single residue
Distance Prediction	$\mathcal{L}_{(i,j,r)} = (f_{\text{dist}}(\boldsymbol{h}'_i, \boldsymbol{h}'_j) - \Vert\boldsymbol{x}_i - \boldsymbol{x}_j\Vert_2)^2$	Single edge
Angle Prediction	$\mathcal{L}_{(i,j,r_1),(j,k,r_2)} = \text{CE}(f_{\text{angle}}(\boldsymbol{h}'_i, \boldsymbol{h}'_j, \boldsymbol{h}'_k), \text{bin}(\angle ijk))$	Adjacent edge pairs
Dihedral Prediction	$\mathcal{L}_{(i,j,r_1),(j,k,r_2),(k,t,r_3)} = \text{CE}(f_{\text{dih}}(\boldsymbol{h}'_i, \boldsymbol{h}'_j, \boldsymbol{h}'_k, \boldsymbol{h}'_t), \text{bin}(\angle ijkt))$	Adjacent edge triplets

- Masked residue type prediction is widely used for pretraining protein language models.
- Residue types, Distances, Angles and Dihedrals

ESM-GearNet

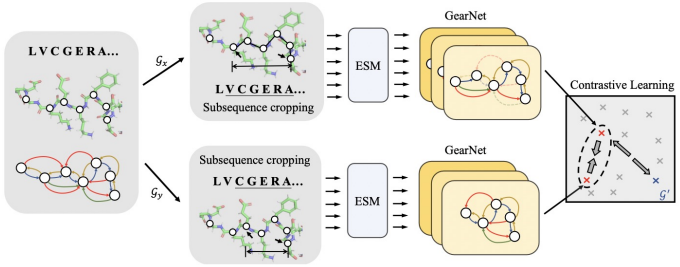


Figure 2: High-level illustration of ESM-GearNet pre-trained with Multiview Contrast. For each protein, we randomly sample subsequences $\mathcal{G}_x$ and $\mathcal{G}_y$ and randomly mask some edges to add noises. Encoding with ESM-GearNet, their representations are aligned in the latent space while minimizing its similarity with a negative sample $\mathcal{G}'$.

Enhancing protein language model with structure-based encoder and pre-training (MLDD workshop, ICLR 2023)

- Fuse protein sequence and structure encoder together.
- Sequence- and Structure-based pre-training method.

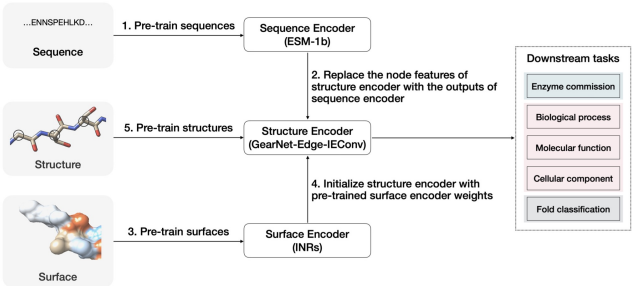
ProteinINR-Surface

Comparison of different protein encoders with and without sequence, structure, or surface pre-training

Method	Sequence Encoder	Structure Encoder	Sequence Pre-training	Structure Pre-training	Surface Encoder	Surface Pre-training
CNN	✓					
Transformer	✓					
GVP		✓				
GearNet		✓				
ESM-1b	✓		✓			
ProtBert	✓		✓			
DeepFRI	✓	✓	✓			
LM-GVP	✓	✓	✓			
ESM-GearNet	✓	✓	✓			
GearNet-MC		✓		✓		
GearNet-DP		✓		✓		
ESM-GearNet-MC	✓	✓	✓	✓		
ESM-GearNet-INR-MC (Ours)	✓	✓	✓	✓	✓	✓

- Advantages of surface feature:
  - Consider side-chain information
  - Traditional protein structure encoders only contain alpha carbon or backbone atoms.
- Methods: Proposes Implicit Neural Representations (INRs) as an effective mechanism for learning surface characteristics of proteins.

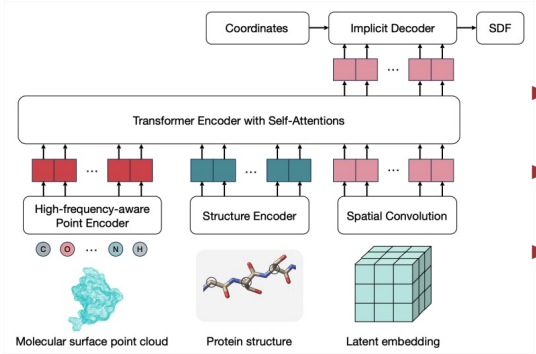
ProteinINR framework



An illustration for pre-training sequences, structures, and surfaces to solve downstream tasks.

- 1. Pre-train the sequences and use the node feature as input for the structure encoder.
- 2. Initialize structure encoder with pre-trained surface encoder weights.
- 3. Pre-train the structure encoder through multi-view contrastive learning based on GearNet.

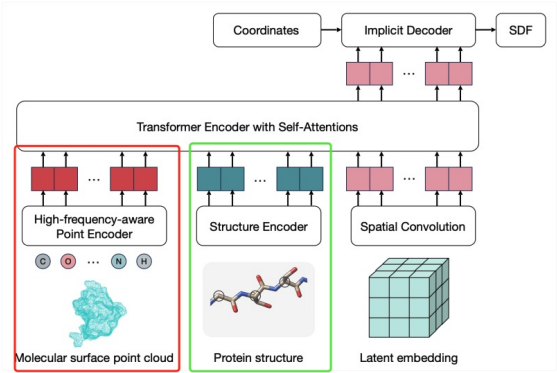
ProteinINR architecture



- Point and structure encoder
  - Structure embeddings: $\mathbf{s} \in R^{R \times h}$
  - Point embeddings: $\mathbf{p} \in R^{M \times h}$
- Spatially arranged latent representations
  - Latent feature: $\mathbf{z} \in R^{L \times h}$
- Transformer encoder for INRs
  - $\mathbf{h} = Concat(\boldsymbol{p}, \boldsymbol{s}, \boldsymbol{z}) \in R^{(M+R+L) \times h}$
- INR decoder and SDF regression
  - $\tilde{\boldsymbol{s}} = D_\phi(\boldsymbol{x}, \boldsymbol{z})$
  - $min_{\psi,\boldsymbol{z}} \frac{1}{NK_n} \sum_{n=1}^{N} \sum_{i=1}^{K_n} \|clamp(\boldsymbol{s}, \delta) - clamp(\tilde{\boldsymbol{s}}, \delta)\|_2^2$

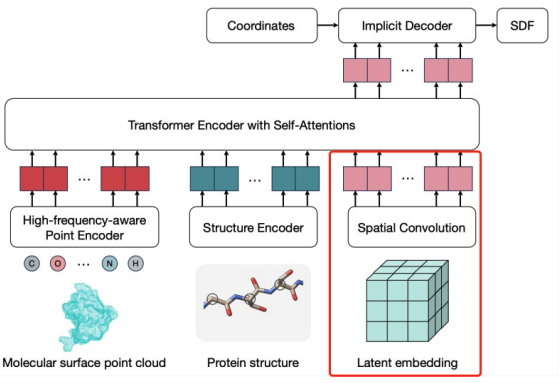
Performance on downstream tasks

Point and structure encoder



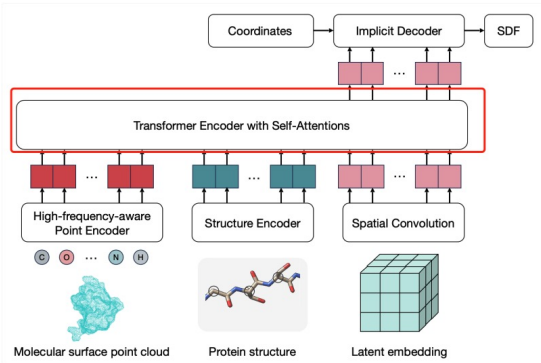
- The input is a protein point cloud $P \in R^{N\times 3}$.
  - N denotes the number of points in the point cloud of the protein molecular surface, and we randomly sampled 16,384 points to input the point encoder in our experiments.
- Structure embeddings: $\mathbf{s} \in R^{R\times h}$
  - R denotes the length of residues
- Point embeddings: $\mathbf{p} \in R^{M\times h}$
  - Downsample the points into a reduced set of M by Kernel Point Convolution (KPConv) networks (Thomas et al., 2019)

Spatially arranged latent representations



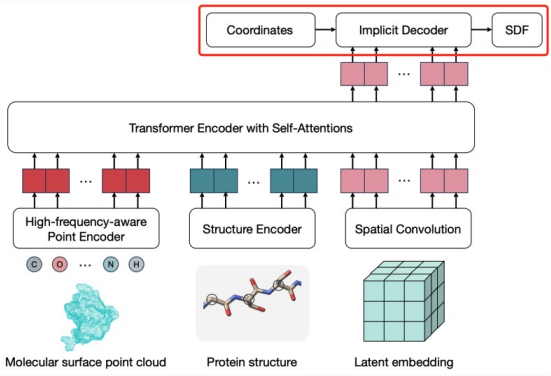
- 3D convolutions(Spatial Functa (Bauer et al., 2023)):
  - Latent feature: $\mathbf{z} \in R^{L \times h}$
  - Rearranged into a Three-dimensional voxel grid $\mathbf{z} \in R^{i \times j \times k \times c}$
  - 3D convolutions layer
  - Rearranged into $\mathbf{z} \in R^{L \times h}$

Transformer encoder for INRs



- The latent representation (referred to as $\boldsymbol{z}$) of a protein instance's surface is obtained using a transformer encoder.
- The input is $\mathbf{h} = Concat(\boldsymbol{p}, \boldsymbol{s}, \boldsymbol{z}) \in R^{(M+R+L)\times h}$

INR decoder and SDF regression



- In ProteinINR, the locality-aware INR decoder D_ϕ utilizes the latent code $\boldsymbol{z}$ to predict the SDF $\tilde{\boldsymbol{s}}$ for K query coordinates $\boldsymbol{x} \in R^{K\times 3}$ near molecular surface of N protein samples(DeepSDF).
- INR decoder: $\tilde{\boldsymbol{s}} = D_\phi(\boldsymbol{x}, \boldsymbol{z})$
- $\tilde{\boldsymbol{s}}$:predicted SDF values; $\boldsymbol{s}$: SDF values
- $min_{\psi,\boldsymbol{z}} \frac{1}{NK_n} \sum_{n=1}^{N} \sum_{i=1}^{K_n} \|clamp(\boldsymbol{s}, \delta) - clamp(\tilde{\boldsymbol{s}}, \delta)\|_2^2$

INR training

- INR decoder is a Implicit Neural Representations (INRs):
  - The SDF is a mathematical expression that assigns a scalar value to a given coordinate x, expressing the distance d between the spatial point and the closest point on the shape’ s surface as follow: $F(\boldsymbol{x}) = s : \boldsymbol{x} \in R^3, s \in R$, We define the inside surface as $d < 0$ and the outside surface as d > 0.
- The SDF values are their distances from the nearest vertices point of a given molecular surface mesh.
- We utilized the MSMS program (Connolly, 1983;Sanner et al., 1996), which is well-established triangulation software for molecular surfaces. The sample points are sampled near to the molecular mesh obtained via MSMS, we computed the SDF values for the points acquired by the sampling approach utilized in DeepSDF.
- In this work, 500,000 points were generated for SDF training, serving as the SDF points independent of the protein point cloud input.

Dataset

- Pre-training datasets:

Table 5: The number of protein structures used per species

Proteome ID	Taxonomy	# structures
UP000006548	Arabidopsis thaliana	27393
UP000001940	Caenorhabditis elegans	19658
UP000000559	Candida albicans	5956
UP000000437	Danio rerio	24595
UP000002195	Dictyostelium discoideum	12592
UP000000803	Drosophila melanogaster	13424
UP000000625	Escherichia coli	4363
UP000008827	Glycine max	55747
UP000005640	Homo sapiens	23280
UP000008153	Leishmania infantum	7903
UP000000805	Methanocaldococcus jannaschii	1772
UP000000589	Mus musculus	21571
UP000001584	Mycobacterium tuberculosis	3988
UP000059680	Oryza sativa subsp. japonica	43623
UP000001450	Plasmodium falciparum	5162
UP000002494	Rattus norvegicus	21209
UP000002311	Saccharomyces cerevisiae	6026
UP000002485	Schizosaccharomyces pombe	5123
UP000008816	Staphylococcus aureus	2885
UP000002296	Trypanosoma cruzi	18992
UP000007305	Zea mays	39258
Swiss-Prot	-	541938
Total	-	906458

- Downstream Classification task

Table 6: The number of datasets for downstream tasks.

Dataset	# Train	# Validation	# Test
Enzyme Commission	15,170	1,686	1,860
Gene Ontology	28,305	3,139	3,148
Fold Classification	12,312	736	718

- Enzyme Commission(EC): predicts EC numbers of proteins
- Gene Ontology(GO): predicts whether a protein is associated with a specific GO term.
- Fold cclassification(FC): predicts fold labels of proteins.

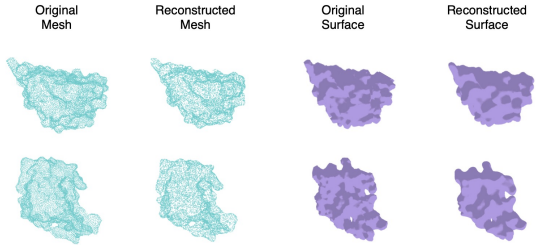
Experiments

Performance on downstream tasks

Method	EC		GO-BP		GO-MF		GO-CC		FC	Sum
	F_{max}	AUPR	F_{max}	AUPR	F_{max}	AUPR	F_{max}	AUPR	Acc	
ESM-1b†	86.9	88.4	45.2	33.2	65.9	63.0	47.7	32.4	-	-
ESM-2†	87.4	88.8	47.2	**34.0**	66.2	**64.3**	47.2	35.0	-	-
GearNet	81.6	83.7	44.8	25.2	60.4	52.9	43.3	26.8	46.8	465.5
GearNet-INR	81.4	83.7	44.7	26.5	59.9	52.1	43.0	27.2	47.6	466.1
GearNet-MC	87.2	88.9	49.9	26.4	64.6	55.8	46.9	27.1	51.5	498.3
GearNet-INR-MC	86.9	88.9	49.8	26.0	65.4	56.1	47.7	26.6	51.1	498.5
ESM-GearNet-MC	89.0	89.7	**53.5**	27.5	**68.7**	57.9	49.4	32.4	**53.8**	521.9
ESM-GearNet-INR	89.0	90.3	50.8	33.4	67.8	62.6	**50.6**	**36.9**	48.9	**530.3**
ESM-GearNet-INR-MC	**89.6**	**90.3**	51.8	33.2	68.3	58.0	50.4	35.7	50.8	528.1

- ESM-GearNet-INR-MC and ESM-GearNet-INR outperform the previous state-of-the-art model, ESM-GearNet-MC, when taking the summation of all scores.

Experiments



Representing protein surface shapes using ProteinINR

- The procedure for acquiring a triangular mesh that corresponds to a specific protein using INR parameters from ProteinINR is outlined.
  - Initially, the SDFs are calculated for the vertices of a voxel grid with a regular size of 128. Following this, the marching cubes algorithm (Chernyaev, 1995) is employed to compute the mesh.Protein surface samples reconstructed using ProteinINR are depicted in the Figure.
- ProteinINR effectively preserves intricate information, even hole or ring-like shapes.

Conclusion

- Propose a pre-training strategy that incorporates information from protein sequences, structures, and surfaces.
- Utilize Implicit Neural Representations (INRs) as an effective mechanism for learning surface characteristics of proteins.