Pre-Training Protein Bi-level Representation Through Span Mask Strategy On 3D Protein Chains

Jiale Zhao [1 2] **Wanru Zhuang** [1 3] **Jia Song** [1 4] **Yaqi Li** [1] **Shuqi Lu** [1]

Abstract

In recent years, there has been a surge in the development of 3D structure-based pre-trained protein models, representing a significant advancement over pre-trained protein language models in various downstream tasks. However, most existing structure-based pre-trained models primarily focus on the residue level, i.e., alpha carbon atoms, while ignoring other atoms like side chain atoms. We argue that modeling proteins at both residue and atom levels is important since the side chain atoms can also be crucial for numerous downstream tasks, for example, molecular docking. Nevertheless, we find that naively combining residue and atom information during pre-training typically fails. We identify a key reason is the information leakage caused by the inclusion of atom structure in the input, which renders residue-level pre-training tasks trivial and results in insufficiently expressive residue representations. To address this issue, we introduce a span mask pre-training strategy on 3D protein chains to learn meaningful representations of both residues and atoms. This leads to a simple yet effective approach to learning protein representation suitable for diverse downstream tasks. Extensive experimental results on binding site prediction and function prediction tasks demonstrate our proposed pre-training approach significantly outperforms other methods. Our code will be made public.

1. Introduction

Protein modeling is essential, as proteins play critical roles in various cellular processes, such as transcription, translation, signaling, and cell cycle regulation. In recent years, advances in deep learning have significantly contributed to the development of pre-trained models for generating high-quality protein representations, enabling the prediction of diverse properties, like protein classification and function. Although many previous studies have focused on pre-training based on protein sequences (Rao et al., 2021; Lin et al., 2023; Chen et al., 2023a), the significance of protein structure as a determinant of protein function has led to the emergence of 3D structure-based models (Zhang et al., 2022b; 2023; Lee et al., 2023). The development of these models has been greatly enhanced by recent breakthroughs in highly accurate deep learning-based protein structure prediction methods (Jumper et al., 2021).

A common pre-training approach in previous studies leverages advances in self-prediction techniques within the field of natural language processing (Brown et al., 2020; Devlin et al., 2018). In this context, the objective for a given protein can be framed as predicting a specific segment of the protein based on the information from the remaining structure. Standard pre-training tasks often involve randomly masking certain residues, predicting the types and positions of the masked residues (i.e., the coordinates of alpha carbon atoms), and the angles between them and other residues (Zhang et al., 2022b; Guo et al., 2022; Chen et al., 2023b). Through this process, the model effectively captures the information of the residues, thereby obtaining a high-quality representation of the protein's residues.

Although many pre-trained models have demonstrated success in modeling the 3D structure of proteins, the majority focus on the residue level, utilizing only the geometric positions of alpha carbon atoms or main chain atoms. However, side chain atoms are also essential in numerous downstream tasks, such as molecular docking, due to their interactions with small molecules (Krishna et al., 2023). Thus, it is imperative to incorporate information from all atoms in protein modeling. In our empirical study, we find that naive atom-level modeling typically fails: (1) Simply replacing residue input with atom input, conducting pre-training tasks at the atom level without considering residue level, such as predicting atom coordinates and angles, does not yield significant improvements. This suggests that residue-based modeling is indispensable. (2) Naively combining residue

[*]Equal contribution [1]DP Technology, Beijing, China [2]Institute of Computing Technology,UCAS, Beijing, China [3]Xiamen University, School of Informatics, Xiamen, China [4]Xiamen University, Institute of Artificial Intelligence, Xiamen, China. Correspondence to: Shuqi Lu <lusq@dp.tech>.

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

and atom information and conducting pre-training tasks at both atom and residue levels does not lead to enhanced performance. We identified a key reason to be the information leakage of residue-level tasks caused by the inclusion of atom structure in the input, which renders residue-level pre-training tasks trivial and results in insufficiently expressive residue representations. As shown in Figure 1, the current residue structure can be easily predicted when surrounded by atom structure. This indicates the necessity of properly modeling residue information. To address this, we introduce a Span Mask strategy on 3D Protein Chains (SMPC). We mask the residue type of consecutive biologically meaningful substructures, retaining only the alpha carbon atom for the span-masked residues while eliminating all other atoms. This approach increases the difficulty of residue tasks, making it impossible to infer residue type and structure solely from side-chain and backbone atoms, thus, prompting the model to learn meaningful residue representations.

Our analysis leads to a simple yet effective approach. In this paper, we propose a **V**ector **A**ware **B**ilevel **S**parse Attention Network (Vabs-Net), a pre-trained model that simultaneously models residues and atoms. Vabs-Net employs a carefully designed edge vector encoding module and a two-track sparse attention module which comprise an atom-atom track and a residue-residue track, to encode atoms and residues. These tracks interact via alpha carbon atoms. To ensure meaningful tasks at the residue level, we adopt the SMPC pre-training strategy. At the atom level, we employ a random noise strategy. Through a series of structure pre-training tasks, such as position and torsion angle prediction, our model effectively learns residue and atom representations jointly, enabling comprehensive protein modeling at both levels. An overview of our approach is shown in Figure 2.

We design a series of downstream tasks to assess the impact of Vabs-Net. These tasks include Enzyme Commission (EC) number prediction and Gene Ontology (GO) term prediction, which focus on the global properties of proteins, as well as binding site prediction, which emphasizes protein local properties. Furthermore, we incorporated molecular docking, a task known to rely on atom modeling of proteins, as each atom may interact with small molecules. We incorporate the atom-level representation generated by our model and input it into an existing docking model, subsequently evaluating its effectiveness through performance improvement in molecular docking tasks. Across these tasks, our model, Vabs-Net, significantly outperforms previous baselines, demonstrating its superior efficacy. In summary, our contributions are as follows:

1. We present a novel pre-training model Vabs-Net that learns effective residue and atom representations simultaneously.
2. We propose SMPC strategy to enhance the pre-training task at the residue level, resulting in a significant improvement in performance.
3. Vabs-Net achieves state-of-the-art results on various downstream tasks, including the molecular docking task. These results demonstrate the efficacy of the protein representations generated by Vabs-Net.

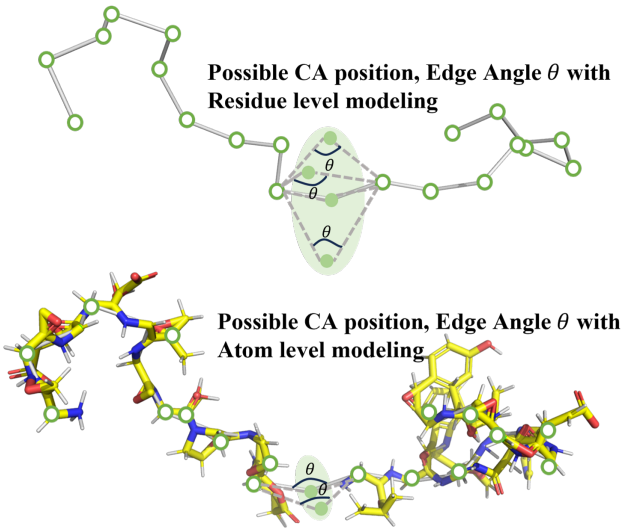

Figure 1. With all-atom added, the possible range for residue position is limited, thus resulting in easier prediction for residue position and angle between edges, etc. When all atoms are utilized, the prediction of residue positions and inter-edge angles relies predominantly on other atoms rather than residues themselves.

2. Related Works

Pre-trained Protein Models Pre-training using protein sequences has recently attracted significant attention due to its potential applications. A family of models, which includes ProtTrans (Elnaggar et al., 2020), ESM-1b (Rao et al., 2021), and ESM2 (Lin et al., 2023), utilize individual protein sequences as input and undergo pre-training by optimizing the masked language model objective. A prompt-based pre-trained protein model (Wang et al., 2022b) has been introduced to guide multi-task pre-training. Additionally, xTrimoPGLM (Chen et al., 2023a) is proposed to further explore the potential of a unified pre-training strategy.

Recent advancements in structure-based pre-training models have demonstrated significant improvements in performance across a diverse range of downstream tasks. The prevailing protein pre-training models primarily focus on predicting a diverse range of physical quantities, including torsion angles, angles between edges, alpha carbon positions, and distances, as well as residue types. (Zhang et al., 2022b;

Guo et al., 2022; Chen et al., 2023b; Wang et al., 2022b). Contrastive learning has demonstrated exceptional performance in a recent study based on GearNet (Zhang et al., 2022b). Additionally, recent attempts for pre-training with protein surfaces have been made (Lee et al., 2023). Numerous diffusion-based models have been proposed (Huang et al., 2023; Zhang et al., 2023). For instance, SiamDiff (Zhang et al., 2023) employs a pre-trained protein encoder through sequence-structure joint diffusion modeling. HotProtein(Chen et al., 2022), a structure-aware pre-training model, is proposed to improve thermostability prediction. A self-supervised pretraining method is also used to predict compound–protein affinity and contact(You & Shen, 2022).

Although significant progress has been made in the development of residue-level pre-training models, there is a scarcity of focus on atom-level pre-training models for proteins. Among very few of them, Siamdiff (Zhang et al., 2023) can function as either a residue or atom-level model.

The GearNet style model (Zhang et al., 2022b; Lee et al., 2023) has demonstrated superior performance compared to other residue-level models (Zhang et al., 2022b;b). In addition to GearNet, we select Siamdiff (Zhang et al., 2023) as another primary benchmark, as it has been reported to be the most effective atom-level pre-training model. Together, these two models provide a comprehensive and robust foundation for our analysis and comparison.

Residue Level Encoder. In addition to numerous pre-training models, several studies attempt to encode protein structures in various downstream tasks without pre-training. To leverage structural information, plenty of models using structure or structure+sequence information have been proposed. GVP (Jing et al., 2021) iteratively updates the scalar and vector representations of a protein to learn its representation. Additionally, CDConv (Fan et al., 2022) employs both irregular and regular approaches to model the geometry and sequence structures. Furthermore, ProNet (Wang et al., 2023) utilizes torsion angles to capture side-chain positions.

Atom Level Encoder. Because of the importance of side chain atoms, there have also been some all-atom-level encoders proposed recently. IEConv (Hermosilla et al., 2020) introduce a novel convolution operator and a hierarchy pooling operator, which facilitated multi-scale protein analysis. FAIR (ZHANG et al., 2023) attempts to encode atom and residue-level information by employing two separate encoders, without incorporating any interaction between them.

3. Method

3.1. Vector Aware Bilevel Sparse Attention Network

Protein Graph Construction We first construct both residue-level and atom-level graphs. A residue level graph can be defined as $\mathcal{G}_{res} = (\mathcal{V}_{CA}, \mathcal{E}_{res})$, where $\mathcal{V}_{CA}$ includes all alpha carbon nodes and $\mathcal{E}_{res}$ includes all residue edges. An atom level graph can be defined as $\mathcal{G}_{atom} = (\mathcal{V}, \mathcal{E}_{atom})$, where $\mathcal{V}$ includes all-atom nodes, i.e. $\mathcal{V}_{CA} \subset \mathcal{V}$ and $\mathcal{E}_{atom}$ includes all atom level edges. The construction of edges involves the selection of k nearest neighbors in the space for both residue and atom-level graph nodes. The residue-level graph shares alpha carbon nodes with the atom-level graph. Consequently, the bilevel graph of protein can be written as $\mathcal{G} = (\mathcal{V}, \mathcal{E}_{res}, \mathcal{E}_{atom})$. The model architecture can be seen in Figure2. This approach facilitates information exchange between atom and residue levels, thereby not only allowing atoms to acquire an expanded receptive field but also enabling residues to obtain detailed structural information.

To comprehensively encapsulate global information on protein, we introduce a virtual origin point positioned at the geometry center. This point is connected with every atom and residue, integrating overall protein representation.

Node Encoding In our model, node embeddings incorporate both atom type and residue type. Additionally, we utilize the Large language model (LLM) version ESM (Lin et al., 2023) to leverage sequence information. The representation of atom nodes can be seen as follows:

$$\boldsymbol{x} = \text{Embeding}(ATOM) + \text{Embeding}(RES) + \boldsymbol{W_E} \boldsymbol{v}_E,$$

where $ATOM$ and RES stand for the type of atom and residue. $\boldsymbol{W_E}$ maps the shape of ESM repr. to node shape.

Edge Distance Encoding To encode the distance between atoms or residues, we use the Gaussian kernel.

$$g_{i,j}^k = \frac{1}{\sigma^k \sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\alpha_{i,j}\|\boldsymbol{r}_i - \boldsymbol{r}_j\| - \mu^k + \beta_{i,j}}{\sigma^k}\right)^2\right),$$
$$\boldsymbol{g}_{i,j} = \text{concat}(g_{ij}^k), \quad k = \{1, 2, \ldots, K\}$$

where $\boldsymbol{g}_{i,j}^k$ is the k-th Gaussian kernel of the nodes pair (i, j), K is the number of kernels. The input 3D coordinate of the i-th atom is represented by $\boldsymbol{r}_i \in \mathbb{R}^3$, and $\alpha_{i,j}$ and $\beta_{i,j}$ are learnable scalars indexed by the pair of node types. μ^k and σ^k are predefined constants. Specifically, $\mu^k = w \times (k-1)/K$ and $\sigma^k = w/K$, where the width w is a hyper-parameter.

Edge Vector Encoding In previous protein-pretraining modeling, structural information is traditionally encoded using the distance between residues. While this distance-based structural encoding may suffice for residue modeling, encoding distance is not informative enough for atom modeling since the number of atoms is much larger than that of the residueB. Encoding edge directions in both the residue local coordinate system and the absolute global coordinate system can alleviate this problem. The residue local coordinate system can be formulated as following steps with three

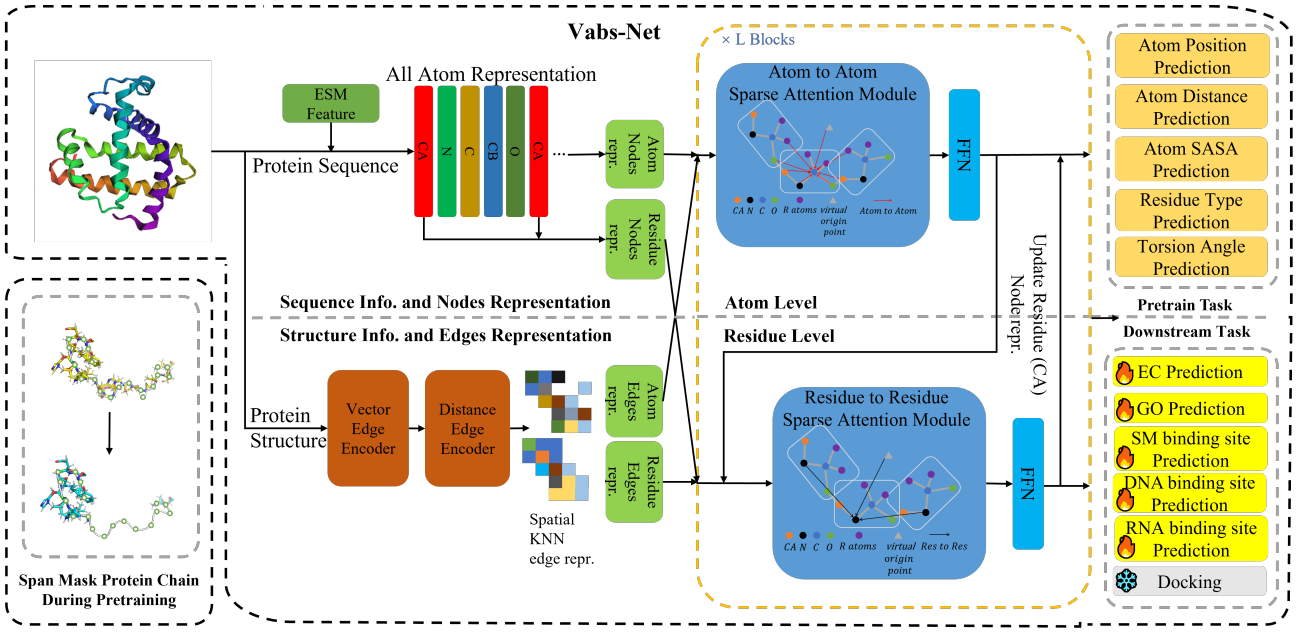

Figure 2. An overview of Vabs-Net architecture. We use atom type, residue type, and preprocessed ESM features to encode atom nodes. Residue nodes share representation with their corresponding alpha carbon. Encoding of edges is through vector edge encoder and distance edge encoder to encode direction and distance of edges. We input node and edge encoding into a two-track sparse attention module. Each track includes a sparse attention module and a feedforward neural network. This module first updates atom representations with the atom-atom track and then updates alpha carbon atom nodes by residue-residue track. In this way, two tracks interact through alpha carbon atom nodes. Finally, representations of nodes and edges are used for various pre-training and downstream tasks. In addition, we show the span mask protein chain strategy on the left. Atom nodes other than alpha carbon are removed in the masked area of the span mask protein chain method during pre-training.

backbone atoms(CA, C, N) in each residue. We use $\boldsymbol{r}$ to indicate the position of atoms and set $\boldsymbol{v}_N = \boldsymbol{r}_N - \boldsymbol{r}_{CA}$ and $\boldsymbol{v}_C = \boldsymbol{r}_C - \boldsymbol{r}_{CA}$ as the vector of edges between N and CA, CA and C in the absolute global coordinate system. More generally, We define $\boldsymbol{v}_l$ and $\boldsymbol{v}_g = \boldsymbol{r}_{atom} - \boldsymbol{r}_{CA}$ are the vectors of edges from CA to an atom in the same residue in the local residue coordinate system and global coordinate system, respectively. They serve to encode edge directions in subsequent steps. Then we can get a local coordinate system with a rotational matrix $\boldsymbol{R}$, which is used to transform vectors in global coordinates to local coordinates. $\boldsymbol{R}$ is constructed with unit vector $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ in XYZ axis of residue local coordinate.

$$\boldsymbol{u} = \frac{\boldsymbol{v}_N - \boldsymbol{v}_C}{\|\boldsymbol{v}_N - \boldsymbol{v}_C\|}, \quad \boldsymbol{v} = \frac{\boldsymbol{v}_N \times \boldsymbol{v}_C}{\|\boldsymbol{v}_N \times \boldsymbol{v}_C\|}, \quad \boldsymbol{w} = \boldsymbol{u} \times \boldsymbol{v},$$
$$\boldsymbol{R} = [\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}], \quad \boldsymbol{v}_l = \boldsymbol{R}^\top \boldsymbol{v}_g.$$

Empirical evidence (Yifan et al., 2022) suggests that this higher-dimensional Fourier encoding is more apt for subsequent processing by neural networks. The Fourier encoded representation is defined as:

$$\gamma(\theta) = \left[\sin\left(2^0 \frac{\pi}{\theta}\right), \cos\left(2^0 \frac{\pi}{\theta}\right), \ldots, \sin\left(2^{L-1} \frac{\pi}{\theta}\right), \cos\left(2^{L-1} \frac{\pi}{\theta}\right)\right].$$

Thus we encode the direction of an edge from atom i to atom j by concatenating Fourier encoding angle between edges and xy, yz, and xz planes.

$$\boldsymbol{f}_{i,j} = \left[\gamma(\phi_{ij}^{xy}) \| \gamma(\phi_{ij}^{yz}) \| \gamma(\phi_{ij}^{xz}) \| \gamma(\varphi_{ij}^{xy}) \| \gamma(\varphi_{ij}^{yz}) \| \gamma(\varphi_{ij}^{xz})\right]$$

We set $\phi_{ij}^{xy} = \arcsin(\boldsymbol{v_g} \cdot \hat{\boldsymbol{z}}_{\boldsymbol{g}})$ the angle between the vector from node i to node j and the global absolute xy plane, where $\hat{\boldsymbol{z}}_{\boldsymbol{g}}$ is the unit vector in the z-direction in the global absolute coordinate system. $\varphi_{ij}^{xy} = \arcsin(\boldsymbol{v_l} \cdot \hat{\boldsymbol{z}}_{\boldsymbol{l}})$ is the angle between the residue local vector from node i to node j and local residue xy plane, where $\hat{\boldsymbol{z}}_{\boldsymbol{l}}$ is the unit vector in local coordinate. Other angles can be obtained using the same method.

Learnable positional encoding is also applied to encode sequential positions. As a result, the representation of edge between atoms or residues can be summarized as follows:

$$\boldsymbol{e}_{i,j} = \boldsymbol{W_g} \boldsymbol{g}_{i,j} + \boldsymbol{W_f} \boldsymbol{f}_{i,j} + \boldsymbol{l}(i-j),$$

where $\boldsymbol{l}(i-j)$ is the learnable positional encoding based on the sequence distance between two residues. $\boldsymbol{W_g}$ and $\boldsymbol{W_f}$ maps the dimension of distance and direction encoding to dimension of edge.

Sparse Attention Module(SAM) Numerous pre-training models for molecules based on Graphormer have achieved state-of-the-art performance in downstream tasks (Zhou et al., 2023). However, a fully connected Graphormer model can pose significant computational challenges (Lu et al., 2023). To tailor Graphormer's attention mechanism to the all-atom level graph, we introduce a Sparse Graph Attention Module aimed at reducing computational load and preventing overfitting in downstream tasks. The sparse attention module is formulated as follows:

$$a_{i,j}^{l} = \text{softmax}_{j\in\mathcal{N}_i}\left(\frac{\boldsymbol{W_Q}\boldsymbol{x}_i^l(\boldsymbol{W_K}\boldsymbol{x}_j^l)^\top}{\sqrt{d_h}} + \boldsymbol{W_B}\boldsymbol{e}_{i,j}\right),$$

$$\boldsymbol{x}_i^{l+1} = \sum_{j\in\mathcal{N}_i} a_{i,j}^{l}\boldsymbol{W_V}\boldsymbol{x}_j^{l},$$

where $\boldsymbol{x}_i^l$ is the representation of node i in the lth layer. $a_{i,j}^l$ is the attention weight. Set $\mathcal{N}_i$ includes all nodes connected to node i. $\boldsymbol{W_B}$ is a linear layer for bias calculation. $\boldsymbol{e}_{i,j}$ is the edge representation between node i and node j. The h dimension in multi-head attention is omitted.

3.2. Pre-training Tasks

Span Mask Protein Chain(SMPC) Residue Type Prediction During the pre-training phase, span masking is applied to mask residue types of consecutive biologically meaningful substructures. Specifically, only the alpha carbon atom is retained for the span-masked residues, while all other atoms are eliminated. This aims to prevent information leakage from side chain atoms and other backbone atoms, which could make residue type prediction and torsion angle prediction trivial. We find even backbone atoms can leak residue type to some extent. Subsequently, only alpha carbon representation is employed for residue type prediction. Cross-entropy loss serves as the objective function for predicting residue types.

Span Mask Protein Chain(SMPC) Side Chain and Backbone Torsion Angle Prediction Drawing upon the feature of alpha carbon atoms in SMPC part, Vabs-Net predicts the cosine and sine components for seven side chain and backbone torsion angle angles. In addition to the L1 norm loss for cosine and sine values, the loss function encompasses a term that ensures the normalization of the sum of squared sine and cosine values to promote accurate angular representation. The loss function for this task is similar to the one used in AlphaFold2 (Jumper et al., 2021).

Atom Position and Distance Prediction Apart from the residue level pre-training tasks. To learn atom-level structural information, We randomly choose consecutively residues independent of SMPS. For these residues, we simply add Gaussian noise to atoms in those residues instead of removing atoms. For these atoms, positions are predicted by a movement prediction head.

The position movement prediction head we adopted is similar to the one used in Graphormer-3D (Shi et al., 2022; Lu et al., 2023).

$$a_{i,j} = \text{softmax}_{j\in\mathcal{N}_i}\left(\frac{\boldsymbol{W'_Q}\boldsymbol{x}_i(\boldsymbol{W'_K}\boldsymbol{x}_j)^\top}{\sqrt{d_h}} + \boldsymbol{W'_B}\boldsymbol{e}_{i,j}\right),$$

$$\boldsymbol{b}_{i,j} = \sum_{j\in\mathcal{N}_i} a_{i,j}(\boldsymbol{r}_i^N - \boldsymbol{r}_j^N)\boldsymbol{W'_V}\boldsymbol{x}_j,$$

$$\boldsymbol{r}_i^P = \boldsymbol{r}_i^N + [\boldsymbol{W_{px}}\boldsymbol{b}_{i,j}^x \| \boldsymbol{W_{py}}\boldsymbol{b}_{i,j}^y \| \boldsymbol{W_{pz}}\boldsymbol{b}_{i,j}^z],$$

$$\mathcal{L}_{dist} = \frac{1}{n^2}\sum_{i\in\Omega|j\in\Omega} \|d_{ij}^P - d_{ij}^R\|_1,$$

$$\mathcal{L}_{pos} = \frac{1}{n}\sum_{i\in\Omega} \|\boldsymbol{r}_i^P - \boldsymbol{r}_i^N\|_2,$$

where $\boldsymbol{W'_Q}, \boldsymbol{W'_K}, \boldsymbol{W'_V}, \boldsymbol{W'_B}$ are projection heads for movement prediction head. $\boldsymbol{r_i^N}$ is the coordinate of node i with noise. $\boldsymbol{r_i^R}$is the real coordinate of node i. Ω is the set of nodes with noise. $\boldsymbol{x}_i$ and $\boldsymbol{e}_{i,j}$ are representations of node i and representation of edge between node i and node j. $d_{i,j}^P = \|\boldsymbol{r}_i^P - \boldsymbol{r}_j^P\|_2$ and $d_{i,j}^R = \|\boldsymbol{r}_i^R - \boldsymbol{r}_j^R\|_2$ are predicted distance and real distance between node i and node j.

Solvent Accessible Surface Area(SASA) Prediction SASA describes the surface area of a biomolecule, such as a protein or nucleic acid, that is accessible to a solvent. It could be used to understand the shape of a protein. We used freesasa (Mitternacht, 2016) to calculate SASA of each atom. The loss function for SASA prediction is L1 loss.

4. Experiments

In this section, we report our experiment setup and results in training and evaluation of our models for pre-training and downstream tasks. Detailed information about downstream task settings can be seen in Appendix D.

4.1. Pre-training

Setting During this pre-training SMPC phase, we mask consecutive spans of the protein residues. The span mask lengths follow a Poisson distribution with a mean (λ) of 6 and cumulatively makeup 30% of the protein chain. For those span-masked residues, we only retain alpha carbon atoms. We then construct a protein KNN graph after SMPC. Also, we randomly sample 30% of atoms of consecutive residues and add Gaussian noise. The consecutive span of

Table 1. Model performance on EC numbers and GO terms prediction tasks(F_{max}). SMPC stands for span mask protein chain.

METHOD	BP	MF	CC	EC
MTL(WANG ET AL., 2022B)	0.445	0.640	0.503	0.869
GRADNORM(CHEN ET AL., 2018; WANG ET AL., 2022B)	0.466	0.643	0.504	0.874
LM-GVP(WANG ET AL., 2022A)	0.417	0.545	0.527	0.664
ROTOGRAD(JAVALOY & VALERA, 2022)	0.470	0.638	0.509	0.876
CDCONV(FAN ET AL., 2022)	0.453	0.654	0.479	0.820
PROMPTPRO.(WANG ET AL., 2022B)	0.495	0.677	0.551	0.888
ESM_1B(LIN ET AL., 2023)	0.470	0.657	0.488	0.864
GEARNET(ZHANG ET AL., 2022B)	0.490	0.650	0.486	0.874
SIAMDIFF(ZHANG ET AL., 2023)	-	-	-	0.857
GEARNET-ESM(ZHANG ET AL., 2022B)	0.516	0.684	0.506	0.890
SIAMDIFF-ESM(ZHANG ET AL., 2023)	-	-	-	0.897
GEARNET-ESM-INR-MC(LEE ET AL., 2023)	0.518	0.683	0.504	0.896
VABS-NET-NO-SMPC	0.496	0.667	0.552	0.876
VABS-NET	**0.531**	**0.695**	**0.579**	**0.900**

residues also follows the same Poisson distribution with a mean of 6. Gaussian noise is added to atoms with a scale of 0.5Å. More detailed information about our model can be seen in Appendix C. The pre-training dataset is constructed from the Protein Data Bank and structures predicted by AlphaFold (Jumper et al., 2021). To obtain high-quality data, structures from the Protein Data Bank Database (Rose et al., 2016) with a resolution greater than 9 are filtered out. Structures from the AlphaFold2 Database with a pLDDT lower than 70 are filtered out. Additionally, MMSeq2 (Mirdita et al., 2021) is utilized to cluster the dataset. Finally, we get 163412 clusters from 1.3 million structures to accelerate pre-training by preventing the model from learning repetitive similar structures. During the training process, for each epoch, we randomly sample one structure from each cluster, thus leading to 163412 samples in one epoch. This leads to a more efficient pre-training, where we only need to train 100 epochs (containing only 16w clusters/samples in one epoch) to reach the best performance.

4.2. Protein Function Prediction

Enzyme Commission (EC) number prediction involves the anticipation of the EC numbers associated with diverse proteins, delineating their role in catalyzing biochemical reactions. EC numbers are drawn from the third and fourth tiers of the EC tree, resulting in the formation of 538 binary classification tasks (Hermosilla et al., 2020). Moreover, Gene Ontology (GO) term prediction focuses on determining whether a protein is associated with specific GO terms. These terms categorize proteins into interconnected functional classes within three ontologies: molecular function (MF), biological process (BP), and cellular component (CC).

Setting For EC and GO prediction, we use the same datasets as former researches (Zhang et al., 2022b). We utilize the protein-level F_{max} to assess performance.

Result The results of our model on BP, MF, CC, and EC can be seen in Table 1, which clearly shows that our Vabs-Net model exhibits superior performance in comparison to all baselines with respect to BP, MF, CC, and EC. A substantial enhancement compared to GearNet-ESM demonstrates the efficacy of atom-level encoding. The superior performance over Siamdiff-ESM shows the importance of residue-level encoding and the effectiveness of the proposed SMPC pre-training strategy in learning efficient residue-level representations. To further validate SMPC's efficacy, we compared our model with and without SMPC integration during pre-training. The findings demonstrate that incorporating SMPC contributes to effective residue-level representation learning.

4.3. Protein Binding Site Prediction

Precise prediction of protein-ligand binding site forms the bedrock of comprehending diverse biological activities and facilitating the design of novel drugs (Pei et al., 2023). The binding site prediction task is to predict whether an atom or a residue is a binding site (binary classification task) without inputting ligands. The localization of the binding site relies heavily on the fine-grained local structure.

The prediction of binding sites is hindered by the limited training data, as evident in the constrained sizes of the training sets for DNA and RNA, standing at 573 and 495 samples, respectively (Zhang et al., 2022a). Traditional methods typically entail an extensive input of features into the model such as atom mass, B-factor, electronic charge, whether it is in a ring, and the van der Waals radius of the atom, among others (Xia et al., 2021). Hence, our aim is to leverage pre-training methods to learn meaningful atom representations and mitigate the risk of overfitting.

Prior research efforts generally focus on handling single tasks like small molecules or nucleic acid. Comprehensive binding site prediction experiments are made on small

Table 2. Small Molecule Binding Site Prediction Result in terms of IoU(%).

METHOD	PRE-TRAIN	B277	DT198	ASTEX85	CHEN251	COACH420
FPOCKET(LE GUILLOUX ET AL., 2009)	×	31.5	23.2	34.1	25.4	30.0
SITEHOUND(HERNANDEZ ET AL., 2009)	×	36.4	23.1	38.9	29.4	34.9
METAPOCKET2(MACARI ET AL., 2019)	×	37.3	25.8	37.5	32.8	37.7
DEEPSITE(JIMÉNEZ ET AL., 2017)	×	34.0	29.1	37.4	27.4	33.9
P2RANK(KRIVÁK & HOKSZA, 2018)	×	49.8	38.6	47.4	**56.5**	45.3
ESM2_150M(LIN ET AL., 2023)	√	19.6	16.6	20.5	18.9	22.0
GEARNET(ZHANG ET AL., 2022B)	√	39.9	35.8	41.0	36.4	41.3
SIAMDIFF(ZHANG ET AL., 2023)	√	37.7	31.0	40.7	35.3	40.3
VABS-NET	×	57.7	48.6	57.8	53.2	61.4
VABS-NET	√	**60.1**	**52.0**	**58.8**	56.3	**64.1**

Table 3. DNA Binding Site Prediction Result trained on DNA-573 Train, tested on DNA-129 Test.

METHOD	PRE-TRAIN	AUC
TARGETDNA(HU ET AL., 2016)	×	0.825
DNAPRED(ZHU ET AL., 2019)	×	0.845
SVMNUC(SU ET AL., 2019)	×	0.812
COACH-D(WU ET AL., 2018)	×	0.761
NUCBIND(SU ET AL., 2019)	×	0.797
DNABIND(LIU & HU, 2013)	×	0.858
GRAPHBIND(XIA ET AL., 2021)	×	<u>0.927</u>
ESM2_150M(LIN ET AL., 2023)	√	0.779
GEARNET(ZHANG ET AL., 2022B)	√	0.849
SIAMDIFF(ZHANG ET AL., 2023)	√	0.823
VABS-NET	×	0.912
VABS-NET	√	**0.940**

molecules, DNA, and RNA to test our model.

Setting We have constructed a large high-quality small molecule binding site dataset, and we use this dataset to train our model and other baselines, which will be open access to the public. For all-atom models, the label of Alpha C is the residue label. In DNA and RNA binding site prediction, we use the AUC for evaluation. Baselines are finetuned using the same setting as their original paper.

In the preparation of our small molecule training dataset, we utilize three distinct datasets: CASF-2016 coreset, PDBBind v2020 refined set, and MOAD. During dataset preparation, we follow the original ligand records in these datasets to extract the protein and ligand components based on the corresponding PDB ID from RCSB. The extracted segments undergo a structural repair process using in-house scripts. For proteins, this process includes the repair of missing residues, replenishment of absent heavy atoms, and addition of hydrogen atoms. In the case of ligands, the process involves repairing bond orders, adding hydrogen atoms, and determining the correct protonation states based on the pocket environment. A comparison of our dataset and frequently used scPDB can be seen in Appendix A. To avoid leakage, MMSeqs2 is used to filter out high protein sequence similarity(similarity above 40% which is also used in AlphaFold2 (Jumper et al., 2021)) with test sets.

Datasets for DNA and RNA are downloaded from the Biolip and Graphbind website.

To construct the validation set, we used MMSeqs2 to cluster the training set so that the sequence similarity between the validation set and the training set is lower than 40%.

Result Tables 2, 3 and 4 provide results of pre-trained models and none pre-trained models on binding site prediction. We found that Vabs-Net outperforms all of the baseline models. Baseline models for binding site prediction are typically not pre-trained. Therefore, we compared the best baselines with Vabs-Net without pre-training, revealing that Vabs-Net performs competitively with the best baselines when not pre-trained, thereby demonstrating the effectiveness of our backbone. Moreover, incorporating pre-training substantially enhances performance, surpassing other pre-trained models considerably. The comparison between other pre-training models and best baselines without pre-training shows the importance of direction encoding.

Table 4. RNA Binding Site Prediction Result trained on RNA-495 Train, tested on RNA-117 Test

METHOD	PRE-TRAIN	AUC
RNABIND+(WALIA ET AL., 2014)	×	0.717
SVMNUC(SU ET AL., 2019)	×	0.729
COACH-D(WU ET AL., 2018)	×	0.663
NUCBIND(SU ET AL., 2019)	×	0.715
AARNA(MIAO & WESTHOF, 2015)	×	0.771
NUCLEICNET(LAM ET AL., 2019)	×	0.788
GRAPHBIND(XIA ET AL., 2021)	×	<u>0.854</u>
ESM2_150M(LIN ET AL., 2023)	√	0.699
GEARNET(ZHANG ET AL., 2022B)	√	0.778
SIAMDIFF(ZHANG ET AL., 2023)	√	0.735
VABS-NET	×	0.834
VABS-NET	√	**0.880**

4.4. Molecular Docking

Predicting the binding structure of a small molecule ligand to a protein, a task known as molecular docking, is crucial for drug design (Pei et al., 2023). Equibind (Stärk et al., 2022) and Diffdock (Corso et al., 2023) are the two most commonly used models for this purpose. However, the diffusion-based Diffdock requires substantial computational resources; thus, we employ Equibind (Stärk et al., 2022) to evaluate the efficacy of features from our model. Equibind is trained using features extracted from various structural pre-training models. To be more specific, we trained 4 distinct Equibind models. The first one is Equibind itself. For the other three models, we add features from GearNet (pre-trained by multiview contrast learning), Siamdiff (atom-level), and our Vabs-Net to the node representation of Equibind. Those features are pre-processed before training. It is also important to note that Equibind is a residue-level model. To optimize the utilization of atom-level features from both the Siamdiff model and our own, we implement a single attention layer to aggregate atom features to the residue node representation of Equibind.

Setting For molecular docking, due to time constraints, we adopt a faster and more stable version of Equibind, which can also be seen in the official repository. More detail is in Appendix E. To conduct a fair comparison between our model and other pre-training models, we incorporate ESM features into all the models under consideration. Both Equibind and its variant enhanced with pre-trained features are trained for 300 epochs. We use PDBBind as a dataset, which is also used by Equibind (Stärk et al., 2022).

Result The effects of incorporating features from various structural protein pre-training models on the molecular docking task can be observed in Table 5. These findings demonstrate that the atom and residue representation acquired by our model outperforms those of other pre-training models. Incorporating atom-level encoding, our model demonstrates superior performance compared to GearNet. The improved results of both our model and Siamdiff over GearNet further emphasize the significance of atom-level encoding. Owing to the SMPC pretraining method, which aids in learning a more refined residue-level representation, our model surpasses Siamdiff in performance.

Table 5. Molecular docking results with Equibind.

METHOD	LIGAND RMSD	CENTROID RMSD
EQUIBIND	8.91	6.02
EQUIBIND+GEARNET	7.82	5.34
EQUIBIND+SIAMDIFF	7.75	5.11
EQUIBIND+VABS-NET	**7.23**	**4.11**

4.5. Ablation Studies

To analyze the effect of different components, we choose a protein function prediction task(EC) and a binding site prediction task(small molecule(SM)) for the ablation study.

We investigate different pre-training strategies and model configurations, with the findings presented in Table 6. (1) A comparison between models No.0 and No.1 demonstrates the significance of atom-level encoding, particularly in binding site prediction tasks where side chain atoms play a crucial role. (2) Upon comparing models No.0 and No.2, it becomes clear that residue-level encoding is also important for enhancing the receptive field and capturing residue-level representations. (3) Furthermore, the comparison between models No.0 and No.3 confirms the effectiveness of our proposed SMPC pre-training method in enhancing residue-level representations. (4) The notable decrease in performance observed after removing the vector edge encoder (No.3 and No.4) underscores the importance of direction encoding, rather than solely encoding distance. (5) While leveraging ESM can enhance performance (No.5 and No.3), the impact is not as significant as that of the vector edge encoder. (6) Increasing the KNN parameter does not yield a substantial improvement in the model's performance (No.6 and No.3), possibly due to KNN=30 being adequate for capturing most structural motifs (Tateno et al., 1997). (7) Our pre-training strategy significantly enhances the model's performance, as evidenced by the comparisons between No.7 and No.0. (8) To demonstrate the effectiveness of our Sparse Attention Module, we replaced it with the AttMLP module used in PiFold (Gao et al., 2022) in experiment No. 8. The comparison between No.0 and No.8 clearly illustrates the superior performance of our Sparse Attention Module. (9) The results of ablation studies No.9 and No.10 indicate that neither randomly masking protein chains (not necessarily consecutive) nor only masking side chain atoms (while retaining backbone atoms) can completely prevent information leakage.

Table 6. Ablation studies of our model in small molecule binding site prediction task(COACH420 test set) and EC prediction task.

NO.	METHOD	SM	EC
0	VABS-NET	**64.1**	**0.900**
1	NO ATOM LEVEL	56.3	0.896
2	NO RESIDUE LEVEL	62.2	0.886
3	NO SMPC	63.0	0.876
4	NO SMPC,VECTOR EDGE ENCODER	60.9	0.874
5	NO SMPC,ESM	61.9	0.867
6	NO SMPC,KNN=90	62.7	0.876
7	NO PRE-TRAIN	61.4	0.826
8	SAM→ATTMLP(GAO ET AL., 2022)	62.3	0.881
9	SPAN MASK SIDE CHAIN ATOMS	63.6	0.892
10	RANDOMLY MASK	63.3	0.884

5. Limitation

Compared to the sequential pre-training dataset (1103M in xTrimoPGLM), our structural pre-training dataset (130K) is much smaller. The AlphaFold database now contains over 100 million structures. Also, our pre-training is limited to single-chain pre-training, neglecting interaction between chains. In our subsequent efforts, we seek to tackle the limitations above.

6. Conclusion

In this study, we introduce the Vabs-Net model with the span mask strategy 3D protein chains pre-training technique, aiming to learn atom-level representation and improve residue-level representation. We have conducted extensive experiments on various task types to assess the effectiveness of our Vabs-Net model and the span mask protein chains pre-training approach. Our Vabs-Net model demonstrates superior performance, surpassing previous state-of-the-art models. In the subsequent phase of our research, we plan to enhance the integration of sequence and structural features.

Acknowledgments

This study was supported by the National Key Research and Development Program of China (2022YFA1004304).

Impact Statement

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

References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chen, B., Cheng, X., Geng, Y.-a., Li, S., Zeng, X., Wang, B., Gong, J., Liu, C., Zeng, A., Dong, Y., Tang, J., and Song, L. xTrimoPGLM: Unified 100B-Scale Pre-trained Transformer for Deciphering the Language of Protein, July 2023a. Pages: 2023.07.05.547496 Section: New Results.

Chen, C., Zhou, J., Wang, F., Liu, X., and Dou, D. Structure-aware protein self-supervised learning, 2023b.

Chen, T., Gong, C., Diaz, D. J., Chen, X., Wells, J. T., Wang, Z., Ellington, A., Dimakis, A., Klivans, A., et al. Hot-protein: A novel framework for protein thermostability prediction and editing. In *The Eleventh International Conference on Learning Representations*, 2022.

Chen, Z., Badrinarayanan, V., Lee, C.-Y., and Rabinovich, A. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks, 2018.

Corso, G., Stärk, H., Jing, B., Barzilay, R., and Jaakkola, T. DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking, February 2023. arXiv:2210.01776 [physics, q-bio].

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

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., and Rost, B. Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. *CoRR*, abs/2007.06225, 2020.

Fan, H., Wang, Z., Yang, Y., and Kankanhalli, M. Continuous-discrete convolution for geometry-sequence modeling in proteins. In *The Eleventh International Conference on Learning Representations*, 2022.

Gao, Z., Tan, C., Chacón, P., and Li, S. Z. Pifold: Toward effective and efficient protein inverse folding. *arXiv preprint arXiv:2209.12643*, 2022.

Guo, Y., Wu, J., Ma, H., and Huang, J. Self-supervised pre-training for protein embeddings using tertiary structures. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6):6801–6809, 2022.

Hermosilla, P., Schäfer, M., Lang, M., Fackelmann, G., Vázquez, P.-P., Kozlikova, B., Krone, M., Ritschel, T., and Ropinski, T. Intrinsic-extrinsic convolution and pooling for learning on 3d protein structures, 2020.

Hernandez, M., Ghersi, D., and Sanchez, R. Sitehound-web: a server for ligand binding site identification in protein structures. *Nucleic acids research*, 37(suppl_2): W413–W416, 2009.

Hu, J., Li, Y., Zhang, M., Yang, X., Shen, H.-B., and Yu, D.-J. Predicting protein-dna binding residues by weightedly combining sequence-based features and boosting multiple svms. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(6):1389–1398, 2016.

Huang, Y., Wu, L., Lin, H., Zheng, J., Wang, G., and Li, S. Z. Data-Efficient Protein 3D Geometric Pretraining via Refinement of Diffused Protein Structure Decoy, February 2023. arXiv:2302.10888 [cs, q-bio].

Javaloy, A. and Valera, I. Rotograd: Gradient homogenization in multitask learning, 2022.

Jiménez, J., Doerr, S., Martínez-Rosell, G., Rose, A. S., and De Fabritiis, G. Deepsite: protein-binding site predictor using 3d-convolutional neural networks. *Bioinformatics*, 33(19):3036–3042, 2017.

Jing, B., Eismann, S., Soni, P. N., and Dror, R. O. Equivariant Graph Neural Networks for 3D Macromolecular Structure, July 2021. arXiv:2106.03843 [cs, q-bio].

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021. Number: 7873 Publisher: Nature Publishing Group.

Krishna, R., Wang, J., Ahern, W., Sturmfels, P., Venkatesh, P., Kalvet, I., Lee, G. R., Morey-Burrows, F. S., Anishchenko, I., Humphreys, I. R., McHugh, R., Vafeados, D., Li, X., Sutherland, G. A., Hitchcock, A., Hunter, C. N., Baek, M., DiMaio, F., and Baker, D. Generalized Biomolecular Modeling and Design with RoseTTAFold All-Atom, October 2023. Pages: 2023.10.09.561603 Section: New Results.

Krivák, R. and Hoksza, D. P2rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure. *Journal of cheminformatics*, 10:1–12, 2018.

Lam, J. H., Li, Y., Zhu, L., Umarov, R., Jiang, H., Héliou, A., Sheong, F. K., Liu, T., Long, Y., Li, Y., et al. A deep learning framework to predict binding preference of rna constituents on protein surface. *Nature communications*, 10(1):4941, 2019.

Le Guilloux, V., Schmidtke, P., and Tuffery, P. Fpocket: An open source platform for ligand pocket detection. *Bioinformatics*, 10(1):168, 2009. ISSN 1471-2105. doi: 10.1186/1471-2105-10-168.

Lee, Y., Yu, H., Lee, J., Kim, J., and Brain, K. Pre-training sequence, structure, and surface features for comprehensive protein representation learning. 2023.

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.

Liu, R. and Hu, J. Dnabind: A hybrid algorithm for structure-based prediction of dna-binding residues by combining machine learning-and template-based approaches. *PROTEINS: structure, Function, and Bioinformatics*, 81(11):1885–1899, 2013.

Lu, S., Gao, Z., He, D., Zhang, L., and Ke, G. Highly accurate quantum chemical property prediction with unimol+. *arXiv preprint arXiv:2303.16982*, 2023.

Macari, G., Toti, D., and Polticelli, F. Computational methods and tools for binding site recognition between proteins and small molecules: from classical geometrical approaches to modern machine learning strategies. *Journal of computer-aided molecular design*, 33:887–903, 2019.

Miao, Z. and Westhof, E. A large-scale assessment of nucleic acids binding site prediction programs. *PLoS computational biology*, 11(12):e1004639, 2015.

Mirdita, M., Steinegger, M., Breitwieser, F., Söding, J., and Levy Karin, E. Fast and sensitive taxonomic assignment to metagenomic contigs. *Bioinformatics*, 37(18):3029–3031, 2021.

Mitternacht, S. FreeSASA: An open source c library for solvent accessible surface area calculations. *F1000Research*, 5:189, 2016.

Pei, Q., Gao, K., Wu, L., Zhu, J., Xia, Y., Xie, S., Qin, T., He, K., Liu, T.-Y., and Yan, R. Fabind: Fast and accurate protein-ligand binding. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

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

Rose, P. W., Prlić, A., Altunkaya, A., Bi, C., Bradley, A. R., Christie, C. H., Costanzo, L. D., Duarte, J. M., Dutta, S., Feng, Z., et al. The rcsb protein data bank: integrative view of protein, gene and 3d structural information. *Nucleic acids research*, pp. gkw1000, 2016.

Shi, Y., Zheng, S., Ke, G., Shen, Y., You, J., He, J., Luo, S., Liu, C., He, D., and Liu, T.-Y. Benchmarking graphormer on large-scale molecular modeling datasets, 2022. URL https://arxiv.org/abs/2203.04810.

Stärk, H., Ganea, O.-E., Pattanaik, L., Barzilay, R., and Jaakkola, T. EquiBind: Geometric Deep Learning for Drug Binding Structure Prediction, June 2022. arXiv:2202.05146 [cs, q-bio].

Su, H., Liu, M., Sun, S., Peng, Z., and Yang, J. Improving the prediction of protein–nucleic acids binding residues via multiple sequence profiles and the consensus of complementary methods. *Bioinformatics*, 35(6):930–936, 2019.

Tateno, Y., Ikeo, K., Imanishi, T., Watanabe, H., Endo, T., Yamaguchi, Y., Suzuki, Y., Takahashi, K., Tsunoyama, K., Kawai, M., et al. Evolutionary motif and its biological and structural significance. *Journal of molecular evolution*, 44:S38–S43, 1997.

Walia, R. R., Xue, L. C., Wilkins, K., El-Manzalawy, Y., Dobbs, D., and Honavar, V. Rnabindrplus: a predictor that combines machine learning and sequence homology-based methods to improve the reliability of predicted rna-binding residues in proteins. *PloS one*, 9(5):e97725, 2014.

Wang, L., Liu, H., Liu, Y., Kurtin, J., and Ji, S. Learning hierarchical protein representations via complete 3d graph networks. In *International Conference on Learning Representations (ICLR)*, 2023.

Wang, Z., Combs, S. A., Brand, R., Calvo, M. R., Xu, P., Price, G., Golovach, N., Salawu, E. O., Wise, C. J., Ponnapalli, S. P., and Clark, P. M. LM-GVP: an extensible sequence and structure informed deep learning framework for protein property prediction. *Scientific Reports*, 12 (1):6832, April 2022a. Number: 1 Publisher: Nature Publishing Group.

Wang, Z., Zhang, Q., Shuang-Wei, H., Yu, H., Jin, X., Gong, Z., and Chen, H. Multi-level protein structure pre-training via prompt learning. In *The Eleventh International Conference on Learning Representations*, 2022b.

Wu, Q., Peng, Z., Zhang, Y., and Yang, J. Coach-d: improved protein–ligand binding sites prediction with refined ligand-binding poses through molecular docking. *Nucleic acids research*, 46(W1):W438–W442, 2018.

Xia, Y., Xia, C.-Q., Pan, X., and Shen, H.-B. Graphbind: protein structural context embedded rules learned by hierarchical graph neural networks for recognizing nucleic-acid-binding residues. *Nucleic acids research*, 49(9): e51–e51, 2021.

Yifan, W., Doersch, C., Arandjelović, R., Carreira, J., and Zisserman, A. Input-level inductive biases for 3d reconstruction, 2022.

You, Y. and Shen, Y. Cross-modality and self-supervised protein embedding for compound–protein affinity and contact prediction. *Bioinformatics*, 38(Supplement_2): ii68–ii74, 2022.

Zhang, C., Shine, M., Pyle, A. M., and Zhang, Y. Us-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes. *Nature methods*, 19(9): 1109–1115, 2022a.

Zhang, Z., Xu, M., Jamasb, A., Vijil, V., Lozano, A., Das, P., and Tang, J. Protein representation learning by geometric structure pretraining. In *International Conference on Machine Learning*, 2022b.

ZHANG, Z., Lu, Z., Hao, Z., Zitnik, M., and Liu, Q. Full-atom protein pocket design via iterative refinement. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Zhang, Z., Xu, M., Lozano, A., Chenthamarakshan, V., Das, P., and Tang, J. Pre-training protein encoder via siamese sequence-structure diffusion trajectory prediction. In *Annual Conference on Neural Information Processing Systems*, 2023.

Zhou, G., Gao, Z., Ding, Q., Zheng, H., Xu, H., Wei, Z., Zhang, L., and Ke, G. Uni-mol: A universal 3d molecular representation learning framework. In *The Eleventh International Conference on Learning Representations*, 2023.

Zhu, Y.-H., Hu, J., Song, X.-N., and Yu, D.-J. Dnapred: accurate identification of dna-binding sites from protein sequence by ensembled hyperplane-distance-based support vector machines. *Journal of chemical information and modeling*, 59(6):3057–3071, 2019.

A. Small Molecule Dataset.

During the construction of our dataset, after eliminating systems with unsuccessful protein preparation, ligand bond order repair failures, and high similarity records, our dataset comprises a total of 22995 samples for training and 1006 samples for validation.

scPDB seems to be the most popular small molecule binding site dataset currently. Table 8 shows the performance of pre-training baselines and our model on scPDB and our dataset. Our dataset containing 22995 samples is also significantly larger than scPDB which only contains 5564 samples.

Table 7. Small Molecule Binding Site Prediction Result in terms of IoU(%).

DATASET	METHOD	AVE.	B277	DT198	ASTEX85	CHEN251	COACH420
SCPDB	ESM2_150M	18.4	18.8	15.6	17.2	18.5	21.8
	GEARNET	29.6	28.8	26.2	31.2	29.0	32.8
	SIAMDIFF_ATOM	26.6	25.2	21.6	28.1	27.8	30.1
	VABS-NET	**54.8**	**57.1**	**48.8**	**55.1**	**52.8**	**60.0**
OUR DATASET	ESM2_150M	19.5	19.6	16.6	20.5	18.9	22.0
	GEARNE	38.9	39.9	35.8	41.0	36.4	41.3
	SIAMDIFF	37.0	37.7	31.0	40.7	35.3	40.3
	VABS-NET	**58.3**	**60.1**	**52.0**	**58.8**	**56.3**	**64.1**

The IoU results of prediction for the binding site of small molecules can be seen in the following table. We will add these results to our appendix. Results show that IoU drops slightly.

Table 8. Results of small molecules vary with the number of residues.

NUMBER OF RESIDUES	0-200	200-399	400-599	600-799	800-999
AVERAGE IOU	0.663	0.675	0.612	0.501	0.614

B. Vector Encoding.

In most protein pretraining models, distance is frequently utilized to encode protein structure. Theoretically, given KNN=30 (as detailed in our paper) as input, we should be capable of reconstructing a protein chain's configuration (referring to the protein's shape or the 3D coordinates of its atoms within a specific coordinate system.). Multidimensional scaling represents a widely employed algorithm for translating distances between each pair of n objects in a set into a configuration of n points, which are then mapped into an abstract Cartesian space. The computational complexity of this process is $O(N^3)$, where N represents the number of points. Consequently, as the number of atom nodes increases, reconstructing the configuration of a protein becomes more challenging. Notably, the number of atoms is typically eight times greater (based on our pretraining dataset) than the number of residues, which suggests that distance may not be sufficient to encode all-atom configuration. This motivates us to add edge direction features to edges.

C. More Detailed Information About Model Architecture.

Our model, not being SO(3) equivariant, requires additional steps to ensure robustness to rotation. To achieve this, we randomly rotate each sampled chain from the dataset and translate its center to the coordinate origin.

We try to achieve SO(3) equivariance in our model through the following modifications: (1) eliminating the embedding of the edge vector in the global coordinate system, and (2) removing the virtual origin point, as it lacks a relative coordinate system. To pre-train a SO(3) equivalent pretraining model, we also modify the movement head module to make it SO(3) equivalent. We find that making our model SO(3) equivalent can impair the performance of our model as shown in Table 9.

In addition, the ESM model of size 650M is used for pre-training and downstream finetuning.

Table 9. Performance of Vabs-Net with or without SO(3) equivariance.

METHOD	SM	EC
VABS-NET	**64.1**	**0.900**
NO SMPC	63.0	0.876
NO SMPC SO(3)	60.9	0.874

Throughout the fine-tuning process, with the exception of molecular docking, all parameters remained unfrozen. Additionally, in conventional practice, two evolutionary conversation profiles (PSSM and HMM) are traditionally utilized in DNA and RNA binding site prediction (Xia et al., 2021). A more comprehensive description of the model is provided in Table 11. The sizes of DNA and RNA molecules are considerably larger than those of small molecules, necessitating the selection of a k-nearest neighbor (knn) value of 90 to encompass long-range information. In contrast, the pre-training model maintains a knn value of 30. Due to the restricted size of the training set for DNA and RNA binding site prediction, we utilized a model with 256-node dimensions.

Table 10. Model parameters.

METHOD	DNA,RNA BINDING SITE	EC,GO	SMALL MOLECULE BINDING SITE
LAYERS	12	12	12
NODE DIM.	256	768	768
EDGE DIM.	128	128	128
FFN DIM.	512	768	768
BATCH SIZE	32	32	64
KNN	90	30	30
TOTAL STEP	50K	200K	100K
WARMUP STEP	5K	5K	5K
LEARNING RATE	1E-5	5E-5	1E-5
OPTIMIZER	ADAM	ADAM	ADAM

D. More Detailed Information About Downstream Task Experiments.

In all downstream tasks, we introduce random Gaussian noise at a scale of 0.5Å to 20% of the residues in order to prevent overfitting and enhance the robustness of our model.

We compare our model with two important pre-trained models. The first one is the well-known residue-level GearNet pre-trained by multiview contrast learning. The second one is the all-atom-level Siamdiff. We used the checkpoints downloaded from their respective GitHub pages and employed the original configurations to obtain results for small molecules, DNA, and RNA binding sites tasks.

Notably, all of these models are trained using 16 Tesla A100 GPUs.

E. More Detailed Information About Equibind Training.

A faster and more stable version of Equibind in the official repository features 5 layers, 20 attention heads, and 1e-5 weight decay as opposed to the original version using 8 layers, 30 attention heads, and 1e-4 weight decay.

F. Computational Efficiency

All training experiments are conducted with 16 Tesla A100. For the downstream task, we choose small molecule binding site prediction. All inference experiments are conducted with 8 V100 on COACH420 test set(409 samples).

Table 11. Time Spent to Pre-train, Finetune, Inference Our All-atom Version and Residue-level Version vabs-net.

SETTINGS	ATOM PT	RESIDUE PT	ATOM FT	RESIDUE FT	ATOM INFER	RESIDUE INFER
TIME	4D16H	2D2H	8H	3.5H	22S	7S
EPOCHS	190	190	50	50	1	1