Learning Complete Protein Representation by Deep Coupling of Sequence and Structure

Bozhen Hu  
Westlake University  
hubozhen@westlake.edu.cn

Cheng Tan  
Westlake University  
tancheng@westlake.edu.cn

Jun Xia  
Westlake University  
xiajun@westlake.edu.cn

Jiangbin Zheng  
Westlake University  
zhengjiangbin@westlake.edu.cn

Yufei Huang  
Westlake University  
huangyufei@westlake.edu.cn

Lirong Wu  
Westlake University  
wulirong@westlake.edu.cn

Yue Liu  
National University of Defense Technology  
yueliu199907310163.com

Yongjie Xu  
Westlake University  
xuyongjie@westlake.edu.cn

Stan Z. Li*  
Westlake University  
stan.zq.li@westlake.edu.cn

Abstract

Learning effective representations is crucial for understanding proteins and their biological functions. Recent advancements in language models and graph neural networks have enabled protein models to leverage primary or tertiary structure information to learn representations. However, the lack of practical methods to deeply co-model the relationships between protein sequences and structures has led to suboptimal embeddings. In this work, we propose CoupleNet, a network that couples protein sequence and structure to obtain informative protein representations. CoupleNet incorporates multiple levels of features in proteins, including the residue identities and positions for sequences, as well as geometric representations for tertiary structures. We construct two types of graphs to model the extracted sequential features and structural geometries, achieving completeness on these graphs, respectively, and perform convolution on nodes and edges simultaneously to obtain superior embeddings. Experimental results on a range of tasks, such as protein fold classification and function prediction, demonstrate that our proposed model outperforms the state-of-the-art methods by large margins.

1 Introduction

Proteins are the fundamental building blocks of life and play essential roles in a diversity of applications, from therapeutics to materials. They are composed of 20 different basic amino acids, which are lined by peptide bonds and form a sequence. The one-dimensional (1D) sequence of a
Figure 1: Illustration of the protein sequence and structure. 1) The primary structure comprises $n$ amino acids. 2) The tertiary structure with atom arrangement in Euclidean space is presented, where each atom has a specific 3D coordinate. Amino acids have fixed backbone atoms ($\text{C}_\alpha, \text{C}, \text{N}, \text{O}$) and side-chain atoms that vary depending on the residue types. GLU: Glutamic acid. Complete geometries can be obtained based on these coordinates. The sequence and structure provide different information types and data categories.

protein determines its structure, which in turn determines its biochemical function \[40\]. Due to recent progress in protein sequencing \[34\], massive numbers of protein sequences are now available. For example, the UniProt \[3\] database contains over 200 million protein sequences with annotations, e.g., gene ontology (GO) terms, similar proteins, family and domains. Notably, the development of large-scale language models (LMs) in natural language processing has substantially benefited protein research owing to similarities between human language and protein sequences \[16, 27\]. For instance, models like ProtTrans \[14\] and ESM-series \[39, 33\] in learning protein representations have proven successful utility of pre-training protein LMs with self-supervision to process protein sequences.

Thanks to the recent significant progress made by AlphaFold2 \[30\] in three-dimensional (3D) structure prediction, a large number of protein structures from their sequence data are now made available. The latest release of AlphaFold protein structure database \[43\] provides broad coverage of UniProt \[3\]. Recently proposed structure-based protein encoders become to utilize geometric features \[25, 24, 53\], e.g., ProNet \[47\] learns representations of proteins with 3D structures at different levels, like the amino acid, backbone or all-atom levels. There also exists a group of methods that build graph neural networks and LMs (LSTMs or attention models) to process sequence and structure \[53, 50, 19\], for example, GearNet \[53\] encodes sequential and spatial features by alternating node and edge message passing on protein residue graphs.

The 1D sequence and 3D structure of a protein provide different types of information, in detail, as shown in Figure 1 compared with the 1D sequential order and amino acids in peptide chains, the tertiary structure provides 3D coordinates of each atom in protein residues, which allow them to perform precise functions. Although a protein’s sequence determines its structure, various works have demonstrated the effectiveness of learning from either sequence or structure \[43, 25\]. However, rich constraints between the sequence and structure of a protein, which may be critical for protein tasks \[4\], have yet to be fully explored. Most protein sequence-structure modeling methods cannot deeply integrate the information behind sequence and structure for the reason that they tend to fuse representations together, extracted from sequence and structure encoders, respectively, by message passing mechanism \[8\] or by simple concatenation operations.

In this work, we aim to learn protein representations by deeply coupling the protein sequences and structures. Considering the relative positions of residues in the sequence and the spatial arrangement of atoms in the Euclidean space, the proposed CoupleNet constructs two categories of graphs for them, respectively. The complete representations are obtained at the base and backbone levels on the two graphs, which are used as node and edge features to learn the final graph-level representations. Rather than concatenating sequence and structure representations, we take advantage of graph convolutions, performing node and edge convolutions simultaneously. The contributions of this paper are threefold:

- We propose a novel two-graph-based approach for representing the sequence and the 3D geometric structure of a protein, which is an effective way to guarantee completeness.
• We propose CoupleNet, a model that performs convolutions on nodes and edges of graphs to effectively integrate protein sequence and structure. This can better model the node-edge relationships and utilize the intrinsic associations between sequences and structures.

• Practically, the proposed model is verified by obtaining new state-of-the-art experimental results compared with current mainstream protein representation learning methods on a range of tasks, including protein fold classification, enzyme reaction classification, GO term prediction, domain prediction, and enzyme commission number prediction.

2 Related Work

Protein Representation Learning Protein representation learning has become an active and promising direction in biology, which is essential to various downstream tasks in protein science. Because of the different levels of protein structures, existing methods mainly fall into three categories: protein LMs for sequences, structure models for geometry, and hybrid methods for both of them. As proteins are sequences of amino acids, considering their similarities with human languages, UniRep [1], UDSMProt [22] and SeqVec [23] use LSTM or its variants to learn sequence representations and long-range dependencies. TAPE [37] benchmarks a group of protein models, e.g., 1D CNN, LSTM, and Transformer by various tasks. Elmagar et al. [14] have trained six successful transformer variants on billions of amino acid sequences, like ProtBert, and ProtT5. Similarly, ESM-series [39, 38, 33] employs a transformer architecture and a masked language modeling strategy to train robust representations based on large-scale databases. Besides the protein sequence, as we have stated before, the 3D geometric structure is vital to enhance protein representations. Most methods commonly seek to encode the spatial information of protein structures by convolutional neural networks (CNNs) [11], or graph neural networks [19, 2, 29]. For instance, SPROF [7] employs distance maps to predict protein sequence profiles, and IEConv [25] introduces a convolution operator to capture all relevant structural levels of a protein. GVP-GNN [29] designs the geometric vector perceptrons (GVP) for learning both scalar and vector features in an equivariant and invariant manner, Guo et al. [21] adopt SE(3)-invariant features as the model inputs and reconstruct gradients over 3D coordinates to avoid the usage of complicated SE(3)-equivariant models. ProNet [47] learns hierarchical protein representations at multiple tertiary structure levels of granularity. Moreover, CDConv [15] proposes continuous-discrete convolution using irregular and regular approaches to model the geometry and sequence structures. Some protein learning methods model the multi-level of structures at the same time [53, 6, 15], except for the primary structure and the tertiary structure, the second refers to the 3D form of local segments of proteins (e.g., α-helix, β-strand), the quaternary is a protein multimer comprising multiple polypeptides, for example, PromtProtein [48] adopts a prompt-guided multi-task learning strategy for different protein structures with specific pre-training tasks. While previous works have attempted to combine protein sequence and structure, we focus on profoundly integrating them by specifically designing two types of graphs respectively and conducting convolutions simultaneously to learn protein representations.

Complete Message Passing Mechanism ComENet [46] proposes rotation angles and spherical coordinates to fulfil the global completeness of 3D information on molecular graphs. By incorporating these designed geometric representations into the message passing scheme [18], the complete representation for a whole 3D graph is eventually yielded [47]. Unlike these methods, we couple sequence and structure via corresponding graphs and different geometric representations to obtain completeness representations.

3 Method

3.1 Preliminaries

Notations We represent a 3D graph as \( G = (\mathcal{V}, \mathcal{E}, \mathcal{P}) \), where \( \mathcal{V} = \{v_i\}_{i=1,...,n} \) and \( \mathcal{E} = \{e_{ij}\}_{i,j=1,...,n} \) denote the vertex and edge sets with \( n \) nodes in total, respectively, and \( \mathcal{P} = \{P_i\}_{i=1,...,n} \) is the set of position matrices, where \( P_i \in \mathbb{R}^{k_i \times 3} \) represents the position matrix for node \( v_i \). We treat each amino acid as a graph node for a protein, then \( k_i \) depends on the number of atoms in the \( i \)-th amino acid. The node feature matrix is \( X = [x_i]_{i=1,...,n} \), where \( x_i \in \mathbb{R}^{d_i} \) is
the feature vector of node \( v_i \). The edge feature matrix is \( E = [e_{ij}]_{i,j=1,...,n} \), where \( e_{ij} \in \mathbb{R}^{d_e} \) is the feature vector of edge \( e_{ij} \). \( d_v \) and \( d_e \) denote the dimensions of feature vectors \( x_i \) and \( e_{ij} \).

**Invariance and Equivariance** We consider affine transformations that preserve the distance between any two points, i.e., the isometric group \( SE(3) \) in the Euclidean space. This is called the symmetry group, and it turns out that \( SE(3) \) is the special Euclidean group that includes 3D translations and the 3D rotation group \( SO(3) \) [17] [12]. The matrix form of \( SE(3) \) is provided in Appendix A.1.

Given the function \( f : \mathbb{R}^m \rightarrow \mathbb{R}^{m'} \), assuming the given symmetry group \( G \) acts on \( \mathbb{R}^m \) and \( \mathbb{R}^{m'} \), then \( f \) is \( G \)-equivariant if,

\[
f(T_g x) = S_g f(x), \quad \forall x \in \mathbb{R}^m, g \in G
\]

where \( T_g \) and \( S_g \) are the transformations. For the \( SE(3) \) group, when \( m' = 1 \), the output of \( f \) is a scalar, we have

\[
f(T_g x) = f(x), \quad \forall x \in \mathbb{R}^m, g \in G
\]

thus \( f \) is \( SE(3) \)-invariant.

**Complete Geometric Representations** A geometric transformation \( \mathcal{F}(\cdot) \) is complete if two 3D graphs \( G^1 = (V, E, \mathcal{P}^1) \) and \( G^2 = (V, E, \mathcal{P}^2) \), there exists \( T_g \in SE(3) \) such that the representations

\[
\mathcal{F}(G^1) = \mathcal{F}(G^2) \iff P_i^1 = T_g(P_i^2), \quad \text{for } i = 1, \ldots, n
\]

The operation \( T_g \) would not change the 3D conformation of a 3D graph [46]. Positions can generate geometric representations, which can also be recovered from them.

**Message Passing Paradigm** Message passing mechanism is mainly applied in graph convolutional networks (GCNs) [32], which follows an iterative scheme of updating node representations based on the feature aggregation from nearby nodes.

\[
\begin{align*}
    h_i^{(0)} &= BN(FC(x_i)), \\
    u_i^{(l)} &= f_{Agg}^{(l)}(h_j^{(l-1)} | v_j \in \mathcal{N}(v_i)), \\
    h_i^{(l)} &= f_{Update}^{(l)}(h_j^{(l-1)}, u_i^{(l)})
\end{align*}
\]

where \( FC(\cdot) \) and \( BN(\cdot) \) mean the linear transformation and batch normalization respectively. \( \mathcal{N}(v_i) \) denotes the neighbours of node \( v_i \). \( f_{Agg}^{(l)} \) and \( f_{Update}^{(l)} \) are aggregation and transformation functions at the \( l \)-th layer, which are permutation invariant and equivariant of node representations.

### 3.2 Sequence-Structure Graph Construction

Specifically, we represent each amino acid as a node, considering the residue types and their positions \( i = 1, 2, \ldots, n \) (See Figure 1) in the sequence, we define the sequential graph primarily on the sequence, if \( |i - j| < l \), the edge \( e_{ij} \) exists, where \( l \) is a hyper-parameter. Besides the sequential graph, we predefined a radius \( r \), and build the radius graph, and there exists an edge between node \( v_i \) and \( v_j \) if \( ||P_i, C_\alpha - P_j, C_\alpha|| < r \), where \( P_i, C_\alpha \) denotes the 3D position of \( C_\alpha \) in the \( i \)-th residue.

Firstly, we design a base approach called CoupleNet$_{aa}$ that only uses the \( C_\alpha \) positions of the structures. Inspired by Ingraham et al. [28], we construct a local coordinate system (LCS) for each residue, as shown in Figure 2.

\[
Q_i = [b_i, n_i, b_i \times n_i]
\]

![Figure 2: The local coordinate system.](image-url)
where \( u_i = \frac{P_i,Ca - P_{i-1},Ca}{\|P_i,Ca - P_{i-1},Ca\|}, \quad b_i = \frac{u_i \times u_{i+1}}{\|u_i \times u_{i+1}\|}, \quad n_i = \frac{u_i}{\|u_i\|} \). Then we can get the geometric representations at the base level of a protein 3D graph,

\[
F(G)_{ij,aa} = (\|P_i,Ca - P_j,Ca\|, Q_i^T, \frac{P_i,Ca - P_j,Ca}{\|P_i,Ca - P_j,Ca\|}, Q_j^T \cdot Q_j)
\]

where \( \cdot \) is the matrix multiplication, this implementation is SE(3)-equivariant and obtains complete representations at the base level; as if we have \( Q_i \), the LCS \( Q_j \) can be easily obtained by \( F(G)_{ij,aa} \).

For a node \( v_i \), the node features \( x_{i,aa} \) in the baseline approach is the concatenation of the one-hot embeddings of the amino acid types and the physicochemical properties of each residue, namely, a steric parameter, hydrophobicity, volume, polarizability, isoelectric point, helix probability and sheet probability [11][22], which provide quantitative insights into the biochemical nature of each amino acid. And \( F(G)_{ij,aa} \) is set as edge features for CoupleNet.

Secondly, we consider all backbone atoms \( C_\alpha, C, N, O \) in CoupleNet. In detail, the peptide bond exhibits partial double-bond character due to resonance [20], indicating that the three non-hydrogen atoms comprising the bond (the carbonyl oxygen, carbonyl carbon, and amide nitrogen) are coplanar, as shown in Figure 5. There is some rotation about the connection. The \( \Omega_i - C_\alpha_i \) and \( C_\alpha_i - C_i \) bonds are the two bonds in the basic repeating unit of the polypeptide backbone. These single bonds allow unrestricted rotation until sterically restricted by side chains [35][45]. Since the coordinates of \( C_\alpha \) can be obtained as we have the complete representations at the base level, the coordinates of other backbone atoms based on these rigid bond lengths and angles are able to be determined with the remaining degree of the backbone torsion angles \( \Phi_i, \Psi_i, \Omega_i \). The omega torsion angle around the \( C - N \) peptide bond is typically restricted to nearly 180° (trans) but can approach 0° (cis) in rare instances. Other than the bond lengths and angles presented in Figure 3, all the H bond lengths measure approximately 1 Å.

For the sequential graph, we compute the sine and cosine values of \( \Phi_i, \Psi_i, \Omega_i \) for each amino acid \( i \), and use them as another part of nodes features for node \( v_i \).

\[
x_i = x_{i,aa} \| (\sin \wedge \cos) (\Phi_i, \Psi_i, \Omega_i) \)
\]

where \( \| \) denotes concatenation. There is no isolated node for the designed graph, which means the backbone atoms can be determined one by one along the polypeptide chain based on the positions of \( C_\alpha \) and these three backbone dihedral angles. Therefore, the existing presentations \( \|F(G)_{ij,aa}\|_{i,j=1,\ldots,n} \) and \( |x_i|_{i=1,\ldots,n} \) are complete at the backbone level for the sequential graph.

For the radius graph, we want to get the positions of backbone atoms in any two amino acids \( i \) and \( j \). Inspired by trRosetta [52], the relative rotation and distance are computed including the distance \( (d_{ij},C_\alpha) \), three dihedral angles \( (\omega_{ij}, \theta_{ij}, \theta_{ij}) \) and two planar angles \( (\phi_{ij}, \psi_{ij}) \), as shown in Figure 4 where \( d_{ij,C_\alpha} = d_{ij,C_\alpha}, \omega_{ij} = \omega_{ij}, \text{but } \theta \text{ and } \phi \text{ values depend on the order of residues. These interresidue geometries define the relative locations of the backbone atoms of two residues in all their details [52], because the torsion angles of \( N_j - C_\alpha_i \) and \( C_\alpha_i - C_j \) do not influence their positions. Therefore, these six geometries are complete for amino acids at the backbone level for the
radius graph. The graph edges contain the relative spatial information between any two neighboring amino acids $e_{ij} = F(G)_{ij,aa} \parallel F(G)_{ij,bb}$, where

$$F(G)_{ij,bb} = (d_{ij,C\alpha}, (\sin \wedge \cos)(\omega_{ij}, \theta_{ij}, \varphi_{ij}))$$

(8)

The designed node and edge features, $x_i$ and $e_{ij}$, for the sequential and radius graphs, provide a new perspective to represent protein sequences and structures. Such integration can bring better performance for the following graph-based learning tasks.

3.3 Sequence-Structure Graph Convolution

Inspired by the message passing paradigm and continuous-discrete convolution [15], sequences and structures are encoded successfully together by convolutions. To deeply couple sequences and structures of proteins and encode them jointly, we employ convolution to embed them simultaneously, exploring their relationships to generate comprehensive and effective embeddings. Different from previous works, we innovatively construct two categories of graphs for sequence and structure and design various sequential and structural representations to achieve completeness on them at the base and backbone levels. We then convolve node and edge features with the help of the message passing mechanism.

In order to implement convolution on nodes and edges simultaneously between sequence and structure, we set $\varepsilon_{ij}$ to exist if the following conditions are satisfied

$$\|i - j\| < l \quad \text{and} \quad \|P_{i,C\alpha} - P_{j,C\alpha}\| < r$$

(9)

The existing node and edge feature matrices ($X, E$) are complete representations of a protein 3D graph to reconstruct its backbone atom positions. Compared with the equation Eq. 4, the proposed CoupleNet first apply a FC($\cdot$) layer and a BN($\cdot$) layer to the node features to obtain the initial encoded representation. Then the $f_{Agg}^{(l)}$ is applied to gather neighboring features of nodes and edges by convolution, where $\sigma(\cdot)$ is the activation function, and $W$ is the learnable convolutional kernel matrix. We use the dropout and add a residual connection from the previous layer as $f_{Update}^{(l)}$. For the consideration that the spatial arrangement and tight positioning of specific amino acids, which may be spaced widely apart on the linear polypeptide, are necessary for proteins to operate as intended [10], $l$ is set to be a relatively large number, see Appendix A.2 for details.

$$h_{i}^{(0)} = \text{BN}(\text{FC}(x_i)),$$

$$u_{i}^{(l)} = \sigma(\text{BN}(\sum_{v_j \in N(v_i)} We_{ij}h_{j}^{(l-1)})),$$

$$h_{i}^{(l)} = h_{i}^{(l)} + \text{Dropout}(u_{i}^{(l)})$$

(10)

3.4 Model Architecture

Building upon the sequence-structure graph convolution, we build the CoupleNet, as shown in Figure 5. The inputs to the graph are the calculated sequential and structural representations ($X, E$). Following the existing protein graph models [15, 25, 47], our CoupleNet employs graph pooling layers to obtain deeply encoded, graph-level representations. After pooling, due to the decrease in nodes, we increase the predefined radius $r$ to include more neighbors. The message passing mechanism only executes on nodes for the consideration of reducing model complexity. Another reason is that representations of sequences and structures have already been coupled by equation Eq. 4. A detailed description of the model architecture is provided in Appendix A.2.
4 Experiments

4.1 Datasets and Settings

The models are trained with the Adam optimizer [31] using the PyTorch and PyTorch Geometric libraries. Detailed descriptions of the datasets and experimental settings are provided in Appendix A.3. Following the tasks in IECovn [25], GearNet [53] and CDConv [15], here, we evaluate the CoupleNet on four protein tasks: protein fold classification, enzyme reaction classification, GO term prediction and enzyme commission (EC) number prediction.

Fold Classification Protein fold is to predict the fold class label given a protein, which is crucial for understanding how protein structure and protein evolution interact [26]. In total, this dataset contains 16,712 proteins with 1,195 fold classes. There are three test sets available, Fold: Training excludes proteins from the same superfamily. Superfamily: Training does not include proteins from the same family. Family: Proteins from the same family are included in the training.

Enzyme Reaction Classification Reaction categorization aims to predict a protein’s class of enzyme-catalyzed reactions, according to all four levels of the EC number [49, 36]. Following the setting in [25], this dataset has 37,248 proteins from 384 four-level EC numbers [5].

GO Term Prediction The goal of GO term prediction is to foretell whether a protein is related to a certain GO term. Following [19], these proteins are organized into three ontologies: molecular function (MF), biological process (BP), and cellular component (CC), which are hierarchically connected, functional classes. MF describes activities that occur at the molecular level, BP represents the larger processes, and CC describes the parts of a cell or its extracellular environment [3].

EC Number Prediction This task seeks to predict the 538 EC numbers from the third level and fourth levels of different proteins [19], which describe their catalysis of biochemical reactions.

4.2 Baselines

We compare our proposed method with existing protein representation learning methods, which are classified into three categories based on their inputs, which could be a sequence (amino acid sequence), 3D structure or both sequence and structure. 1) Sequence-based encoders, including CNN [41], ResNet [37], LSTM [57] and Transformer [37]. 2) Structure-based methods (GCN [32], GAT [44], 3DCNN_MQA [11], IECovn (atom level) [25]). 3) Sequence-structure based models, e.g., GVP [29], ProNet [47], GearNet [53], CDConv [15], etc. GearNet-IEConv and GearNetEdge-IEConv [53] add the IEConv layer based on GearNet, which is found important in fold classification.

4.3 Results of Fold and Reaction Classification.

Table 1 provides the comparisons on the fold and enzyme reaction classification. The results are reported in terms of accuracy (%) for these two tasks. From this table, we can see that the proposed model CoupleNet achieves the best performance across all four test sets on the fold and enzyme reaction classification compared with recent state-of-the-art methods. Especially on the Fold and SuperFamily test sets, CoupleNet improves the results by about 4%, showing that CoupleNet is proficient at learning the mapping between protein sequences, structures and functions. Moreover,
Table 1: Accuracy (%) on fold classification and enzyme reaction classification. [*] means the results are taken from [15]. The best and suboptimal results are shown in bold and underline.

<table>
<thead>
<tr>
<th>Input</th>
<th>Method</th>
<th>Fold Classification</th>
<th>Enzyme Reaction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Fold</td>
<td>SuperFamily</td>
</tr>
<tr>
<td>Sequence</td>
<td>CNN [41]*</td>
<td>11.3</td>
<td>13.4</td>
</tr>
<tr>
<td></td>
<td>ResNet [37]*</td>
<td>10.1</td>
<td>7.21</td>
</tr>
<tr>
<td></td>
<td>LSTM [37]*</td>
<td>6.41</td>
<td>4.33</td>
</tr>
<tr>
<td></td>
<td>Transformer [37]*</td>
<td>9.22</td>
<td>8.81</td>
</tr>
<tr>
<td>Structure</td>
<td>GCN [32]*</td>
<td>16.8</td>
<td>21.3</td>
</tr>
<tr>
<td></td>
<td>GAT [44]*</td>
<td>12.4</td>
<td>16.5</td>
</tr>
<tr>
<td></td>
<td>3DCNN_MQA [11]*</td>
<td>31.6</td>
<td>45.4</td>
</tr>
<tr>
<td></td>
<td>IEConv (atom level)  [25]*</td>
<td>45.0</td>
<td>69.7</td>
</tr>
<tr>
<td>Sequence-Structure</td>
<td>GraphQA [2]*</td>
<td>23.7</td>
<td>32.5</td>
</tr>
<tr>
<td></td>
<td>GVP [29]*</td>
<td>16.0</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>ProNet-Amino Acid [47]</td>
<td>51.5</td>
<td>69.9</td>
</tr>
<tr>
<td></td>
<td>ProNet-Backbone [47]</td>
<td>52.7</td>
<td>70.3</td>
</tr>
<tr>
<td></td>
<td>ProNet-All-Atom [47]</td>
<td>52.1</td>
<td>69.0</td>
</tr>
<tr>
<td></td>
<td>IEConv (residue level) [25]*</td>
<td>47.6</td>
<td>70.2</td>
</tr>
<tr>
<td></td>
<td>GearNet [53]</td>
<td>28.4</td>
<td>42.6</td>
</tr>
<tr>
<td></td>
<td>GearNet-IEConv [53]</td>
<td>42.3</td>
<td>64.1</td>
</tr>
<tr>
<td></td>
<td>GearNet-Edge [53]</td>
<td>44.0</td>
<td>66.7</td>
</tr>
<tr>
<td></td>
<td>GearNet-Edge-IEConv [53]</td>
<td>48.3</td>
<td>70.3</td>
</tr>
<tr>
<td></td>
<td>CDConv [15]</td>
<td>56.7</td>
<td>77.7</td>
</tr>
<tr>
<td></td>
<td>CoupleNet (Proposed)</td>
<td>60.6</td>
<td>82.1</td>
</tr>
</tbody>
</table>

CDConv [15] ranks second among these methods, both CDConv and our method are implemented by sequence-structure convolution. This phenomenon illustrates that deeply coupling sequences and structures of proteins is conducive to learning better protein embeddings. And our proposed CoupleNet model utilizes complete geometric representations and the specially designed message passing mechanism, achieving new state-of-the-art results.

4.4 Results of GO Term and EC Prediction

We follow the split method in [19, 53] to guarantee that the test set only comprises PDB chains with sequence identity no higher than 95% to the training set for GO term and EC number prediction. Table 2 compares different protein modeling methods on GO term prediction and EC number prediction. The results are reported in terms of $F_{\text{max}}$, which considers both precision and recall for evaluation, the equation of $F_{\text{max}}$ is provided in Appendix A.4. The proposed model, CoupleNet yields the highest $F_{\text{max}}$ across these four test sets of two tasks, outperforming other state-of-the-art models. This indicates CoupleNet can effectively predict the functions, locations, and enzymatic activities of proteins. These results once again illustrate the importance of deeply coupled sequences and structures. The improvements of CoupleNet over the suboptimal CDConv [15] model indicate the advanced modeling power of CoupleNet.

We employ different cutoff splits following [19, 15], which means that the proteins in the test set are divided into groups that have, respectively, 30%, 40%, 50%, 70%, and 95% similarity to the training set for GO term and EC number prediction, as shown in Figure 6 and Appendix A.5.
Table 2: \( F_{\text{max}} \) on GO term and EC number prediction. [∗] means the results are taken from [15]. The best and suboptimal results are shown in bold and underline.

<table>
<thead>
<tr>
<th>Category</th>
<th>Method</th>
<th>GO-BP</th>
<th>GO-MF</th>
<th>GO-CC</th>
<th>EC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence</td>
<td>CNN [41]</td>
<td>0.244</td>
<td>0.354</td>
<td>0.287</td>
<td>0.545</td>
</tr>
<tr>
<td></td>
<td>ResNet [37]</td>
<td>0.280</td>
<td>0.405</td>
<td>0.304</td>
<td>0.605</td>
</tr>
<tr>
<td></td>
<td>LSTM [37]</td>
<td>0.225</td>
<td>0.321</td>
<td>0.283</td>
<td>0.425</td>
</tr>
<tr>
<td></td>
<td>Transformer [37]</td>
<td>0.264</td>
<td>0.211</td>
<td>0.405</td>
<td>0.238</td>
</tr>
<tr>
<td>Structure</td>
<td>GCN [32]</td>
<td>0.252</td>
<td>0.195</td>
<td>0.329</td>
<td>0.320</td>
</tr>
<tr>
<td></td>
<td>GAT [44]</td>
<td>0.284</td>
<td>0.317</td>
<td>0.385</td>
<td>0.368</td>
</tr>
<tr>
<td></td>
<td>3DCNN_MQA [11]</td>
<td>0.240</td>
<td>0.147</td>
<td>0.305</td>
<td>0.077</td>
</tr>
<tr>
<td>Sequence-Structure</td>
<td>GraphQA [2]</td>
<td>0.308</td>
<td>0.329</td>
<td>0.413</td>
<td>0.509</td>
</tr>
<tr>
<td></td>
<td>GVP [29]</td>
<td>0.326</td>
<td>0.426</td>
<td>0.420</td>
<td>0.489</td>
</tr>
<tr>
<td></td>
<td>IEConv (residue level) [25]</td>
<td>0.421</td>
<td>0.624</td>
<td>0.431</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>GearNet [53]</td>
<td>0.356</td>
<td>0.503</td>
<td>0.414</td>
<td>0.730</td>
</tr>
<tr>
<td></td>
<td>GearNet-IEConv [53]</td>
<td>0.381</td>
<td>0.563</td>
<td>0.422</td>
<td>0.800</td>
</tr>
<tr>
<td></td>
<td>GearNet-Edge [53]</td>
<td>0.403</td>
<td>0.580</td>
<td>0.450</td>
<td>0.810</td>
</tr>
<tr>
<td></td>
<td>GearNet-Edge-IEConv [53]</td>
<td>0.400</td>
<td>0.581</td>
<td>0.430</td>
<td>0.810</td>
</tr>
<tr>
<td></td>
<td>CDConv [15]</td>
<td>0.453</td>
<td>0.654</td>
<td>0.479</td>
<td>0.820</td>
</tr>
<tr>
<td></td>
<td>CoupleNet (Proposed)</td>
<td><strong>0.467</strong></td>
<td><strong>0.669</strong></td>
<td><strong>0.494</strong></td>
<td><strong>0.866</strong></td>
</tr>
</tbody>
</table>

Table 3: Ablation of our proposed method

<table>
<thead>
<tr>
<th>Method</th>
<th>Fold Classification</th>
<th>Enzyme Reaction</th>
<th>GO</th>
<th>EC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fold</td>
<td>Superfamily</td>
<td>Family</td>
<td>Reaction</td>
</tr>
<tr>
<td>CoupleNet</td>
<td>60.6</td>
<td>82.1</td>
<td>99.7</td>
<td>89.0</td>
</tr>
<tr>
<td>CoupleNet_{aa}</td>
<td>57.8</td>
<td>78.7</td>
<td>99.6</td>
<td>88.6</td>
</tr>
<tr>
<td>w/o ( \Phi, \Psi, \Omega )</td>
<td>60.3</td>
<td>81.3</td>
<td>99.6</td>
<td>88.7</td>
</tr>
<tr>
<td>w/o ( d, \omega, \theta, \varphi )</td>
<td>60.4</td>
<td>81.5</td>
<td>99.7</td>
<td>88.9</td>
</tr>
</tbody>
</table>

Figure 6, we can see that our proposed model CoupleNet achieves the highest \( F_{\text{max}} \) scores across all cutoffs, especially when the cutoffs are at 30% to 50%. There is a larger margin compared with GearNet, GearNet-Edge [53] and CDConv [15]. This demonstrates that CoupleNet, which utilizes complete geometric representations, is more robust, especially when there is a low similarity between the training and test sets.

### 4.5 Ablation Study

Table 3 presents an ablation study of the proposed CoupleNet model on the four protein tasks. We examined the impact of removing the backbone torsion angles (w/o \( \Phi, \Psi, \Omega \)) and removing the interresidue geometric structure representations (w/o \( d_{C\beta}, \omega, \theta, \varphi \)). The former is designed for the sequential graph, and the latter is for the radius graph to achieve completeness at the protein backbone level. However, we combine the two types of graphs together to enhance the relationships between sequence and structure. From Table 3, we can also find that these complete geometries provide complementary information to amino acid position features, with one of their removals leading to minor performance drops for the reason that they both provide complete geometries from different perspectives. Removing \( \Phi, \Psi, \Omega \) causes larger performance degradation compared with removing \( d_{C\beta}, \omega, \theta, \varphi \). Such comparisons indicate that the backbone dihedral angles may have more effects on learning protein representations in these experimental settings. CoupleNet_{aa} is a base model that only adopts the \( C\alpha \) position of protein structures. Compared with CoupleNet_{aa}, CoupleNet achieves significant improvements on the four tasks, demonstrating the importance of complete structural representations at the backbone level in learning protein embeddings.
5 Conclusions and Limitations

In this work, we propose CoupleNet, a novel protein representation learning method that deeply fuses protein sequences and multi-level structures by conducting convolution on graph nodes and edges simultaneously. We design the sequential graph and the radius graph, achieving completeness on them at different protein structure levels. Our approach achieves new state-of-the-art results on the protein tasks, which demonstrates the superiority of the proposed method. A limitation is that the detailed inter-relationships between sequence and structures remain to be explored and uncovered. We leave such research for future work.

While our model can enable advanced protein analyses and provide effective representations, there may exist broader impacts and harmful activities. The representations could potentially be misused, e.g., for designing harmful molecules or proteins.
References


A Appendix

A.1 SE(3)

The collection of $4 \times 4$ real matrices of the SE(3) is shown as:

$$
\begin{bmatrix}
R & t \\
0 & 1
\end{bmatrix} =
\begin{bmatrix}
r_{11} & r_{12} & r_{13} & t_1 \\
r_{21} & r_{22} & r_{23} & t_2 \\
r_{31} & r_{32} & r_{33} & t_3 \\
0 & 0 & 0 & 1
\end{bmatrix},
$$

(11)

where $R \in SO(3)$ and $t \in \mathbb{R}^3$, SO(3) is the 3D rotation group. $R$ satisfying $R^T R = I$ and $\det(R) = 1$.

A.2 Details of Model Architecture

As stated in Sec. 3.3 on sequence-structure graph convolution, $l$ is set to be a constant number 11. We increase the predefined radius $r$ to $2r$ after one pooling layer, and the number of feature channels for node embeddings is also doubled. We use a Leaky ReLU function [13] as the activation $\sigma(\cdot)$ in the message passing layers.

We design the sequential and radius graph instead of the $k$-nearest neighbour graph because a constant $k$ make some neighbor nodes far away from the center node. As shown in Figure 7 the distances of a group of neighbor nodes ($\|P_{i,C_{\alpha}} - P_{j,C_{\beta}}\|$) are larger than 20 Å, which cannot be seen as contacts [9]. Therefore, the radius is initially set to 4, enlarging to 16 in deeper layers. There are four message passing and pooling layers. In this condition, when the number of nodes decreases, $l$ is constant, $r$ increases, neighbours of center nodes gradually cover more distant nodes.

![Figure 7: The histogram of distance statistics of $k = 30$ nearest neighbor nodes of a protein dataset (CATH [29]). The horizontal axis denotes the distance in terms of exponents of 10, and the vertical axis represents the number of neighbor nodes with this distance.](image-url)
Table 4: Dataset statistics. # X means the number of X.

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Train</th>
<th># Validation</th>
<th># Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enzyme Commission</td>
<td>15,550</td>
<td>1,729</td>
<td>1,919</td>
</tr>
<tr>
<td>Gene Ontology</td>
<td>29,898</td>
<td>3,322</td>
<td>3,415</td>
</tr>
<tr>
<td>Fold Classification - Fold</td>
<td>12,312</td>
<td>736</td>
<td>718</td>
</tr>
<tr>
<td>Fold Classification - Superfamily</td>
<td>12,312</td>
<td>736</td>
<td>1,254</td>
</tr>
<tr>
<td>Fold Classification - Family</td>
<td>12,312</td>
<td>736</td>
<td>1,272</td>
</tr>
<tr>
<td>Reaction Classification</td>
<td>29,215</td>
<td>2,562</td>
<td>5,651</td>
</tr>
</tbody>
</table>

A.3 Details of Datasets and Training Setup

For all datasets, we use a data augmentation strategy by adding noise for the training set to increase the variability of data. For example, we update the position of $C_{\alpha_i}, P_i, C_{\alpha_i}$ as:

$$P_{i,C_{\alpha_i}} \leftarrow P_{i,C_{\alpha_i}} + N(\mu_N, \sigma^2_N)$$ (12)

where $\mu_N, \sigma^2_N$ are the mean (expectation) and variance of the normal distribution $N$, which are set to 0 and 0.1 in experiments. Dataset statistics [53] of our four downstream tasks are summarized in Table 4.

Settings The proposed models are conducted on a single NVIDIA-SMI A100 GPU, through PyTorch 1.13+cu117 and PyTorch Geometric 2.3.1 with CUDA 11.2. The number of the initial feature channels is 256. The learning rate is set to 0.001. More details about implementation is shown in Table 5.

Table 5: More details of training setup

<table>
<thead>
<tr>
<th>Hyper-parameter</th>
<th>Fold</th>
<th>Enzyme Reaction</th>
<th>GO</th>
<th>EC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch size</td>
<td>4</td>
<td>4</td>
<td>24</td>
<td>64</td>
</tr>
<tr>
<td>Epoch</td>
<td>400</td>
<td>400</td>
<td>500</td>
<td>500</td>
</tr>
</tbody>
</table>

A.4 Evaluation Metric $F_{\text{max}}$

$F_{\text{max}}$ is calculated by first determining the precision and recall for each protein, then averaging these results over all proteins [53, 15, 19]. $p_{ij}$ is the prediction probability for the $j$-th class of the $i$-th protein, given the decision threshold $t \in [0, 1]$, the precision and call are given as:

$$\text{precision}_i(t) = \frac{\sum_j \mathbb{I}[(p_{ij}^t \geq t) \cap b_i^j]}{\sum_j \mathbb{I}[p_{ij}^t \geq t]} , \quad \text{recall}_i(t) = \frac{\sum_j \mathbb{I}[(p_{ij}^t \geq t) \cap b_i^j]}{\sum_j b_i^j}$$

where $b_i^j \in \{0, 1\}$ is the corresponding binary class label, and $\mathbb{I} \in \{0, 1\}$ is an indicator function. If there are $N$ proteins in total, then the average precision and recall are defined as:

$$\text{precision}(t) = \frac{\sum_i \text{precision}_i(t)}{\sum_i \left(\sum_j (p_{ij}^t \geq t) \geq 1\right)} , \quad \text{recall}(t) = \frac{\sum_i \text{recall}_i(t)}{N}$$

Finally, $F_{\text{max}}$ is defined as the maximum value of F-score over all thresholds,

$$F_{\text{max}} = \max_t \left\{ \frac{2 \cdot \text{precision}(t) \cdot \text{recall}(t)}{\text{precision}(t) + \text{recall}(t)} \right\}$$ (13)

A.5 More Results of GO Term Prediction

For GO term prediction, we also apply different cutoff splits. Proteins in the test set are categorized into five groups based on their similarity to the training set (30%, 40%, 50%, 70%, and 95%). As
shown in Figure 6, the results of GO term prediction are presented in Figure 8(a)-(c). The proposed model CoupleNet achieves the highest $F_{\text{max}}$ scores across all cutoffs on these tasks. Even when there is a low similarity between the training and test sets, our model also has higher scores, which demonstrates the superiority and robustness of the proposed model.

### A.6 Completeness Analysis

Given a protein 3D graph $G = (\mathcal{V}, \mathcal{E}, \mathcal{P})$, we capture the geometric representations based on the atoms’ 3D positions and use sequential and structural representations as the node and edge features. For a 3D structure, based on the definition of completeness in Sec. 3.1 and the rigorously demonstrated method to show the calculated geometries can achieve completeness for structures [47], we guarantee the completeness of the selected geometric representations at the base and backbone levels of structures.

The geometric representations are SE(3) invariant (distances, angles) and SE(3) equivariant (directions, orientations). Therefore, it is natural for Eq. 3 to hold from right to left. To demonstrate Eq. 3 holding from left to right, we need to show $F(G) \Rightarrow T_{\gamma}(\mathcal{P})$, where $T_{\gamma}$ does not change the 3D conformation of a 3D graph. Thus we need to show positions can be determined by $F(G)$.

The base approach CoupleNet$_{\text{aa}}$ only considers the C$_\alpha$ coordinates and constructs LCS for each residue. $F(G)_{\text{aa}}$ provides complete representations. First, when $n = 1$, it holds. Assume the case $n = k$ holds such that $F(G)_{\text{aa}}$ is complete. Then we need to prove the case $n = k + 1$ still holds. This is obvious because if $v_i$ is the $(k + 1)$-th node connected to node $v_i$ among the existing $k$ nodes, the LCS $Q_j$ can be easily obtained from $Q_i$ and $F(G)_{\text{aa}}$.

When considering the backbone atoms C$_\alpha$, C, N, O, $F(G)_{\text{oa}}$ is complete. As shown in Figure 5, the remaining degree of freedom at the backbone level is the rotation angles $\Phi$, $\Psi$, $\Omega$ based on the rigid bond lengths and angles. Such backbone torsion angles are calculated and concatenated with $x_{i,\text{oa}}$ into $x_i$. Besides, for any residues $i$ and $j$, the calculated six inter-residue geometries fully define the relative locations of backbone atoms. Therefore, there are no other remaining degrees of freedom. Consequently, the obtained geometric representations at the backbone level are complete.
A.7 More Results of Ablation Study

Table 3 presents an ablation study of the proposed CoupleNet model. Apart from removing $\Phi, \Psi, \Omega$ or $d, \omega, \theta, \varphi$ and using the base model CoupleNet$_{aa}$, we conduct more ablation experiments on the four tasks. The results are shown in Table 6.

Compared with the full model, we consider removing either the sequence or structure information to analyze their importance. Removing the sequence information means removing the encoding of amino acid types for each node. Removing the structure information means removing features related to protein geometry ($F(G)_{aa}, \Phi, \Psi, \Omega, d, \omega, \theta, \varphi$, and we omit related subscripts for brevity).

As shown in Table 6, removing either sequence or structure causes a performance drop on all tasks, demonstrating that both types of information are critical for the proposed method. When removing the structure, the performance decreases more significantly, suggesting that structural information provides more important and comprehensive clues compared with sequence information alone. Combining these diverse data sources leads to optimal predictive performance.