Enhancing Protein Function Prediction: Integrating Pretraining and Fine-Tuning within Geometric-Aware Graph Neural Networks

Chenxi Hu
Tsinghua University
2024310688
hucx24@mails.tsinghua.edu.cn

Fei Long
Tsinghua University
2024316091
longf24@mails.tsinghua.edu.cn

Renrui Tian
Tsinghua University
2024310636
trr24@mails.tsinghua.edu.cn

Abstract

Proteins are essential to biological processes, and accurate function prediction is vital for advancing molecular biology and therapeutic development. Traditional methods often face challenges with low-similarity proteins and novel families, while recent deep learning approaches leveraging sequence and structural data have shown promise. To address limitations in existing methods, we propose a geometric-aware graph neural network (GNN) framework that explicitly models protein structures through node and edge features, incorporating radial basis functions and Fourier encoding to capture spatial relationships. Combined with a self-supervised pretraining task on large-scale, unlabeled structural datasets, our method demonstrates competitive performance across benchmarks, suggesting its potential to enhance protein function prediction and contribute to further progress in the field.

1 Introduction

Proteins are fundamental biomolecules responsible for a wide array of essential tasks in living organisms, ranging from catalyzing biochemical reactions to maintaining cellular structure. Their unique amino acid sequences determine their three-dimensional structures, which in turn dictate their biological functions. Protein functions are remarkably diverse, encompassing enzymatic activity, structural support, molecular transport, cell signaling, and immune response. Accurate prediction of protein function is crucial not only for advancing our understanding of molecular biology but also for driving innovations in drug discovery and disease treatment [1].

The advent of high-throughput sequencing technologies has led to an exponential increase in the number of protein sequences in databases. However, the majority of these sequences remain unannotated due to the high costs and labor-intensive nature of experimental methods. This challenge has created a pressing need for efficient and accurate computational approaches for protein function prediction. Traditional methods, such as sequence homology alignment, transfer functional annotations from known to unknown proteins using tools like BLAST [2], Position-Specific Scoring Matrices (PSSM) [3], and Multiple Sequence Alignment [4]. While effective for proteins with high sequence similarity, these methods struggle to capture functional relationships for proteins with low sequence identity or novel families, limiting their applicability across diverse proteomes.

Preprint. Under review.

Recent advances in deep learning have ushered in a new era of protein function prediction methods, which can be broadly categorized into sequence-based and structure-based approaches. Sequence-based methods leverage 1D convolutional neural networks (CNNs) or Transformer architectures to generate task-specific representations from protein sequences [5, 6]. Later developments have incorporated homology information alongside sequence data, yielding significant improvements [7, 8]. Meanwhile, breakthroughs in protein structure prediction, such as AlphaFold2 [9] and RoseTTAFold [10], have enabled researchers to obtain highly accurate three-dimensional structures from protein sequences. These advancements have paved the way for structure-based methods, which often employ graph neural networks (GNNs) to extract features from protein structures [11, 12, 13]. In these approaches, residues are represented as nodes, and signals are propagated across geometric neighborhoods via message passing, with graph pooling layers aggregating the information into protein-level representations for classification.

Despite these developments, several limitations in GNN-based protein function prediction remain unresolved:

1. Existing methods, such as HEAL [13], primarily focus on learning node information and updating it based on simple connectivity. However, they often overlook crucial factors like distance and orientation, which are essential for accurately modeling protein structures [9]. This simplistic treatment of edges limits the expressiveness of the learned representations.
2. The emergence of advanced structural prediction tools like AlphaFold [9, 14] has introduced a wealth of novel protein structures that could significantly enhance function prediction. However, many of these structures lack labels due to their sequence novelty and low similarity to known proteins. While current methods like HEAL attempt to transfer labels via sequence homology alignment, such approaches fail to fully utilize the large volume of unlabeled and novel structures, posing challenges for accurate function prediction.

To address these challenges, we propose a novel framework for protein function prediction. Inspired by the application of GNNs in materials science [15, 16] and building on HEAL [13], we developed a geometric-aware GNN that explicitly models protein structures through both nodes and edges. This model incorporates radial basis functions (RBFs) [17] and Fourier feature encoding [18] to account for the varying distances and orientations of edges. By sequentially updating edge and node information during message passing, our approach enables the GNN to capture the intricate structural properties of proteins more effectively.

Additionally, inspired by methods like GearNet-Edge [19], we designed a self-supervised pretraining task to further enhance the geometric-aware GNN. This pretraining task allows the model to learn generalizable representations from large-scale, unlabeled protein structural datasets, equipping it with universal knowledge about protein structures that can be leveraged for downstream tasks, including function prediction.

We evaluated our framework against a range of baseline methods, including BLAST [2], DeepGO [5], DeepFRI [11], DeepGO-Plus [7], DeepGO-SE [20], and HEAL [13]. On most evaluation metrics, our method consistently outperformed state-of-the-art approaches. This highlights the effectiveness of explicitly modeling protein structures and leveraging large-scale structural datasets for pretraining in improving protein function prediction accuracy.

2 Problem Definition

Protein. Proteins are intricate macromolecules composed of sequences of amino acid residues that fold into unique three-dimensional structures through various non-covalent interactions, such as hydrogen bonds, hydrophobic interactions, and van der Waals forces. The primary structure of a protein is determined by its amino acid sequence, with each amino acid possessing distinct chemical properties that influence the protein's final conformation. This sequence can be represented as $S = [s_1, s_2, \ldots, s_n]$, where each s_i is an integer from the set $0, 1, \ldots, 19$, corresponding to one of the 20 standard amino acids. The three-dimensional structure, which plays a crucial role in determining the protein's function, is denoted as $\mathcal{X} = [x_1, x_2, \ldots, x_m]$. Each $x_i \in \mathbb{R}^{L\times 3}$ represents the 3D coordinates of the atoms within the i-th amino acid residue, where L is the number of atoms in that residue. Both the sequence and structure are essential for characterizing protein function, as they directly influence how the protein performs its specific biological roles within the cell.

Protein Function. Various standards exist for classifying protein functions, including the Gene Ontology (GO) [21], Enzyme Commission (EC) [22], Kyoto Encyclopedia of Genes and Genomes (KEGG) [23], and Pfam [24]. In this work, we adopt the GO database to describe protein functions, as it is one of the most widely used and successful ontologies in biology [21]. GO annotations classify protein functions into three main categories: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC). Within each category, GO employs a Directed Acyclic Graph (DAG) structure to represent the hierarchical relationships between protein functions. Terms within the DAG are connected through "is-a" or "part-of" relationships, linking specific, lower-level terms to broader, higher-level categories. This structure effectively represents protein functions in a hierarchical manner. Notably, when a protein is assigned a specific GO term, it is automatically associated with all its ancestor terms, ensuring comprehensive and consistent functional annotation.

Protein Function Prediction. Protein function prediction is commonly formulated as a classification problem, where the objective is to determine the functional annotations of a protein based on its sequence S and three-dimensional structure $\mathcal{X}$.

3 Methodology

3.1 Overview

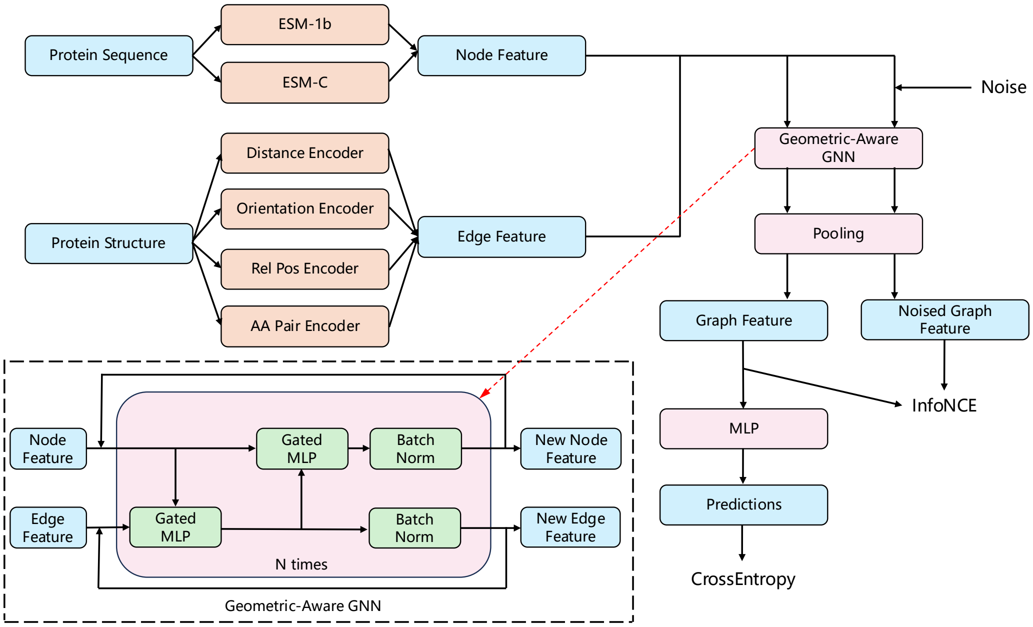

Figure 1: Overview of our method

Our method overview is illustrated in Figure 1, which presents a neural network framework for protein function prediction by integrating sequence and structural information. Protein sequences are processed using pre-trained models (ESM-1b [25] and ESM-C [26]) to generate node features, while geometric encoders (distance, orientation, relative position, and amino acid pair encoders) encode protein structures to produce edge features. These features are input to a Geometric-Aware GNN, which iteratively updates them using gated MLPs and batch normalization (details in Section 3.2). The updated graph features are pooled and passed through an MLP for downstream predictions, supervised by a CrossEntropy loss. To enhance representation learning, a contrastive learning task with InfoNCE loss is applied by introducing noise to the graph features (details in Section 3.3).

3.2 Geometric-Aware Graph Neural Network

To more effectively capture the structural properties of proteins during message passing, we propose a geometric-aware graph neural network (GNN) that incorporates both node and edge features.

Graph Featurizer The graph features for protein structures consist of nodes representing individual amino acid residues, and edges capturing interactions between residues within a cutoff distance (i.e., 1nm).

For node features, we utilize pre-trained models ESM-1b[25] and ESM-C[26], which encode protein sequence information. ESM-1b is a deep learning model developed for protein sequence analysis, trained on a large dataset of protein sequences using a transformer-based architecture. It learns rich feature representations that capture both local and global sequence patterns, which are critical for understanding protein structure and function. ESM-C is an enhanced version of ESM-1b that scales up both the data and training compute. These models generate embeddings that capture intricate sequence-level relationships and contribute to a deeper understanding of the protein's structural and functional properties.

For edge features, we consider four key attributes: sequence distance, spatial distance, orientation angle, and amino acid pair type. Sequence distance and amino acid pair type are encoded using an embedding layer, transforming categorical features into continuous representations. Spatial distance is modeled using a Gaussian basis function with a specified maximum radius and a set number of basis functions, enabling the capture of spatial proximity and its influence on residue interactions. Orientation angles, which represent the relative rotation between residue pairs in three-dimensional space, are encoded with a sinusoidal function, using Fourier feature encoding to map angular values into a higher-dimensional space.

Message Passing The message passing mechanism updates both node and edge features at each layer. For the edge feature update, the node features from both ends of the edge, together with the current edge feature, are concatenated and passed through a GatedMLP[27]. This is followed by a weighted update, where a learned weight is applied to the edge features, guided by an initial edge feature, $\mathbf{e}_0$. The formula for updating the edge features is as follows:

$$\mathbf{e}_i' = \mathbf{e}_i + \text{GatedMLP}_{\text{edge}}\left([\mathbf{h}_{u_i}, \mathbf{h}_{v_i}, \mathbf{e}_i]\right) \cdot \mathbf{w}_{\text{edge}}\left(\mathbf{e}_i^{\text{init}}\right)$$

Here, $\mathbf{e}_i$ represents the current edge feature for edge i, and $\mathbf{h}_{u_i}$ and $\mathbf{h}_{v_i}$ are the node features of the two nodes, u_i and v_i, connected by edge i. The multi-layer perceptron $\text{GatedMLP}_{\text{edge}}$ processes the concatenated node and edge features, and the initial edge feature $\mathbf{e}_i^{\text{init}}$ guides the update with a learned weight $\mathbf{w}_{\text{edge}}$.

Similarly, node features are updated by aggregating the node features from neighboring nodes, along with the edge features. The aggregated features are passed through a separate GatedMLP for nodes, and the updated node features are aggregated using a scatter operation, ensuring that information from all neighbors is properly integrated. The formula for updating the node features is as follows:

$$\mathbf{h}_u' = \mathbf{h}_u + \sum_v \text{GatedMLP}_{\text{node}}\left([\mathbf{h}_u, \mathbf{h}_v, \mathbf{e}_i]\right) \cdot \mathbf{w}_{\text{node}}\left(\mathbf{e}_i^{\text{init}}\right)$$

By directly integrating geometric information into the GNN, our approach captures the complex structural relationships within proteins, significantly enhancing the prediction of protein functions based on both sequence and structure.

3.3 Contrastive Learning

Following HEAL[13], we apply graph contrastive learning to regularize and improve the quality of graph representations in our model. By adding random noise to both the node and edge features in the latent space, we generate multiple augmented views of the same protein graph, ensuring that crucial residues and interactions are retained. This approach enables us to introduce diverse yet meaningful perspectives of the graph without losing essential structural information.

To achieve this, we perturb the node and edge features of each protein graph and aim to maximize the similarity between the graph-level representations of two augmented views of the same graph, while minimizing their similarity to representations from other graphs. We optimize this objective using the InfoNCE loss function[28], which encourages the model to learn discriminative graph representations. The InfoNCE loss is formally defined as:

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{M}\sum_{i=1}^{M} \log \frac{\exp(\text{sim}(z_i, \tilde{z}_i)/\tau)}{\sum_{j=1}^{N} \exp(\text{sim}(z_i, \tilde{z}_j)/\tau)}$$

Here, $\text{sim}(x, y)$ denotes the cosine similarity between two graph features x and y, and τ is a temperature parameter that controls the sharpness of the softmax function, typically set to 0.5 as recommended in prior work[29].

3.4 Pretraining and Fine-Tuning Framework

To bolster the ability of Geometric-Aware GNNs to capture protein structures and provide more robust representations, we employ a large-scale pre-training dataset to conduct self-supervised pre-training on our GNNs. This approach allows our model to learn intricate patterns and features from the data without the need for manual labeling, which is a significant step towards enhancing the model's capacity to understand and predict complex protein structures.

The pre-training task is inspired by the Residue Type Prediction task in GearNet[19], which adopts a training strategy akin to masked language models[30]. Specifically, we introduce a masking mechanism to the dataset that has been pre-processed by ESM-1b and ESM-C. This mechanism obscures the original types of amino acid sequences and their corresponding node features in the feature sequence. Subsequently, we utilize our Geometric-Aware GNN coupled with a MLP to output logits for the masked positions, predicting the amino acid types at these locations. To elevate the difficulty of the pre-training task, we opt to mask 70% of the data, challenging the model to make predictions based on the limited remaining visible information.

3.5 Model Training

During the pre-training phase, we train our model for 50,000 iterations on the pre-training dataset with a learning rate of 1×10^{-4}. The entire training process is conducted in parallel across 8 NVIDIA A100 GPUs to ensure computational efficiency and expedite the training speed. Through this pre-training approach, our model is able to learn deeper protein structural features, laying a solid foundation for subsequent downstream tasks.

For both training from scratch and fine-tuning tasks, we utilize the Adam optimizer[31] with a learning rate of 1×10^{-4} and a batch size of 32 to train our Geometric-aware GNN. All models are implemented using PyTorch and the PyTorch Geometric library[32, 33]. The experiments are conducted on a single NVIDIA A100 GPU with 80 GB of memory.

4 Experiments

4.1 Datasets

For pre-training, we utilized a subset of the AlphaFold protein structure database [9], which consists of 400,000 proteome-wide predictions generated by AlphaFold2. For model training, we selected a subset of the dataset employed by HEAL [13], containing 36,404 protein structures from the Protein Data Bank (PDB). This subset includes representative PDB chains with at least one functional annotation and high-resolution structures. The dataset was divided into training, validation, and test sets, with an 8:1:1 ratio. Protein graphs were constructed for each sequence, with functional annotations derived from SIFTS [34] and UniProtKB [35]. Each sequence was annotated with 489 molecular function (MF), 1,943 biological process (BP), and 320 cellular component (CC) terms. The sequences in the test set were grouped based on sequence homology, and experiments were conducted at a sequence identity threshold of 95%.

4.2 Baselines

We compared our approach with several baseline methods, including Blast [36], DeepGO [5], DeepFRI [11], DeepGO-SE [20], and HEAL [13]. We replicated the experimental setup used in the HEAL study, while reusing the evaluation results of the other baseline methods as reported in TAWFN [37].

4.3 Metrics

To evaluate the performance of our method, we employed AUPR (Area Under the Precision-Recall Curve) [38]. AUPR measures the area under the precision-recall curve, reflecting the model's performance across varying prediction thresholds. Higher AUPR values indicate better overall performance, as the metric accounts for both precision and recall, particularly in imbalanced datasets.

4.4 Results

Method	**MF**	**BP**	**CC**
BLAST	0.136	0.067	0.096
DeepGO	0.391	0.189	0.258
DeepFRI	0.495	0.265	0.274
DeepGO-SE	0.495	0.233	**0.423**
HEAL	0.604	0.296	0.401
GACNN(scratch)	0.615	0.309	0.414
GACNN(pretrained)	**0.621**	**0.314**	0.415

Table 1: Comparison of different methods in terms of AUPR.

We evaluate the performance of our model on three gene ontology domains (MF, BP, CC) individually. As shown in Table 1, our Geometric-Aware GNN (GAGNN) achieves AUPR scores of 0.621, 0.314, and 0.415 for the MF, BP, and CC tasks, respectively. GAGNN outperforms BLAST, DeepGO, DeepFRI, and HEAL in all three domains, and achieves comparable results to DeepGO-SE on the CC task. These results highlight our method's ability to effectively capture the complex structural features of proteins, leading to a noticeable improvement in protein function prediction across various gene ontology domains.

5 Conclusion

Accurate protein function prediction is critical for understanding molecular biology and driving innovations in drug discovery. To address challenges posed by the abundance of unannotated protein sequences and structures, we propose a geometric-aware GNN framework that explicitly models both nodes and edges in protein structures. By incorporating edge-to-edge interactions alongside traditional node and edge modeling, the framework captures crucial geometric properties such as distances and orientations using RBFs and Fourier feature encoding. Furthermore, we introduce a pretraining strategy, which is a self-supervised task that leverages large-scale unlabeled protein datasets to learn universal structural representations. These enhancements enable our model to better generalize across diverse proteins and improve function prediction accuracy. Looking ahead, this framework provides a foundation for more robust protein modeling. Future efforts may explore integrating additional biochemical properties and refining pretraining strategies to further enhance its predictive power.

References

[1] David Eisenberg, Edward M Marcotte, Ioannis Xenarios, and Todd O Yeates. Protein function in the post-genomic era. *Nature*, 405(6788):823–826, 2000.

[2] Thomas Madden. The blast sequence analysis tool. *The NCBI handbook*, 2(5):425–436, 2013.

[3] Jong cheol Jeong, Xiaotong Lin, and Xue-Wen Chen. On position-specific scoring matrix for protein function prediction. *IEEE/ACM transactions on computational biology and bioinformatics*, 8(2):308–315, 2010.

[4] Robert C Edgar and Serafim Batzoglou. Multiple sequence alignment. *Current opinion in structural biology*, 16(3):368–373, 2006.

[5] Maxat Kulmanov, Mohammed Asif Khan, and Robert Hoehndorf. Deepgo: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics*, 34(4):660–668, 2018.

[6] Rui Fa, Domenico Cozzetto, Cen Wan, and David T Jones. Predicting human protein function with multi-task deep neural networks. *PloS one*, 13(6):e0198216, 2018.

[7] Maxat Kulmanov and Robert Hoehndorf. Deepgoplus: improved protein function prediction from sequence. *Bioinformatics*, 37(8):1187–1187, 2021.

[8] Yue Cao and Yang Shen. Tale: Transformer-based protein function annotation with joint sequence–label embedding. *Bioinformatics*, 37(18):2825–2833, 2021.

[9] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[10] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

[11] Vladimir Gligorijević, P Douglas Renfrew, Tomasz Kosciolek, Julia Koehler Leman, Daniel Berenberg, Tommi Vatanen, Chris Chandler, Bryn C Taylor, Ian M Fisk, Hera Vlamakis, et al. Structure-based protein function prediction using graph convolutional networks. *Nature communications*, 12(1):3168, 2021.

[12] Boqiao Lai and Jinbo Xu. Accurate protein function prediction via graph attention networks with predicted structure information. *Briefings in Bioinformatics*, 23(1):bbab502, 2022.

[13] Zhonghui Gu, Xiao Luo, Jiaxiao Chen, Minghua Deng, and Luhua Lai. Hierarchical graph transformer with contrastive learning for protein function prediction. *Bioinformatics*, 39(7):btad410, 2023.

[14] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3, 2024.

[15] Han Yang, Chenxi Hu, Yichi Zhou, Xixian Liu, Yu Shi, Jielan Li, Guanzhi Li, Zekun Chen, Shuizhou Chen, Claudio Zeni, et al. Mattersim: A deep learning atomistic model across elements, temperatures and pressures. *arXiv preprint arXiv:2405.04967*, 2024.

[16] Chi Chen and Shyue Ping Ong. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science*, 2(11):718–728, 2022.

[17] Martin Dietrich Buhmann. Radial basis functions. *Acta numerica*, 9:1–38, 2000.

[18] Yang Li, Si Si, Gang Li, Cho-Jui Hsieh, and Samy Bengio. Learnable fourier features for multi-dimensional spatial positional encoding. *Advances in Neural Information Processing Systems*, 34:15816–15829, 2021.

[19] Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022.

[20] Maxat Kulmanov, Francisco J Guzmán-Vega, Paula Duek Roggli, Lydie Lane, Stefan T Arold, and Robert Hoehndorf. Protein function prediction as approximate semantic entailment. *Nature Machine Intelligence*, 6(2):220–228, 2024.

[21] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[22] Amos Bairoch. The enzyme database in 2000. *Nucleic acids research*, 28(1):304–305, 2000.

[23] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[24] Jaina Mistry, Sara Chuguransky, Lowri Williams, Matloob Qureshi, Gustavo A Salazar, Erik LL Sonnhammer, Silvio CE Tosatto, Lisanna Paladin, Shriya Raj, Lorna J Richardson, et al. Pfam: The protein families database in 2021. *Nucleic acids research*, 49(D1):D412–D419, 2021.

[25] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *PNAS*, 2019.

[26] ESM Team. ESM Cambrian: Revealing the mysteries of proteins with unsupervised learning. `https://evolutionaryscale.ai/blog/esm-cambrian`, December 2024. EvolutionaryScale Website.

[27] Hanxiao Liu, Zihang Dai, David So, and Quoc V Le. Pay attention to mlps. *Advances in neural information processing systems*, 34:9204–9215, 2021.

[28] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[29] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[30] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[31] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[33] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

[34] Jose M Dana, Aleksandras Gutmanas, Nidhi Tyagi, Guoying Qi, Claire O'Donovan, Maria Martin, and Sameer Velankar. Sifts: updated structure integration with function, taxonomy and sequences resource allows 40-fold increase in coverage of structure-based annotations for proteins. *Nucleic acids research*, 47(D1):D482–D489, 2019.

[35] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2025. *Nucleic Acids Research*, page gkae1010, 11 2024.

[36] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

[37] Lu Meng and Xiaoran Wang. Tawfn: a deep learning framework for protein function prediction. *Bioinformatics*, 40(10):btae571, 2024.

[38] J Davis and M Goadrich. Icml'06: Proceedings of the 23rd international conference on machine learning. *New York, NY, USA: ACM*, pages 233–240, 2006.