FAFormer: Frame Averaging Transformer for Predicting Nucleic Acid-Protein Interactions

Tinglin Huang[1]* **Zhenqiao Song**[2] **Rex Ying**[1] **Wengong Jin**[3]

[1]Yale University, [2]University of California, Santa Barbara,
[3]Broad Institute of MIT and Harvard

Abstract

Frame averaging (FA), a recent progress in geometric deep learning, is a general framework that endows a given architecture with the ability to transform data equivariantly. However, serving FA as a model wrapper introduces additional computation that grows linearly with the group's cardinality and may hinder the exploitation of 3D structures, making it challenging to model macro-molecules such as proteins and nucleic acids. In this paper, we present FAFormer, an equivariant Transformer model that incorporates FA as a basic component within each layer. Such incorporation allows FAFormer to model the coordinates in the latent space directly without using other elaborate geometric features. Building on this foundation, we introduce an equivariant cross-attention module to FAFormer to capture the interactions between node and coordinate representations. Besides, an equivariant feed-forward network is proposed for enhancing the communication between them. To evaluate FAFormer's performance, we establish two benchmark datasets for nucleic acid-protein contact prediction and compare FAFormer with 8 different baseline models. With these two innovations, FAFormer outperforms all the baselines and achieves state-of-the-art performance.

1 Introduction

Machine learning methods have recently shown promise in modeling and understanding biological molecules, such as predicting the protein tertiary structure [20, 4], designing the molecule with high binding affinity [9, 17], and approximating the quantum mechanics [14]. One key factor contributing to the success of these methods is their capacity to exploit molecular symmetry by learning transformations that are equivariant to specific symmetry groups [27, 22, 11, 13, 29, 28, 24, 19]. For example, some of them solely rely on the invariant features extracted from the molecules to exhibit invariant transformations while others achieve this by mapping the coordinate system into the spherical harmonics space.

Recently, a novel line of research focuses on designing an encoder-agnostic equivariant framework with frame averaging (FA) $\langle\cdot\rangle_{\mathcal{F}}$ [25, 34]:

$$\langle\Phi\rangle_{\mathcal{F}} = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\rho_2(g)\Phi\left(\rho_1(g)^{-1}X\right) \tag{1}$$

where $\Phi : V \to W$ is the mapping function between normed vector spaces V and W, $\rho_1(g)$ and $\rho_2(g)$ are representations of the group G over V and W, and $\mathcal{F} : V \to 2^G$ is the frame that maps the vector space into a group. If a frame is G-equivariant, then any given map Φ can achieve equivariance

*This work was done when Tinglin was an intern at Broad Institute.

Machine Learning for Structural Biology Workshop, NeurIPS 2023.

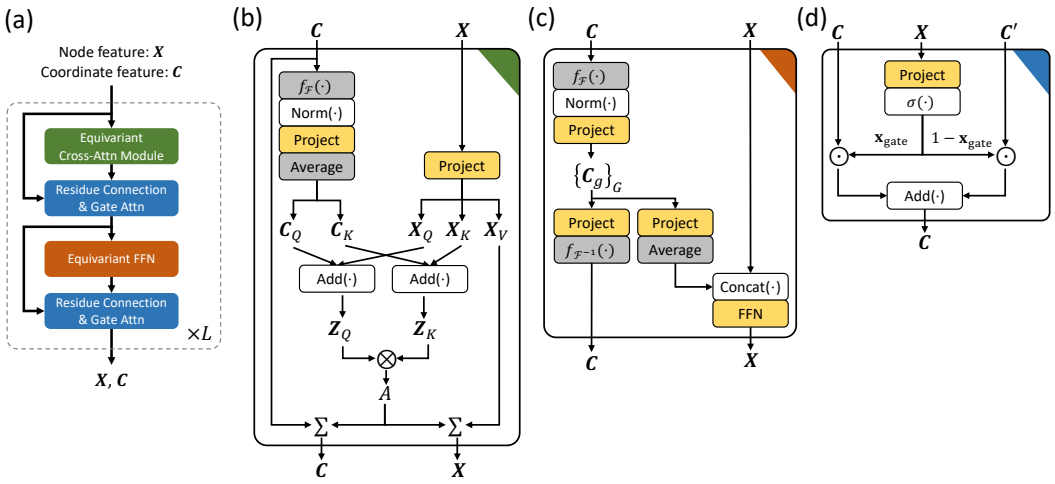

Figure 1: Overview of our proposed FAFormer architecture. **(a)** The input consists of the node features and coordinate features, and FAFormer includes a stack of equivariant cross-attention modules and equivariant FFNs. **(b)** Equivariant cross-attention module includes the projection, FA transformation, and query/key embedding fusion. **(c)** Equivariant FFN includes standard FFN for node representations and two transformations for coordinate representations. **(d)** Gate attention utilizes node representations to module the fusion between updated and original coordinate representations. $\sum$ deontes aggregation, $\odot$ denotes element-wise multiplication, and $\otimes$ denotes multiplication. Gray cells indicate the operation related to FA.

(or invariance) by averaging predictions across that frame. As an example, one can simply encode the mapped coordinates with a vanilla Transformer [32] and average the results over the group to exhibit an equivariance transformation.

However, despite its generality and ease of use, the computation of the backbone model scales linearly with the cardinality of G [11, 18] due to the independent encoding for different group elements. Specifically, applying Principal Component Analysis (PCA) to the frame results in an 8x increase in computation, making it intractable for modeling macro-molecules such as protein and nucleic acid. Besides, merely using FA as a model wrapper and encoding the geometric features together with node features may diminish the exploitation of the geometric information, as shown in our experiments.

In this work, we present **FAFormer**, an equivariant Transformer architecture for nucleic acid-protein complexes modeling. Instead of serving FA as an external geometric wrapper on top of the model, FAFormer instantiates FA as an integral geometric module within each layer, which eliminates the need for separate encoding under distinct group elements, yet preserves the equivariant transformation. Moreover, it incorporates the equivariant cross-attention module, equivariant feed-forward network, and gate activation to enhance expressive power and allow communication between node and coordinate representations. To evaluate its performance, we clean up and construct two contact prediction benchmarks for DNA-Protein and RNA-Protein complexes from multiple sources [6, 5, 1], and FAFormer outperforms all baseline models, achieving state-of-the-art results.

2 Architecture

The main idea of FAFormer is incorporating FA as a basic component into Transformer, allowing it to directly represent coordinates in the latent space. Such a strategy enables effective modeling of molecules without depending on elaborate spatial features. Specifically, the input of FAFormer comprises the node features $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ and coordinate features $\boldsymbol{C} \in \mathbb{R}^{N \times 3}$, where N is the number of the residues or nucleotides and D is the hidden size. FAFormer processes and updates the input features at each layer:

$$\boldsymbol{X}^{(l+1)}, \boldsymbol{C}^{(l+1)} = f^{(l)}(\boldsymbol{X}^{(l)}, \boldsymbol{C}^{(l)}) \tag{2}$$

where $f^{(l)}(\cdot)$ represents l-th layer of FAFormer. Each layer contains an *Equivariant Cross-Attention Module* presented in Section 2.1 and an *Equivariant Feed-Forward Network* presented in Section 2.2.

Frame. Here $f_{\mathcal{F}}(\cdot)$ is used to denote the mapping based on the frame $\mathcal{F}$. Specifically, it projects a given set of coordinates into different vector spaces using the group G computed by $\mathcal{F}$:

$$\{\boldsymbol{C}_g\}_G = f_{\mathcal{F}}(\boldsymbol{C}) \tag{3}$$

where $\boldsymbol{C}_g \in \mathbb{R}^{N\times 3}$ is the coordinates transformed by the group element g. The notation $|G|$ is denoted as the cardinality of G. Following the previous methods [25, 11], the three principle components $\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3$ computed by PCA construct the group with 8 group elements:

$$G = \mathcal{F}(\boldsymbol{C}) = \{[\pm\boldsymbol{u}_1, \pm\boldsymbol{u}_2, \pm\boldsymbol{u}_3]\} \tag{4}$$

Besides, $f_{\mathcal{F}^{-1}}(\cdot)$ denotes the inverse mapping which first maps each projected coordinates C_g using the inverse group element g^{-1}, then averages the results over the group to equivariantly obtain the coordinates:

$$\boldsymbol{C}' = f_{\mathcal{F}^{-1}}(\{\boldsymbol{C}_g\}_G) \tag{5}$$

2.1 Equivariant Cross-Attention Module

Motivated by recent studies on multi-modality learning [7, 16, 10], we propose Equivariant Cross-Attention Module to enable the node and coordinate representations to complement each other through the shared cross attention map. In general, it processes them with separate branches and fuses the query/key embeddings for generating the attention map biased by their interactions.

Feature Transformation. Similar to the Transformer, the attention module of FAFormer first transforms the node features $\boldsymbol{X} \in \mathbb{R}^{N\times D}$ to the query, key, and value matrices:

$$\boldsymbol{X}_Q = \boldsymbol{X}\boldsymbol{W}_Q, \quad \boldsymbol{X}_K = \boldsymbol{X}\boldsymbol{W}_K, \quad \boldsymbol{X}_V = \boldsymbol{X}\boldsymbol{W}_V \tag{6}$$

where $\boldsymbol{W}_Q, \boldsymbol{W}_K, \boldsymbol{W}_V \in \mathbb{R}^{D\times D'}$ are the learnable projections. With frame averaging, we can also transform the coordinates features $\boldsymbol{C} \in \mathbb{R}^{N\times 3}$ equivariantly and model the representations in the latent space similar to the node features. Specifically, the coordinates are first projected using Equ.3, then transformations are averaged over the group:

$$\boldsymbol{C}_Q = \frac{1}{|G|}\sum_g \text{Norm}(\boldsymbol{C}_g)\boldsymbol{W}'_{Q,g}, \quad \boldsymbol{C}_K = \frac{1}{|G|}\sum_g \text{Norm}(\boldsymbol{C}_g)\boldsymbol{W}'_{K,g} \tag{7}$$

where $\boldsymbol{W}'_{Q,g}, \boldsymbol{W}'_{K,g} \in \mathbb{R}^{3\times D'}$ are the learnable projections for each group element g, and $\text{Norm}(\cdot)$ is the normalization which scales the coordinates such that their root-mean-square norm is one [19]:

$$\boldsymbol{C}/\sqrt{\frac{1}{\nu}||\boldsymbol{C}||_2^2} = \text{Norm}(\boldsymbol{C}) \tag{8}$$

where ν is a non-learnable scalar.

Attention Map Calculation. To enable a cross-attention between node and coordinate representations, we fuse their query/key matrices and compute the attention map with dot-product operation:

$$\boldsymbol{Z}_Q = \boldsymbol{X}_Q + \boldsymbol{C}_Q, \quad \boldsymbol{Z}_K = \boldsymbol{X}_K + \boldsymbol{C}_K, \tag{9}$$

$$\boldsymbol{A} = \text{Softmax}(\boldsymbol{Z}_Q\boldsymbol{Z}_K^T/\sqrt{D}) \tag{10}$$

where $\boldsymbol{A} \in \mathbb{R}^{N\times N}$. It can be observed that the resulting attention map $\boldsymbol{A}$ is biased by the interactions between these two representations, which can align the node representation of a residue/nucleotide with its geometric context. Since $\boldsymbol{C}_Q, \boldsymbol{C}_K$ are obtained through an equivariant transformation, the attention calculation is also equivariant to the 3D structure.

Aggregation. Node representations are updated with the attention map and the residue connection:

$$\boldsymbol{X}' = \text{LN}\left((\boldsymbol{A}\boldsymbol{X}_v)\boldsymbol{W}_{\text{output}}\right) + \boldsymbol{X} \tag{11}$$

where LN is the layernorm [2] and $\boldsymbol{W}_{\text{output}} \in \mathbb{R}^{D' \times D}$ is a learnable matrix. As for the coordinate representations, we additionally employ gate attention [20, 3] which applies the node representations as input to modulate the aggregation:

$$\boldsymbol{C}' = \mathbf{x}_{\text{gate}}^{\text{Attn}} \odot \boldsymbol{A}\boldsymbol{C} + (1 - \mathbf{x}_{\text{gate}}^{\text{Attn}}) \odot \boldsymbol{C}, \tag{12}$$

$$\text{where} \quad \mathbf{x}_{\text{gate}}^{\text{Attn}} = \sigma\left(\boldsymbol{X}\mathbf{a}_{\text{gate}}^{\text{Attn}} + \mathbf{b}_{\text{gate}}^{\text{Attn}}\right) \tag{13}$$

where $\mathbf{a}_{\text{gate}}^{\text{Attn}} \in \mathbb{R}^{D \times 1}, \mathbf{b}_{\text{gate}}^{\text{Attn}} \in \mathbb{R}^{N \times 1}$ are learnable gating vectors, $\sigma(\cdot)$ is the sigmoid function, and $\odot$ is the element-wise multiplication operation.

2.2 Equivariant Feed-Forward Network

To further enhance the communication, we design an Equivariant Feed-Forward Network that updates both node and coordinate representations. We first project the coordinates $\boldsymbol{C}$ into $\{\boldsymbol{C}'_g\}_G$ using Equ.3 and transform it into latent space:

$$\{\boldsymbol{C}_g\}_G = \{\text{Norm}(\boldsymbol{C}'_g)\boldsymbol{W}_g\}_G \tag{14}$$

where $\boldsymbol{C}_g \in \mathbb{R}^{N \times D'}, \boldsymbol{W}_g \in \mathbb{R}^{3 \times D'}$. It is then split up into two types of features through separate transformations:

$$\boldsymbol{C}_u = \frac{1}{|G|}\sum_g \boldsymbol{C}_g\boldsymbol{W}_u, \quad \boldsymbol{C}_v = f_{\mathcal{F}^{-1}}\left(\{\boldsymbol{C}_g\boldsymbol{W}_v\}_G\right) \tag{15}$$

where $\boldsymbol{C}_u \in \mathbb{R}^{N \times D}, \boldsymbol{C}_v \in \mathbb{R}^{N \times 3}, \boldsymbol{W}_u \in \mathbb{R}^{D' \times D}, \boldsymbol{W}_v \in \mathbb{R}^{D' \times 3}$. The final updated node representations $\boldsymbol{X}'$ are calculated as a standard FFN [32] projection of the concatenated node features and the transformed coordinate features, resulting in:

$$\boldsymbol{X}' = \text{FFN}([\boldsymbol{X}, \boldsymbol{C}_u]) + \boldsymbol{X}, \tag{16}$$

This allows the node representations to engage with geometric features via the cross-attention map while also directly integrating spatial information into the feature vectors. As for the coordinate representations, we also employ the gate attention to encourage consistency between node and coordinate representations:

$$\boldsymbol{C}' = \mathbf{x}_{\text{gate}}^{\text{FFN}} \odot \boldsymbol{C}_v + (1 - \mathbf{x}_{\text{gate}}^{\text{FFN}}) \odot \boldsymbol{C}, \tag{17}$$

$$\text{where} \quad \mathbf{x}_{\text{gate}}^{\text{FFN}} = \sigma\left(\boldsymbol{X}\mathbf{a}_{\text{gate}}^{\text{FFN}} + \mathbf{b}_{\text{gate}}^{\text{FFN}}\right) \tag{18}$$

where $\mathbf{a}_{\text{gate}}^{\text{FFN}} \in \mathbb{R}^{D \times 1}, \mathbf{b}_{\text{gate}}^{\text{FFN}} \in \mathbb{R}^{N \times 1}$.

3 Experiments

To show the effectiveness of FAFormer, we collect two benchmarks for nucleic acid-protein contact prediction and compare the model with 8 baseline methods. During the experiments, only the coordinates of the C_α atoms from each protein residue and the C_3 atoms from each nucleotide of the nucleic acids are used as coordinate features. For node feature generation, we employ ESM2 [23] for proteins and RNA-FM [8] for RNA. The one-hot embedding is utilized as DNA's node feature.

Dataset. Different from the previous studies [33, 26, 35] which only focus on modeling the protein and identifying the binding residues, the task of our datasets is to predict the exact contact pairs between protein $\{S_i\}_N$ and nucleic acid $\{S'_j\}_{N'}$:

$$\text{Model}(S_i, S'_j) = \begin{cases} 1, & S_i \text{ contacts with } S'_j \\ 0, & \text{Other} \end{cases} \tag{19}$$

The DNA/RNA-Protein Complex data, i.e., DPC and RPC, is collected from PDB [6], NDB [5] and RNASolo [1] databases. We filter out the complexes with the sequence length less than 5 or greater than 800. Following the previous studies on Protein-Protein Interaction [30, 31], a residue-nucleotide pair is determined to be in contact if any of their atoms are within 6Å from each other. We split complexes to ensure that no protein in the validation or test datasets shares over 50% sequence identity with any protein in the training dataset[2]. The statistics are shown in Table 1.

[2]Note that we don't use 30% as the threshold since it results in a very limited validation and test set.

	#Train	#Val	#Test	Label
DPC	2,941	195	192	1.144%
RPC	1,084	104	108	1.344%

Table 1: Dataset statistics where "Label" is the average ratio of the contact pair over all pairs.

Baselines. As for the contact prediction task, we compare FAFormer with two classes of models: 1) graph neural network-based methods, including EGNN [27] and GVP-GNN [19]; 2) Transformer-based methods, including Transformer [32], FA [25] with Transformer, se3Transformer [13] and Equiformer [22]. We further compare GraphBind [33] and GraphSite [35] on the task of binding site prediction. For each model, we individually embed the protein and DNA/RNA. The representations of residues and nucleotides are concatenated from all pairs and fed into a classifier to conduct prediction.

Results. To comprehensively evaluate the performance on label-imbalanced datasets, we apply F1 and PRAUC as the evaluation metrics. The comparison results on contact prediction are presented in Table 2 from which FAFormer achieves the best performance over all the baseline models. It also can be observed that Transformer exhibits the worse performance in most cases, highlighting the importance of structural information in this task. Furthermore, even when FA equips the Transformer with the capability to handle structural information, its performance on the RPC dataset suggests that merely serving FA as a wrapper might not be adequate to fully exploit the geometric structure.

	Metric	Transformer	FA	se3Transformer	Equiformer	GVP-GNN	EGNN	FAFormer
DPC	F1	0.0848	0.1458	0.0196	0.0638	0.1442	0.1109	**0.1651**
	PRAUC	0.1166	0.1377	0.1001	0.0992	0.1403	0.1357	**0.1477**
RPC	F1	0.0396	0.0446	0.0539	0.0940	0.0995	0.0647	**0.1177**
	PRAUC	0.0970	0.1019	0.0929	0.0977	0.1046	0.0959	**0.1188**

Table 2: Comparison results on complex contact prediction.

We also show the results of the binding site prediction task, which only uses protein as input and predicts the potential binding sites on it. As shown in Table 3, FAFormer still achieves the best performance over the current SOTA methods GraphBind and GraphSite, demonstrating the expressive power of FAFormer to model the geometric structure.

	Metric	GraphBind	GraphSite	FAFormer
DPC	F1	0.5194	0.4969	**0.5565**
	PRAUC	0.6137	0.6259	**0.6334**
RPC	F1	0.4786	0.3905	**0.5019**
	PRAUC	0.5072	0.5138	**0.5375**

Table 3: Comparison results on protein binding site prediction.

4 Conclusion

In this study, we embed the frame averaging (FA) mechanism as a geometric module rather than using it as an external model wrapper, and present FAFormer. FAFormer directly models the coordinates in latent space without extracting additional geometric features, and incorporates several modules to allow the communication between the node and geometric features effectively. We establish two benchmark datasets for predicting contact pairs in nucleic acid-protein complex, and FAFormer achieves the SOTA performance over all the baseline models.

References

[1] Bartosz Adamczyk, Maciej Antczak, and Marta Szachniuk. Rnasolo: a repository of cleaned pdb-derived rna 3d structures. *Bioinformatics*, 38(14):3668–3670, 2022.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Minkyung Baek, Ivan Anishchenko, Ian Humphreys, Qian Cong, David Baker, and Frank DiMaio. Efficient and accurate prediction of protein structure using rosettafold2. *bioRxiv*, pages 2023–05, 2023.

[4] Minkyung Baek and David Baker. Deep learning and protein structure modeling. *Nature methods*, 19(1):13–14, 2022.

[5] Helen M Berman, Catherine L Lawson, and Bohdan Schneider. Developing community resources for nucleic acid structures. *Life*, 12(4):540, 2022.

[6] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[7] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366, 2021.

[8] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *bioRxiv*, pages 2022–08, 2022.

[9] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

[10] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34:19822–19835, 2021.

[11] Alexandre Agm Duval, Victor Schmidt, Alex Hernández-Garcı'a, Santiago Miret, Fragkiskos D Malliaros, Yoshua Bengio, and David Rolnick. Faenet: Frame averaging equivariant gnn for materials modeling. In *International Conference on Machine Learning*, pages 9013–9033. PMLR, 2023.

[12] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

[13] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

[14] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[15] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[16] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[17] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Antibody-antigen docking and design via hierarchical equivariant refinement. *arXiv preprint arXiv:2207.06616*, 2022.

[18] Wengong Jin, Siranush Sarkizova, Xun Chen, Nir Hacohen, and Caroline Uhler. Unsupervised protein-ligand binding energy prediction via neural euler's rotation equation. *arXiv preprint arXiv:2301.10814*, 2023.

[19] Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael JL Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. *arXiv preprint arXiv:2009.01411*, 2020.

[20] 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.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. *arXiv preprint arXiv:2206.11990*, 2022.

[23] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[24] Shengchao Liu, Weitao Du, Yanjing Li, Zhuoxinran Li, Zhiling Zheng, Chenru Duan, Zhiming Ma, Omar Yaghi, Anima Anandkumar, Christian Borgs, et al. Symmetry-informed geometric representation for molecules, proteins, and crystalline materials. *arXiv preprint arXiv:2306.09375*, 2023.

[25] Omri Puny, Matan Atzmon, Heli Ben-Hamu, Ishan Misra, Aditya Grover, Edward J Smith, and Yaron Lipman. Frame averaging for invariant and equivariant network design. *arXiv preprint arXiv:2110.03336*, 2021.

[26] Rahmatullah Roche, Bernard Moussad, Md Hossain Shuvo, Sumit Tarafder, and Debswapna Bhattacharya. Equipnas: improved protein-nucleic acid binding site prediction using protein-language-model-informed equivariant deep graph neural networks. *bioRxiv*, pages 2023–09, 2023.

[27] Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

[28] Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24), 2018.

[29] Philipp Thölke and Gianni De Fabritiis. Torchmd-net: equivariant transformers for neural network based molecular potentials. *arXiv preprint arXiv:2202.02541*, 2022.

[30] Raphael Townshend, Rishi Bedi, Patricia Suriana, and Ron Dror. End-to-end learning on 3d protein structure for interface prediction. *Advances in Neural Information Processing Systems*, 32, 2019.

[31] Raphael JL Townshend, Martin Vögele, Patricia Suriana, Alexander Derry, Alexander Powers, Yianni Laloudakis, Sidhika Balachandar, Bowen Jing, Brandon Anderson, Stephan Eismann, et al. Atom3d: Tasks on molecules in three dimensions. *arXiv preprint arXiv:2012.04035*, 2020.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[33] Ying Xia, Chun-Qiu Xia, Xiaoyong Pan, and Hong-Bin Shen. 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.

[34] Dmitry Yarotsky. Universal approximations of invariant maps by neural networks. *Constructive Approximation*, 55(1):407–474, 2022.

[35] Qianmu Yuan, Sheng Chen, Jiahua Rao, Shuangjia Zheng, Huiying Zhao, and Yuedong Yang. Alphafold2-aware protein–dna binding site prediction using graph transformer. *Briefings in Bioinformatics*, 23(2):bbab564, 2022.

A Experimental Detail

Running environment. The experiments are conducted on a single Linux server with The AMD EPYC 7513-32 Core Processor, 1024G RAM, and 8 RTX A5000-24GB. Our method is implemented on PyTorch 1.13.1 and Python 3.9.6.

Training details. For all the baseline models and FAFormer, we fix the batch size as 8, number of layers as 3, and train the model for 50 epochs. Adam [21] with a learning rate of 0.001 is used as the optimizer. The number of nearest neighbors is set as 30 for all the GNN-based methods. Binary cross-entropy loss is used for contact identification tasks with a positive weight of 4. We report the model's performance on the test set using the best-performing model selected based on its performance on the validation set.

Hyperparameters. Here we show the hyperparameters of all the baseline models and FAFormer:

- FAFormer: The hidden size, dropout rate, and attention dropout rate are set as 64, 0.3, and 0.3 respectively. We initialize the weight of the gate module with zero weights, and bias with a constant value of 1, ensuring a mostly-opened gate. GELU [15] is used as the activation function.
- GVP-GNN: The hidden size of node scalar feature, node vector feature, edge scalar feature, and edge vector feature are all set as 32. The dropout rate is fixed as 0.2. For a fair comparison, we only extract the geometric feature based on C_α, i.e., the forward and reverse unit vectors oriented in the direction of C_α between neighbor residues.
- EGNN: The hidden size is set as 64. We apply gate attention to each edge update module and residue connection to the node update module. SiLU [12] is used as the activation function.
- Equiformer&se3Transformer: The hidden size, number of attention heads, and the hidden size of each attention head are set as 64, 4, 16. We exclude the neighbor nodes with a distance greater than 100Å. Based on our experiments, we set the degree of spherical harmonics to 1, as higher degrees tend to lead to performance collapse according to our experiments.
- Transformer&FA+Transformer: We set the hidden size, dropout rate, and attention dropout rate as 64, 0.2, and 0.2. The number of attention heads is set as 1 since we don't observe any improvement with more attention heads.
- GraphBind: The hidden size and dropout ratio are set as 64 and 0.5. We apply addition aggregation to the node and edge update module, following the suggested setting presented in the paper.
- GraphSite: The hidden size and dropout ratio are set as 64 and 0.2. The number of attention layers and attention heads are 2 and 4 respectively. Besides, we additionally use the DSSP features as the node features, as suggested in the paper.