Energy-Adaptive Equivariant State Space Models for Noise-Robust Protein Structure Representation

Zhongyue Zhang* **Runze Ma*** **Yanjie Huang** **Shuangjia Zheng†**
Global Institute of Future Technology, Shanghai Jiao Tong University
Shanghai, China

Abstract

Structure-informed protein representation learning is essential for effective protein function annotation and *de novo* design. With the rapid progress of protein structure prediction and experimental structure determination, a central challenge in the field is shifting from merely obtaining structural models to reliably leveraging predicted or measured structures for downstream learning. However, both crystal and AlphaFold-like predicted structures can contain structural uncertainty, which can bias local neighborhood construction and make robust protein representation learning challenging. To address these issues, we propose a novel equivariant Transformer-State Space Model (SSM) hybrid framework, termed E^3former, designed for efficient and robust protein representation. Our approach uses energy function-based receptive fields to construct proximity graphs that adaptively stabilize neighborhood selection under structural deviations, and incorporates an equivariant high-tensor-elastic selective SSM within the transformer architecture. These components allow the model to adapt to complex geometric interactions and integrate high-order structural features across network depth. Empirical results demonstrate that our model outperforms or matches the compared geometric representation-learning baselines on structure-intensive tasks, such as inverse folding and binding site prediction, with the clearest gains on predicted structures and on harder generalization splits. Our approach offers a novel perspective for conducting biological function research and drug discovery using imperfect but increasingly available protein structure data. Our code is available on `https://github.com/GENTEL-lab/E3former`.

1 Introduction

A wide range of downstream tasks, such as protein interaction prediction, functional annotation, inverse folding, and protein binder design, depend critically on robust structure-informed representations [39, 6, 24]. With recent progress in both experimental structure measurement and protein structure prediction, the central challenge is increasingly shifting from obtaining structural models to reliably using predicted or measured structures for downstream learning.

Recent advances in experimental techniques, along with progress in protein structure prediction and related modeling systems such as AlphaFold [19, 1] and other AlphaFold-like models, including Boltz [41], Protenix [35], and AuroBind [43], have greatly expanded access to 3D protein structures. This progress makes robustness a downstream bottleneck: many models now receive structural inputs, but those inputs are not always equally reliable. Both measured and predicted structures can contain structural uncertainty, which may affect geometry-driven learning [2, 26]. It is therefore important to

*Equal contribution.
†Corresponding author: `shuangjia.zheng@sjtu.edu.cn`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

develop geometric learning methods that can use imperfect structures without being overly sensitive to local coordinate deviations.

Equivariant neural networks have shown strong capabilities for learning from 3D protein structures [9]. However, many residue-level geometric encoders, including EGNN [29], GearNet [44], spherical harmonics-based architectures such as Equiformer [21], and protein-specific models such as GCPNet [27], still build message-passing neighborhoods from the input coordinates. In this setting, coordinate deviations can affect not only continuous geometric features, but also the discrete graph on which message passing is performed. For example, a residue close to a cutoff boundary may enter or leave the local neighborhood after a small coordinate change, altering the structural context available to the node before any equivariant update is applied. Figure 1 illustrates this coordinate-to-graph instability.

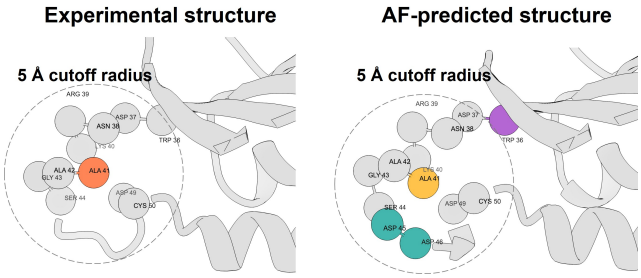

Figure 1: Coordinate-to-graph instability in chain A of 3R2E under a 5 Å cutoff radius. Left: experimental crystal structure. Right: corresponding AlphaFold-predicted structure for the same query residue. Green residues are gained neighbors in the AlphaFold graph, while purple residues are lost neighbors relative to the crystal-structure cutoff graph.

Here we introduce E^3former, an Energy-Adaptive Elastic Equivariant Transformer–SSM hybrid architecture for robust structure-informed protein representation learning. It combines an energy-aware adaptive radius graph for stable residue-neighborhood construction with an elastic SSM that selectively integrates high-order equivariant features into compact scalar representations. We evaluate E^3former on paired benchmark variants built from experimentally determined structures and AlphaFold-predicted structures, focusing on structure-sensitive tasks including inverse folding [16] and binding-site prediction [11].

Our contributions are summarized as follows: (1) Inspired by molecular interaction potentials [12], we introduce an **energy-aware adaptive radius function** with a **radius sampler** that adjusts receptive fields according to local structural context, reducing reliance on a single fixed cutoff in protein proximity graph construction. (2) We propose a **novel equivariant elastic selective SSM** that allows high-order equivariant interactions while compressing selected intermediate tensor features into type-0 equivariant summaries before downstream prediction. (3) We construct new **AlphaFold-predicted variants** of established CATH and PPBS benchmarks, enabling paired evaluation of protein representation models on measured and predicted structural inputs. (4) Experiments show competitive performance among the compared representation-learning baselines, with the clearest gains on AlphaFold-predicted structures and harder binding-site generalization splits. On CATH-AF under +Seq, E^3former reduces perplexity by **9.7%** relative to the strongest compared baseline. Collectively, these results suggest that adaptive neighborhood construction and controlled equivariant tensor integration are useful ingredients for learning from imperfect but increasingly available protein structures.

2 Related Work

2.1 Structure-based Protein Representation Learning

With progress in experimental structure determination and structure prediction, protein structure data have expanded rapidly [19, 42, 4]. This has motivated a broad range of structure-based representation learning methods, while also making the reliable use of predicted and measured structures increasingly important.

Voxelized representations map 3D protein structures into volumetric grids and encode them with 3D convolutional neural networks. Methods such as [28, 3, 25] demonstrate the utility of voxelized structural representations across diverse tasks. Metal3D [8] further enriches this representation by incorporating physical and chemical properties.

Graph-structured representations model proteins as proximity graphs over residues or atoms, and use graph neural networks to capture structural interactions [15]. GVP [18] performs equivariant message passing for protein design and model quality assessment. SchNet [30] and ScanNet [39] integrate spatial and chemical features within graph-based frameworks, while EGNN [29], GearNet-Edge [44], GCPNet [27], and related models further improve geometric protein representation learning. Despite these advances, most graph-based encoders construct local neighborhoods directly from input coordinates, typically using fixed cutoff or nearest-neighbor rules. This makes graph construction itself an important design choice when structural inputs are imperfect.

Structure-aware language models. ESMFold [23] uses protein language-model representations to predict atomic structures directly from sequence. SaProt [34] combines residue identities with Foldseek-derived 3Di structure tokens, masks structural tokens in low-confidence regions, and also evaluates representations derived from ESMFold-predicted structures. These methods complement our focus on adaptive neighborhood construction and equivariant feature integration over continuous coordinates.

2.2 Equivariant Neural Networks

Equivariant neural networks have achieved strong performance in modeling 3D atomic systems, including small molecules and biological macromolecules [10, 5, 21, 22, 38, 6]. Cartesian-based equivariant models operate on vectors or Cartesian tensors while preserving equivariance. For example, GVP [18] separates protein features into scalar and vector channels, TorchMD-Net [36] combines equivariance with transformer-style message passing, and TensorNet [32] uses Cartesian rank-2 tensor representations for efficient equivariant message passing.

Spherical harmonics-based models instead use irreducible representations to decompose geometric features by tensor degree. TFN [37] constructs filters from spherical harmonics, SE(3)-Transformer [10] adapts this framework to attention, and Equiformer [21] and EquiformerV2 [22] incorporate equivariant graph attention into Transformer blocks.

These methods improve the expressivity of 3D geometric modeling, but they also introduce a design question: how should intermediate tensor features be reused for downstream prediction? Higher-order equivariant interactions can provide useful geometric information, but effective use of these features requires compact aggregation and compression. This motivates architectures that preserve high-order interactions while selectively integrating them into the final representation.

3 E^{3}former

In this section, we introduce an adaptive **E**nergy-aware **E**lastic **E**quivariant Transformer-SSM hybrid architecture, termed E^3former. The energy-aware radius graph module will be detailed in Section 3.1 , while the high-order tensor elastic compression SSM module and the discussion of equivariance will be described in Section 3.2. Section 3.3 summarizes the overall architecture and equivariant operations. Figure 2 provides an overview.

3.1 Energy-aware Protein Graph

We represent a protein as a residue-level graph $G = (V, E)$, where each node is an amino acid and edges define local structural neighborhoods. A standard cutoff graph with a top-k constraint is

$$\begin{aligned} E_c &= \{e_{j\to i} : i \neq j,\ j \in \mathcal{N}_c(i)\}, \\ \mathcal{N}_c(i) &= \{j : j \neq i,\ d_{ij} \leq r_c,\ j \in \mathcal{N}_{\text{top-}k}(i)\}. \end{aligned} \tag{1}$$

where d_{ij} is the distance between residues i and j, and $\mathcal{N}_{\text{top-}k}(i)$ keeps the closest k candidates. This locality prior is useful, but a fixed cutoff can be brittle: small coordinate deviations may add or remove residues near the boundary, changing the structural context used for message passing, as illustrated in Figure 2(a).

Energy-aware radius function. To stabilize neighborhood selection under coordinate perturbation, we propose an energy-aware radius graph that adaptively adjusts the receptive field using a simple local energy proxy. Intuitively, it expands the field when informative neighbors may be displaced and shrinks it when neighborhoods are locally dense, reducing sensitivity to local graph flips without

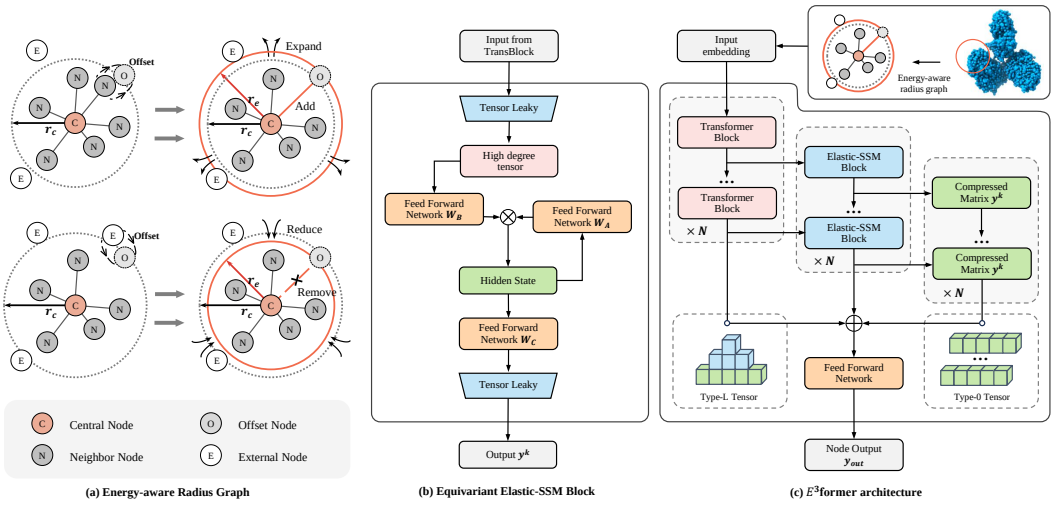

Figure 2: Overview of E^3former. (a) The energy-aware radius graph adjusts residue neighborhoods using a distance-derived local structural score. (b) The equivariant Elastic-SSM block integrates selected high-order tensor features across Transformer depth and projects them into type-0 summaries. (c) The full architecture combines adaptive graph construction, equivariant Transformer blocks, and elastic feature integration.

introducing a task-specific heuristic. We use a Lennard-Jones potential form as a generic interaction proxy:

$$\mathcal{E}_i = \sum_{j \in \mathcal{N}_c(i)} \epsilon_c \left[\left(\frac{\sigma}{d_{ij}} \right)^{12} - \left(\frac{\sigma}{d_{ij}} \right)^{6} \right], \tag{2}$$

where $\mathcal{E}_i$ is the neighbor-summed local energy proxy of node i. We use this Lennard-Jones-shaped form as an invariant structural score rather than a calibrated physical energy. We keep ϵ_c and σ fixed across tasks to avoid over-tuning.

Energy-based radius sampler. We first normalize the local energy within each protein:

$$\mathcal{E}_i^{norm} = \frac{\log\left(\mathcal{E}_i - \mathcal{E}_{min} + 1\right)}{\log\left(\mathcal{E}_{max} - \mathcal{E}_{min} + 1\right) + \varepsilon}, \qquad \mathcal{E}_i^{norm} \in [0, 1]. \tag{3}$$

Here, $\mathcal{E}_{min}$ and $\mathcal{E}_{max}$ are the minimum and maximum local energy values within the protein, the logarithm smooths the energy scale across residues, and $\varepsilon > 0$ is a numerical-stability constant.

Based on the normalized energy $\mathcal{E}_i^{norm}$, the energy-driven sampler produces a normalized adaptive radius:

$$r_i^{\mathcal{E}} \sim \mathrm{Beta}\left(\alpha + \beta(1 - \mathcal{E}_i^{norm}), \alpha + \beta \mathcal{E}_i^{norm}\right). \tag{4}$$

This parameterization biases residues with lower normalized local energy toward larger receptive fields, while residues with higher normalized local energy receive more conservative neighborhoods. The reported experiments use sampled radii during both training and inference. The parameters α and β are shared across tasks. Appendix B.3 quantifies graph instability, and Appendix B.1 reports controlled graph and radius-sampling comparisons.

3.2 Equivariant high-tensor-elastic selective SSM

In equivariant architectures built on irreducible representations (irreps), node features are organized by tensor degree L: $L = 0$ are scalars, $L = 1$ are vectors, and larger L correspond to higher-order spherical tensors [21]. Higher-order channels increase expressivity and help represent complex 3D geometric interactions. However, when the input structure is imperfect, the intermediate tensor features are also derived from uncertain coordinates and local neighborhoods. Directly propagating all high-order intermediate channels to the final prediction head can therefore make the representation high-capacity and difficult to regularize.

Algorithm 1 Energy-based radius sampling process

input: coordinate set $X = \{x_1, x_2, ..., x_N\}$, cut-off radius r_c, max neighbors number k
Initialize energy set $\mathcal{E} \leftarrow \emptyset$
Initialize energy radius set $\mathcal{R} \leftarrow \emptyset$
for $i = 1$ **to** N **do**
  $\mathcal{N}_c(i) = \{j : j \neq i,\ d_{ij} \leq r_c,\ j \in \mathcal{N}_{\text{top-}k}(i)\}$.
  $\mathcal{E}_i = \sum_{j \in \mathcal{N}_c(i)} \epsilon_c \left[\left(\frac{\sigma}{d_{ij}} \right)^{12} - \left(\frac{\sigma}{d_{ij}} \right)^{6} \right]$.
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{\mathcal{E}_i\}$.
end for
for $i = 1$ **to** N **do**
  $\mathcal{E}_i^{norm} = \frac{\log(\mathcal{E}_i - \mathcal{E}_{min} + 1)}{\log(\mathcal{E}_{max} - \mathcal{E}_{min} + 1) + \varepsilon}$.
  $r_i^{\mathcal{E}} \sim \text{Beta}\left(\alpha + \beta(1 - \mathcal{E}_i^{norm}), \alpha + \beta \mathcal{E}_i^{norm}\right)$.
  $\mathcal{R} \leftarrow \mathcal{R} \cup \{r_i^{\mathcal{E}}\}$.
end for
return $\mathcal{R}$

To address this expressivity–robustness tension, we introduce an equivariant elastic high-tensor SSM block. The key idea is to treat selected high-order components extracted from successive Transformer layers as a sequence along depth, and then apply a parameter-sharing state space model to integrate them selectively. This provides a controlled path for high-order equivariant information: the Transformer blocks are allowed to compute expressive high-order interactions, while the elastic SSM compresses selected intermediate tensor features into compact type-0 geometric summaries before downstream prediction. The interaction between the Transformer and SSM blocks is shown in Figure 2(c).

High-tensor leaky layers. Let T_k^{out} denote the output of the k-th equivariant Transformer layer. We form a depth-wise sequence from the non-final Transformer layers and pass each element through a high-tensor leaky layer to obtain the SSM input:

$$X_k^{in} = L^{leaky}_{(L_{min}, L_{max})}\left(T_k^{out}\right), \qquad k = 1, \ldots, N-1. \tag{5}$$

$L^{leaky}_{(L_{min}, L_{max})}$ keeps only tensor degrees L_{min} through L_{max}, isolating the selected high-order channels for elastic SSM filtering.

Equivariant elastic selective SSM. State space models provide a recurrent framework for sequence modeling [14]. A basic continuous formulation maps an input sequence $x(t)$ and hidden state $h(t)$ to output $y(t)$:

$$\begin{aligned} h'(t) &= \boldsymbol{A}h(t) + \boldsymbol{B}x(t), \\ y(t) &= \boldsymbol{C}h(t), \end{aligned} \tag{6}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are the state, input, and output matrices. In the elastic SSM module, we discretize the sequence over Transformer depth and use $\{X_k^{in}\}_{k=1}^{N-1}$ as input:

$$\begin{aligned} h_k &= \bar{\boldsymbol{A}} h_{k-1} + \bar{\boldsymbol{B}} X_k^{in}, \\ y_k &= \bar{\boldsymbol{C}} h_k. \end{aligned} \tag{7}$$

We further replace $\bar{\boldsymbol{A}}$, $\bar{\boldsymbol{B}}$, and $\bar{\boldsymbol{C}}$ with parameter-sharing equivariant tensors $\boldsymbol{W}_A$, $\boldsymbol{W}_B$, and $\boldsymbol{W}_C$:

$$\begin{aligned} h_k &= (\boldsymbol{W}_A \otimes_{dtp} h_{k-1}) + (\boldsymbol{W}_B \otimes_{dtp} X_k^{in}), \\ y_k &= L^{leaky}_{(0,0)}(\boldsymbol{W}_C \otimes_{dtp} h_k). \end{aligned} \tag{8}$$

The depth-wise tensor product $\otimes_{dtp}$ defines the output tensor channels while preserving equivariance [21]. The $L^{leaky}_{(0,0)}$ operation projects the SSM output into type-0 geometric summaries. The summaries from multiple depths are concatenated with the final Transformer output:

$$\boldsymbol{Y} = (y_1 \| y_2 \| \cdots \| y_{N-1}), \qquad \boldsymbol{Z} = (\boldsymbol{Y} \| \boldsymbol{T}_N), \tag{9}$$

where $\|$ denotes concatenation. The fused representation $\boldsymbol{Z}$ is then used for node-level prediction.

Proof of Equivariance. We briefly verify that the tensor-leaky and elastic SSM operations preserve equivariance. For a mapping $F : X \to Y$ between tensor spaces X and Y, given group representations $D_X(g)$ and $D_Y(g)$, equivariance requires $D_Y(g)f(x) = f(D_X(g)x)$ for any $g \in E(3)$. For

$$f(x) = L^{leaky}_{(L_1,L_2)}(\boldsymbol{W}_C \otimes_{dtp} x), \tag{10}$$

the tensor product is equivariant by construction in e3nn-style irrep operations [13]:

$$\boldsymbol{W}_C \otimes_{dtp} D_X(g)x = D_Y(g)y'. \tag{11}$$

The operation $L^{leaky}_{(L_1,L_2)}$ selects tensor channels from L_1 to L_2. Since channel selection is independent of rotation and translation, equivariance is preserved:

$$L^{leaky}_{(L_1,L_2)}(D_Y(g)y') = D_Y(g)y. \tag{12}$$

Thus the elastic SSM preserves equivariance, and $L^{leaky}_{(0,0)}$ produces type-0 invariant summaries for node-level prediction.

Table 1: Performance on AlphaFold-predicted structures. Values are mean ± standard deviation. A dash denotes a run that did not converge. PPBS-AF All combines the four test partitions.

	Features	SchNet	TFN	EGNN	Equiformer	GearNet	GCPNet	E^3**former**
CATH-AF(↓)	+Seq	9.80 ± 0.09	7.30 ± 0.04	8.32 ± 0.05	6.09 ± 0.04	-	6.20 ± 0.08	$\mathbf{5.50 \pm 0.05}$
	$+\kappa, \alpha$	8.71 ± 0.07	7.28 ± 0.04	7.72 ± 0.06	5.62 ± 0.03	-	6.19 ± 0.09	$\mathbf{5.44 \pm 0.04}$
	$+\phi, \psi, \omega$	7.32 ± 0.08	5.11 ± 0.02	6.07 ± 0.05	3.64 ± 0.03	-	3.91 ± 0.07	$\mathbf{3.54 \pm 0.04}$
PPBS-AF(↑) 70	+Seq	0.520 ± 0.01	0.507 ± 0.01	0.526 ± 0.00	0.596 ± 0.02	0.597 ± 0.02	0.554 ± 0.02	$\mathbf{0.607 \pm 0.01}$
	$+\kappa, \alpha$	0.532 ± 0.02	0.574 ± 0.01	0.557 ± 0.00	0.573 ± 0.00	$\mathbf{0.607 \pm 0.02}$	0.567 ± 0.00	0.560 ± 0.00
	$+\phi, \psi, \omega$	0.554 ± 0.01	0.591 ± 0.01	0.569 ± 0.01	0.592 ± 0.01	0.591 ± 0.01	0.549 ± 0.02	$\mathbf{0.598 \pm 0.01}$
PPBS-AF(↑) Homology	+Seq	0.398 ± 0.03	0.487 ± 0.01	0.483 ± 0.00	0.537 ± 0.00	0.538 ± 0.02	0.422 ± 0.03	$\mathbf{0.546 \pm 0.01}$
	$+\kappa, \alpha$	0.403 ± 0.00	0.507 ± 0.02	0.503 ± 0.04	$\mathbf{0.547 \pm 0.03}$	0.536 ± 0.02	0.419 ± 0.01	0.525 ± 0.01
	$+\phi, \psi, \omega$	0.428 ± 0.01	0.516 ± 0.00	0.518 ± 0.00	0.547 ± 0.04	0.541 ± 0.00	0.425 ± 0.02	$\mathbf{0.551 \pm 0.02}$
PPBS-AF(↑) Topology	+Seq	0.408 ± 0.02	0.547 ± 0.00	0.570 ± 0.00	0.595 ± 0.01	0.520 ± 0.00	0.403 ± 0.03	$\mathbf{0.599 \pm 0.00}$
	$+\kappa, \alpha$	0.410 ± 0.03	0.566 ± 0.00	0.574 ± 0.00	0.590 ± 0.00	0.530 ± 0.00	0.411 ± 0.00	$\mathbf{0.598 \pm 0.01}$
	$+\phi, \psi, \omega$	0.416 ± 0.01	0.542 ± 0.00	0.602 ± 0.02	0.588 ± 0.00	0.525 ± 0.01	0.406 ± 0.01	$\mathbf{0.602 \pm 0.00}$
PPBS-AF(↑) None	+Seq	0.269 ± 0.00	0.383 ± 0.01	0.392 ± 0.00	0.394 ± 0.00	0.366 ± 0.00	0.269 ± 0.00	$\mathbf{0.405 \pm 0.01}$
	$+\kappa, \alpha$	0.264 ± 0.00	0.371 ± 0.00	0.384 ± 0.00	0.385 ± 0.00	0.357 ± 0.00	0.263 ± 0.00	$\mathbf{0.409 \pm 0.01}$
	$+\phi, \psi, \omega$	0.274 ± 0.01	0.357 ± 0.00	0.407 ± 0.01	0.394 ± 0.01	0.366 ± 0.00	0.271 ± 0.00	$\mathbf{0.412 \pm 0.00}$
PPBS-AF(↑) All	+Seq	0.390 ± 0.02	0.483 ± 0.03	0.494 ± 0.01	0.531 ± 0.01	0.509 ± 0.01	0.399 ± 0.02	$\mathbf{0.541 \pm 0.01}$
	$+\kappa, \alpha$	0.390 ± 0.02	0.503 ± 0.03	0.504 ± 0.01	0.520 ± 0.01	0.509 ± 0.02	0.401 ± 0.03	$\mathbf{0.525 \pm 0.03}$
	$+\phi, \psi, \omega$	0.410 ± 0.03	0.504 ± 0.01	0.526 ± 0.03	0.533 ± 0.00	0.513 ± 0.03	0.406 ± 0.02	$\mathbf{0.541 \pm 0.02}$

3.3 Architecture of E^{3}former

The E^3former architecture comprises an energy graph module (Section 3.1), an equivariant Elastic-SSM module (Section 3.2), and an $SE(3)$-equivariant Transformer module [21]. In the protein graph $G = (V, E)$, each node $v_i \in V$ corresponds to an amino acid. To reduce computational cost, we adopt a C_α-based residue graph and incorporate structural features following prior protein representation benchmarks [17]. Each node v_i has a feature encoding h_v^i, including:

- Residue type for binding-site prediction; residue identity is not used as an input for inverse folding.
- A 16-dimensional transformer-style positional encoding [40].
- Backbone dihedral angles $\phi, \psi, \omega \in \mathbb{R}^6$.
- Virtual torsion and bond angles $\kappa, \alpha \in \mathbb{R}^4$ defined over C_α atoms.
- Edge displacement vector $\vec{r}_{ij}$ between connected residues.

During training and inference, the energy-driven radius sampler establishes the adaptive adjacency E. The resulting graph and protein features are processed by the equivariant Transformer blocks to produce intermediate representations $\{T_k\}_{k=1}^N$. For non-final blocks, the tensor-leaky operation extracts selected tensor channels as Elastic-SSM inputs X_k. The Elastic-SSM outputs $\{y_k\}_{k=1}^{N-1}$ are concatenated with the final Transformer representation T_N, and the fused representation is mapped to the final node output H_{out}.

Table 2: Performance on experimental structures. Values are mean ± standard deviation.

	Features	SchNet	TFN	EGNN	Equiformer	GearNet	GCPNet	E^3**former**
CATH(↓)	+Seq	11.78 ± 0.08	10.34 ± 0.03	10.28 ± 0.04	7.80 ± 0.03	12.79 ± 0.17	8.35 ± 0.08	$\mathbf{7.53 \pm 0.02}$
	$+\kappa, \alpha$	11.03 ± 0.03	10.02 ± 0.05	9.84 ± 0.07	8.09 ± 0.00	12.35 ± 0.05	8.80 ± 0.09	$\mathbf{7.97 \pm 0.02}$
	$+\phi, \psi, \omega$	9.97 ± 0.09	8.73 ± 0.02	8.89 ± 0.04	6.91 ± 0.04	11.61 ± 0.12	7.56 ± 0.11	$\mathbf{6.64 \pm 0.03}$
PPBS(↑) 70	+Seq	0.648 ± 0.01	$\mathbf{0.781 \pm 0.01}$	0.721 ± 0.00	0.766 ± 0.01	0.768 ± 0.01	0.741 ± 0.02	0.777 ± 0.01
	$+\kappa, \alpha$	0.660 ± 0.00	$\mathbf{0.783 \pm 0.00}$	0.739 ± 0.01	0.771 ± 0.03	0.765 ± 0.03	0.739 ± 0.00	0.775 ± 0.01
	$+\phi, \psi, \omega$	0.658 ± 0.01	0.783 ± 0.02	0.752 ± 0.02	0.780 ± 0.02	$\mathbf{0.787 \pm 0.02}$	0.747 ± 0.01	0.783 ± 0.01
PPBS(↑) Homology	+Seq	0.552 ± 0.03	0.674 ± 0.01	0.680 ± 0.02	0.724 ± 0.02	0.688 ± 0.00	0.692 ± 0.03	$\mathbf{0.732 \pm 0.02}$
	$+\kappa, \alpha$	0.569 ± 0.00	0.697 ± 0.02	0.690 ± 0.01	0.721 ± 0.00	0.688 ± 0.01	0.691 ± 0.02	$\mathbf{0.727 \pm 0.02}$
	$+\phi, \psi, \omega$	0.568 ± 0.03	0.694 ± 0.00	0.702 ± 0.02	0.727 ± 0.03	0.692 ± 0.01	0.702 ± 0.00	$\mathbf{0.732 \pm 0.01}$
PPBS(↑) Topology	+Seq	0.532 ± 0.03	0.651 ± 0.03	0.716 ± 0.02	0.725 ± 0.01	0.636 ± 0.00	0.713 ± 0.01	$\mathbf{0.743 \pm 0.02}$
	$+\kappa, \alpha$	0.530 ± 0.02	0.675 ± 0.00	0.718 ± 0.03	0.716 ± 0.02	0.642 ± 0.01	0.716 ± 0.03	$\mathbf{0.745 \pm 0.00}$
	$+\phi, \psi, \omega$	0.519 ± 0.01	0.668 ± 0.00	0.733 ± 0.02	0.728 ± 0.01	0.640 ± 0.03	0.721 ± 0.02	$\mathbf{0.743 \pm 0.03}$
PPBS(↑) None	+Seq	0.455 ± 0.02	0.542 ± 0.04	0.596 ± 0.05	0.618 ± 0.04	0.540 ± 0.01	0.588 ± 0.04	$\mathbf{0.637 \pm 0.03}$
	$+\kappa, \alpha$	0.464 ± 0.01	0.567 ± 0.04	0.602 ± 0.04	0.604 ± 0.04	0.540 ± 0.03	0.581 ± 0.00	$\mathbf{0.628 \pm 0.02}$
	$+\phi, \psi, \omega$	0.448 ± 0.03	0.564 ± 0.02	0.598 ± 0.01	0.615 ± 0.02	0.548 ± 0.03	0.589 ± 0.00	$\mathbf{0.631 \pm 0.01}$
PPBS(↑) All	+Seq	0.537 ± 0.02	0.654 ± 0.01	0.674 ± 0.04	0.708 ± 0.03	0.656 ± 0.00	0.676 ± 0.02	$\mathbf{0.724 \pm 0.02}$
	$+\kappa, \alpha$	0.546 ± 0.02	0.674 ± 0.00	0.684 ± 0.03	0.703 ± 0.01	0.658 ± 0.01	0.669 ± 0.04	$\mathbf{0.719 \pm 0.02}$
	$+\phi, \psi, \omega$	0.539 ± 0.00	0.672 ± 0.00	0.692 ± 0.01	0.712 ± 0.03	0.660 ± 0.01	0.685 ± 0.01	$\mathbf{0.726 \pm 0.02}$

4 Experiments

4.1 Dataset

We evaluate E^3former on inverse folding and binding-site prediction using experimental structures and their AlphaFold-predicted counterparts. Dataset-construction and split details are provided in Appendix A.1.

CATH. We provide the dataset derived from CATH 4.2 [16], in which all protein structures with 40% nonredundancy are partitioned by their CATH (class, architecture, topology/fold, homologous superfamily) and kept the same data settings as a benchmark of protein representation learning [17]. These data are split based on random assignment of the CATH topology classifications based on an 80/10/10 split.

CATH-AF. We replace experimental structural inputs with corresponding AlphaFold-predicted structures retrieved from the AlphaFold Protein Structure Database. Construction details are provided in Appendix A.1.

PPBS. We provide the dataset derived from PPBS (prediction of protein–protein binding sites dataset) curated by ScanNet [39], which constructed a nonredundant dataset of 20K representative protein chains with annotated binding sites derived from the Dockground database of protein complexes [20].

PPBS-AF. We retrieve corresponding AlphaFold-predicted structures and transfer the experimental binding-site labels by sequence alignment. Construction details are provided in Appendix A.1.

4.2 AlphaFold-predicted dataset tasks

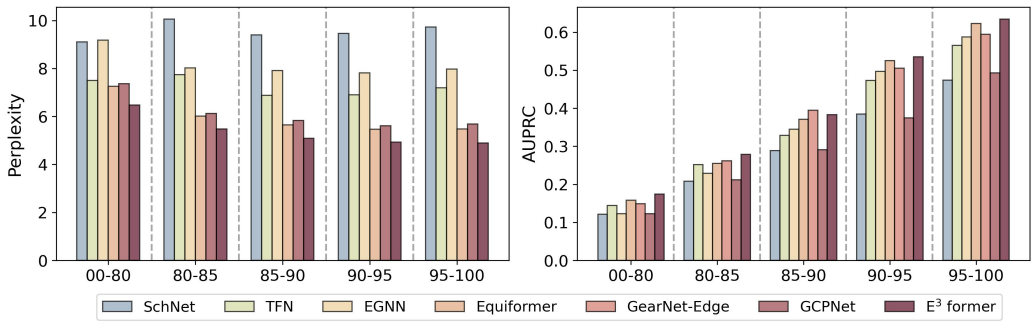

Figure 3: Performance stratified by AlphaFold confidence. Bins contain different proteins rather than repeated perturbations of the same inputs. Sample counts are reported in Table S7. Left: CATH-AF, perplexity↓. Right: PPBS-AF All, AUPRC↑.

We compare E^3former with representative geometric protein representation-learning baselines following the benchmark of Jamasb et al. [17]: SchNet [30], TFN [37], EGNN [29], Equiformer [21], GearNet [44], and GCPNet [27]. We use the benchmark's default settings for most baselines. Task-specific comparisons with ProteinMPNN and ScanNet, together with their input and evaluation settings, are reported in Appendix B.2.

To evaluate robustness under predicted structures, we run all methods on AlphaFold-based datasets. We report perplexity for inverse folding [18] and AUPRC for binding site prediction due to label imbalance.

Table 1 shows that E^3former achieves the lowest inverse-folding perplexity across the three feature settings on CATH-AF. Under +Seq, it reduces PPL from 6.09 for the strongest compared baseline to 5.50, a 9.7% relative reduction. On PPBS-AF, the gains vary across feature settings and generalization splits, with competitive results on the harder Topology and None splits.

4.3 Experimental dataset tasks

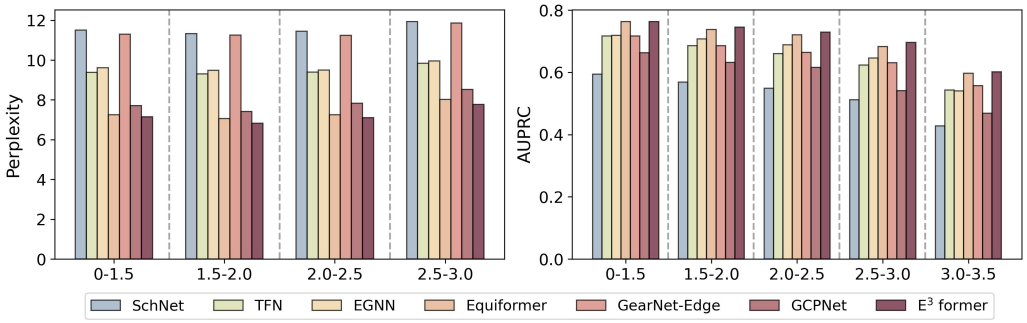

Figure 4: Performance stratified by crystallographic resolution (Å). Resolution is a property of the structural measurement, not a directly controlled coordinate-error magnitude. The five nonempty CATH bins contain 102, 383, 298, 114, and 34 proteins; all bin counts are reported in Table S7. Left: CATH, perplexity↓. Right: PPBS All, AUPRC↑.

Table 2 shows that E^3former obtains the lowest CATH perplexity in all three feature settings, including a 3.5% reduction over Equiformer under +Seq. On PPBS, it leads on the Homology, Topology, None, and All splits, while other methods match or exceed it on the 70 split. Adding κ, α features does not consistently improve performance, as also observed by Jamasb et al. [17].

4.4 Evaluation across structural confidence and resolution

To examine performance across structural quality levels, we stratify AlphaFold-predicted test structures by confidence and experimental test structures by crystallographic resolution (Figures 3 and 4). The results show competitive performance on lower-confidence predicted structures and lower-resolution experimental structures. Each bin contains different proteins, so the observed trends also reflect protein composition and sample size; crystallographic resolution is not a direct measure of coordinate error. The five nonempty CATH resolution bins contain 102, 383, 298, 114, and 34 proteins, respectively (Table S7).

4.5 Ablation Study

In this section, we ablate the two core components of E^3former: the energy-aware graph module and the elastic selective SSM. For the graph ablation, we replace the adaptive energy-aware graph with a standard radius-cutoff k-NN graph. For the SSM ablation, we remove the elastic selective SSM and use the equivariant Transformer backbone alone.

Table 3 reports component ablations on predicted structures. Experimental-structure ablations are reported in Appendix B.8. Removing either module generally degrades performance. Additional controlled comparisons in Appendix B.1 show that adaptive neighborhoods improve over a fixed radius and that Elastic-SSM benefits from intermediate high-order features. The gains specific to

Table 3: Component ablations on AlphaFold-predicted structures. Values are mean ± standard deviation.

Method	Features	CATH-AF(↓)	PPBS-AF(↑)				
			70	Homology	Topology	None	All
E^3former	+Seq	$\mathbf{5.50 \pm 0.05}$	$\mathbf{0.607 \pm 0.01}$	$\mathbf{0.546 \pm 0.01}$	$\mathbf{0.599 \pm 0.00}$	$\mathbf{0.405 \pm 0.01}$	$\mathbf{0.541 \pm 0.01}$
	$+\kappa, \alpha$	$\mathbf{5.44 \pm 0.04}$	0.560 ± 0.00	0.525 ± 0.01	$\mathbf{0.598 \pm 0.01}$	$\mathbf{0.409 \pm 0.01}$	$\mathbf{0.525 \pm 0.03}$
	$+\phi, \psi, \omega$	$\mathbf{3.54 \pm 0.04}$	$\mathbf{0.598 \pm 0.01}$	$\mathbf{0.551 \pm 0.02}$	$\mathbf{0.602 \pm 0.00}$	$\mathbf{0.412 \pm 0.00}$	$\mathbf{0.541 \pm 0.02}$
w/o Energy	+Seq	5.82 ± 0.06	0.603 ± 0.02	0.542 ± 0.03	0.597 ± 0.01	0.399 ± 0.02	0.533 ± 0.03
	$+\kappa, \alpha$	5.61 ± 0.03	0.571 ± 0.01	0.528 ± 0.02	0.593 ± 0.03	0.402 ± 0.01	0.523 ± 0.02
	$+\phi, \psi, \omega$	3.58 ± 0.03	0.597 ± 0.02	0.549 ± 0.01	0.598 ± 0.02	0.404 ± 0.01	0.536 ± 0.03
w/o Elastic	+Seq	5.71 ± 0.04	0.601 ± 0.02	0.543 ± 0.03	0.596 ± 0.01	0.403 ± 0.02	0.536 ± 0.02
	$+\kappa, \alpha$	5.52 ± 0.04	0.563 ± 0.01	0.541 ± 0.02	0.596 ± 0.03	0.405 ± 0.01	0.525 ± 0.02
	$+\phi, \psi, \omega$	$\mathbf{3.54 \pm 0.02}$	0.593 ± 0.02	0.549 ± 0.01	0.595 ± 0.02	0.407 ± 0.02	0.539 ± 0.03
w/o EE	+Seq	6.09 ± 0.01	0.596 ± 0.02	0.537 ± 0.03	0.595 ± 0.01	0.394 ± 0.02	0.531 ± 0.02
	$+\kappa, \alpha$	5.62 ± 0.05	$\mathbf{0.573 \pm 0.01}$	$\mathbf{0.547 \pm 0.02}$	0.590 ± 0.03	0.385 ± 0.01	0.520 ± 0.02
	$+\phi, \psi, \omega$	3.64 ± 0.05	0.592 ± 0.02	0.547 ± 0.01	0.588 ± 0.02	0.394 ± 0.01	0.533 ± 0.03

the Lennard-Jones score over inverse-distance adaptation and to Elastic-SSM over attention-based aggregation are modest. Computational profiling is reported in Appendix A.3.

5 Conclusion and Limitations

In this work, we proposed E^3former, an energy-aware elastic equivariant Transformer–SSM hybrid architecture for protein structure representation learning under imperfect structural inputs. Our key perspective is that reliable use of imperfect structures requires addressing two connected issues: coordinate deviations from experimental uncertainty or prediction errors can destabilize local neighborhood selection in proximity graphs, and high-order equivariant interactions, while expressive, should be integrated into the final representation in a controlled way. E^3former addresses these issues by combining an energy-aware graph module that adaptively stabilizes receptive fields with an equivariant elastic selective SSM that filters and compresses high-order features via parameter-sharing equivariant operations, supporting representation learning from imperfect structural inputs. Experiments on structure-sensitive benchmarks show strong performance on both predicted and experimental structures, suggesting that targeting neighborhood instability and high-order feature fragility is an effective route to reliable protein representation learning in the post-AlphaFold era.

The main limitation of this work is its residue-level structural scope. E^3former does not yet fully incorporate richer multimodal information, such as sequence annotations, chemical bond details, or protein functional labels, and it has not been evaluated on atom-level side-chain-sensitive tasks, ligand-specific modeling, or multi-chain complex generation.

Acknowledgments and Disclosure of Funding

Support for this work was provided through the National Natural Science Foundation of China (Grant No. 62041209), the Natural Science Foundation of Shanghai (Grant No. 24ZR1440600), and the Science and Technology Commission of Shanghai Municipality (Grant No. 24510714300). The authors declare no competing interests.

References

[1] 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*, 630(8016):493–500, 2024.

[2] K Ravi Acharya and Matthew D Lloyd. The advantages and limitations of protein crystal structures. *Trends in pharmacological sciences*, 26(1):10–14, 2005.

[3] Namrata Anand, Raphael Eguchi, Irimpan I Mathews, Carla P Perez, Alexander Derry, Russ B Altman, and Po-Ssu Huang. Protein sequence design with a learned potential. *Nature communications*, 13(1):746, 2022.

[4] 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.

[5] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1): 2453, 2022.

[6] Anton Bushuiev, Roman Bushuiev, Anatolii Filkin, Petr Kouba, Marketa Gabrielova, Michal Gabriel, Jiri Sedlar, Tomas Pluskal, Jiri Damborsky, Stanislav Mazurenko, et al. Learning to design protein-protein interactions with enhanced generalization. *arXiv preprint arXiv:2310.18515*, 2023.

[7] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[8] Simon L Dürr, Andrea Levy, and Ursula Rothlisberger. Metal3d: a general deep learning framework for accurate metal ion location prediction in proteins. *Nature Communications*, 14 (1):2713, 2023.

[9] Alexandre Duval, Simon V Mathis, Chaitanya K Joshi, Victor Schmidt, Santiago Miret, Fragkiskos D Malliaros, Taco Cohen, Pietro Lio, Yoshua Bengio, and Michael Bronstein. A hitchhiker's guide to geometric gnns for 3d atomic systems. *arXiv preprint arXiv:2312.07511*, 2023.

[10] 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.

[11] Pablo Gainza, Freyr Sverrisson, Frederico Monti, Emanuele Rodola, Davide Boscaini, Michael M Bronstein, and Bruno E Correia. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. *Nature Methods*, 17(2):184–192, 2020.

[12] Isidro Lorenzo Geada, Hadi Ramezani-Dakhel, Tariq Jamil, Marialore Sulpizi, and Hendrik Heinz. Insight into induced charges at metal surfaces and biointerfaces using a polarizable lennard–jones potential. *Nature communications*, 9(1):716, 2018.

[13] Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022.

[14] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[15] Jiaqi Han, Jiacheng Cen, Liming Wu, Zongzhao Li, Xiangzhe Kong, Rui Jiao, Ziyang Yu, Tingyang Xu, Fandi Wu, Zihe Wang, et al. A survey of geometric graph neural networks: Data structures, models and applications. *arXiv preprint arXiv:2403.00485*, 2024.

[16] John Ingraham, Vikas Garg, Regina Barzilay, and Tommi Jaakkola. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

[17] Arian R Jamasb, Alex Morehead, Chaitanya K Joshi, Zuobai Zhang, Kieran Didi, Simon V Mathis, Charles Harris, Jian Tang, Jianlin Cheng, Pietro Liò, et al. Evaluating representation learning on the protein structure universe. *arXiv preprint arXiv:2406.13864*, 2024.

[18] Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2020.

[19] 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.

[20] Petras J Kundrotas, Ivan Anishchenko, Taras Dauzhenka, Ian Kotthoff, Daniil Mnevets, Matthew M Copeland, and Ilya A Vakser. Dockground: a comprehensive data resource for modeling of protein complexes. *Protein Science*, 27(1):172–181, 2018.

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

[22] Yi-Lun Liao, Brandon Wood, Abhishek Das, and Tess Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. *arXiv preprint arXiv:2306.12059*, 2023.

[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] Sidney Lyayuga Lisanza, Jake Merle Gershon, Sam Tipps, Lucas Arnoldt, Samuel Hendel, Jeremiah Nelson Sims, Xinting Li, and David Baker. Joint generation of protein sequence and structure with rosettafold sequence space diffusion. *bioRxiv*, pages 2023–05, 2023.

[25] Qinqing Liu, Peng-Shuai Wang, Chunjiang Zhu, Blake Blumenfeld Gaines, Tan Zhu, Jinbo Bi, and Minghu Song. Octsurf: Efficient hierarchical voxel-based molecular surface representation for protein-ligand affinity prediction. *Journal of Molecular Graphics and Modelling*, 105: 107865, 2021.

[26] Peter B Moore, Wayne A Hendrickson, Richard Henderson, and Axel T Brunger. The protein-folding problem: Not yet solved. *Science*, 375(6580):507–507, 2022.

[27] Alex Morehead and Jianlin Cheng. Geometry-complete perceptron networks for 3d molecular graphs. *Bioinformatics*, 40(2):btae087, 2024.

[28] Guillaume Pagès, Benoit Charmettant, and Sergei Grudinin. Protein model quality assessment using 3d oriented convolutional neural networks. *Bioinformatics*, 35(18):3313–3319, 2019.

[29] 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.

[30] 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.

[31] Ian Sillitoe, Nicola Bordin, Natalie Dawson, Vaishali P Waman, Paul Ashford, Harry M Scholes, Camilla SM Pang, Laurel Woodridge, Clemens Rauer, Neeladri Sen, et al. Cath: increased structural coverage of functional space. *Nucleic acids research*, 49(D1):D266–D273, 2021.

[32] Guillem Simeon and Gianni De Fabritiis. Tensornet: Cartesian tensor representations for efficient learning of molecular potentials. *Advances in Neural Information Processing Systems*, 36, 2024.

[33] Mark P Styczynski, Kyle L Jensen, Isidore Rigoutsos, and Gregory Stephanopoulos. Blosum62 miscalculations improve search performance. *Nature biotechnology*, 26(3):274–275, 2008.

[34] Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. Saprot: Protein language modeling with structure-aware vocabulary. In *International Conference on Learning Representations*, volume 2024, pages 6987–7009, 2024.

[35] ByteDance AML AI4Science Team, Xinshi Chen, Yuxuan Zhang, Chan Lu, Wenzhi Ma, Jiaqi Guan, Chengyue Gong, Jincai Yang, Hanyu Zhang, Ke Zhang, et al. Protenix-advancing structure prediction through a comprehensive alphafold3 reproduction. *BioRxiv*, pages 2025–01, 2025.

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

[37] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[38] Raphael JL Townshend, Stephan Eismann, Andrew M Watkins, Ramya Rangan, Masha Karelina, Rhiju Das, and Ron O Dror. Geometric deep learning of rna structure. *Science*, 373(6558): 1047–1051, 2021.

[39] Jérôme Tubiana, Dina Schneidman-Duhovny, and Haim J Wolfson. Scannet: an interpretable geometric deep learning model for structure-based protein binding site prediction. *Nature Methods*, 19(6):730–739, 2022.

[40] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[41] Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Liam Atkinson, Tally Portnoi, Itamar Chinn, et al. Boltz-1 democratizing biomolecular interaction modeling. *BioRxiv*, pages 2024–11, 2025.

[42] wwPDB consortium. Protein Data Bank: the single global archive for 3D macromolecular structure data. *Nucleic Acids Research*, 47(D1):D520–D528, 10 2018. ISSN 0305-1048. doi: 10.1093/nar/gky949. URL `https://doi.org/10.1093/nar/gky949`.

[43] Zhongyue Zhang, Jiahua Rao, Jie Zhong, Weiqiang Bai, Dongxue Wang, Shaobo Ning, Lifeng Qiao, Sheng Xu, Runze Ma, Will Hua, et al. Fitness aligned structural modeling enables scalable virtual screening with aurobind. *arXiv preprint arXiv:2508.02137*, 2025.

[44] 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.

Appendix contents

A Experimental Details 14
A.1 Datasets and splits 14
A.2 Training and hyperparameters 14
A.3 Computational profiling 14
B Additional Results 15
B.1 Controlled design comparisons 15
B.2 Task-specific baselines 16
B.3 Graph instability 16
B.4 Stratified sample counts 17
B.5 Case study 17
B.6 Cross-source evaluations 17
B.7 Convergence analysis 19
B.8 Ablations on experimental structures 20
C Broader Impact 20

A Experimental Details

A.1 Datasets and splits

CATH. We follow the CATH 4.2 40% nonredundant inverse-folding benchmark [16, 17], retain chains with at most 500 residues, and use topology-separated splits with 18,024 training, 608 validation, and 1,120 test structures. The task predicts one of 20 amino acid identities at each residue without providing the target residue identity as input.

PPBS. We follow the ScanNet dataset and split protocol [39], using PDB chains with interfaces derived from Dockground [20]. Chains shorter than 10 residues and designed proteins are excluded. The 70 split contains chains with at least 70% sequence similarity to a training sample. Homology contains remaining chains sharing a superfamily with a training sample; Topology contains remaining chains sharing a CATH topology [31]; None contains chains satisfying none of these criteria. All combines the four partitions. Each residue has a binary interface label.

AlphaFold-predicted counterparts. We map PDB identifiers to UniProt identifiers using the PDBe SIFTS API[3] and retrieve the corresponding structures from the AlphaFold Protein Structure Database[4]. Entries without a usable mapping or available predicted structure are excluded. For PPBS-AF, global sequence alignment with BLOSUM62 scoring [33] maps the experimental interface labels onto the predicted structures. The predicted-structure datasets retain the original split assignments for the included entries. Table S1 summarizes the datasets.

Table S1: Dataset sizes and evaluation metrics.

Dataset	Train	Validation	Test	Metric
CATH	18,024	608	1,120	PPL
CATH-AF	16,468	559	1,015	PPL
PPBS	12,577	3,178	3,984	AUPRC
PPBS-AF	11,345	2,876	3,636	AUPRC

A.2 Training and hyperparameters

Inverse folding uses cross-entropy loss over the 20 residue types, and binding-site prediction uses binary cross-entropy loss. Models are trained for at most 50 epochs with early stopping based on validation performance. The learning rate is 0.001, batch size is 16 or 32, and dropout is 0. The prediction head has two hidden layers of 128 units. Other baseline settings follow Jamasb et al. [17].

Across tasks, the adaptive graph uses $\epsilon_c = 1$, $\sigma = 3.8$, $\alpha = 1$, and $\beta = 8$. The backbone has six Transformer blocks and passes tensor degrees up to $L = 2$ between blocks. Tensor degrees $L = 3, 4, 5$ from the five non-final blocks are supplied to Elastic-SSM, whose outputs have $L = 0$. The reported main results use sampled radii during training and inference. Sensitivity to (α, β) is reported in Appendix B.1.

A.3 Computational profiling

We profile all models on one A100 80GB GPU with batch size 32, using 18,024 training samples and 931 samples for inference profiling. The profiled mean residue counts are 217 for training and 161 for inference; graph-size statistics are reported using the same convention as the profiled batches.

[3] `https://www.ebi.ac.uk/pdbe/api/doc/sifts.html`

[4] `https://alphafold.ebi.ac.uk`

Table S2: Computational profiling on CATH. Memory values are peak GPU memory; inference time is the total time for the profiled inference set. Edge columns report mean retained graph-size statistics.

	Params	Memory (GB)		Time (s)		Edges	
Model	(M)	Train	Infer	Epoch	Infer	Train	Infer
SchNet	2.68	7.89	0.62	47	9	3479	2577
EGNN	11.24	52.84	3.69	115	13	3479	2577
TFN	8.86	77.81	29.65	712	17	4034	2999
GearNet-Edge	12.10	31.37	4.96	141	16	3479	2577
GCPNet	2.17	12.27	1.41	239	25	3479	2577
Equiformer	2.26	33.05	8.94	328	21	3358	2445
E^3former	2.75	38.67	10.39	387	28	3864	2876

Relative to Equiformer, E^3former adds 0.49M parameters, 5.62 GB of peak training memory, and 1.45 GB of peak inference memory. The top-k neighbor cap bounds the retained edge count by k times the number of residues, while Elastic-SSM operates over a fixed number of Transformer layers. In an additional memory stress test, inputs of approximately 2,000 residues were processed on an A100 80GB GPU with batch size 8 without out-of-memory errors. This tests memory feasibility only; the CATH accuracy evaluation is restricted to proteins of at most 500 residues.

B Additional Results

B.1 Controlled design comparisons

Table S3 compares graph construction and cross-layer aggregation on CATH-AF under +Seq. Graph variants share the backbone, input features, radius range, and training protocol. Aggregation variants share the backbone, high-order tensor inputs, type-0 projection, and prediction head. Values are mean $\pm$ standard deviation over repeated runs.

Table S3: Controlled comparisons of graph construction and cross-layer aggregation on CATH-AF under +Seq.

Configuration	PPL $\downarrow$
Graph construction	
Fixed-radius graph	5.82 ± 0.06
Density-selected three-radius graph	5.79 ± 0.04
Inverse-distance adaptive score	5.56 ± 0.03
Lennard-Jones adaptive score	5.50 ± 0.05
Cross-layer aggregation	
Elastic-SSM	5.50 ± 0.05
Mean aggregation	5.57 ± 0.01
Sum aggregation	5.59 ± 0.02
Gated aggregation	5.58 ± 0.02
Attention-based aggregation	5.55 ± 0.03
Inter-layer Transformer	5.62 ± 0.02
Simple recurrent aggregation	5.58 ± 0.06
1D convolution across depth	5.61 ± 0.03
Without Elastic-SSM	5.71 ± 0.04

The Lennard-Jones score gives the lowest mean PPL among the tested graph variants, and Elastic-SSM gives the lowest mean PPL among the tested aggregation methods.

Across $(\alpha, \beta) \in \{(0.5, 8), (1, 4), (1, 8), (1, 16), (2, 8)\}$, mean PPL ranges from 5.50 to 5.53; the default $(1, 8)$ gives 5.50 ± 0.05. Supplying features from early $\{1, 2\}$, middle $\{3\}$, late $\{4, 5\}$, or alternating $\{1, 3, 5\}$ layers gives PPLs of 5.62 ± 0.04, 5.69 ± 0.04, 5.64 ± 0.06, and 5.58 ± 0.04, respectively, compared with 5.50 ± 0.05 using all five non-final layers. Only the supplied layer set changes in this comparison.

B.2 Task-specific baselines

We complement the geometric-encoder benchmarks with task-specific models for inverse folding and binding-site prediction. For inverse folding, we evaluate the official CA-only pretrained ProteinMPNN checkpoint [7] in its CA-only scoring mode. We also fine-tune the same checkpoint for 15 epochs on the CATH-AF training split while retaining CA-only structural inputs. Table S4 reports these results. CA-only inputs align the available structural coordinates with our +Seq setting, while the models retain their respective prediction and pretraining procedures.

Table S4: Task-specific inverse-folding comparisons on CATH-AF. Values are mean ± standard deviation.

Method	PPL ↓
Equiformer, +Seq	6.09 ± 0.04
ProteinMPNN, official CA-only checkpoint	7.15 ± 0.06
ProteinMPNN, CA-only fine-tuned	6.54 ± 0.04
E^3former, +Seq	5.50 ± 0.05

For binding-site prediction, Table S5 compares our geometric models under $+\phi, \psi, \omega$ on experimental PPBS structures with the published ScanNet results [39]. Full ScanNet uses evolutionary information, while the ablated variant omits it.

Table S5: Binding-site AUPRC on experimental PPBS structures. ScanNet values are reported results; geometric-model rows use $+\phi, \psi, \omega$.

Method	70	Homology	Topology	None	All
Equiformer	0.780	0.727	0.728	0.615	0.712
ScanNet	0.732	0.712	0.735	0.605	0.694
ScanNet, no evolutionary info.	0.672	0.648	0.685	0.565	0.639
E^3former	0.783	0.732	0.743	0.631	0.726

B.3 Quantifying graph instability under predicted structures

We quantify neighborhood instability using 1007 matched CATH/CATH-AF proteins for which the AlphaFold-predicted structure can be aligned to the corresponding experimental CATH structure. For each predicted structure, we construct either a fixed-radius graph or an adaptive graph, and compare it against the fixed-radius graph built from the matched experimental structure. Edge mismatch is reported as the percentage of mismatched edges relative to the reference edge count.

Table S6: Graph edge mismatch on matched CATH/CATH-AF structures. RMSD is in Å; mismatch and relative reduction are percentages.

RMSD bin	Proteins	Fixed	Adaptive	Reduction
All	1007	9.28	9.07	2.3
< 1	459	5.83	5.69	2.4
1–3	327	9.47	9.15	3.4
3–5	69	13.07	12.43	4.9
5–10	88	16.91	15.81	6.5
> 10	64	21.43	19.67	8.2

These results show that AlphaFold-predicted coordinate deviations induce measurable neighborhood instability, and that the adaptive graph consistently reduces this mismatch. The reduction is more pronounced in higher-RMSD bins, supporting the role of adaptive neighborhood construction under larger structural deviations.

B.4 Confidence- and resolution-stratified sample counts

We report the test-sample counts used in Figures 3 and 4. The bins contain different proteins.

Table S7: Test-sample counts for the confidence- and resolution-stratified evaluations.

AlphaFold confidence

Split	0–80	80–85	85–90	90–95	95–100
CATH-AF	157	118	239	350	151
PPBS-AF 70	56	47	70	115	190
PPBS-AF Homology	133	120	169	415	541
PPBS-AF Topology	77	39	72	189	475
PPBS-AF None	148	96	143	312	229
PPBS-AF All	414	302	454	1031	1435

Crystallographic resolution (Å)

Split	0–1.5	1.5–2.0	2.0–2.5	2.5–3.0	3.0–3.5	3.5–9.0
CATH	102	383	298	114	34	0
PPBS 70	23	178	171	113	43	17
PPBS Homology	105	428	508	284	129	21
PPBS Topology	77	327	313	157	29	8
PPBS None	56	329	334	194	100	40
PPBS All	261	1262	1326	748	301	86

B.5 Case study

Figure S1 visualizes the residue-level binding-site probabilities predicted by E^3former for chain A of 1KPK and chain E of 3DXA. Color intensity indicates the predicted binding-site probability.

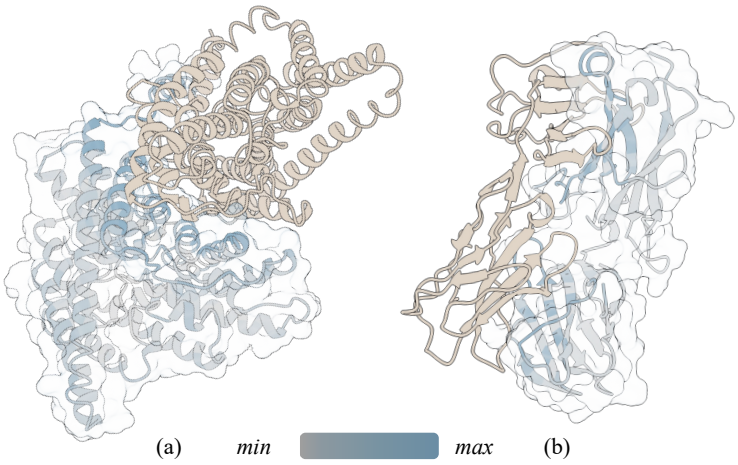

Figure S1: Binding-site predictions for (a) 1KPK and (b) 3DXA. Color intensity represents the predicted probability at each residue.

B.6 Cross-source evaluations

We evaluate transfer between experimental PPBS structures and their AlphaFold-predicted counterparts. Table S8 reports training on PPBS-AF and testing on experimental PPBS structures, while Table S9 reports the reverse direction.

Table S8: Binding-site AUPRC after training on PPBS-AF and evaluating on experimental PPBS structures. Bold denotes the highest value in each row.

	Features	SchNet	TFN	EGNN	Equiformer	GearNet	GCPNet	E^3**former**
PPBS 70	+Seq	0.44953	0.64124	0.62695	0.66591	0.65592	0.64219	**0.68544**
	$+\kappa, \alpha$	0.48296	0.65132	0.64405	**0.68343**	0.64709	0.63064	0.68185
	$+\phi, \psi, \omega$	0.52123	0.63546	0.62501	**0.67828**	0.64118	0.62339	0.67672
PPBS Homology	+Seq	0.37548	0.56347	0.59807	0.63228	0.61499	0.58236	**0.65478**
	$+\kappa, \alpha$	0.39545	0.60154	0.59935	0.64031	0.59546	0.56641	**0.65088**
	$+\phi, \psi, \omega$	0.4434	0.57120	0.58794	0.64042	0.60408	0.57923	**0.65221**
PPBS Topology	+Seq	0.37378	0.59578	0.65274	0.67387	0.60586	0.57741	**0.69854**
	$+\kappa, \alpha$	0.41137	0.65262	0.64796	0.67406	0.59483	0.61239	**0.69284**
	$+\phi, \psi, \omega$	0.43746	0.60515	0.66177	0.67785	0.60587	0.58901	**0.69788**
PPBS None	+Seq	0.28093	0.43247	0.51527	0.51266	0.48786	0.44592	**0.53728**
	$+\kappa, \alpha$	0.32368	0.49414	0.51055	0.51803	0.45168	0.47998	**0.53678**
	$+\phi, \psi, \omega$	0.35349	0.45055	0.50075	0.51098	0.47044	0.43271	**0.52693**
PPBS All	+Seq	0.35806	0.53397	0.59119	0.61592	0.58545	0.56649	**0.64033**
	$+\kappa, \alpha$	0.38917	0.59224	0.59253	0.62204	0.56501	0.58232	**0.63571**
	$+\phi, \psi, \omega$	0.42553	0.55304	0.58419	0.62065	0.57698	0.56612	**0.63465**

Table S9: Binding-site AUPRC after training on experimental PPBS structures and evaluating on PPBS-AF. Bold denotes the highest value in each row.

	Features	SchNet	TFN	EGNN	Equiformer	GearNet	GCPNet	E^3**former**
PPBS-AF 70	+Seq	0.20168	**0.36930**	0.28549	0.32871	0.35460	0.22138	0.33082
	$+\kappa, \alpha$	0.21044	0.31716	0.20516	0.33254	0.32297	0.23769	**0.34731**
	$+\phi, \psi, \omega$	0.28123	**0.37468**	0.26797	0.33381	0.33172	0.29985	0.34371
PPBS-AF Homology	+Seq	0.14265	0.25322	0.22522	0.26929	0.23107	0.17286	**0.27231**
	$+\kappa, \alpha$	0.14966	0.23096	0.14952	0.26761	0.21692	0.18126	**0.27503**
	$+\phi, \psi, \omega$	0.20689	0.27209	0.20752	0.28433	0.22315	0.21829	**0.29445**
PPBS-AF Topology	+Seq	0.22531	0.35238	0.39045	0.40333	0.30934	0.26523	**0.41394**
	$+\kappa, \alpha$	0.23966	0.34851	0.23546	0.39530	0.31184	0.26981	**0.41967**
	$+\phi, \psi, \omega$	0.26613	0.36855	0.26359	0.41227	0.30314	0.29314	**0.42827**
PPBS-AF None	+Seq	0.10728	0.15740	0.15415	0.15649	0.13569	0.11267	**0.17847**
	$+\kappa, \alpha$	0.11008	0.15162	0.10940	0.16689	0.13152	0.12901	**0.16882**
	$+\phi, \psi, \omega$	0.13614	0.17037	0.13929	0.18115	0.12171	0.12691	**0.18519**
PPBS-AF All	+Seq	0.16702	0.26963	0.25671	0.28902	0.24462	0.17968	**0.29870**
	$+\kappa, \alpha$	0.17084	0.25210	0.17031	0.29183	0.23460	0.19982	**0.29493**
	$+\phi, \psi, \omega$	0.21646	0.29070	0.21599	0.31065	0.23602	0.22435	**0.31575**

Across both transfer directions, E^3former achieves the highest AUPRC on the Homology, Topology, None, and All partitions under all three feature settings. Results on the 70 partition vary across methods and feature settings. These experiments show that the learned representations remain effective under transfer between the two structural sources.

B.7 Convergence analysis

Figures S2 and S3 show validation-loss trajectories on predicted and experimental structures. In the displayed runs, E^3former and Equiformer converge relatively early. The curves also identify early stopping and runs with abnormal loss changes.

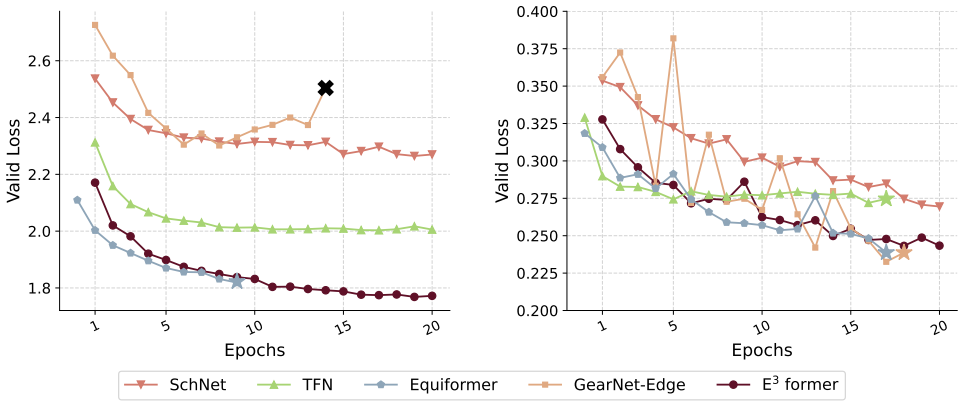

Figure S2: Validation loss across training epochs on AlphaFold-predicted structures. Crosses indicate abnormal loss changes; stars indicate early stopping.

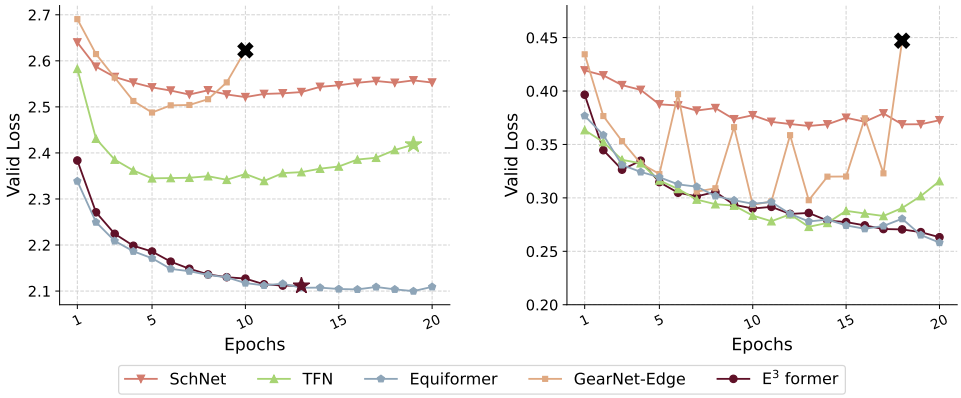

Figure S3: Validation loss across training epochs on experimental structures.

B.8 Ablations on experimental structures

Table S10 complements the component ablations on predicted structures in Section 4.5. Removing either module increases CATH perplexity and reduces PPBS AUPRC across the reported feature settings and test partitions, supporting contributions from both adaptive neighborhoods and Elastic-SSM. The relative effects of the two modules vary across tasks and feature settings.

Table S10: Component ablations on experimental structures. Values are mean ± standard deviation. Bold denotes the best value in each column within each feature setting.

Method	Features	CATH(↓)	PPBS(↑)				
			70	Homology	Topology	None	All
E^3former	+Seq	$\mathbf{7.53 \pm 0.02}$	$\mathbf{0.777 \pm 0.01}$	$\mathbf{0.732 \pm 0.02}$	$\mathbf{0.743 \pm 0.02}$	$\mathbf{0.637 \pm 0.03}$	$\mathbf{0.724 \pm 0.02}$
	$+\kappa, \alpha$	$\mathbf{7.97 \pm 0.02}$	$\mathbf{0.775 \pm 0.01}$	$\mathbf{0.727 \pm 0.02}$	$\mathbf{0.745 \pm 0.00}$	$\mathbf{0.628 \pm 0.02}$	$\mathbf{0.719 \pm 0.02}$
	$+\phi, \psi, \omega$	$\mathbf{6.64 \pm 0.03}$	$\mathbf{0.783 \pm 0.01}$	$\mathbf{0.732 \pm 0.01}$	$\mathbf{0.743 \pm 0.03}$	$\mathbf{0.631 \pm 0.01}$	$\mathbf{0.726 \pm 0.02}$
w/o Energy	+Seq	7.61 ± 0.02	0.771 ± 0.01	0.725 ± 0.03	0.729 ± 0.02	0.625 ± 0.04	0.711 ± 0.03
	$+\kappa, \alpha$	8.07 ± 0.01	0.773 ± 0.00	0.724 ± 0.03	0.732 ± 0.00	0.613 ± 0.02	0.709 ± 0.01
	$+\phi, \psi, \omega$	6.76 ± 0.01	0.779 ± 0.01	0.727 ± 0.02	0.734 ± 0.00	0.623 ± 0.01	0.719 ± 0.03
w/o Elastic	+Seq	7.68 ± 0.02	0.773 ± 0.03	0.728 ± 0.00	0.735 ± 0.01	0.631 ± 0.00	0.715 ± 0.01
	$+\kappa, \alpha$	8.03 ± 0.00	0.772 ± 0.00	0.723 ± 0.00	0.729 ± 0.02	0.619 ± 0.03	0.712 ± 0.00
	$+\phi, \psi, \omega$	6.79 ± 0.00	0.782 ± 0.00	0.729 ± 0.01	0.739 ± 0.02	0.627 ± 0.03	0.722 ± 0.01
w/o EE	+Seq	7.80 ± 0.00	0.765 ± 0.00	0.722 ± 0.00	0.723 ± 0.02	0.615 ± 0.00	0.703 ± 0.02
	$+\kappa, \alpha$	8.12 ± 0.03	0.771 ± 0.00	0.719 ± 0.00	0.713 ± 0.00	0.602 ± 0.01	0.701 ± 0.00
	$+\phi, \psi, \omega$	6.89 ± 0.01	0.778 ± 0.02	0.726 ± 0.01	0.725 ± 0.03	0.617 ± 0.00	0.711 ± 0.02

C Broader Impact

This work develops residue-level protein representations and paired benchmarks using experimental and AlphaFold-predicted structures. Improved use of imperfect structural inputs may support protein design and function prediction. Structural errors can nevertheless lead to incorrect downstream predictions, which should be checked experimentally before use.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the main contributions and scope of the work. The claims are limited to residue-level protein representation learning on the CATH/CATH-AF and PPBS/PPBS-AF benchmarks evaluated in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a limitations discussion covering the evaluated task scope, structure sources, residue-level setting, and computational cost-performance trade-off.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The equivariant tensor operations and the transformation rules of Elastic-SSM are described and derived in Section 3.2.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper describes the datasets, model architecture, training objectives, hyperparameters, evaluation metrics, and ablation settings needed to reproduce the main experimental results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code repository is linked in the abstract, and Appendix A.1 describes the construction of CATH-AF and PPBS-AF from public resources.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Dataset splits, task definitions, input features, losses, hyperparameters, and baseline settings are described in the experiments section and appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The main result tables report mean performance and standard deviations across repeated runs using explicit $\pm$ notation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix reports runtime comparisons on a single A100 80GB GPU, including training and inference time for the proposed model and baselines.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses public protein structure and benchmark datasets and does not involve private personal data, human subjects, or deployment decisions affecting individuals.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The impact statement discusses potential benefits for protein representation learning and notes that predictions from imperfect structures should not replace experimental validation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The work does not release high-risk generative models, scraped human data, or assets with a direct high-risk misuse pathway requiring special safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the original sources of the existing datasets, tools, and baselines used, including CATH, PPBS/ScanNet, AlphaFold DB, and related model implementations where applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces CATH-AF and PPBS-AF benchmark variants and documents their construction procedure, data sources, filtering steps, and task definitions.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing, user studies, or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

Justification: The paper does not involve human subjects research, so IRB approval or equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as an important, original, or non-standard component of the core method or experiments.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.