VTPE: VIRTUAL TRAJECTORY POSITION ENCODING FOR BIOMOLECULAR COMPLEX STRUCTURE PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

AlphaFold-style co-folding methods have achieved near-experimental accuracy on protein monomers, yet structure prediction for biomolecular complexes remains challenging. A key difficulty is that biomolecular complexes contain many atoms that engage in long-range interactions: the position of one atom constrains the positions of numerous other atoms at varying distances, all at long range. Diffusion-based structure prediction resolves these constraints progressively over denoising steps, but supervision is defined only on the final denoised structure, leaving the relationships among intermediate states unmodeled. Consequently, the atomic adjustment process and the coordination among distant atoms remain difficult to capture. In this paper, we view these intermediate structures as a virtual trajectory and propose Virtual Trajectory Position Encoding (VTPE), which encodes atomic coordinates along this virtual trajectory as positional encoding. This enables the model to represent how distant atoms move relative to one another over the course of denoising. Since long-range interactions are precisely the couplings that produce such coordinated adjustments, exposing the model to these adjustments provides a direct signal for learning long-range interactions. We evaluate VTPE at multiple parameter scales of Protenix, the open-source reproduction of AlphaFold 3 (AF3), and observe consistent improvements over the corresponding baselines across benchmarks with limited additional overhead. Experiments confirm the contribution of virtual trajectory position encoding to complex structure prediction.

1 INTRODUCTION

Protein structure prediction is a fundamental task in structural biology and an important tool for structure-based drug discovery. With advances in deep learning, AI-for-drug-discovery (AIDD) methods are increasingly applied to this problem Jumper et al. (2021). Models such as ESMFold Lin et al. (2023) and AlphaFold2 Jumper et al. (2021) have achieved near-experimental accuracy in predicting single-chain protein structures. However, many biological functions and therapeutic mechanisms depend on interactions among multiple biomolecular components; multimeric proteins and other biomolecular complexes are therefore of central importance Abramson et al. (2024).

Increasing attention has shifted toward complex structure prediction, which is emerging as a foundational task for foundation models in AIDD. This role parallels that of natural language tasks in multimodal foundation models, and it underpins a wide range of downstream applications Passaro et al. (2025); Zhang et al. (2026), including affinity prediction, structure-based drug design (SBDD), and molecular generation. Accurate complex-structure representations can support downstream tasks Passaro et al. (2025). Alphafold 3 (AF3) Abramson et al. (2024) performs complex structure prediction. Boltz-2 Passaro et al. (2025) builds on the AF3 architecture with enlarged training data and a classification head for affinity prediction, retaining AF3's capabilities while adding affinity estimation. Protenix Zhang et al. (2026), the open-source reproduction of AF3, integrates extensive datasets and model acceleration, thereby improving complex structure prediction accuracy. It further augments AF3 with multiple classification heads to enable SBDD and molecular generation, expanding the foundation model's downstream task repertoire.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The task of complex structure prediction aims to generate the three-dimensional (3D) atomic coordinates of input complexes Abramson et al. (2024). AlphaFold-style co-folding methods have achieved near-experimental accuracy on protein monomers, yet biomolecular complex structure prediction remains challenging Zhang et al. (2026). A key difficulty in biomolecular complex prediction arises from long-range interactions, whereby the position of one atom imposes constraints on many distal atoms. Because these constraints are non-local, local updates alone cannot account for the coordinated rearrangements required for accurate structures. Diffusion-based structure prediction resolves these constraints progressively over denoising steps, but supervision is defined only on the final denoised structure Abramson et al. (2024); Zhang et al. (2026); Passaro et al. (2025), leaving the relationships among intermediate states unmodeled. Consequently, the atomic adjustment process and the coordination among distant atoms remain difficult for the model to capture.

Similar long-range challenges also arise in the Large Language Models (LLMs) domain, where, when the input context is particularly long, the goal is to enable self-attention-based models to capture textual relationships across varying distances Wang et al. (2024; 2026). Positional encoding algorithms in LLMs have greatly alleviated the challenges caused by long-range problems Wang et al. (2024; 2026); Su et al. (2024); Wei et al. (2025). The core idea of positional encoding in LLMs is to exploit the inherent sequential order of text, assign each token a unique position index, and enable the model to perceive token-level position indices, so that it can distinguish the relative distance between tokens during both training and inference, thereby enhancing its long-range understanding Su et al. (2024). Inspired by this idea from LLMs, we aim to develop a positional encoding algorithm for biological data to enhance the model's long-range reasoning ability. However, two discrepancies exist: (1) the lack of atom-level position indices and (2) the dimensional gap between text and biological data.

Motivated by prior work Zelikman et al. (2024); Li et al. (2024); Wang et al. (2025); Su et al. (2024); Wang et al. (2024; 2026), we view the intermediate states in structure prediction models as a *virtual trajectory* and propose Virtual Trajectory Position Encoding (**VTPE**), which bridges these two discrepancies and enhances the model's ability to capture atomic interactions within complexes. To the best of our knowledge, this is the first work that applies intermediate states as positional encoding for biomolecular complex structure prediction. Empirically, we find that the virtual trajectory gradually stabilizes across layers, a phenomenon we term *layer-wise stabilization*. VTPE encodes the 3D coordinates of atoms along the virtual trajectory as position indices and injects atomic position indices into attention layers via 3D symmetry-preserving, norm-preserving orthogonal rotations. This enables the model to represent how atoms at different distances move relative to each other during denoising. As this process is naturally compatible with self-attention, it exposes the model to such adjustments with limited additional memory and computational overhead, thereby providing a direct signal for learning long-range interactions. Our contributions are summarized as follows:

- **Method.** We externalize the model's intermediate states as a virtual trajectory and use per-atom 3D coordinates along the trajectory as position indices in our proposed VTPE. This naturally injects interatomic relative distances and directionality into the self-attention mechanism, enhancing the model's understanding of complex atomic interactions with negligible additional memory and compute overhead.
- **Analysis.** We mathematically characterize the layer-wise stabilization of virtual trajectory and analyze the 3D symmetry and data-adaptive behavior of VTPE, thereby providing interpretability for virtual-conformation injection and complex structure perception.
- **Experiments.** Our extensive evaluation spans multiple open-source AF3 variants, including Protenix-base, Protenix-mini, and Protenix-tiny. It demonstrates improved performance over existing methods, with the additional advantage of VTPE on three challenging complex structure prediction benchmarks: RecentPDB, PoseBusters, and PXM-Legacy.

2 Related Work

Complex Structure Prediction. Given the ubiquity of molecular interactions in drug discovery, research has increasingly shifted toward complex structure prediction in recent years Abramson et al. (2024). As this transition unfolds, current work primarily focuses on adapting complex structure prediction models to broader drug discovery tasks, for instance, enabling molecular docking or

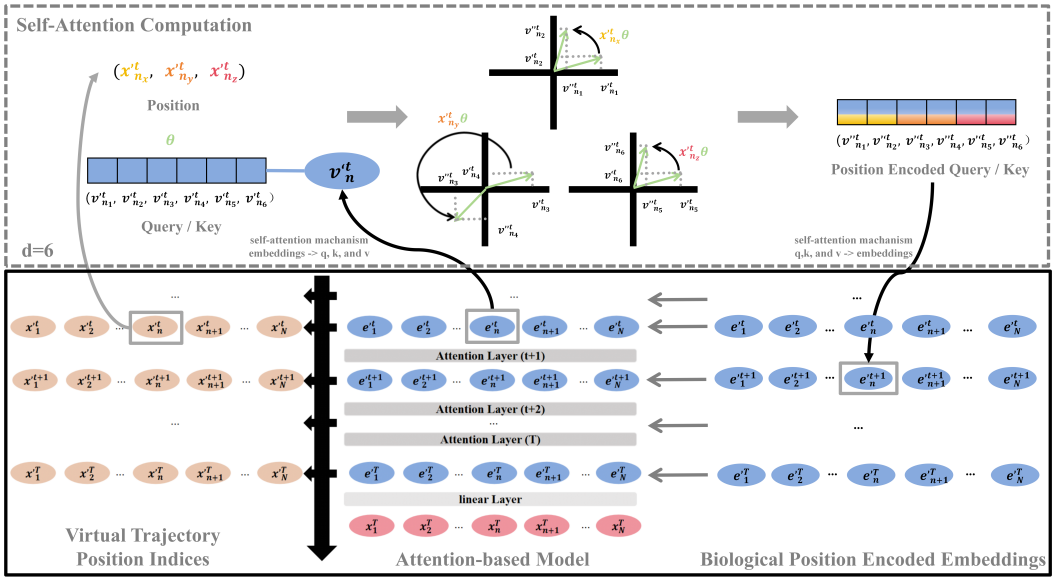

Figure 1: Overview. Virtual trajectory derived from an attention-based model serve as position indices. The black lines crossing the top and bottom panels twice represent the computation of the self-attention mechanism.

molecular design on top of these models Passaro et al. (2025); Zhang et al. (2026). However, insufficient attention has been paid to the challenges arising from the modeling of long-range atomic interactions in complexes, particularly the larger number of atoms that attention-based models must handle and the performance degradation relative to protein monomer structure prediction.

Model Hidden States. Many tasks involve a series of intermediate states before the final answer is produced. During training, these intermediate states are naturally embedded within the model. For instance, textual reasoning tasks involve intermediate thought states, and externalizing these states and injecting them into the model in an appropriate manner often improves model performance Li et al. (2024); Liu et al. (2022); Zhao et al. (2024); Loos & Linhardt (2025); Lu et al. (2025). Inspired by these methods, we propose to construct the model's internal intermediate states as a virtual trajectory, thereby providing layer-wise, atom-level position indices for biological complexes.

Positional Encoding. Self-attention is inherently position-agnostic Vaswani et al. (2017). Prior work injects positional information so that self-attention can distinguish token positions Devlin et al. (2019); Yun et al. (2019); Lan et al. (2019). Position-aware models generally exhibit stronger long-range modeling capabilities Dai et al. (2025); Zhu et al. (2025). RoPE Su et al. (2024), the dominant positional encoding in LLMs Wei et al. (2025); Bai et al. (2025), enables length generalization from shorter training sequences to longer inference sequences. Inspired by the long-range capability and memory efficiency of RoPE, we propose a biological positional encoding tailored to biomolecular data.

3 Method

3.1 Overview

To formulate the problem, we denote a complex as $c = (\mathbf{z}, X)$, where $\mathbf{z} = \{z_i\}_{i=1}^{N}$ denotes the atomic types of the N atoms and $X = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{N\times 3}$ denotes their 3D coordinates. Given a complex c, a foundation model G_θ produces atomic representations $E = G_\theta(c) = [\mathbf{e}_1, \ldots, \mathbf{e}_N]$, $\mathbf{e}_i \in \mathbb{R}^d$, where $\mathbf{e}_i$ denotes the representation of atom i conditioned on the complex context. For complex structure prediction, we consider supervised pairs $\langle c, X^*(c)\rangle \in \mathcal{C} \times \mathcal{Y}$, where $X^* : \mathcal{C} \to \mathcal{Y}$ maps a complex to its target atomic coordinates and $\mathcal{Y} \subseteq \mathbb{R}^{N\times 3}$ denotes the space of complex structures.

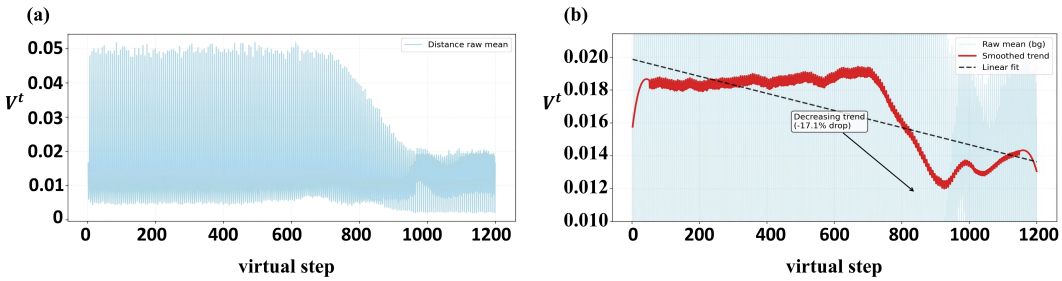

Figure 2: Layer-wise stabilization of virtual trajectory. Inference results of Protenix-base and VTPE on PoseBusters. Left: layer-wise change $V'^t = \Delta D^t_{\mathrm{pl}}$. Right: the same quantity with linear and nonlinear fits. The magnitude of V'^t decreases in later layers and then fluctuates at a low level, indicating a transient-to-quasi-steady crossover.

To expose layer-wise geometric information and improve long-range structural modeling in foundation models, we make the model's internal intermediate states via positional encoding, thereby enhancing intermediate structures perception and long-range aibility with negligible additional training overhead. The method is illustrated in Figure. 1. Specifically, we first map the tokens in the attention layer of the atom sequence through a lightweight linear layer to obtain a virtual trajectory of the complex binding process. We then use the 3D coordinates of the virtual trajectory as positional indices for the VTPE method to apply orthogonal rotations to the tokens in 3D space. The resulting position-aware tokens are fed into the next layer. No additional loss or supervision signal is introduced throughout training.

3.2 Virtual Trajectory Position Encoding

Existing foundation models contain multiple self-attention layers. Let

$$G_{A_\theta}(c) = \langle A^1, A^2, \ldots, A^T \rangle$$

denote the sequence of complex representations produced across these layers, where

$$A^t = \langle e^t_1, e^t_2, \ldots, e^t_N \rangle$$

is the sequence of latent atomic representations at virtual step t, with $e^t_n \in \mathbb{R}^d$. To expose these intermediate states in geometric form, we project each A^t to a three-dimensional atomic conformation using a lightweight linear layer.

Definition 1 (Virtual trajectory)**.** For each virtual step $t \in \{1, \ldots, T-1\}$, the corresponding *virtual conformation* is defined as

$$X'^t = L^t_\theta(A^t) \in \mathbb{R}^{N\times 3}.$$

The ordered sequence

$$X'^{1:T-1} = \langle X'^1, X'^2, \ldots, X'^{T-1} \rangle \tag{1}$$

is called the *virtual trajectory* of the complex. It is an internal layer-wise geometric representation rather than a trajectory in physical time, and no direct supervision is imposed on its intermediate conformations.

We then treat each virtual conformation X'^t as atom-level 3D position indices for the next self-attention layer, inspired by positional encoding in LLMs. Specifically,

$$X'^t = \langle \mathbf{x}'^t_1, \ldots, \mathbf{x}'^t_N \rangle, \qquad \mathbf{x}'^t_m = (m_x, m_y, m_z),$$

and we associate the m-th atom with a coordinate-dependent rotary operator. In this way, the layer-wise virtual geometry is injected into self-attention with limited additional memory and computational overhead.

At virtual step t, the self-attention layer $G^t_{A_\theta}$ outputs $A^t = \langle e'^t_1, e'^t_2, \cdots, e'^t_N \rangle$. Based on A^t and the virtual conformation X'^t at this step, by incorporating the designed VTPE with the self-attention

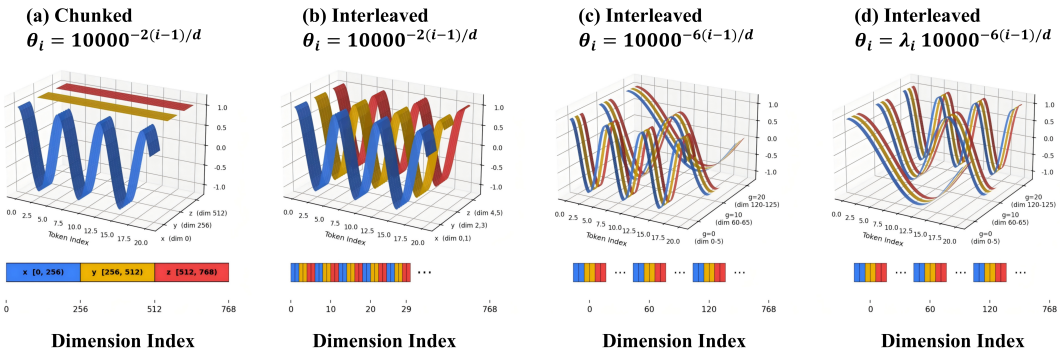

Figure 3: Rotation symmetry and awareness. (a) In vanilla RoPE with $u = 2$, sequentially assigning x, y, and z. (b) Assigning x, y, and z in an interleaved manner. (c) Building on (b), setting $u = 6$ ensures rotational symmetry among x, y, and z. (d) Introducing learnable parameters λ_i.

mechanism, we obtain the atomic representation at the next step, $A^{t+1} = \langle e'^{t+1}_1, e'^{t+1}_2, \cdots, e'^{t+1}_N \rangle$. We write $\mathbf{X}'^t = \langle \mathbf{x}'^t_1, \mathbf{x}'^t_2, \cdots, \mathbf{x}'^t_N \rangle$, where $\mathbf{x}'^t_m = (m_x, m_y, m_z)$, $m \in \{1, 2, \ldots, N\}$, and $m_x, m_y, m_z \in \mathbb{R}$. We use these virtual coordinates as position indices by associating each atom with the following coordinate-dependent rotary transformation.

Definition 2 (VTPE rotation matrix)**.** For the m-th atom with virtual coordinate

$$\mathbf{m} = (m_x, m_y, m_z) = \mathbf{x}'^t_m,$$

let

$$\mathbf{R}(\mathbf{m}, \theta_i) = \mathrm{diag}(\mathbf{R}(m_x\theta_i), \mathbf{R}(m_y\theta_i), \mathbf{R}(m_z\theta_i)), \tag{2}$$

where

$$\mathbf{R}(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}, \qquad \theta_i = 10000^{-\frac{u(i-1)}{d}}. \tag{3}$$

Here, d denotes the dimension of the atomic encoding vector, and each frequency group consists of three two-dimensional rotary blocks, corresponding respectively to the x-, y-, and z-coordinates.

Assuming d is divisible by 6, as in our implementation, there are $d/6$ such frequency groups. We denote their frequency set by

$$\Theta = \{\theta_1, \theta_2, \ldots, \theta_{d/6}\}.$$

The complete VTPE rotation matrix associated with virtual coordinate $\mathbf{m}$ is defined as

$$\mathbf{R}^d_{\Theta,\mathbf{m}} = \mathrm{diag}\big(\mathbf{R}(\mathbf{m}, \theta_1), \mathbf{R}(\mathbf{m}, \theta_2), \ldots, \mathbf{R}(\mathbf{m}, \theta_{d/6})\big) \in \mathbb{R}^{d\times d}. \tag{4}$$

When referring to axis-specific rotary blocks, we use i_x, i_y, and i_z to denote the block indices assigned to the x-, y-, and z-coordinates, respectively.

We follow Su et al. (2024) for the base frequency schedule in Eq. 3 and introduce several extensions to adapt it to 3D biomolecular coordinates. These extensions are analyzed in Section 3.3.

Proposition 1 (Relative-position consistency and norm preservation)**.** *For the VTPE rotation matrix* $\mathbf{R}^d_{\Theta,\mathbf{m}}$ *in Definition 2, and any two virtual coordinates* $\mathbf{m} = (m_x, m_y, m_z)$ *and* $\mathbf{n} = (n_x, n_y, n_z)$*,*

$$\left(R^d_{\Theta,m}\right)^\top R^d_{\Theta,m} = I_d,$$

and

$$\left(R^d_{\Theta,m}\right)^\top R^d_{\Theta,n} = R^d_{\Theta,n-m},$$

where

$$n - m = (n_x - m_x,\ n_y - m_y,\ n_z - m_z).$$

Consequently, for any $v \in \mathbb{R}^d$*,*

$$\left\|R^d_{\Theta,m} v\right\|_2 = \|v\|_2.$$

The first identity shows that VTPE is norm preserving, while the second shows that the interaction between two VTPE rotations depends only on their relative virtual coordinate. A proof is provided in Appendix B.2.

With the relative-position structure of VTPE established, we instantiate it in self-attention.

Definition 3 (VTPE-augmented self-attention)**.** At virtual step t, let $\mathbf{e}'^{(t)}_m \in \mathbb{R}^d$ be the representation of atom m with virtual coordinate $\mathbf{m} = (m_x, m_y, m_z) = \mathbf{x}'^t_m$. Using $\mathbf{R}^d_{\Theta,\mathbf{m}}$ from Definition 2, we define

$$\mathbf{q}^{(t)}_m = \mathbf{R}^d_{\Theta,\mathbf{m}} \mathbf{W}_{q^t} \mathbf{e}'^{(t)}_m, \qquad \mathbf{k}^{(t)}_m = \mathbf{R}^d_{\Theta,\mathbf{m}} \mathbf{W}_{k^t} \mathbf{e}'^{(t)}_m, \qquad \mathbf{v}^{(t)}_m = \mathbf{W}_{v^t} \mathbf{e}'^{(t)}_m. \tag{5}$$

The attention update is

$$a^{(t)}_{mn} = \frac{\exp\left(\mathbf{q}^{(t)\top}_m \mathbf{k}^{(t)}_n / \sqrt{d}\right)}{\sum_{r=1}^{N} \exp\left(\mathbf{q}^{(t)\top}_m \mathbf{k}^{(t)}_r / \sqrt{d}\right)}, \qquad \mathbf{e}'^{(t+1)}_m = \mathbf{o}^{(t)}_m = \sum_{n=1}^{N} a^{(t)}_{mn} \mathbf{v}^{(t)}_n. \tag{6}$$

Corollary 1 (Relative virtual geometry in VTPE attention)**.** Under Definition 3, for atoms m and n with virtual coordinates $\mathbf{m} = (m_x, m_y, m_z)$ and $\mathbf{n} = (n_x, n_y, n_z)$,

$$\mathbf{q}^{(t)\top}_m \mathbf{k}^{(t)}_n = \mathbf{e}'^{(t)\top}_m \mathbf{W}^{\top}_{q^t} \mathbf{R}^d_{\Theta,\mathbf{n}-\mathbf{m}} \mathbf{W}_{k^t} \mathbf{e}'^{(t)}_n, \qquad \mathbf{n} - \mathbf{m} = (n_x - m_x,\ n_y - m_y,\ n_z - m_z). \tag{7}$$

Moreover, with $\mathbf{J} = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$, each two-dimensional rotary block satisfies

$$\mathbf{a}^{\top} \mathbf{R}(\theta_i \Delta x) \mathbf{b} = (\mathbf{a}^{\top} \mathbf{b}) \cos(\theta_i \Delta x) + (\mathbf{a}^{\top} \mathbf{J} \mathbf{b}) \sin(\theta_i \Delta x), \tag{8}$$

with analogous forms for Δy and Δz. Hence, VTPE injects the signed relative 3D displacement into the attention mechanism as a content-conditioned, multi-frequency sinusoidal encoding, rather than collapsing the geometry into a scalar interatomic distance. See Figure 1 for an illustration.

Our proposed VTPE, while introducing limited additional memory or computational overhead, enables the model to perceive the directionality and relativity of each atom in the complex. Moreover, through the designed frequency allocation strategy and the θ_i in Equation 3, we ensure the symmetry and data-awareness of the rotations. Details are provided in Section 3.3.

3.3 Property Analysis

Having formalized VTPE and its relative-position structure in Section 3.2, we next examine three properties directly associated with its design. We first study the layer-wise behavior of the learned virtual trajectory, then analyze how the 3D frequency allocation controls directional sensitivity, and finally characterize how learnable rotation magnitudes affect the stability of VTPE under virtual-coordinate perturbations.

VLayer-wise Stabilization. As defined in Definition 1, the virtual trajectory $X'^{1:T-1}$ is a layer-wise sequence of intermediate 3D conformations rather than a trajectory in physical time. We empirically observe that its geometric changes become progressively smaller in later layers, suggesting a stabilization behavior of the learned virtual trajectory.

To quantify this tendency, for sample i at virtual step t, we define the protein and ligand centroids as

$$\mathbf{c}^t_{p,i} = \frac{1}{k_i} \sum_{j=1}^{k_i} \mathbf{X}'^t_{\text{protein},i,j}, \qquad \mathbf{c}^t_{l,i} = \frac{1}{l_i} \sum_{j=1}^{l_i} \mathbf{X}'^t_{\text{ligand},i,j},$$

where k_i and l_i are the corresponding numbers of atoms. The average centroid separation and its layer-wise change are

$$D^t_{\text{pl}} = \frac{1}{K} \sum_{i=1}^{K} \left\| \mathbf{c}^t_{p,i} - \mathbf{c}^t_{l,i} \right\|_2, \qquad V'^t = \Delta D^t_{\text{pl}} = D^{t+1}_{\text{pl}} - D^t_{\text{pl}}. \tag{9}$$

As shown in Figure 2, the magnitude of ΔD^t_{pl} generally decreases in later layers.

Rotational Symmetry. When extending rotary position encoding from one-dimensional sequences to 3D coordinates, a key design choice is how the x, y, and z components are assigned to the d feature dimensions. Since the base frequency $\theta_i = 10000^{-u(i-1)/d}$ varies monotonically with the dimension index, a sequential allocation assigns substantially different frequency ranges to the three coordinate axes, introducing an artificial axis-dependent sensitivity; see Figure 3(a). Interleaving the three coordinates alleviates this imbalance, as shown in Figure 3(b).

We further set $u = 6$ so that each coordinate axis follows the same RoPE frequency progression. Specifically, the d dimensions are equally divided among three spatial axes, and each planar rotation occupies two dimensions; hence each axis contains $d/6$ rotary blocks. Applying the standard RoPE frequency schedule independently to each axis gives

$$\theta_i = 10000^{-\frac{2(i-1)}{d/3}} = 10000^{-\frac{6(i-1)}{d}},$$

which corresponds exactly to $u = 6$ in Eq 3. See Appendix B.1 for the full rotation matrix. We next analyze the symmetry induced by the VTPE positional operator.

For a virtual coordinate $\mathbf{m}$ and perturbation $\boldsymbol{\delta} = (\delta_x, \delta_y, \delta_z)$, define the local change of the VTPE rotation by

$$\mathcal{D}_\Theta(\mathbf{m}, \boldsymbol{\delta})^2 = \frac{1}{2} \left\| \mathbf{R}^d_{\Theta,\mathbf{m}+\boldsymbol{\delta}} - \mathbf{R}^d_{\Theta,\mathbf{m}} \right\|_F^2 .$$

Proposition 2 (Frequency balance yields local isotropy)**.** *Let $\Theta_x, \Theta_y, \Theta_z$ be the effective frequency sets of the three axes, and let*

$$S_a = \sum_{\theta \in \Theta_a} \theta^2, \quad a \in \{x, y, z\}.$$

As $\|\boldsymbol{\delta}\|_2 \to 0$,

$$\mathcal{D}_\Theta(\mathbf{m}, \boldsymbol{\delta})^2 = S_x \delta_x^2 + S_y \delta_y^2 + S_z \delta_z^2 + O(\|\boldsymbol{\delta}\|_2^4).$$

Thus the leading-order VTPE positional response is locally isotropic, i.e., depends only on $\|\boldsymbol{\delta}\|_2$, if and only if $S_x = S_y = S_z$. Under the interleaved $u = 6$ construction in Definition 2, $\Theta_x = \Theta_y = \Theta_z$, so $S_x = S_y = S_z = S$ and

$$\mathcal{D}_\Theta(\mathbf{m}, \boldsymbol{\delta})^2 = S\|\boldsymbol{\delta}\|_2^2 + O(\|\boldsymbol{\delta}\|_2^4).$$

Thus, the proposed frequency allocation removes a preferred coordinate axis from the leading-order positional response of VTPE: sufficiently small displacements of equal Euclidean magnitude induce the same local change in the rotary operator regardless of direction, as shown in Figure 3 (c). A detailed proof is provided in Appendix B.3.

Rotation Magnitude Awareness. The base RoPE schedule fixes the spatial frequency θ_i of each rotary block a priori. However, different feature dimensions may require different spatial sensitivities. We therefore introduce a learnable scale λ_i and use the effective frequency

$$\widetilde{\theta}_i = \lambda_i \theta_i,$$

allowing the data to adapt the rotation magnitude across dimensions, as illustrated in Figure 3(d). The following result characterizes how these frequencies control the sensitivity of VTPE to perturbations of the virtual coordinates.

Proposition 3 (Lipschitz stability under virtual-coordinate perturbations)**.** *Let*

$$\mathbf{q}_m^{(t)} = \mathbf{W}_{q^t} \mathbf{e}_m'^{(t)}, \qquad \mathbf{k}_n^{(t)} = \mathbf{W}_{k^t} \mathbf{e}_n'^{(t)},$$

and let

$$s_{mn}^{(t)} = \frac{\mathbf{q}_m^{(t)\top} \mathbf{k}_n^{(t)}}{\sqrt{d}}$$

be the attention logit between atoms m and n. Suppose their virtual coordinates are perturbed as

$$\mathbf{m} \mapsto \mathbf{m} + \boldsymbol{\varepsilon}_m, \qquad \mathbf{n} \mapsto \mathbf{n} + \boldsymbol{\varepsilon}_n,$$

while the content representations remain fixed. With $\Omega = \max_i |\widetilde{\theta}_i|$, the perturbed logit $\widehat{s}_{mn}^{(t)}$ satisfies

$$\left| \widehat{s}_{mn}^{(t)} - s_{mn}^{(t)} \right| \le \frac{\|\mathbf{q}_m^{(t)}\|_2 \|\mathbf{k}_n^{(t)}\|_2}{\sqrt{d}} \, \Omega \, \|\boldsymbol{\varepsilon}_n - \boldsymbol{\varepsilon}_m\|_2 .$$

Table 1: Evaluation results on PoseBusters, RecentPDB, and PXM-Legacy benchmarks.

Model	training steps	PoseBusters			RecentPDB			PXM-Legacy		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	47.68	37.02	16.63	32.04	8.18	27.24	22.90	0.25	32.38
	15000	61.07	57.14	11.51	40.94	19.08	22.44	31.07	10.18	25.24
	20000	63.61	57.49	11.58	42.61	19.25	22.11	32.52	12.75	24.49
Protenix-base & VTPE	10000	**60.39**	**56.78**	**11.54**	**41.05**	**25.47**	**19.63**	**32.94**	**19.87**	**21.09**
	15000	**66.00**	**66.07**	**10.16**	**43.68**	**28.51**	**18.93**	**33.10**	**19.91**	**21.39**
	20000	**67.34**	**67.50**	**9.82**	**44.98**	**30.56**	**18.18**	**34.19**	**23.15**	**20.34**
Protenix-mini	10000	35.43	23.24	18.58	26.65	8.00	25.17	22.18	6.52	25.65
	15000	43.98	31.78	16.85	30.28	8.69	25.22	24.63	5.56	26.32
	20000	45.98	27.50	17.81	32.13	7.32	25.51	26.24	5.71	26.60
Protenix-mini & VTPE	10000	**49.06**	**44.05**	**13.86**	**34.54**	**19.12**	**21.26**	**29.48**	**15.60**	**22.14**
	15000	**54.71**	**56.55**	**11.29**	**37.45**	**23.18**	**20.18**	**30.61**	**18.88**	**21.44**
	20000	**55.36**	**54.88**	**11.80**	**37.83**	**23.13**	**20.10**	**30.91**	**18.66**	**21.41**
Protenix-tiny	10000	29.60	32.98	15.99	23.39	11.25	23.82	19.85	7.96	25.51
	15000	30.17	37.38	15.57	24.19	13.53	23.76	21.47	11.80	24.96
	20000	30.38	41.78	**14.04**	23.96	14.80	22.64	20.27	10.86	24.73
Protenix-tiny & VTPE	10000	**33.58**	**38.93**	**15.16**	**27.05**	**16.20**	**21.98**	**23.91**	**14.09**	**23.17**
	15000	**33.02**	**42.26**	**14.12**	**27.04**	**17.23**	**22.02**	**23.92**	**15.36**	**22.52**
	20000	**32.17**	**41.79**	14.16	**26.25**	**16.80**	**21.74**	**23.30**	**14.33**	**22.56**

Table 2: Effect of different training crop size: PoseBusters evaluated by LDDT and RMSD

Model	crop size	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	128	9.60	18.50	33.15	54.32	28.85	21.67	18.18
Protenix-base	256	17.77	31.91	54.25	80.20	45.99	47.50	13.09
Protenix-base	384	16.63	30.42	52.76	78.33	44.58	48.45	13.21
Protenix-base & VTPE	128	**19.47**	**34.12**	**55.52**	**78.32**	**46.81**	**49.05**	**13.31**
Protenix-base & VTPE	256	**26.62**	**45.41**	**68.76**	**87.01**	**56.86**	**65.95**	**10.09**
Protenix-base & VTPE	384	**22.85**	**39.03**	**60.47**	**79.79**	**50.47**	**60.72**	**10.62**
Protenix-mini	128	8.38	16.50	31.29	55.28	27.90	12.38	20.64
Protenix-mini	256	12.45	23.51	41.00	62.43	34.80	38.45	14.88
Protenix-mini	384	10.19	19.21	33.58	52.35	28.80	29.88	16.79
Protenix-mini & VTPE	128	**14.24**	**26.86**	**47.20**	**72.76**	**40.23**	**42.15**	**14.12**
Protenix-mini & VTPE	256	**22.35**	**40.11**	**64.05**	**84.77**	**52.75**	**62.62**	**10.13**
Protenix-mini & VTPE	384	**22.59**	**40.13**	**63.58**	**83.59**	**52.37**	**66.19**	**9.56**
Protenix-tiny	128	6.36	12.67	25.08	48.38	23.12	7.62	21.92
Protenix-tiny	256	8.43	16.73	32.54	59.45	29.29	32.14	16.26
Protenix-tiny	384	9.04	17.91	34.60	61.76	30.84	33.22	16.47
Protenix-tiny & VTPE	128	**9.51**	**18.77**	**35.83**	**62.39**	**31.62**	**38.22**	**14.97**
Protenix-tiny & VTPE	256	**9.99**	**19.67**	**37.23**	**63.46**	**32.58**	**50.00**	**12.29**
Protenix-tiny & VTPE	384	**10.82**	**21.27**	**40.09**	**67.08**	**34.80**	**50.12**	**12.26**

Proposition 3 shows that the attention logit depends on the virtual coordinates only through their relative displacement, and that its sensitivity is Lipschitz, with a constant controlled by the projected content norms and the maximum VTPE frequency. Larger effective frequencies yield finer spatial sensitivity but also amplify the worst-case response to virtual-coordinate noise; the learnable λ_i therefore lets VTPE trade off these two effects in a data-adaptive manner, as shown in Figure 3 (d). A detailed proof is provided in Appendix B.4.

Together with the relative-position property in Section 3.2, these analyses show that VTPE injects structured 3D virtual geometry into self-attention through a layer-wise geometric signal, an axis-balanced local positional response, and data-adaptive spatial sensitivity. We next evaluate whether these properties translate into improved complex structure prediction.

4 Experiments

4.1 Experimental Setup

Training data and compute budget. We train VTPE on subsets of WeightedPDB Abramson et al. (2024); Zhang et al. (2026), the same open-source training set used by AF3 and Protenix; see Appendix C.1 for details. For the benchmark experiments in Section 4.2, we use a 10k subset, training on 4 GPUs for 20k steps (eight passes), whereas AF3 and Protenix train on the full dataset for $6-7$ passes. For the sequence-length experiments in Section 4.3, we use a 2k subset, reused for the ablations in Section 4.4 for fair comparison; these are run on 2 GPUs for 5k steps due to limited compute. All baselines and VTPE use identical compute budgets.

Implementation Details. All models are initialized from Protenix. We orthogonally incorporate our proposed VTPE as an add-on module on top of the base architecture. In all other respects, we adhere as closely as possible to the training settings of AF3 and Protenix, including the optimizer, learning rate, and loss functions, keeping them strictly identical to those of the baselines. Detailed training configurations are provided in Appendix C.4. The default training crop size is 128, and the inference crop size follows the setting of Zhang et al. (2026).

Table 3: Effect of different inference crop size: PoseBusters evaluated by LDDT and RMSD

Model	crop size	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	128	**49.07**	69.24	85.71	95.59	74.89	**100.00**	2.66
Protenix-base	256	45.57	67.10	84.72	94.38	73.04	**93.18**	4.17
Protenix-base	384	41.09	62.35	80.90	91.66	69.12	82.31	6.19
Protenix-base	default	35.59	55.68	75.25	87.92	63.61	57.50	11.58
Protenix-base & VTPE	128	48.50	**72.75**	**90.82**	**98.34**	**77.52**	**100.00**	**2.03**
Protenix-base & VTPE	256	**47.60**	**72.02**	**89.07**	**96.53**	**76.23**	92.05	**3.64**
Protenix-base & VTPE	384	**42.84**	**66.91**	**85.70**	**94.84**	**72.48**	**90.17**	**4.66**
Protenix-base & VTPE	default	**36.86**	**59.74**	**80.60**	**92.55**	**67.34**	**67.50**	**9.82**

Table 4: Ablations: PoseBusters evaluated by LDDT and RMSD

Model	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base & VTPE	**19.47**	**34.12**	**55.52**	**78.32**	**46.81**	**49.05**	**13.31**
Protenix-base & VTPE w/o λ_i	16.08	28.95	48.86	73.28	41.75	40.71	14.53
Protenix-base & VTPE w/o $u = 6$	18.24	32.11	52.48	74.45	44.32	43.85	14.10
Protenix-base & VTPE w/o Interleaved and $u = 6$	15.17	26.89	44.95	67.88	38.70	37.14	15.08

Evaluation Benchmarks. We evaluate our method on two comprehensive benchmarks for complex structure prediction: RecentPDB Zhang et al. (2026) and PXM-Legacy Ma et al. (2025). From PXM-Legacy, we use four selected subsets and exclude the rest due to the prohibitive evaluation cost. In addition, given the critical role of protein-ligand interactions, we include PoseBusters Buttenschoen et al. (2024), a benchmark specifically designed for protein-ligand complex structure prediction. Detailed descriptions of the evaluation metrics are provided in Appendix C.3.

4.2 Results on various Benchmarks

Across model scales and benchmarks, our method consistently achieves superior performance on the majority of evaluated datasets. The two largest models each require 672 NVIDIA A100 GPU-hours, with training steps and wall-clock time comparable to those of prior positional encoding methods Wei et al. (2025); Ostmeier et al. (2024). Experimental results are reported in Table 1.Further details are deferred to the Appendix C.5.

4.3 VARYING TRAINING AND INFERENCE CROP SIZES

Long-range capability is typically assessed along two dimensions Wang et al. (2024; 2026); Su et al. (2024): (1) improved performance under a limited training crop size, and (2) improved performance across different inference crop sizes under the same training crop size. We use Protenix-base and Protenix-base & VTPE from the previous subsection, inference with different crop sizes. In addition, we train models with varying training crop sizes. Our method improves improves most reported metric on the same model and enables the smaller Protenix-Mini & VTPE to outperform Protenix-base, Experimental results are presented in Tables 2 and 3. Further details are deferred to the Appendix C.6.

4.4 Ablation Studies

In this section, we conduct a series of ablation studies on the proposed method. The results demonstrate the necessity of treating virtual trajectory as positional encodings and incorporating rotational symmetry and data awareness. Results are reported in Tables 4. Further details are deferred to the Appendix C.7.

5 Conclusion

In this work, we study how to enable foundation models to capture long-range biological dependencies in biomolecular complexes. We introduce VTPE, which models virtual intermediate states rather than only the final conformation, yielding a virtual trajectory with *layer-wise stabilization*, whose conformations serve as biological positional indices. We further analyze this stabilization and its relative geometry and 3D symmetry, showing that these properties resemble atomic interactions in real complexes, thereby enhancing long-range modeling and structural awareness. VTPE consistently outperforms baselines on most benchmarks. Future work includes scaling VTPE and extending it to broader AIDD applications, such as molecular dynamics and molecule generation.

AI USE STATEMENT

Kimi was used exclusively for language polishing, including grammar correction and clarity enhancement.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our results, we provide:

- Detailed descriptions of training configurations in Appendix C.4.
- The source code used for all experiments will be made publicly available as soon as possible.
- Complete dataset descriptions in Appendix C.1 and Appendix C.2.

REFERENCES

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.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Martin Buttenschoen, Garrett M Morris, and Charlotte M Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical science*, 15(9):3130–3139, 2024.

Chang Dai, Hongyu Shan, Mingyang Song, and Di Liang. Hope: Hyperbolic rotary positional encoding for stable long-range dependency modeling in large language models. *arXiv preprint arXiv:2509.05218*, 2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

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.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

Quan Li, Xike Xie, Chao Wang, and S Kevin Zhou. Prompt learning with extended kalman filter for pre-trained language models. In *IJCAI*, pp. 4452–4460, 2024.

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.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 61–68, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.8. URL https://aclanthology.org/2022.acl-short.8/.

Jonas Loos and Lorenz Linhardt. Latent diffusion u-net representations contain positional embeddings and anomalies. *arXiv preprint arXiv:2504.07008*, 2025.

Wenquan Lu, Yuechuan Yang, Kyle Lee, Yanshu Li, and Enqi Liu. Latent chain-of-thought? decoding the depth-recurrent transformer. *arXiv preprint arXiv:2507.02199*, 2025.

Wenzhi Ma, Zhenyu Liu, Jincai Yang, Chan Lu, Hanyu Zhang, and Wenzhi Xiao. From dataset curation to unified evaluation: Revisiting structure prediction benchmarks with pxmeter. *bioRxiv*, 2025. doi: 10.1101/2025.07.17.664878. URL https://www.biorxiv.org/content/early/2025/07/22/2025.07.17.664878.

Valerio Mariani, Marco Biasini, Alessandro Barbato, and Torsten Schwede. lddt: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21):2722–2728, 2013.

Sophie Ostmeier, Brian Axelrod, Maya Varma, Michael E Moseley, Akshay Chaudhari, and Curtis Langlotz. Liere: Lie rotational positional encodings. *arXiv preprint arXiv:2406.10322*, 2024.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

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.

Jianing Wang, Jin Jiang, Yang Liu, Mengdi Zhang, and Xunliang Cai. Prejudge-before-think: Enhancing large language models at test-time by process prejudge reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 4656–4673, 2025.

Xindi Wang, Mahsa Salmani, Parsa Omidi, Xiangyu Ren, Mehdi Rezagholizadeh, and Armaghan Eshaghi. Beyond the limits: A survey of techniques to extend the context length in large language models. *arXiv preprint arXiv:2402.02244*, 2024.

Zhaowei Wang, Wenhao Yu, Xiyu Ren, Jipeng Zhang, Yu Zhao, Rohit Saxena, Liang Cheng, Ginny Wong, Simon See, Pasquale Minervini, et al. Mmlongbench: Benchmarking long-context vision-language models effectively and thoroughly. *Advances in Neural Information Processing Systems*, 38, 2026.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Jian Tong, Haodong Duan, Qipeng Guo, Jiaqi Wang, et al. Videorope: What makes for good video rotary position embedding? *arXiv preprint arXiv:2502.05173*, 2025.

Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank J Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*, 2019.

Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah D Goodman. Quiet-star: Language models can teach themselves to think before speaking. *arXiv preprint arXiv:2403.09629*, 2024.

Yuxuan Zhang, Chengyue Gong, Jinyuan Sun, Jiaqi Guan, Milong Ren, Song Xue, Hanyu Zhang, Wenzhi Ma, Zhenyu Liu, Xinshi Chen, et al. Protenix-v2: Broadening the reach of structure prediction and biomolecular design. *bioRxiv*, pp. 2026–04, 2026.

Zheng Zhao, Yftah Ziser, and Shay B Cohen. Layer by layer: Uncovering where multi-task learning happens in instruction-tuned large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15195–15214, Miami, Florida, USA, November 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.emnlp-main.847.

Jiajun Zhu, Peihao Wang, Ruisi Cai, Jason D. Lee, Pan Li, and Zhangyang Wang. Rethinking addressing in language models via contextualized equivariant positional encoding. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 80155–80186. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/zhu25t.html.

A Appendix

B Details of Method

B.1 Details of rotation matrix

For $d = 768$, the complete rotation matrix is given in Equation 10:

$$\underbrace{\begin{pmatrix} q^{(0)} \\ q^{(1)} \\ q^{(6)} \\ q^{(7)} \\ \vdots \\ q^{(762)} \\ q^{(763)} \end{pmatrix}^{\top} \begin{pmatrix} \cos\theta_0\Delta x & -\sin\theta_0\Delta x & 0 & 0 & \cdots & 0 & 0 \\ \sin\theta_0\Delta x & \cos\theta_0\Delta x & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos\theta_1\Delta x & -\sin\theta_1\Delta x & \cdots & 0 & 0 \\ 0 & 0 & \sin\theta_1\Delta x & \cos\theta_1\Delta x & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos\theta_{127}\Delta x & -\sin\theta_{127}\Delta x \\ 0 & 0 & 0 & 0 & \cdots & \sin\theta_{127}\Delta x & \cos\theta_{127}\Delta x \end{pmatrix} \begin{pmatrix} k^{(0)} \\ k^{(1)} \\ k^{(6)} \\ k^{(7)} \\ \vdots \\ k^{(762)} \\ k^{(763)} \end{pmatrix}}_{\text{modeling } x \text{ dependency with interleaved frequency}}$$

$$+ \underbrace{\begin{pmatrix} q^{(2)} \\ q^{(3)} \\ q^{(8)} \\ q^{(9)} \\ \vdots \\ q^{(764)} \\ q^{(765)} \end{pmatrix}^{\top} \begin{pmatrix} \cos\theta_0\Delta y & -\sin\theta_0\Delta y & 0 & 0 & \cdots & 0 & 0 \\ \sin\theta_0\Delta y & \cos\theta_0\Delta y & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos\theta_1\Delta y & -\sin\theta_1\Delta y & \cdots & 0 & 0 \\ 0 & 0 & \sin\theta_1\Delta y & \cos\theta_1\Delta y & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos\theta_{127}\Delta y & -\sin\theta_{127}\Delta y \\ 0 & 0 & 0 & 0 & \cdots & \sin\theta_{127}\Delta y & \cos\theta_{127}\Delta y \end{pmatrix} \begin{pmatrix} k^{(2)} \\ k^{(3)} \\ k^{(8)} \\ k^{(9)} \\ \vdots \\ k^{(764)} \\ k^{(765)} \end{pmatrix}}_{\text{modeling } y \text{ dependency with interleaved frequency}}$$

$$+ \underbrace{\begin{pmatrix} q^{(4)} \\ q^{(5)} \\ q^{(10)} \\ q^{(11)} \\ \vdots \\ q^{(766)} \\ q^{(767)} \end{pmatrix}^{\top} \begin{pmatrix} \cos\theta_0\Delta z & -\sin\theta_0\Delta z & 0 & 0 & \cdots & 0 & 0 \\ \sin\theta_0\Delta z & \cos\theta_0\Delta z & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos\theta_1\Delta z & -\sin\theta_1\Delta z & \cdots & 0 & 0 \\ 0 & 0 & \sin\theta_1\Delta z & \cos\theta_1\Delta z & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos\theta_{127}\Delta z & -\sin\theta_{127}\Delta z \\ 0 & 0 & 0 & 0 & \cdots & \sin\theta_{127}\Delta z & \cos\theta_{127}\Delta z \end{pmatrix} \begin{pmatrix} k^{(4)} \\ k^{(5)} \\ k^{(10)} \\ k^{(11)} \\ \vdots \\ k^{(766)} \\ k^{(767)} \end{pmatrix}}_{\text{modeling } z \text{ dependency with interleaved frequency}} \tag{10}$$

B.2 Proof of Proposition 1

Proof. For clarity of exposition, we prove the case $d = 6$; the general even-dimensional case follows by applying the same argument blockwise. Fix $\Theta = (\theta_1, \theta_2, \theta_3)$, and for $m = (m_x, m_y, m_z) \in \mathbb{Z}^3$ define

$$R^6_{\Theta,m} = \operatorname{diag}(R(m_x\theta_1), R(m_y\theta_2), R(m_z\theta_3)), \qquad R(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}.$$

Then

$$(R^6_{\Theta,m})^{\top} R^6_{\Theta,n} = \operatorname{diag}\big(R(m_x\theta_1)^{\top} R(n_x\theta_1), R(m_y\theta_2)^{\top} R(n_y\theta_2), R(m_z\theta_3)^{\top} R(n_z\theta_3)\big).$$

For each block, using the identity $R(a)^\top R(b) = R(b-a)$, we have

$$\begin{aligned} R(m_x\theta_1)^\top R(n_x\theta_1) &= \begin{pmatrix} \cos(m_x\theta_1) & \sin(m_x\theta_1) \\ -\sin(m_x\theta_1) & \cos(m_x\theta_1) \end{pmatrix} \begin{pmatrix} \cos(n_x\theta_1) & -\sin(n_x\theta_1) \\ \sin(n_x\theta_1) & \cos(n_x\theta_1) \end{pmatrix} \\ &= \begin{pmatrix} \cos((n_x-m_x)\theta_1) & -\sin((n_x-m_x)\theta_1) \\ \sin((n_x-m_x)\theta_1) & \cos((n_x-m_x)\theta_1) \end{pmatrix} \\ &= R\big((n_x-m_x)\theta_1\big). \end{aligned}$$

The same calculation applies to the y- and z-blocks. Therefore,

$$(R^6_{\Theta,m})^\top R^6_{\Theta,n} = \mathrm{diag}(R((n_x-m_x)\theta_1), R((n_y-m_y)\theta_2), R((n_z-m_z)\theta_3)) = R^6_{\Theta,n-m}.$$

This establishes relative-position consistency for $d = 6$. The general even-dimensional case follows by applying the same blockwise argument to each two-dimensional rotary block. In particular, when $n = m$, we obtain

$$(R^6_{\Theta,m})^\top R^6_{\Theta,m} = R^6_{\Theta,0} = I_6.$$

□

B.3 Proof of Proposition 2

Proof. By Definition 2, the VTPE rotation matrix $\mathbf{R}^d_{\Theta,\mathbf{m}}$ is block diagonal. For each axis $a \in \{x, y, z\}$ and each frequency $\theta \in \Theta_a$, there is a 2×2 rotation block

$$R(\theta m_a) = \begin{pmatrix} \cos(\theta m_a) & -\sin(\theta m_a) \\ \sin(\theta m_a) & \cos(\theta m_a) \end{pmatrix}.$$

Since the Frobenius norm decomposes over block-diagonal entries,

$$\mathcal{D}_\Theta(\mathbf{m}, \boldsymbol{\delta})^2 = \sum_{a\in\{x,y,z\}} \sum_{\theta\in\Theta_a} \frac{1}{2} \left\| R(\theta(m_a+\delta_a)) - R(\theta m_a) \right\|_F^2.$$

Fix one block. Write

$$\varphi = \theta m_a, \qquad \Delta = \theta\delta_a.$$

Then

$$R(\varphi+\Delta) - R(\varphi) = \begin{pmatrix} \cos(\varphi+\Delta) - \cos\varphi & -\sin(\varphi+\Delta) + \sin\varphi \\ \sin(\varphi+\Delta) - \sin\varphi & \cos(\varphi+\Delta) - \cos\varphi \end{pmatrix}.$$

Therefore, by the definition of the Frobenius norm,

$$\begin{aligned} \|R(\varphi+\Delta) - R(\varphi)\|_F^2 &= \big[\cos(\varphi+\Delta) - \cos\varphi\big]^2 + \big[-\sin(\varphi+\Delta) + \sin\varphi\big]^2 \\ &\quad + \big[\sin(\varphi+\Delta) - \sin\varphi\big]^2 + \big[\cos(\varphi+\Delta) - \cos\varphi\big]^2 \\ &= 2\Big(\big[\cos(\varphi+\Delta) - \cos\varphi\big]^2 + \big[\sin(\varphi+\Delta) - \sin\varphi\big]^2\Big). \end{aligned}$$

Using the sum-to-product identities,

$$\cos(\varphi+\Delta) - \cos\varphi = -2\sin\left(\varphi + \frac{\Delta}{2}\right)\sin\left(\frac{\Delta}{2}\right),$$

and

$$\sin(\varphi+\Delta) - \sin\varphi = 2\cos\left(\varphi + \frac{\Delta}{2}\right)\sin\left(\frac{\Delta}{2}\right).$$

Substituting these into the previous display gives

$$\begin{aligned} \|R(\varphi+\Delta) - R(\varphi)\|_F^2 &= 2\left(4\sin^2\left(\frac{\Delta}{2}\right)\sin^2\left(\varphi+\frac{\Delta}{2}\right) + 4\sin^2\left(\frac{\Delta}{2}\right)\cos^2\left(\varphi+\frac{\Delta}{2}\right)\right) \\ &= 8\sin^2\left(\frac{\Delta}{2}\right)\left[\sin^2\left(\varphi+\frac{\Delta}{2}\right) + \cos^2\left(\varphi+\frac{\Delta}{2}\right)\right] \\ &= 8\sin^2\left(\frac{\Delta}{2}\right). \end{aligned}$$

Hence

$$\frac{1}{2}\left\|R(\varphi+\Delta)-R(\varphi)\right\|_F^2 = 4\sin^2\left(\frac{\Delta}{2}\right) = 2(1-\cos\Delta).$$

Using the Taylor expansion

$$\cos\Delta = 1-\frac{\Delta^2}{2}+\frac{\Delta^4}{24}+O(\Delta^6),$$

we obtain

$$2(1-\cos\Delta) = \Delta^2-\frac{1}{12}\Delta^4+O(\Delta^6).$$

Since $\Delta=\theta\delta_a$, it follows that

$$\frac{1}{2}\left\|R(\theta(m_a+\delta_a))-R(\theta m_a)\right\|_F^2 = \theta^2\delta_a^2-\frac{1}{12}\theta^4\delta_a^4+O(\theta^6\delta_a^6).$$

Summing over all axes and all frequencies,

$$\begin{aligned}\mathcal{D}_\Theta(\mathbf{m},\boldsymbol{\delta})^2 &= \sum_{a\in\{x,y,z\}}\sum_{\theta\in\Theta_a}\left(\theta^2\delta_a^2-\frac{1}{12}\theta^4\delta_a^4+O(\theta^6\delta_a^6)\right)\\ &= \sum_{a\in\{x,y,z\}}\left(\sum_{\theta\in\Theta_a}\theta^2\right)\delta_a^2-\frac{1}{12}\sum_{a\in\{x,y,z\}}\left(\sum_{\theta\in\Theta_a}\theta^4\right)\delta_a^4+O\left(\sum_{a\in\{x,y,z\}}\sum_{\theta\in\Theta_a}\theta^6\delta_a^6\right).\end{aligned}$$

By definition,

$$S_a=\sum_{\theta\in\Theta_a}\theta^2,\qquad a\in\{x,y,z\}.$$

Since the frequency sets are finite, the sums $\sum_{\theta\in\Theta_a}\theta^4$ and $\sum_{\theta\in\Theta_a}\theta^6$ are bounded. Moreover, $|\delta_a|\le\|\boldsymbol{\delta}\|_2$, so

$$\delta_a^4\le\|\boldsymbol{\delta}\|_2^4,\qquad \delta_a^6\le\|\boldsymbol{\delta}\|_2^6.$$

Therefore the quartic and higher-order terms are absorbed into $O(\|\boldsymbol{\delta}\|_2^4)$, and hence

$$\mathcal{D}_\Theta(\mathbf{m},\boldsymbol{\delta})^2 = S_x\delta_x^2+S_y\delta_y^2+S_z\delta_z^2+O(\|\boldsymbol{\delta}\|_2^4).$$

Thus, at leading order, the local positional response is the quadratic form

$$Q(\boldsymbol{\delta}) = S_x\delta_x^2+S_y\delta_y^2+S_z\delta_z^2.$$

Local isotropy means that this leading-order response depends only on $\|\boldsymbol{\delta}\|_2$, not on the direction of $\boldsymbol{\delta}$. Equivalently, there exists a constant c such that

$$Q(\boldsymbol{\delta}) = c\|\boldsymbol{\delta}\|_2^2$$

for all sufficiently small $\boldsymbol{\delta}$. Taking $\boldsymbol{\delta}=\epsilon\mathbf{e}_a$ and letting $\epsilon\to0$ gives

$$S_a=c,\qquad a\in\{x,y,z\}.$$

Hence $S_x=S_y=S_z$. Conversely, if $S_x=S_y=S_z=S$, then

$$Q(\boldsymbol{\delta}) = S(\delta_x^2+\delta_y^2+\delta_z^2) = S\|\boldsymbol{\delta}\|_2^2,$$

which is isotropic. Therefore the VTPE positional operator is locally isotropic if and only if $S_x=S_y=S_z$.

Under the interleaved $u=6$ construction in Definition 2, the three axes share the same frequency schedule, so

$$\Theta_x=\Theta_y=\Theta_z.$$

Therefore

$$S_x=S_y=S_z=S,$$

and the expansion becomes

$$\mathcal{D}_\Theta(\mathbf{m},\boldsymbol{\delta})^2 = S\|\boldsymbol{\delta}\|_2^2+O(\|\boldsymbol{\delta}\|_2^4).$$

□

B.4 Proof of Proposition 3

Proof. Let

$$\mathbf{q}_m = \mathbf{W}_{q^t}\mathbf{e}'^{(t)}_m, \qquad \mathbf{k}_n = \mathbf{W}_{k^t}\mathbf{e}'^{(t)}_n, \qquad \boldsymbol{\delta} = \boldsymbol{\varepsilon}_n - \boldsymbol{\varepsilon}_m.$$

By the VTPE construction,

$$s^{(t)}_{mn} = \frac{1}{\sqrt{d}}\mathbf{q}_m^\top \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}\mathbf{k}_n, \qquad \widehat{s}^{(t)}_{mn} = \frac{1}{\sqrt{d}}\mathbf{q}_m^\top \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}+\boldsymbol{\delta}}\mathbf{k}_n.$$

The VTPE operator is a block-wise rotation, so it is multiplicative in its positional argument:

$$\mathbf{R}_{\Theta,\mathbf{a}+\mathbf{b}} = \mathbf{R}_{\Theta,\mathbf{a}}\mathbf{R}_{\Theta,\mathbf{b}} = \mathbf{R}_{\Theta,\mathbf{b}}\mathbf{R}_{\Theta,\mathbf{a}}.$$

Applying this with $\mathbf{a} = \mathbf{n} - \mathbf{m}$ and $\mathbf{b} = \boldsymbol{\delta}$ gives

$$\mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}+\boldsymbol{\delta}} = \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}\mathbf{R}_{\Theta,\boldsymbol{\delta}}.$$

Therefore

$$\begin{aligned}\widehat{s}^{(t)}_{mn} - s^{(t)}_{mn} &= \frac{1}{\sqrt{d}}\mathbf{q}_m^\top \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}\mathbf{R}_{\Theta,\boldsymbol{\delta}}\mathbf{k}_n - \frac{1}{\sqrt{d}}\mathbf{q}_m^\top \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}\mathbf{k}_n \\ &= \frac{1}{\sqrt{d}}\mathbf{q}_m^\top \mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}\left(\mathbf{R}_{\Theta,\boldsymbol{\delta}} - \mathbf{I}\right)\mathbf{k}_n.\end{aligned}$$

Since $\mathbf{R}_{\Theta,\mathbf{n}-\mathbf{m}}$ is orthogonal, Cauchy–Schwarz and the spectral-norm bound give

$$\left|\widehat{s}^{(t)}_{mn} - s^{(t)}_{mn}\right| \le \frac{1}{\sqrt{d}}\|\mathbf{q}_m\|_2\|\mathbf{k}_n\|_2 \left\|\mathbf{R}_{\Theta,\boldsymbol{\delta}} - \mathbf{I}\right\|_2.$$

It remains to bound the last term. The matrix $\mathbf{R}_{\Theta,\boldsymbol{\delta}}$ is block diagonal, so its spectral norm is the maximum spectral norm over its blocks:

$$\left\|\mathbf{R}_{\Theta,\boldsymbol{\delta}} - \mathbf{I}\right\|_2 = \max_{i,\, a\in\{x,y,z\}} \left\|R(\widetilde{\theta}_i\delta_a) - \mathbf{I}\right\|_2.$$

For a single 2×2 rotation block with angle α, we have

$$R(\alpha) - \mathbf{I} = \begin{pmatrix}\cos\alpha - 1 & -\sin\alpha \\ \sin\alpha & \cos\alpha - 1\end{pmatrix}.$$

Its squared spectral norm is the largest eigenvalue of $(R(\alpha)-\mathbf{I})^\top(R(\alpha)-\mathbf{I})$. Using $R(\alpha)^\top R(\alpha) = \mathbf{I}$ and $R(\alpha) + R(\alpha)^\top = 2\cos\alpha\,\mathbf{I}$, we obtain

$$\begin{aligned}(R(\alpha) - \mathbf{I})^\top(R(\alpha) - \mathbf{I}) &= R(\alpha)^\top R(\alpha) - R(\alpha)^\top - R(\alpha) + \mathbf{I} \\ &= 2\mathbf{I} - R(\alpha) - R(\alpha)^\top \\ &= 2(1-\cos\alpha)\mathbf{I}.\end{aligned}$$

Hence

$$\|R(\alpha) - \mathbf{I}\|_2 = \sqrt{2(1-\cos\alpha)} = 2\left|\sin\frac{\alpha}{2}\right|,$$

where we used $1 - \cos\alpha = 2\sin^2(\alpha/2)$. Setting $\alpha = \widetilde{\theta}_i\delta_a$ and applying $|\sin x| \le |x|$ yields

$$\left\|R(\widetilde{\theta}_i\delta_a) - \mathbf{I}\right\|_2 = 2\left|\sin\frac{\widetilde{\theta}_i\delta_a}{2}\right| \le |\widetilde{\theta}_i|\,|\delta_a|.$$

Since $|\delta_a| \le \|\boldsymbol{\delta}\|_2$ and $|\widetilde{\theta}_i| \le \Omega = \max_i|\widetilde{\theta}_i|$, it follows that

$$\left\|R(\widetilde{\theta}_i\delta_a) - \mathbf{I}\right\|_2 \le |\widetilde{\theta}_i|\,\|\boldsymbol{\delta}\|_2 \le \Omega\|\boldsymbol{\delta}\|_2.$$

Taking the maximum over all blocks gives

$$\left\|\mathbf{R}_{\Theta,\boldsymbol{\delta}} - \mathbf{I}\right\|_2 \le \Omega\|\boldsymbol{\delta}\|_2.$$

Substituting $\boldsymbol{\delta} = \boldsymbol{\varepsilon}_n - \boldsymbol{\varepsilon}_m$, $\mathbf{q}_m = \mathbf{W}_{q^t}\mathbf{e}'^{(t)}_m$, and $\mathbf{k}_n = \mathbf{W}_{k^t}\mathbf{e}'^{(t)}_n$.

Therefore,

$$\left|\widehat{s}^{(t)}_{mn} - s^{(t)}_{mn}\right| \le \frac{1}{\sqrt{d}}\|\mathbf{q}_m\|_2\|\mathbf{k}_n\|_2\left\|\mathbf{R}_{\Theta,\boldsymbol{\delta}} - \mathbf{I}\right\|_2 \le \frac{\|\mathbf{q}_m\|_2\|\mathbf{k}_n\|_2}{\sqrt{d}}\Omega\|\boldsymbol{\delta}\|_2.$$

□

C Details of Experiments

C.1 Training Datasets

We use subsets of the publicly released WeightedPDB training set as the training data for all experiments. WeightedPDB serves as the primary PDB-derived training corpus for both AlphaFold 3 Abramson et al. (2024) and Protenix Zhang et al. (2026).

WeightedPDB is constructed from ground-truth experimental structures curated from the wwPDB archive, with training cutoff dates aligned to those used by AlphaFold 3 and Protenix (e.g., 2021-09-30) to prevent evaluation leakage Zhang et al. (2026). The dataset therefore contains a broad spectrum of biomolecular complexes, including protein–protein, protein–nucleic acid (DNA and RNA), protein–ligand, protein–ion, and multi-chain assemblies. Protein and nucleic acid chains are organized into sequence-identity clusters (e.g., 40% identity threshold for proteins) so that sampling can be performed in a redundancy-reduced manner Zhang et al. (2026).

The overall size of WeightedPDB exceeds 2.4 million chain and interface entries, drawn from wwPDB structures released before the training cutoff. Due to computational resource constraints, we do not train on the full WeightedPDB corpus. Instead, we extract two smaller subsets containing 1×10^4 and 2×10^3 entries, respectively, and use them as the training sets for different experimental configurations. The 1×10^4-entry subset is used for our main experiments, while the 2×10^3-entry subset is reserved for other experiments. Both subsets are drawn from the same WeightedPDB pool and retain the original chain/interface composition and structural annotations.

C.2 Evaluation Benchmarks

PoseBusters. PoseBusters Buttenschoen et al. (2024) is a benchmark of 308 protein–ligand cocrystal complexes curated from the PDB, all deposited after 2021, with one unique ligand per complex. It is accompanied by a test suite that checks the chemical and geometric validity of ligand poses, including molecular formula, connectivity, chirality, bond lengths and angles, planarity, steric clashes, and protein–ligand clashes.

RecentPDB. RecentPDB Ma et al. (2025) is a benchmark curated from recently deposited PDB structures, designed to evaluate generalization to novel sequences through strict homology filtering against common training sets. It comprises low-homology protein subsets, including antibody–antigen complexes, monomers, and other subsets.

PXM-Legacy. PXM-Legacy is a benchmark collection released alongside PXMeter Ma et al. (2025), comprising evaluation sets from earlier benchmark versions. It includes dsDNA–Protein, RNA–Protein, and other subsets, with evaluation metrics including RMSD and LDDT. The benchmark data is released under the CC0 1.0 Universal license.

Evaluation task definitions. RecentPDB and PXM-Legacy contain numerous test sets for structure prediction of various complexes and monomers. Their specific definitions are as follows: `rna_prot` denotes an RNA–protein complex; `antibody` denotes an antibody-only case; `intra_dna` denotes a DNA monomer; `intra_rna` denotes an RNA monomer; `prot_prot` denotes a protein–protein complex; `antibody_antigen` denotes an antibody–antigen complex; `dna_prot` denotes a DNA–protein complex; `intra_prot` denotes a protein monomer; and `ligand_prot` denotes a protein–ligand complex.

C.3 Evaluation Metrics

LDDT. The Local Distance Difference Test (LDDT) is an alignment-free metric that measures local geometric agreement Mariani et al. (2013). For atoms i and j, let

$$d_{ij}^{\text{ref}} = \left\|\mathbf{x}_i^{\text{ref}} - \mathbf{x}_j^{\text{ref}}\right\|_2, \qquad d_{ij}^{\text{pred}} = \left\|\mathbf{x}_i^{\text{pred}} - \mathbf{x}_j^{\text{pred}}\right\|_2, \tag{11}$$

where $\|\cdot\|_2$ denotes the Euclidean norm. Let $\mathcal{P} = \{(i,j) : d_{ij}^{\text{ref}} < R\}$ be the set of atom pairs within an inclusion radius R in the reference structure (typically $R = 15$ Å). For a tolerance $t \in$

$\{0.5, 1, 2, 4\}$ Å,

$$\text{LDDT-t} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \mathbf{1}\big[\,|d_{ij}^{\text{pred}} - d_{ij}^{\text{ref}}| < t\,\big], \tag{12}$$

The LDDT-mean is the average over the four thresholds:

$$\text{LDDT-mean} = \frac{1}{4} \sum_{t\in\{0.5,1,2,4\}} \text{LDDT-t}. \tag{13}$$

RMSD. We measure the geometric deviation between a predicted ligand pose and the crystal reference with the symmetry-aware RMSD, denoted as *RMSD-dist* Lin et al. (2023); Jumper et al. (2021):

$$\text{RMSD-dist} = \min_{\sigma\in S} \sqrt{\frac{1}{N}\sum_{i=1}^{N} \big\|\mathbf{x}_i^{\text{pred}} - \mathbf{x}_{\sigma(i)}^{\text{ref}}\big\|_2^2}, \tag{14}$$

where S is the set of atom permutations that preserve the molecular graph and $\|\cdot\|_2$ is the Euclidean norm; lower values indicate better agreement. We further report *RMSD-u*, the success rate at threshold τ, defined as the fraction of predictions whose RMSD-distance is at most τ:

$$\text{RMSD-}\tau = \frac{1}{M}\sum_{m=1}^{M} \mathbf{1}\big[\,\text{RMSD-dist}_m \le \tau\,\big], \tag{15}$$

where M is the number of predictions and $\tau \in \{5, 10, 15, 20\}$ Å for RMSD-5, RMSD-10, RMSD-15, and RMSD-20, respectively. We define RMSD-mean as the arithmetic mean of these four success rates.

C.4 Training Configurations

For the baseline methods, we adhere strictly to the training configurations of Abramson et al. (2024); Zhang et al. (2026). Owing to the additional operators introduced by our method, certain training details are configured differently. We provide the detailed training configurations in Table 5.

C.5 Details of Results on various Benchmarks

In this section, we supplement the results in Section 4.2 of the main paper and further report evaluation results for PoseBusters, RecentPDB, and PXM-Legacy across multiple models. The corresponding experimental results are presented in Tables 6 – 38.

C.6 DETAILS OF VARYING TRAINING AND INFERENCE CROP SIZES

In this section, we supplement the results in Section 4.3 of the main paper and further report evaluation RMSD results for PoseBusters across multiple models. The corresponding experimental results are presented in Tables 39 and 40.

C.7 Details of ablation

In this section, we supplement the results in Section 4.4 of the main paper and further report evaluation RMSD results for PoseBusters across multiple models. The corresponding experimental results are presented in Table 41.

Table 5: Training configurations used in our experiments.

Configuration	Value
Crop size (tokens)	128
Learning rate	0.0018
LR scheduler	Cosine annealing
Warmup steps	1000
Min LR ratio	0.1

Table 6: Details of various benchmarks. PoseBusters evaluated by LDDT and RMSD.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	19.86	35.20	57.26	78.06	47.68	16.63
	15000	32.26	51.95	72.71	86.91	61.07	11.51
	20000	35.59	55.68	75.25	87.92	63.61	11.58
Protenix-base & VTPE	10000	**29.55**	**50.03**	**72.96**	**89.40**	**60.39**	**11.54**
	15000	**36.14**	**58.13**	**78.85**	**91.33**	**66.00**	**10.16**
	20000	**36.86**	**59.74**	**80.60**	**92.55**	**67.34**	**9.82**
Protenix-mini	10000	11.13	21.89	40.81	67.82	35.43	18.58
	15000	16.33	30.49	52.20	76.69	43.98	16.85
	20000	18.48	33.40	54.82	76.96	45.98	17.81
Protenix-mini & VTPE	10000	**19.51**	**35.60**	**58.83**	**83.57**	**49.06**	**13.86**
	15000	**24.18**	**42.47**	**66.07**	**86.42**	**54.71**	**11.29**
	20000	**25.09**	**43.41**	**66.64**	**86.70**	**55.36**	**11.80**
Protenix-tiny	10000	9.02	17.78	33.78	57.82	29.60	15.99
	15000	9.15	18.03	34.31	59.16	30.17	15.57
	20000	9.29	18.29	34.67	59.24	30.38	**14.04**
Protenix-tiny & VTPE	10000	**10.23**	**20.17**	**38.33**	**65.61**	**33.58**	**15.16**
	15000	**9.95**	**19.40**	**37.50**	**64.98**	**33.02**	**14.12**
	20000	**9.63**	**19.02**	**36.42**	**63.64**	**32.17**	14.16

Table 7: Details of various benchmarks. RMSD results on PoseBusters.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	8.57	35.71	46.66	57.14	37.02	16.63
	15000	31.42	54.76	65.71	76.66	57.14	11.51
	20000	36.19	54.28	63.80	75.71	57.49	11.58
Protenix-base & VTPE	10000	**29.05**	**57.14**	**65.71**	**75.24**	**56.78**	**11.54**
	15000	**51.90**	**62.86**	**70.95**	**78.57**	**66.07**	**10.16**
	20000	**53.81**	**64.29**	**71.90**	**80.00**	**67.50**	**9.82**
Protenix-mini	10000	0.95	10.95	24.38	56.67	23.24	18.58
	15000	6.19	16.19	39.52	65.23	31.78	16.85
	20000	6.67	12.38	31.90	59.04	27.50	17.81
Protenix-mini & VTPE	10000	**11.90**	**35.24**	**53.33**	**75.71**	**44.05**	**13.86**
	15000	**21.43**	**51.43**	**69.05**	**84.29**	**56.55**	**11.29**
	20000	**22.38**	**50**	**66.19**	**80.95**	**54.88**	**11.80**
Protenix-tiny	10000	0	26.19	40.95	64.76	32.98	15.99
	15000	0	35.24	47.14	67.14	37.38	15.57
	20000	0	**39.52**	**54.29**	73.33	41.78	**14.04**
Protenix-tiny & VTPE	10000	**0.95**	**35.24**	**52.39**	**67.14**	**38.93**	**15.16**
	15000	**0.95**	**39.05**	**54.76**	**74.29**	**42.26**	**14.12**
	20000	**0.95**	38.10	53.81	**74.29**	**41.79**	14.16

Table 8: Details of various benchmarks. LDDT and RMSD results on RecentPDB subtasks 1-3.

Model	training steps	RecentPDB: rna_prot, antibody, intra_dna								
		rna_prot			antibody			intra_dna		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	25.57	5.68	30.09	38.18	0.00	29.01	20.19	3.88	29.83
	15000	32.99	15.91	26.00	47.51	0.00	26.94	28.88	22.84	22.05
	20000	32.95	13.64	26.11	49.52	1.31	26.45	32.04	22.85	19.99
Protenix-base & VTPE	10000	**33.57**	**19.32**	**23.01**	**46.39**	**9.21**	**23.99**	**31.47**	**30.17**	**17.05**
	15000	**34.19**	**22.73**	**22.72**	**52.03**	**14.47**	**21.53**	**30.95**	**32.76**	**17.32**
	20000	**35.02**	**25.00**	**21.59**	**53.58**	**11.84**	**20.75**	**32.71**	**34.05**	**16.56**
Protenix-mini	10000	22.83	6.82	27.83	29.50	0.00	28.32	20.05	10.78	22.84
	15000	25.29	5.68	28.24	34.78	0.00	28.30	19.98	5.17	25.28
	20000	27.54	6.82	28.56	36.43	0.00	28.80	21.81	5.17	24.19
Protenix-mini & VTPE	10000	**28.48**	**15.91**	**24.69**	**36.42**	**1.31**	**25.38**	**28.92**	**25.86**	**18.32**
	15000	**31.11**	**18.18**	**23.63**	**40.27**	**3.95**	**24.48**	**29.40**	**27.59**	**17.78**
	20000	**30.98**	**17.05**	**23.16**	**41.12**	**3.95**	**24.51**	**30.23**	**29.31**	**17.25**
Protenix-tiny	10000	21.18	9.09	27.20	25.25	0.00	27.58	20.19	15.09	21.47
	15000	21.49	11.37	26.54	26.21	0.00	28.78	21.04	17.67	20.64
	20000	21.33	12.50	25.68	26.70	0.00	26.56	19.99	21.12	20.29
Protenix-tiny & VTPE	10000	**23.94**	**13.63**	**24.58**	**29.52**	**1.31**	**25.72**	**22.73**	**21.12**	**19.15**
	15000	**24.27**	**12.50**	**25.23**	**29.34**	**1.31**	**26.57**	**22.40**	**23.28**	**18.74**
	20000	**23.64**	**13.63**	**24.72**	**28.38**	**1.31**	**25.81**	**21.89**	**21.98**	**18.83**

Table 9: Details of various benchmarks. LDDT and RMSD results on RecentPDB subtasks 4-6.

Model	training steps	RecentPDB: intra_rna, prot_prot, antibody_antigen								
		intra_rna			prot_prot			antibody_antigen		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	23.51	13.54	27.89	37.11	4.90	28.35	38.42	0.00	28.31
	15000	28.50	16.67	26.29	45.58	16.83	22.21	48.24	2.78	26.69
	20000	**29.58**	17.71	26.15	47.02	17.32	22.08	50.62	2.78	26.51
Protenix-base & VTPE	10000	**29.10**	**23.96**	**21.88**	**45.45**	**23.04**	**19.75**	**46.55**	**8.34**	**23.94**
	15000	**28.72**	**21.88**	**22.37**	**48.66**	**23.37**	**19.80**	**52.48**	**15.28**	**21.24**
	20000	29.17	**29.17**	**21.36**	**50.44**	**28.27**	**18.84**	**54.42**	**12.50**	**20.86**
Protenix-mini	10000	19.78	5.21	29.22	30.65	8.50	24.53	29.92	0.00	28.39
	15000	22.22	11.46	27.23	35.44	10.13	24.32	35.09	0.00	28.06
	20000	24.06	8.34	28.31	36.77	7.35	25.05	36.78	0.00	28.67
Protenix-mini & VTPE	10000	**26.67**	**19.79**	**23.53**	**38.94**	**17.48**	**21.35**	**36.75**	**4.17**	**24.94**
	15000	**28.20**	**19.79**	**23.24**	**41.74**	**22.55**	**20.14**	**41.02**	**5.55**	**24.00**
	20000	**27.47**	**19.79**	**22.58**	**42.13**	**19.94**	**20.83**	**41.97**	**5.55**	**24.10**
Protenix-tiny	10000	19.06	13.54	26.35	25.64	8.99	23.60	25.04	0.00	27.71
	15000	19.76	15.62	25.97	25.83	9.80	24.02	26.09	0.00	28.81
	20000	19.33	14.59	25.50	25.58	12.26	22.85	26.80	0.00	26.76
Protenix-tiny & VTPE	10000	**21.91**	**15.63**	**24.27**	**29.50**	**15.36**	**22.30**	**29.70**	0.00	**25.76**
	15000	**22.88**	**18.75**	**24.38**	**29.07**	**15.68**	**21.86**	**29.67**	**1.39**	**26.44**
	20000	**22.00**	**17.71**	**24.04**	**28.40**	**16.34**	**21.74**	**28.39**	0.00	**25.83**

Table 10: Details of various benchmarks. LDDT and RMSD results on RecentPDB subtasks 7-9.

Model	training steps	RecentPDB: dna_prot, intra_prot, ligand_prot								
		dna_prot			intra_prot			ligand_prot		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	25.38	4.05	28.78	40.42	22.88	21.72	39.60	18.67	21.22
	15000	37.44	26.35	17.67	49.86	37.37	17.11	49.50	33.00	17.03
	20000	39.32	27.03	17.64	51.33	37.63	17.03	51.10	33.00	17.04
Protenix-base & VTPE	10000	**37.97**	**38.51**	**15.32**	**49.65**	**40.99**	**15.71**	**49.26**	**35.67**	**16.01**
	15000	**39.19**	**39.19**	**15.23**	**53.56**	**45.23**	**14.76**	**53.38**	**41.66**	**15.37**
	20000	**40.71**	**43.92**	**14.17**	**54.59**	**46.65**	**14.40**	**54.17**	**43.67**	**15.10**
Protenix-mini	10000	23.81	10.81	21.87	32.33	17.23	21.58	31.00	12.66	21.95
	15000	24.78	6.08	23.89	37.82	21.38	20.65	37.11	18.34	21.02
	20000	27.42	4.73	23.01	39.44	18.82	21.31	38.92	14.67	21.66
Protenix-mini & VTPE	10000	**32.93**	**29.05**	**17.12**	**41.60**	**31.54**	**17.74**	**40.15**	**27.00**	**18.25**
	15000	**34.99**	**37.84**	**15.52**	**45.61**	**38.87**	**16.29**	**44.68**	**34.34**	**16.54**
	20000	**35.76**	**39.87**	**15.29**	**45.92**	**38.34**	**16.37**	**44.88**	**34.33**	**16.77**
Protenix-tiny	10000	21.63	18.25	20.22	26.56	20.58	20.17	26.00	15.67	20.05
	15000	22.97	20.27	19.53	27.10	23.67	19.96	27.24	23.34	19.59
	20000	22.22	22.30	18.69	26.92	26.77	18.74	26.74	23.67	**18.67**
Protenix-tiny & VTPE	10000	**24.89**	**25.68**	**18.63**	**30.44**	**28.09**	**18.53**	**30.83**	**25.00**	**18.84**
	15000	**24.84**	**28.38**	**17.81**	**30.26**	**28.45**	**18.46**	**30.63**	**25.33**	**18.70**
	20000	**24.24**	**27.03**	**17.80**	**29.64**	**28.53**	**18.20**	**29.66**	**24.67**	18.72

Table 11: Details of various benchmarks. LDDT and RMSD results on PXM-Legacy subtasks 1-2.

Model	training steps	PXM-Legacy: intra_dna, rna_prot					
		intra_dna			rna_prot		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	19.83	0.00	34.00	26.13	0.43	30.51
	15000	30.30	9.45	24.84	32.03	10.77	24.63
	20000	32.05	11.89	24.33	33.09	12.93	23.91
Protenix-base & VTPE	10000	**31.03**	**16.92**	**21.40**	**34.90**	**21.12**	**20.54**
	15000	**31.38**	**18.29**	**21.56**	**35.07**	**21.55**	**20.86**
	20000	**32.27**	**21.95**	**20.78**	**36.52**	**25.00**	**19.21**
Protenix-mini	10000	21.04	3.66	26.00	23.23	8.62	24.91
	15000	22.20	0.46	27.37	26.85	9.48	24.95
	20000	24.15	1.68	27.23	28.38	9.05	25.71
Protenix-mini & VTPE	10000	**27.59**	**10.82**	**22.88**	**31.41**	**19.40**	**21.02**
	15000	**28.76**	**16.31**	**22.10**	**32.41**	**21.55**	**20.22**
	20000	**29.26**	**16.61**	**21.95**	**32.57**	**20.26**	**20.50**
Protenix-tiny	10000	19.11	4.88	26.27	20.48	9.91	24.31
	15000	21.04	7.01	25.66	21.88	15.52	23.96
	20000	19.68	6.10	25.37	21.00	14.22	23.76
Protenix-tiny & VTPE	10000	**23.16**	**9.00**	**24.04**	**24.63**	**16.81**	**22.38**
	15000	**23.07**	**10.21**	**23.31**	**24.76**	**19.40**	**21.39**
	20000	**22.49**	**9.30**	**23.30**	**24.10**	**17.67**	**21.56**

Table 12: Details of various benchmarks. LDDT and RMSD results on PXM-Legacy subtasks 3-4.

Model	training steps	PXM-Legacy: dna_prot, intra_rna					
		dna_prot			intra_rna		
		LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)	LDDT_mean (↑)	RMSD_mean (↑)	RMSD_dist (↓)
Protenix-base	10000	19.20	0.00	34.71	26.43	0.58	30.31
	15000	29.41	7.70	26.23	32.52	12.79	25.27
	20000	**31.28**	10.49	25.67	33.68	15.70	24.06
Protenix-base & VTPE	10000	**29.89**	**15.29**	**22.63**	**35.92**	**26.16**	**19.77**
	15000	**30.31**	**15.40**	**22.71**	**35.66**	**24.42**	**20.45**
	20000	30.92	**17.75**	**21.88**	**37.04**	**27.91**	**19.48**
Protenix-mini	10000	20.86	3.90	26.74	23.61	9.89	24.94
	15000	22.13	0.67	28.28	27.33	11.63	24.68
	20000	23.69	2.23	28.06	28.73	9.88	25.40
Protenix-mini & VTPE	10000	**27.05**	**9.49**	**23.93**	**31.89**	**22.67**	**20.74**
	15000	**28.26**	**13.84**	**23.20**	**32.99**	**23.84**	**20.23**
	20000	**28.70**	**14.51**	**23.21**	**33.11**	**23.26**	**19.99**
Protenix-tiny	10000	19.25	4.24	27.24	20.57	12.79	24.21
	15000	21.00	5.47	26.71	21.96	19.19	23.53
	20000	19.60	5.69	26.61	20.79	17.44	23.18
Protenix-tiny & VTPE	10000	**23.02**	**8.48**	**24.91**	**24.84**	**22.09**	**21.35**
	15000	**22.84**	**9.15**	**24.32**	**25.01**	**22.68**	**21.05**
	20000	**22.30**	**8.82**	**24.35**	**24.29**	**21.51**	**21.02**

Table 13: Details of various benchmarks. LDDT results on the intra_rna subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	8.81	15.79	26.85	42.44	23.51
	15000	**11.97**	20.42	32.75	48.76	28.50
	20000	**12.71**	**21.57**	**34.17**	49.62	**29.58**
Protenix-base & VTPE	10000	**11.10**	**20.02**	**33.52**	**51.96**	**29.10**
	15000	11.89	**20.78**	**33.28**	**49.09**	**28.72**
	20000	12.01	21.08	33.90	**49.87**	29.17
Protenix-mini	10000	5.66	11.33	21.99	40.11	19.78
	15000	7.05	13.70	25.23	42.78	22.22
	20000	8.13	15.37	27.45	45.04	24.06
Protenix-mini & VTPE	10000	**8.52**	**16.35**	**30.11**	**51.76**	**26.67**
	15000	**9.41**	**17.82**	**32.11**	**53.54**	**28.20**
	20000	**9.48**	**17.61**	**31.17**	**51.76**	**27.47**
Protenix-tiny	10000	5.49	10.88	21.14	38.71	19.06
	15000	5.67	11.26	21.87	40.22	19.76
	20000	5.59	11.09	21.47	39.16	19.33
Protenix-tiny & VTPE	10000	**6.30**	**12.50**	**24.28**	**44.57**	**21.91**
	15000	**6.52**	**12.95**	**25.24**	**46.81**	**22.88**
	20000	**6.25**	**12.42**	**24.25**	**45.09**	**22.00**

Table 14: Details of various benchmarks. RMSD results on the intra_rna subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	12.50	20.83	20.83	13.54	27.89
	15000	0.00	16.67	20.83	29.17	16.67	26.29
	20000	0.00	16.67	25.00	29.17	17.71	26.15
Protenix-base & VTPE	10000	**4.17**	**25.00**	**29.17**	**37.50**	**23.96**	**21.88**
	15000	0.00	**20.83**	**29.17**	**37.50**	**21.88**	**22.37**
	20000	**4.17**	**20.83**	**37.50**	**54.17**	**29.17**	**21.36**
Protenix-mini	10000	0.00	0.00	4.17	16.67	5.21	29.22
	15000	0.00	0.00	20.83	25.00	11.46	27.23
	20000	0.00	0.00	4.17	**29.17**	8.34	28.31
Protenix-mini & VTPE	10000	0.00	**20.83**	**29.17**	**29.17**	**19.79**	**23.53**
	15000	0.00	**20.83**	**29.17**	**29.17**	**19.79**	**23.24**
	20000	0.00	**20.83**	**29.17**	**29.17**	**19.79**	**22.58**
Protenix-tiny	10000	0.00	0.00	20.83	**33.33**	13.54	26.35
	15000	0.00	8.33	25.00	**29.17**	15.62	25.97
	20000	0.00	4.17	25.00	**29.17**	14.59	25.50
Protenix-tiny & VTPE	10000	0.00	**4.17**	**29.17**	29.17	**15.63**	**24.27**
	15000	0.00	**16.67**	**29.17**	**29.17**	**18.75**	**24.38**
	20000	0.00	**12.50**	**29.17**	**29.17**	**17.71**	**24.04**

Table 15: Details of various benchmarks. LDDT results on the prot_prot subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	15.85	27.37	43.75	61.18	37.11
	15000	22.53	36.60	53.76	69.10	45.58
	20000	23.92	38.28	55.20	70.34	47.02
Protenix-base & VTPE	10000	**20.33**	**35.08**	**53.98**	**72.78**	**45.45**
	15000	**24.07**	**39.73**	**57.68**	**73.62**	**48.66**
	20000	**25.07**	**41.54**	**59.90**	**75.63**	**50.44**
Protenix-mini	10000	9.37	18.52	34.86	59.60	30.65
	15000	12.64	23.67	41.21	64.04	35.44
	20000	14.45	25.96	43.02	63.39	36.77
Protenix-mini & VTPE	10000	**14.38**	**26.55**	**45.51**	**69.53**	**38.94**
	15000	**16.59**	**29.77**	**49.11**	**71.85**	**41.74**
	20000	**17.23**	**30.42**	**49.48**	**71.67**	**42.13**
Protenix-tiny	10000	7.69	15.18	29.01	50.65	25.64
	15000	7.71	15.22	29.11	51.29	25.83
	20000	7.73	15.23	28.98	50.34	25.58
Protenix-tiny & VTPE	10000	**8.80**	**17.38**	**33.27**	**58.58**	**29.50**
	15000	**8.55**	**16.91**	**32.58**	**58.27**	**29.07**
	20000	**8.30**	**16.44**	**31.75**	**57.14**	**28.40**

Table 16: Details of various benchmarks. RMSD results on the `prot_prot` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	2.61	5.88	11.11	4.90	28.35
	15000	2.61	9.80	17.65	37.25	16.83	22.21
	20000	3.92	11.76	16.34	37.25	17.32	22.08
Protenix-base & VTPE	10000	**4.58**	**11.76**	**27.45**	**48.37**	**23.04**	**19.75**
	15000	**4.58**	**15.69**	**29.41**	**43.79**	**23.37**	**19.80**
	20000	**7.19**	**17.65**	**35.29**	**52.94**	**28.27**	**18.84**
Protenix-mini	10000	0.00	1.31	8.50	24.18	8.50	24.53
	15000	0.65	1.96	10.46	27.45	10.13	24.32
	20000	0.65	1.31	7.84	19.61	7.35	25.05
Protenix-mini & VTPE	10000	**0.65**	**8.50**	**17.65**	**43.14**	**17.48**	**21.35**
	15000	**2.61**	**11.76**	**24.84**	**50.98**	**22.55**	**20.14**
	20000	**1.96**	**8.50**	**22.88**	**46.41**	**19.94**	**20.83**
Protenix-tiny	10000	0.00	1.31	7.19	27.45	8.99	23.60
	15000	0.00	3.27	10.46	25.49	9.80	24.02
	20000	0.00	3.27	13.07	32.68	12.25	22.85
Protenix-tiny & VTPE	10000	**0.65**	**7.84**	**16.99**	**35.95**	**15.36**	**22.30**
	15000	0.00	**7.84**	**17.65**	**37.25**	**15.68**	**21.86**
	20000	0.00	**9.15**	**18.30**	**37.91**	**16.34**	**21.74**

Table 17: Details of various benchmarks. LDDT results on the `antibody_antigen` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	15.27	27.23	45.35	65.58	38.42
	15000	22.96	38.46	57.62	73.40	48.24
	20000	24.69	41.23	60.54	75.65	50.62
Protenix-base & VTPE	10000	**20.04**	**35.42**	**55.91**	**75.33**	**46.55**
	15000	**25.86**	**43.48**	**63.06**	**77.93**	**52.48**
	20000	**26.39**	**45.04**	**65.63**	**80.94**	**54.42**
Protenix-mini	10000	8.58	17.26	33.58	60.13	29.92
	15000	11.57	22.28	40.46	65.80	35.09
	20000	13.64	25.24	43.17	64.86	36.78
Protenix-mini & VTPE	10000	**12.22**	**23.48**	**42.55**	**68.90**	**36.75**
	15000	**14.93**	**27.87**	**48.24**	**73.11**	**41.02**
	20000	**15.62**	**28.77**	**49.35**	**74.24**	**41.97**
Protenix-tiny	10000	7.44	14.71	28.20	49.78	25.04
	15000	7.78	15.36	29.41	51.80	26.09
	20000	8.08	15.94	30.39	52.75	26.80
Protenix-tiny & VTPE	10000	**8.78**	**17.35**	**33.34**	**59.35**	**29.70**
	15000	**8.65**	**17.13**	**33.15**	**59.75**	**29.67**
	20000	**8.21**	**16.28**	**31.59**	**57.51**	**28.39**

Table 18: Details of various benchmarks. RMSD results on `antibody_antigen` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	0.00	0.00	28.31
	15000	0.00	0.00	0.00	11.11	2.78	26.69
	20000	0.00	0.00	**5.56**	5.56	2.78	26.51
Protenix-base & VTPE	10000	0.00	0.00	**5.56**	**27.78**	**8.34**	**23.94**
	15000	0.00	0.00	**22.22**	**38.89**	**15.28**	**21.24**
	20000	0.00	0.00	**5.56**	**44.44**	**12.50**	**20.86**
Protenix-mini	10000	0.00	0.00	0.00	0.00	0.00	28.39
	15000	0.00	0.00	0.00	0.00	0.00	28.06
	20000	0.00	0.00	0.00	0.00	0.00	28.67
Protenix-mini & VTPE	10000	0.00	0.00	0.00	**16.67**	**4.17**	**24.94**
	15000	0.00	0.00	0.00	**22.22**	**5.55**	**24.00**
	20000	0.00	0.00	0.00	**22.22**	**5.55**	**24.10**
Protenix-tiny	10000	0.00	0.00	0.00	0.00	0.00	27.71
	15000	0.00	0.00	0.00	0.00	0.00	28.81
	20000	0.00	0.00	0.00	0.00	0.00	26.76
Protenix-tiny & VTPE	10000	0.00	0.00	0.00	0.00	0.00	**25.76**
	15000	0.00	0.00	0.00	**5.56**	**1.39**	**26.44**
	20000	0.00	0.00	0.00	0.00	0.00	**25.83**

Table 19: Details of various benchmarks. LDDT results on `rna_prot` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	10.00	17.74	29.65	44.77	25.57
	15000	14.65	24.76	38.39	53.95	32.99
	20000	15.03	25.05	38.31	53.23	32.95
Protenix-base & VTPE	10000	**13.74**	**24.35**	**39.26**	**57.05**	**33.57**
	15000	**15.43**	**26.24**	**40.08**	**55.50**	**34.19**
	20000	**15.84**	**27.03**	**41.06**	**56.41**	**35.02**
Protenix-mini	10000	6.53	13.08	25.44	46.12	22.83
	15000	8.20	15.80	28.88	48.13	25.29
	20000	9.59	18.00	31.73	50.66	27.54
Protenix-mini & VTPE	10000	**9.49**	**18.02**	**32.52**	**53.96**	**28.48**
	15000	**11.06**	**20.54**	**35.83**	**57.16**	**31.11**
	20000	**11.27**	**20.70**	**35.64**	**56.48**	**30.98**
Protenix-tiny	10000	6.16	12.20	23.59	42.78	21.18
	15000	6.22	12.34	23.88	43.50	21.49
	20000	6.25	12.35	23.82	42.88	21.33
Protenix-tiny & VTPE	10000	**6.94**	**13.76**	**26.65**	**48.43**	**23.94**
	15000	**6.97**	**13.83**	**26.90**	**49.36**	**24.27**
	20000	**6.77**	**13.43**	**26.15**	**48.22**	**23.64**

Table 20: Details of various benchmarks. RMSD results on `rna_prot` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	4.55	9.09	9.09	5.68	30.09
	15000	**4.55**	13.64	18.18	27.27	15.91	26.00
	20000	**4.55**	9.09	18.18	22.73	13.64	26.11
Protenix-base & VTPE	10000	**9.09**	**13.64**	**27.27**	**27.27**	**19.32**	**23.01**
	15000	**4.55**	**18.18**	**27.27**	**40.91**	**22.73**	**22.72**
	20000	**4.55**	**18.18**	**31.82**	**45.45**	**25.00**	**21.59**
Protenix-mini	10000	0.00	0.00	4.55	22.73	6.82	27.83
	15000	0.00	0.00	0.00	22.73	5.68	28.24
	20000	0.00	0.00	4.55	22.73	6.82	28.56
Protenix-mini & VTPE	10000	0.00	**13.64**	**22.73**	**27.27**	**15.91**	**24.69**
	15000	**9.09**	**13.64**	**22.73**	**27.27**	**18.18**	**23.63**
	20000	**4.55**	**13.64**	**22.73**	**27.27**	**17.05**	**23.16**
Protenix-tiny	10000	0.00	0.00	13.64	22.73	9.09	27.20
	15000	0.00	**4.55**	13.64	**27.27**	11.37	26.54
	20000	0.00	**9.09**	13.64	**27.27**	12.50	25.68
Protenix-tiny & VTPE	10000	0.00	**9.09**	**18.18**	**27.27**	**13.63**	**24.58**
	15000	0.00	**4.55**	**18.18**	**27.27**	**12.50**	**25.23**
	20000	0.00	**9.09**	**18.18**	**27.27**	**13.63**	**24.72**

Table 21: Details of various benchmarks. LDDT results on `antibody` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	15.26	27.17	45.12	65.05	38.18
	15000	22.66	38.04	56.78	72.21	47.51
	20000	24.15	40.28	59.18	74.25	49.52
Protenix-base & VTPE	10000	**19.89**	**35.30**	**55.76**	**74.86**	**46.39**
	15000	**25.56**	**42.97**	**62.40**	**77.42**	**52.03**
	20000	**26.00**	**44.31**	**64.59**	**79.83**	**53.58**
Protenix-mini	10000	8.47	17.05	33.13	59.32	29.50
	15000	11.61	22.27	40.19	65.02	34.78
	20000	13.66	25.17	42.78	63.99	36.43
Protenix-mini & VTPE	10000	**12.02**	**23.16**	**42.15**	**68.51**	**36.42**
	15000	**14.52**	**27.20**	**47.34**	**72.26**	**40.27**
	20000	**15.34**	**28.24**	**48.34**	**72.97**	**41.12**
Protenix-tiny	10000	7.53	14.88	28.49	50.09	25.25
	15000	7.82	15.46	29.59	51.92	26.21
	20000	8.07	15.89	30.26	52.56	26.70
Protenix-tiny & VTPE	10000	**8.74**	**17.28**	**33.18**	**58.89**	**29.52**
	15000	**8.57**	**16.96**	**32.79**	**59.03**	**29.34**
	20000	**8.21**	**16.27**	**31.57**	**57.45**	**28.38**

Table 22: Details of various benchmarks. RMSD results on `antibody` subtask of RecentPDB

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	0.00	0.00	29.01
	15000	0.00	0.00	0.00	0.00	0.00	26.94
	20000	0.00	0.00	0.00	5.26	1.31	26.45
Protenix-base & VTPE	10000	0.00	0.00	**5.26**	**31.58**	**9.21**	**23.99**
	15000	0.00	**5.26**	**21.05**	**31.58**	**14.47**	**21.53**
	20000	0.00	0.00	**10.53**	**36.84**	**11.84**	**20.75**
Protenix-mini	10000	0.00	0.00	0.00	0.00	0.00	28.32
	15000	0.00	0.00	0.00	0.00	0.00	28.30
	20000	0.00	0.00	0.00	0.00	0.00	28.80
Protenix-mini & VTPE	10000	0.00	0.00	0.00	**5.26**	**1.31**	**25.38**
	15000	0.00	0.00	0.00	**15.79**	**3.95**	**24.48**
	20000	0.00	0.00	0.00	**15.79**	**3.95**	**24.51**
Protenix-tiny	10000	0.00	0.00	0.00	0.00	0.00	27.58
	15000	0.00	0.00	0.00	0.00	0.00	28.78
	20000	0.00	0.00	0.00	0.00	0.00	26.56
Protenix-tiny & VTPE	10000	0.00	0.00	0.00	**5.26**	**1.31**	**25.72**
	15000	0.00	0.00	0.00	**5.26**	**1.31**	**26.57**
	20000	0.00	0.00	0.00	**5.26**	**1.31**	**25.81**

Table 23: Details of various benchmarks. LDDT results on `intra_dna` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	8.01	14.21	23.49	34.90	20.19
	15000	12.85	21.69	33.65	47.14	28.88
	20000	**14.46**	24.23	37.18	52.07	32.04
Protenix-base & VTPE	10000	**12.44**	**22.20**	**36.52**	**54.95**	**31.47**
	15000	**13.47**	**23.20**	**36.12**	**51.16**	**30.95**
	20000	14.22	**24.60**	**38.21**	**54.07**	**32.71**
Protenix-mini	10000	5.95	11.81	22.56	39.81	20.05
	15000	6.71	12.83	23.05	37.21	19.98
	20000	7.71	14.37	25.09	39.80	21.81
Protenix-mini & VTPE	10000	**9.85**	**18.63**	**33.21**	**54.19**	**28.92**
	15000	**10.43**	**19.43**	**33.95**	**54.00**	**29.40**
	20000	**10.92**	**20.08**	**34.80**	**55.55**	**30.23**
Protenix-tiny	10000	5.86	11.61	22.48	40.81	20.19
	15000	6.06	12.01	23.34	42.75	21.04
	20000	5.80	11.50	22.23	40.40	19.99
Protenix-tiny & VTPE	10000	**6.55**	**12.99**	**25.22**	**46.19**	**22.73**
	15000	**6.41**	**12.73**	**24.79**	**45.66**	**22.40**
	20000	**6.23**	**12.38**	**24.16**	**44.81**	**21.89**

Table 24: Details of various benchmarks. RMSD results on `intra_dna` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	5.17	10.34	3.88	29.83
	15000	0.00	8.62	27.59	55.17	22.84	22.05
	20000	0.00	6.90	31.03	53.45	22.84	19.99
Protenix-base & VTPE	10000	**1.72**	**13.79**	**41.38**	**63.79**	**30.17**	**17.05**
	15000	0.00	**24.14**	**43.10**	**63.79**	**32.76**	**17.32**
	20000	**3.45**	**25.86**	**46.55**	**60.34**	**34.05**	**16.56**
Protenix-mini	10000	0.00	1.72	5.17	36.21	10.78	22.84
	15000	0.00	0.00	3.45	17.24	5.17	25.28
	20000	0.00	0.00	3.45	17.24	5.17	24.19
Protenix-mini & VTPE	10000	0.00	**17.24**	**34.48**	**51.72**	**25.86**	**18.32**
	15000	**1.72**	**13.79**	**39.66**	**55.17**	**27.59**	**17.78**
	20000	0.00	**22.41**	**43.10**	**51.72**	**29.31**	**17.25**
Protenix-tiny	10000	0.00	**6.90**	15.52	37.93	15.09	21.47
	15000	0.00	5.17	18.97	46.55	17.67	20.64
	20000	0.00	6.90	18.97	**58.62**	21.12	20.29
Protenix-tiny & VTPE	10000	0.00	5.17	**32.76**	**46.55**	**21.12**	**19.15**
	15000	0.00	**6.90**	**32.76**	**53.45**	**23.28**	**18.74**
	20000	0.00	**8.62**	**32.76**	46.55	**21.98**	**18.83**

Table 25: Details of various benchmarks. LDDT results on `dna_prot` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	10.76	18.73	30.02	41.80	25.38
	15000	17.38	29.01	43.87	59.15	37.44
	20000	18.79	30.99	45.96	61.32	39.32
Protenix-base & VTPE	10000	**16.05**	**28.15**	**44.61**	**63.18**	**37.97**
	15000	**18.31**	**30.89**	**46.17**	**61.65**	**39.19**
	20000	**18.93**	**32.10**	**48.00**	**64.04**	**40.71**
Protenix-mini	10000	7.29	14.34	27.00	46.50	23.81
	15000	8.68	16.37	28.74	44.87	24.78
	20000	10.20	18.76	31.91	48.32	27.42
Protenix-mini & VTPE	10000	**11.77**	**21.91**	**38.19**	**59.99**	**32.93**
	15000	**13.25**	**24.15**	**40.77**	**61.97**	**34.99**
	20000	**13.76**	**24.83**	**41.63**	**63.06**	**35.76**
Protenix-tiny	10000	6.37	12.61	24.26	43.25	21.63
	15000	6.70	13.27	25.62	46.28	22.97
	20000	6.56	12.98	24.92	44.39	22.22
Protenix-tiny & VTPE	10000	**7.26**	**14.37**	**27.77**	**50.16**	**24.89**
	15000	**7.20**	**14.27**	**27.65**	**50.24**	**24.84**
	20000	**6.98**	**13.83**	**26.88**	**49.27**	**24.24**

Table 26: Details of various benchmarks. RMSD results on `dna_prot` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	2.70	5.41	8.11	4.05	28.78
	15000	0.00	5.41	29.73	70.27	26.35	17.67
	20000	0.00	10.81	37.84	59.46	27.03	17.64
Protenix-base & VTPE	10000	**2.70**	**13.51**	**54.05**	**83.78**	**38.51**	**15.32**
	15000	0.00	**21.62**	**51.35**	**83.78**	**39.19**	**15.23**
	20000	**5.41**	**27.03**	**56.76**	**86.49**	**43.92**	**14.17**
Protenix-mini	10000	0.00	0.00	5.41	37.84	10.81	21.87
	15000	0.00	2.70	5.41	16.22	6.08	23.89
	20000	0.00	0.00	5.41	13.51	4.73	23.01
Protenix-mini & VTPE	10000	0.00	**10.81**	**37.84**	**67.57**	**29.05**	**17.12**
	15000	**2.70**	**16.22**	**56.76**	**75.68**	**37.84**	**15.52**
	20000	0.00	**27.03**	**56.76**	**75.68**	**39.87**	**15.29**
Protenix-tiny	10000	0.00	**5.41**	16.22	51.35	18.25	20.22
	15000	0.00	**5.41**	13.51	62.16	20.27	19.53
	20000	0.00	**10.81**	16.22	62.16	22.30	18.69
Protenix-tiny & VTPE	10000	0.00	**5.41**	**37.84**	**59.46**	**25.68**	**18.63**
	15000	0.00	**5.41**	**40.54**	**67.57**	**28.38**	**17.81**
	20000	0.00	5.41	**37.84**	**64.86**	**27.03**	**17.80**

Table 27: Details of various benchmarks. LDDT results on `intra_prot` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	16.96	29.64	47.83	66.93	40.42
	15000	24.84	40.43	58.95	74.86	49.86
	20000	26.54	42.38	60.38	75.70	51.33
Protenix-base & VTPE	10000	**23.10**	**39.34**	**59.17**	**77.33**	**49.65**
	15000	**27.53**	**44.88**	**63.57**	**78.70**	**53.56**
	20000	**27.95**	**45.94**	**64.94**	**79.88**	**54.59**
Protenix-mini	10000	10.04	19.77	36.96	62.42	32.33
	15000	13.63	25.53	44.25	67.66	37.82
	20000	15.54	28.02	46.39	67.52	39.44
Protenix-mini & VTPE	10000	**15.78**	**29.00**	**49.00**	**72.85**	**41.60**
	15000	**18.92**	**33.61**	**54.07**	**76.05**	**45.61**
	20000	**19.43**	**34.07**	**54.34**	**76.12**	**45.92**
Protenix-tiny	10000	7.98	15.75	30.08	52.41	26.56
	15000	8.11	16.01	30.60	53.68	27.10
	20000	8.13	16.03	30.52	52.97	26.92
Protenix-tiny & VTPE	10000	**9.14**	**18.03**	**34.44**	**60.19**	**30.44**
	15000	**8.97**	**17.73**	**34.07**	**60.31**	**30.26**
	20000	**8.75**	**17.29**	**33.28**	**59.25**	**29.64**

Table 28: Details of various benchmarks. RMSD results on `intra_prot` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	4.95	18.02	26.50	42.05	22.88	21.72
	15000	17.67	33.22	39.58	59.01	37.37	17.11
	20000	18.37	32.86	39.58	59.72	37.63	17.03
Protenix-base & VTPE	10000	**19.08**	**31.45**	**47.00**	**66.43**	**40.99**	**15.71**
	15000	**25.80**	**37.81**	**49.82**	**67.49**	**45.23**	**14.76**
	20000	**27.21**	**38.87**	**51.24**	**69.26**	**46.64**	**14.40**
Protenix-mini	10000	0.71	6.71	20.14	41.34	17.23	21.58
	15000	3.18	12.01	25.80	44.52	21.38	20.65
	20000	2.12	8.83	22.26	42.05	18.81	21.31
Protenix-mini & VTPE	10000	**8.13**	**22.97**	**36.75**	**58.30**	**31.54**	**17.74**
	15000	**15.19**	**29.68**	**43.46**	**67.14**	**38.87**	**16.29**
	20000	**16.61**	**29.33**	**42.40**	**65.02**	**38.34**	**16.37**
Protenix-tiny	10000	0.00	10.95	25.09	46.29	20.58	20.17
	15000	0.00	16.25	30.04	48.41	23.67	19.96
	20000	0.00	19.43	33.22	54.42	26.77	18.74
Protenix-tiny & VTPE	10000	**0.35**	**23.67**	**33.92**	**54.42**	**28.09**	**18.53**
	15000	0.00	**22.61**	**35.34**	**55.83**	**28.45**	**18.46**
	20000	0.00	**22.61**	**35.34**	**56.18**	**28.53**	**18.20**

Table 29: Details of various benchmarks. LDDT results on `ligand_prot` subtask of RecentPDB.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	15.96	28.27	46.75	67.14	39.60
	15000	24.07	39.70	58.71	75.16	49.50
	20000	25.76	41.78	60.35	76.18	51.10
Protenix-base & VTPE	10000	**21.93**	**38.21**	**58.94**	**78.28**	**49.26**
	15000	**26.43**	**44.03**	**63.68**	**79.76**	**53.38**
	20000	**26.71**	**44.83**	**64.80**	**80.78**	**54.17**
Protenix-mini	10000	9.22	18.34	35.06	61.28	31.00
	15000	12.84	24.37	43.18	67.95	37.11
	20000	14.68	26.93	45.65	68.24	38.92
Protenix-mini & VTPE	10000	**14.54**	**27.18**	**47.11**	**71.95**	**40.15**
	15000	**17.73**	**32.07**	**52.91**	**76.32**	**44.68**
	20000	**18.16**	**32.46**	**53.05**	**76.02**	**44.88**
Protenix-tiny	10000	7.76	15.34	29.37	51.53	26.00
	15000	8.11	16.04	30.71	54.06	27.24
	20000	8.06	15.89	30.28	52.70	26.74
Protenix-tiny & VTPE	10000	**9.22**	**18.21**	**34.85**	**61.08**	**30.83**
	15000	**9.05**	**17.90**	**34.44**	**61.14**	**30.63**
	20000	**8.70**	**17.21**	**33.23**	**59.52**	**29.66**

Table 30: Details of various benchmarks. RMSD results on `ligand_prot` subtask of RecentPDB.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	4.00	12.00	20.00	38.67	18.67	21.22
	15000	12.00	**33.33**	36.00	50.67	33.00	17.03
	20000	12.00	**34.67**	34.67	50.67	33.00	17.04
Protenix-base & VTPE	10000	**12.00**	**29.33**	**42.67**	**58.67**	**35.67**	**16.01**
	15000	**22.67**	**33.33**	**45.33**	**65.33**	**41.66**	**15.37**
	20000	**24.00**	**34.67**	**48.00**	**68.00**	**43.67**	**15.10**
Protenix-mini	10000	0.00	5.33	13.33	32.00	12.66	21.95
	15000	2.67	9.33	18.67	42.67	18.34	21.02
	20000	0.00	4.00	16.00	38.67	14.67	21.66
Protenix-mini & VTPE	10000	**6.67**	**18.67**	**28.00**	**54.67**	**27.00**	**18.25**
	15000	**10.67**	**24.00**	**40.00**	**62.67**	**34.34**	**16.54**
	20000	**13.33**	**25.33**	**37.33**	**61.33**	**34.33**	**16.77**
Protenix-tiny	10000	0.00	6.67	18.67	37.33	15.67	20.05
	15000	0.00	**18.67**	26.67	48.00	23.34	19.59
	20000	0.00	**18.67**	26.67	**49.33**	23.67	**18.67**
Protenix-tiny & VTPE	10000	0.00	**17.33**	**30.67**	**52.00**	**25.00**	**18.84**
	15000	0.00	**18.67**	**29.33**	**53.33**	**25.33**	**18.70**
	20000	0.00	**18.67**	**32.00**	48.00	**24.67**	18.72

Table 31: Details of various benchmarks. LDDT results on `dna_prot` subtask of PXM-Legacy.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	7.83	13.66	22.32	32.84	19.20
	15000	**12.92**	21.84	34.19	48.51	29.41
	20000	**13.90**	**23.46**	**36.41**	51.19	**31.28**
Protenix-base & VTPE	10000	**11.54**	**20.80**	**34.66**	**52.70**	**29.89**
	15000	12.84	**22.37**	**35.44**	**50.75**	**30.31**
	20000	12.99	22.75	36.12	**52.00**	30.92
Protenix-mini	10000	5.83	11.73	23.06	42.72	20.86
	15000	7.15	13.76	25.18	42.28	22.13
	20000	8.27	15.40	27.07	43.85	23.69
Protenix-mini & VTPE	10000	**8.66**	**16.65**	**30.66**	**52.33**	**27.05**
	15000	**9.49**	**17.93**	**32.22**	**53.51**	**28.26**
	20000	**9.77**	**18.28**	**32.66**	**54.20**	**28.70**
Protenix-tiny	10000	5.57	11.03	21.38	38.99	19.25
	15000	6.04	11.98	23.26	42.72	21.00
	20000	5.69	11.27	21.80	39.63	19.60
Protenix-tiny & VTPE	10000	**6.62**	**13.13**	**25.51**	**46.83**	**23.02**
	15000	**6.50**	**12.91**	**25.20**	**46.74**	**22.84**
	20000	**6.32**	**12.56**	**24.55**	**45.76**	**22.30**

Table 32: Details of various benchmarks. RMSD results on `dna_prot` subtask of PXM-Legacy.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	0.00	0.00	34.71
	15000	0.00	0.00	4.91	25.89	7.70	26.23
	20000	0.00	0.00	9.38	32.59	10.49	25.67
Protenix-base & VTPE	10000	0.00	**1.34**	**19.20**	**40.62**	**15.29**	**22.63**
	15000	**0.45**	**3.12**	**20.54**	**37.50**	**15.40**	**22.71**
	20000	**0.45**	**2.68**	**23.21**	**44.64**	**17.75**	**21.88**
Protenix-mini	10000	0.00	0.00	0.00	15.62	3.90	26.74
	15000	0.00	0.00	0.00	2.68	0.67	28.28
	20000	0.00	0.00	0.00	8.93	2.23	28.06
Protenix-mini & VTPE	10000	0.00	0.00	**12.50**	**25.45**	**9.49**	**23.93**
	15000	0.00	**3.12**	**16.52**	**35.71**	**13.84**	**23.20**
	20000	0.00	**4.46**	**17.41**	**36.16**	**14.51**	**23.21**
Protenix-tiny	10000	0.00	0.00	0.00	16.96	4.24	27.24
	15000	0.00	0.00	1.34	20.54	5.47	26.71
	20000	0.00	0.00	0.45	22.32	5.69	26.61
Protenix-tiny & VTPE	10000	0.00	0.00	**12.05**	**21.88**	**8.48**	**24.91**
	15000	0.00	0.00	**12.95**	**23.66**	**9.15**	**24.32**
	20000	0.00	0.00	**12.05**	**23.21**	**8.82**	**24.35**

Table 33: Details of various benchmarks. LDDT results on `intra_rna` subtask of PXM-Legacy.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	11.29	19.48	30.94	43.84	26.43
	15000	15.22	25.05	37.70	51.79	32.52
	20000	16.21	26.23	38.94	53.01	33.68
Protenix-base & VTPE	10000	**15.90**	**27.08**	**41.91**	**59.05**	**35.92**
	15000	**17.16**	**28.35**	**41.70**	**55.79**	**35.66**
	20000	**18.02**	**29.75**	**43.27**	**57.36**	**37.04**
Protenix-mini	10000	6.95	13.83	26.51	46.99	23.61
	15000	9.43	17.82	31.52	50.35	27.33
	20000	10.76	19.65	33.30	50.99	28.73
Protenix-mini & VTPE	10000	**11.93**	**21.76**	**36.82**	**57.24**	**31.89**
	15000	**12.87**	**23.02**	**38.03**	**58.22**	**32.99**
	20000	**13.32**	**23.39**	**38.12**	**57.95**	**33.11**
Protenix-tiny	10000	6.03	11.93	23.01	41.28	20.57
	15000	6.38	12.63	24.44	44.37	21.96
	20000	6.11	12.06	23.22	41.76	20.79
Protenix-tiny & VTPE	10000	**7.23**	**14.31**	**27.67**	**50.17**	**24.84**
	15000	**7.24**	**14.35**	**27.80**	**50.68**	**25.01**
	20000	**7.00**	**13.87**	**26.95**	**49.37**	**24.29**

Table 34: Details of various benchmarks. RMSD results on `intra_rna` subtask of PXM-Legacy.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	2.33	0.58	30.31
	15000	0.00	4.65	13.95	32.56	12.79	25.27
	20000	0.00	4.65	20.93	37.21	15.70	24.06
Protenix-base & VTPE	10000	0.00	**25.58**	**37.21**	**41.86**	**26.16**	**19.77**
	15000	0.00	**23.26**	**32.56**	**41.86**	**24.42**	**20.45**
	20000	0.00	**25.58**	**32.56**	**53.49**	**27.91**	**19.48**
Protenix-mini	10000	0.00	0.00	6.98	32.56	9.89	24.94
	15000	0.00	0.00	16.28	30.23	11.63	24.68
	20000	0.00	0.00	9.30	30.23	9.88	25.40
Protenix-mini & VTPE	10000	0.00	**20.93**	**30.23**	**39.53**	**22.67**	**20.74**
	15000	0.00	**20.93**	**32.56**	**41.86**	**23.84**	**20.23**
	20000	0.00	**20.93**	**32.56**	**39.53**	**23.26**	**19.99**
Protenix-tiny	10000	0.00	0.00	18.60	32.56	12.79	24.21
	15000	0.00	18.60	20.93	**37.21**	19.19	23.53
	20000	0.00	6.98	25.58	**37.21**	17.44	23.18
Protenix-tiny & VTPE	10000	0.00	**20.93**	**27.91**	**39.53**	**22.09**	**21.35**
	15000	0.00	**23.26**	**30.23**	**37.21**	**22.68**	**21.05**
	20000	0.00	**20.93**	**27.91**	**37.21**	**21.51**	**21.02**

Table 35: Details of various benchmarks. LDDT results on `intra_dna` subtask of PXM-Legacy.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	8.03	14.08	23.10	33.98	19.83
	15000	13.29	22.52	35.25	49.93	30.30
	20000	**14.26**	**24.07**	37.32	52.42	32.05
Protenix-base & VTPE	10000	**12.09**	**21.74**	**36.09**	**54.36**	**31.03**
	15000	**13.47**	**23.37**	**36.74**	**52.21**	**31.38**
	20000	13.70	23.96	**37.79**	**53.74**	**32.27**
Protenix-mini	10000	5.94	11.93	23.32	42.87	21.04
	15000	7.16	13.81	25.28	42.32	22.20
	20000	8.41	15.69	27.62	44.72	24.15
Protenix-mini & VTPE	10000	**8.90**	**17.09**	**31.35**	**53.12**	**27.59**
	15000	**9.71**	**18.32**	**32.85**	**54.35**	**28.76**
	20000	**10.04**	**18.75**	**33.39**	**54.98**	**29.26**
Protenix-tiny	10000	5.53	10.97	21.25	38.67	19.11
	15000	6.05	12.00	23.31	42.79	21.04
	20000	5.72	11.32	21.89	39.77	19.68
Protenix-tiny & VTPE	10000	**6.66**	**13.21**	**25.67**	**47.11**	**23.16**
	15000	**6.58**	**13.07**	**25.49**	**47.15**	**23.07**
	20000	**6.39**	**12.68**	**24.79**	**46.13**	**22.49**

Table 36: Details of various benchmarks. RMSD results on `intra_dna` subtask of PXM-Legacy.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	0.00	0.00	34.00
	15000	0.00	0.61	5.49	31.71	9.45	24.84
	20000	0.00	0.00	10.37	37.20	11.89	24.33
Protenix-base & VTPE	10000	0.00	0.00	**17.68**	**50.00**	**16.92**	**21.40**
	15000	**0.61**	**2.44**	**22.56**	**47.56**	**18.29**	**21.56**
	20000	**0.61**	**3.66**	**31.71**	**51.83**	**21.95**	**20.78**
Protenix-mini	10000	0.00	0.00	0.00	14.63	3.66	26.00
	15000	0.00	0.00	0.00	1.83	0.46	27.37
	20000	0.00	0.00	0.00	6.71	1.68	27.23
Protenix-mini & VTPE	10000	0.00	**1.22**	**14.63**	**27.44**	**10.82**	**22.88**
	15000	0.00	**4.27**	**19.51**	**41.46**	**16.31**	**22.10**
	20000	0.00	**3.05**	**19.51**	**43.90**	**16.62**	**21.95**
Protenix-tiny	10000	0.00	0.00	1.22	18.29	4.88	26.27
	15000	0.00	0.00	5.49	22.56	7.01	25.66
	20000	0.00	0.00	0.61	**23.78**	6.10	25.37
Protenix-tiny & VTPE	10000	0.00	0.00	**11.59**	**24.39**	**9.00**	**24.04**
	15000	0.00	**0.61**	**14.63**	**25.61**	**10.21**	**23.31**
	20000	0.00	**0.61**	**12.80**	**23.78**	**9.30**	**23.30**

Table 37: Details of various benchmarks. LDDT results on `rna_prot` subtask of PXM-Legacy.

Model	training steps	LDDT-0.5 (↑)	LDDT-1 (↑)	LDDT-2 (↑)	LDDT-4 (↑)	LDDT-mean (↑)
Protenix-base	10000	11.02	19.11	30.60	43.57	26.13
	15000	14.71	24.40	37.18	51.50	32.03
	20000	15.54	25.46	38.37	52.72	33.09
Protenix-base & VTPE	10000	**15.06**	**25.99**	**40.70**	**58.24**	**34.90**
	15000	**16.41**	**27.48**	**41.16**	**55.70**	**35.07**
	20000	**17.44**	**29.13**	**42.78**	**57.10**	**36.52**
Protenix-mini	10000	6.76	13.47	25.94	46.53	23.23
	15000	9.06	17.28	30.92	49.84	26.85
	20000	10.54	19.35	32.92	50.34	28.38
Protenix-mini & VTPE	10000	**11.56**	**21.20**	**36.20**	**56.96**	**31.41**
	15000	**12.31**	**22.23**	**37.41**	**57.99**	**32.41**
	20000	**12.60**	**22.57**	**37.55**	**57.75**	**32.57**
Protenix-tiny	10000	6.01	11.90	22.93	41.07	20.48
	15000	6.36	12.59	24.37	44.20	21.88
	20000	6.17	12.19	23.47	42.16	21.00
Protenix-tiny & VTPE	10000	**7.15**	**14.17**	**27.42**	**49.79**	**24.63**
	15000	**7.14**	**14.16**	**27.49**	**50.24**	**24.76**
	20000	**6.93**	**13.74**	**26.71**	**49.04**	**24.10**

Table 38: Details of various benchmarks. RMSD results on `rna_prot` subtask of PXM-Legacy.

Model	training steps	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	10000	0.00	0.00	0.00	1.72	0.43	30.51
	15000	0.00	1.72	10.34	31.03	10.77	24.63
	20000	0.00	3.45	17.24	31.03	12.93	23.91
Protenix-base & VTPE	10000	0.00	**17.24**	**31.03**	**36.21**	**21.12**	**20.54**
	15000	0.00	**18.97**	**29.31**	**37.93**	**21.55**	**20.86**
	20000	0.00	**18.97**	**31.03**	**50.00**	**25.00**	**19.21**
Protenix-mini	10000	0.00	0.00	6.90	27.59	8.62	24.91
	15000	0.00	0.00	13.79	24.14	9.48	24.95
	20000	0.00	0.00	8.62	27.59	9.05	25.71
Protenix-mini & VTPE	10000	0.00	**15.52**	**27.59**	**34.48**	**19.40**	**21.02**
	15000	0.00	**15.52**	**29.31**	**41.38**	**21.55**	**20.22**
	20000	0.00	**15.52**	**31.03**	**34.48**	**20.26**	**20.50**
Protenix-tiny	10000	0.00	0.00	13.79	25.86	9.91	24.31
	15000	0.00	13.79	17.24	31.03	15.52	23.96
	20000	0.00	8.62	17.24	31.03	14.22	23.76
Protenix-tiny & VTPE	10000	0.00	**15.52**	**18.97**	**32.76**	**16.81**	**22.38**
	15000	0.00	**17.24**	**27.59**	**32.76**	**19.40**	**21.39**
	20000	0.00	**17.24**	**20.69**	**32.76**	**17.67**	**21.56**

Table 39: Details of different training crop size: RMSD on PoseBusters.

Model	crop size	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	128	0.00	1.43	16.19	69.05	21.67	18.18
Protenix-base	256	7.14	41.90	62.38	78.57	47.50	13.09
Protenix-base	384	7.14	44.76	61.90	80.00	48.45	13.21
Protenix-base & VTPE	128	**10.48**	**48.10**	**62.86**	**74.76**	**49.05**	**13.31**
Protenix-base & VTPE	256	**40.00**	**64.76**	**73.81**	**85.24**	**65.95**	**10.09**
Protenix-base & VTPE	384	**24.29**	**61.43**	**70.95**	**86.19**	**60.72**	**10.62**
Protenix-mini	128	0.00	0.00	6.19	43.33	12.38	20.64
Protenix-mini	256	2.38	29.05	45.71	76.67	38.45	14.88
Protenix-mini	384	0.00	15.24	38.10	66.19	29.88	16.79
Protenix-mini & VTPE	128	**7.62**	**30.48**	**50.00**	**80.48**	**42.15**	**14.12**
Protenix-mini & VTPE	256	**29.52**	**54.76**	**71.90**	**94.29**	**62.62**	**10.13**
Protenix-mini & VTPE	384	**36.67**	**58.57**	**77.14**	**92.38**	**66.19**	**9.56**
Protenix-tiny	128	0.00	0.00	1.43	29.05	7.62	21.92
Protenix-tiny	256	0.00	14.29	40.95	73.33	32.14	16.26
Protenix-tiny	384	0.00	19.05	45.71	68.10	33.22	16.47
Protenix-tiny & VTPE	128	0.00	**29.05**	**47.14**	**76.67**	**38.22**	**14.97**
Protenix-tiny & VTPE	256	0.00	**50.95**	**63.81**	**85.24**	**50.00**	**12.29**
Protenix-tiny & VTPE	384	**1.90**	**53.33**	**64.29**	**80.95**	**50.12**	**12.26**

Table 40: Details of different inference crop size: RMSD on PoseBusters.

Model	crop size	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base	128	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	2.66
Protenix-base	256	81.82	**90.91**	**100.00**	**100.00**	**93.18**	4.17
Protenix-base	384	56.18	79.78	94.38	**98.88**	82.31	6.19
Protenix-base	default	36.19	54.28	63.80	75.71	57.50	11.58
Protenix-base & VTPE	128	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**2.03**
Protenix-base & VTPE	256	**86.36**	**90.91**	90.91	**100.00**	92.05	**3.64**
Protenix-base & VTPE	384	**77.53**	**87.64**	**96.63**	**98.88**	**90.17**	**4.66**
Protenix-base & VTPE	default	**53.81**	**64.29**	**71.90**	**80.00**	**67.50**	**9.82**

Table 41: Details of Ablation studies: RMSD on PoseBusters.

Model	RMSD-5 (↑)	RMSD-10 (↑)	RMSD-15 (↑)	RMSD-20 (↑)	RMSD-mean (↑)	RMSD-distance (↓)
Protenix-base & VTPE	**10.48**	**48.10**	**62.86**	74.76	**49.05**	**13.31**
Protenix-base & VTPE w/o λ_i	5.71	32.86	50.00	74.29	40.71	14.53
Protenix-base & VTPE w/o $u = 6$	7.90	38.48	53.33	**75.71**	43.85	14.10
Protenix-base & VTPE w/o Interleaved and $u = 6$	3.33	28.10	43.33	73.81	37.14	15.08