Supplementary Materials

Contents

A Methods

A.1 Model details

A.1.1 Overview

Our model is an E(3)-equivariant, diffusion-based, graph neural network utilizing a coarse-grained representation.

An E(3)-equivariant model transforms the output, y, according to the trans-rotation and parity operations applied to the input x in 3D space [28]. Research has demonstrated that equivariant models can be trained with 1,000 times less data while yielding superior results on the structures of bulk water[29]. Despite substantial advancements in cryo-electron microscopy and crystallography, the existing protein-ligand complex database remains relatively limited, only extending to tens of thousands in size. Consequently, an efficient model is required, capable of discerning the most relevant information and avoiding superficial information that does not hold true upon relocating or rotating the entire structure. The traditional approach to fulfilling the SO(3) symmetry involves exclusively using or predicting invariant quantities, such as the contact map. However, a contact or distance map does not always correlate with physically feasible configurations. For instance, a residue may be predicted to be in contact with two vastly distant atoms. Additionally, a contact map may overlook chirality, a significant aspect in drug discovery[30].

As a diffusion-based model, DynamicBind is trained through a process that incrementally distorts the native conformation at various degrees, enabling the model to learn how to restore the correct conformation. Distorting the original configuration commonly involves adding trans-locational Gaussian noise to the atoms. With bond distance constraints imposed by chemical bonds and excluded volume effects enforced by Van der Waals forces, restoring from such distortions is straightforward when the distortion is relatively small. However, we observed that merely adding Gaussian noise is insufficient to train a model that can predict the transformation from one biologically meaningful configuration to another. To address this, we introduced a morph-like transformation that interpolates between the crystal protein structure and the structure predicted by AlphaFold, thereby reducing the transition barriers between meta-stable configurations, such as the AlphaFold-predicted conformation, and the ligand-bounded holo configuration. Unlike other generative models that train a score function, $\boldsymbol{s}_\theta(\mathbf{x}, t) \approx \nabla \log p_t(\mathbf{x})$, our diffusion architectures aim to map perturbed structures directly back to the original conformations, akin to the consistency model [31, 20]. The outputs of the model are denoted as $\boldsymbol{f}_\theta\left(\mathbf{x}_t, t\right) = -\boldsymbol{\phi}\left(\mathbf{x}_t, t\right)$, where $\boldsymbol{\phi}\left(\mathbf{x}_t, t\right)$ represents the added morph-like transformation to the native conformation.

Traditional methods use an all-atom representation, modeling the coordinates of every atom explicitly. However, atoms do not move independently due to their connections via chemical bonds, and local geometry is highly constrained - for example, a benzene ring is generally flat. To reduce the number of degrees of freedom of these nonphysical configurations, we adopted a coarse-grained representation for both the protein and the ligand. In our model, each protein residue is represented by a node with two vectors - coordinates and directions, and side-chain dihedral angles. More details are provided in Section A.1.2. For the ligand, every heavy atom is represented by a node, and these nodes transform in an extrinsic-to-intrinsic manner, as described in [32]. Additional details can be found in Section A.1.4. Notably, despite being a coarse-grained representation, the coordinates of all non-hydrogen atoms can still be mapped in a one-to-one manner.

The input to our model is the current conformation of the protein and the ligand. The outputs include the predicted updates to k^l scalar torsion angles and two translation-rotation vectors for each ligand, along with updates to k_i^p scalar dihedral angles of the side-chain and two translation-rotation vectors of the backbone for each protein residue. Further details can be found in Section A.1.5. In addition, the model produces two scalar outputs: one to estimate the degree of the native conformation as assessed by cLDDT (contact-LDDT), and another to predict the binding affinity between the protein and ligand.

A.1.2 Featurization

The ligand in our model is the attributed graph $\mathcal{G}^l = (\mathcal{V}^l, \mathcal{E}^l)$, in which each node $\mathfrak{v}_i^l \in \mathcal{V}^l$ represents a heavy atom and the aromatic, single, double, or triple bonds as the edges. The node features of the ligand graph include atomic number, chirality, degree, and formal charge. In addition to bond type, edge length embedding is also used as scalar edge features.

The protein graph is denoted as $\mathcal{G}^p = (\mathcal{V}^p, \mathcal{E}^p)$, where each node $\mathfrak{v}i^p \in \mathcal{V}^p$ corresponds to a residue at the C_α position. The node features in the protein graph include amino acid type, language model embedding from **esm** [33], and side-chain dihedral angles, which are represented as (7×2)-dimensional zero-padded scalar features (five rotatable chi angles [chi1, chi2, ..., chi5] and two symmetric chi angles [altchi1, altchi2] for each amino acid, and these angles are transformed into sine and cosine values). To ensure the uniqueness of the side-chain angles for a given structure, we consistently handle it as $[\max(\text{chi1}, \text{altchi1}), \min(\text{chi1}, \text{altchi1}), \max(\text{chi2}, \text{altchi2}), \min(\text{chi2}, \text{altchi2}),$ chi3, chi4, chi5]. Additionally, the backbone orientation is represented as two unit vector features, which are $\frac{\mathbf{x}_N - \mathbf{x}_{C_\alpha}}{\|\mathbf{x}_N - \mathbf{x}_{C_\alpha}\|}$ and $\frac{\mathbf{x}_C - \mathbf{x}_{C_\alpha}}{\|\mathbf{x}_C - \mathbf{x}_{C_\alpha}\|}$. For edges, length embedding is used as scalar features. Our featurization of the amino acid enable the model to infer the positions of all heavy atoms.

A.1.3 Architecture

DynamicBind is a graph neural network that uses both equivariant and invariant features. It propagates information using tensor products of irreducible representations (irreps) as per the definitions in the e3nn library[28].

The input scalar features of nodes and edges are concatenated with sinusoidal embeddings[34] of diffusion time and then encoded by different multilayer perceptrons (MLPs). For protein node, the two unit vector features of amino acids are combined with the new scalar representations to form the initial features for interaction layers. Similar to DiffDock [15], in each step of the graph propagation process, the ligand and protein graphs undergo one intra-interaction and one inter-interaction. In the ligand's intra-interaction, the representation of each ligand atom is updated by other ligand atoms within a distance of 5Å. For the protein, each amino acid is updated by other amino acids within a distance of 15Å. To reduce the training runtime and memory usage of the model, a maximum of 24 neighbors is allowed for each residue. The edges for inter-interaction are determined based on whether an amino acid is within a distance of $(3\sigma_{tr} + 12)$Å from any ligand atom, where σ_{tr} is the current standard deviation of the diffusion translational noise. This dynamic cutoff is designed to ensure inter-connections exists even when the ligand is far from the receptor when σ_{tr} is large. After the connected graph is determined, the messages of node is updated by the **TensorProductLayer**. Specifically, for each node a belonging to category c_a:

$$\mathbf{h}_a \leftarrow \mathbf{h}_a \bigoplus_{c \in \{\ell, r\}} \mathrm{BN}^{(c_a, c)} \left(\frac{1}{|\mathcal{N}_a^{(c)}|} \sum_{b \in \mathcal{N}_a^{(c)}} Y\left(\mathbf{r}_{ab}\right) \otimes_{\psi_{ab}} \mathbf{h}_b \right)$$
$$\text{with } \psi_{ab} = \Psi^{(c_a, c)}\left(e_{ab}, \mathbf{h}_a^0, \mathbf{h}_b^0\right)$$

Here, $\mathbf{h}_a$ represents the features of a node, and $\mathbf{h}_a^0$ denotes its scalar features. $\mathcal{N}_a^{(c)}$ refers to the neighbors of node a of category c (either ligand, or protein). The spherical harmonics are denoted as Y, and BN represents the (equivariant) batch normalization. The module Ψ is a MLP which contains learnable weights for the tensor product, which are computed based on the edge embeddings, e_{ab}, and scalar features, $\mathbf{h}_a^0, \mathbf{h}_b^0$.

After final interaction layer, the node representations are used to produce the outputs. For generating the cLDDT, binding affinity, ligand's translation and rotation predictions, a convolution of each ligand atom with the geometric center of ligand is employed:

$$\mathbf{v} = \frac{1}{|\mathcal{V}^\ell|} \sum_{a \in \mathcal{V}^\ell} Y\left(\mathbf{r}_{oa}\right) \otimes_{\psi_{oa}} \mathbf{h}_a$$
$$\text{with } \psi_{oa} = \Psi\left(e_{oa}, \mathbf{h}_a^0\right)$$

where e_{oa} is the edge embedding between the geometric center of the ligand and a ligand node a. The output $\mathbf{v}$ consists of 144 even scalars, 2 odd parity vectors and 2 even vectors. The scalars are used for predicting the cLDDT (D) and negative logarithm of the binding affinity (A) as measured in the unit of concentration.

$$D = \text{MLP}(\mathbf{v}_{scalar}[:72])$$
$$A = \text{clamp}(\frac{\text{MLP}(\mathbf{v}_{scalar}[72:144])}{D + \text{eps}}, \min = 0, \max = 15)$$

The odd vectors are used to predict ligand translation, while the even vectors are used to predict ligand rotation:

$$\mathbf{tr}^l = \frac{\overline{\mathbf{v}}^{odd}_{vector}}{\|\overline{\mathbf{v}}^{odd}_{vector}\| + \text{eps}} \times \text{MLP}(\|\overline{\mathbf{v}}^{odd}_{vector}\|, \mathbf{s}_t)$$
$$\mathbf{rot}^l = \frac{\overline{\mathbf{v}}^{even}_{vector}}{\|\overline{\mathbf{v}}^{even}_{vector}\| + \text{eps}} \times \text{MLP}(\|\overline{\mathbf{v}}^{even}_{vector}\|, \mathbf{s}_t)$$
$$\text{with } \overline{\mathbf{v}}_{vector} = \frac{\mathbf{v}_{vector}[0] + \mathbf{v}_{vector}[1]}{2}$$

Here, $\mathbf{s}_t$ is the sinusoidal embeddings of the diffusion time, eps $= 10^{-12}$ is added for numerical stability. Following Jing et al.[32], our model predicts a scalar torsion update for each rotatable bond of ligand. For bond b, the torsion update T^l_b is generated by a convolution of every atom on a radius graph with the bond center o:

$$T^l_b = \text{MLP}(\frac{1}{|\mathcal{N}_b|} \sum_{a \in \mathcal{N}_b} Y(\mathbf{r_{oa}}) \otimes Y^2(\mathbf{r}_b) \otimes_{\gamma_{oa}} \mathbf{h}_a)$$
$$\text{with } \gamma_{oa} = \Gamma\left(e_{oa}, \mathbf{h}^0_a, \mathbf{h}^0_{b_0} + \mathbf{h}^0_{b_1}\right)$$

To predict the conformation changes of protein, we require updates of the side chain chis, translation, and rotation for each protein node. These operations are generated from the final interaction representations $\mathbf{h}_i$ of each amino acid:

$$T^p_i = \text{MLP}(\mathbf{h}^{odd}_{i,scalar}, \mathbf{h}^{even}_{i,scalar})$$
$$\mathbf{tr}^p_i = \frac{\overline{\mathbf{h}}^{odd}_{i,vector}}{\|\overline{\mathbf{h}}^{odd}_{i,vector}\| + \text{eps}} \times \text{MLP}(\|\overline{\mathbf{h}}^{odd}_{i,vector}\|, \mathbf{s}_t)$$
$$\mathbf{rot}^p_i = \frac{\overline{\mathbf{h}}^{even}_{i,vector}}{\|\overline{\mathbf{h}}^{even}_{i,vector}\| + \text{eps}} \times \text{MLP}(\|\overline{\mathbf{h}}^{even}_{i,vector}\|, \mathbf{s}_t)$$
$$\text{with } \overline{\mathbf{h}}_{i,vector} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \mathbf{h}^j_{i,vector}$$

Here, T^p_i is a 5-dimensional scalar outputs representing torsion updates for [chi1, chi2, ..., chi5].

A.1.4 Transformation of the ligand conformation

To update the ligand conformation, we employ a unified global translation $\mathbf{tr}^l \in \mathbb{R}^3$ and rotation $R^l \in \mathbb{R}^{3\times3}$. All atoms of the ligand will be simultaneously translated and rotated around the geometric center of the ligand, which is calculated as $\overline{\mathbf{x}}^l = \frac{1}{n}\sum \mathbf{x}^l_i$, where n is the total number of heavy atoms of the ligand and $\mathbf{x}^l_i$ denotes the position vector of atom i. Specifically, the transformed position vector $\mathbf{x}^l$ is obtained as $\mathbf{x}^l = R^l(\mathbf{x}^l - \overline{\mathbf{x}}^l) + \overline{\mathbf{x}}^l + \mathbf{tr}^l$.

In addition to translation and rotation, torsion angles are also crucial factors in determining the ligand conformation. However, modifying torsion angles can perturb the position of the center of mass of the ligand. To address this issue, Corso et al.[15] demonstrated that performing an RMSD alignment after updating the torsion angles can ensure that the effect of the torsion updates is orthogonal to the roto-translation updates, and thus decouple the consequences of torsional updates and roto-translation updates. Overall, the updated ligand pose is obtained as $\mathbf{x}^l = \text{RMSDAlign}((T^l_0 \circ \cdots T^l_k)(\mathbf{x}^l), R^l(\mathbf{x}^l - \overline{\mathbf{x}}^l) + \overline{\mathbf{x}}^l + \mathbf{tr}^l)$, where T^l_k is the torsion rotation.

A.1.5 Transformation of the protein conformation

Following AlphaFold[1], we use Cα as the residue node to perform global translation and rotation. Additionally, the model predicts the updates of side-chain torsion angles. For $180°$-rotation-symmetric side chain parts, considering symmetry is unnecessary in the inference stage, but we introduce symmetry side chain torsion features during training to correctly compute the loss function. Since the position of the Cα is independent of the side chain torsion angles, rotating the side chain does not affect the residue-level translation and rotation. Thus, we can perform roto-translations and torsion rotations in any order. Finally, the updated conformation of each protein residue is represented as $\mathbf{x}_i^p = (T_{i,0}^p \circ \cdots T_{i,k}^p)(R_i^p(\mathbf{x}_i^p - \mathbf{x}_{i,c_\alpha}^p) + \mathbf{x}_{i,c_\alpha}^p + \mathbf{tr}_i^p)$, where $T_{i,k}^p$ is the side chain torsion rotation of i-th residue.

A.1.6 Training and inference

During the training process, the input are the protein structure in decoy conformation constructed by adding morph-like transformation to the native conformation and the ligand structure in conformation with Gaussian noise added. The expected output are the denoising operations. The input protein structure at time t is defined as $\mathbf{x}_t^p = \phi(\mathbf{x}^{holo}, t)$. Specifically, for the i-th amino acid, the Kabsch algorithm[35] is used to calculate the translation $\mathbf{tr}_i^*$ and rotation $\mathbf{rot}_i^*$ around $C\alpha$ that aligns the backbone atoms $N - C\alpha - C$ of the holo structure to the apo structure:

$$\mathbf{tr}_i^*, \mathbf{rot}_i^* = \text{Kabsch}(\mathbf{x}_{i,(N,C_\alpha,C)}^{holo} - \mathbf{x}_{i,C_\alpha}^{holo}, \mathbf{x}_{i,(N,C_\alpha,C)}^{apo} - \mathbf{x}_{i,C_\alpha}^{holo})$$

Considering the differences in torsion angles, we can draw the conformation changes of i-th residue:

$$\mathbf{x}_i^{apo} = \phi(\mathbf{x}_i^{holo}) = (T_{i,0}^* \circ \cdots T_{i,k}^*)(R_i^*(\mathbf{x}_i^{holo} - \mathbf{x}_{i,C_\alpha}^{holo}) + \mathbf{x}_{i,C_\alpha}^{holo} + \mathbf{tr}_i^*)$$

Here, $T_{i,k}^* = T_{i,k}^{apo} - T_{i,k}^{holo}$ are in radian and the R_i^* is the rotation matrix of $\mathbf{rot}_i^*$. At any given moment, we aim to perturb the protein structure using a factor, denoted as $u(t)$, such that the perturbed data is an intermediate state between the holo structure and apo structure:

$$\begin{aligned}\phi(\mathbf{x}_i^{holo}, t) = &(\Delta T_{i,0}^p \circ \cdots \Delta T_{i,k}^p)(\Delta R_i^p(\mathbf{x}_i^{holo} - \mathbf{x}_{i,C_\alpha}^{holo}) + \mathbf{x}_{i,C_\alpha}^{holo} + \Delta\mathbf{tr}_i^p)\\ \text{with} \quad &\Delta\mathbf{tr}_i^p = u(t)\mathbf{tr}_i^*\\ &\Delta R_i^p = \text{Rotation matrix of } u(t)\mathbf{rot}_i^*\\ &\Delta T_{i,k}^p = u(t)T_{i,k}^* + \mathcal{N}(0, 0.3)\\ &u(t) = \text{clamp}(\tau_{min}^p + (\tau_{max}^p - \tau_{min}^p) \times (5t)^{0.3}, \text{min} = 0, \text{max} = 1)\end{aligned}$$

where τ_{min}^p and τ_{max}^p represent the parameters of the diffusion noise.

To overcome the distribution shift between training and inference that arises from the use of RDKit-generated conformations as starting points in the inference process, we replace the training objective with the conformation $\mathbf{x}_0^l$ that matched to the ground-truth pose $\mathbf{x}^{gt}$[32, 15]. At time t, the input ligand pose is a random perturbed conformation:

$$\begin{aligned}\mathbf{x}_t^l = &(\Delta T_0^l \circ \cdots \Delta T_k^l)(\Delta R^l(\mathbf{x}_0^l - \overline{\mathbf{x}}_0^l) + \overline{\mathbf{x}}_0^l + \Delta\mathbf{tr}^l)\\ \text{with} \quad &\Delta\mathbf{tr}^l = (\mathcal{N}(0, \sigma_{tr}^l), \mathcal{N}(0, \sigma_{tr}^l), \mathcal{N}(0, \sigma_{tr}^l))\\ &\Delta R^l = \text{Rotation matrix of sampling from } p(\omega)\hat{\omega}\\ &\Delta T_k^l = \mathcal{N}(0, \sigma_{tor}^l)\\ &p(\omega) = \frac{1 - cos(\omega)}{\pi} \sum_{l=0}^{\infty} (2l+1) \exp\left(-l(l+1)(\sigma_{rot}^l)^2\right) \frac{\sin((l+1/2)\omega)}{\sin(\omega/2)}\end{aligned}$$

Here, $\overline{\mathbf{x}}_0^l$ is the geometric center of $\mathbf{x}_0^l$, $p(\omega)$ is the isotropic Gaussian distribution on SO(3) and the $\hat{\omega}$ is a unit vector generated by random sampling.

The network is trained with eight losses. The total loss can be defined as follows

$$\mathcal{L} = \frac{1}{3}\mathcal{L}_{\mathbf{tr}}^l + \frac{1}{3}\mathcal{L}_{\mathbf{rot}}^l + \frac{1}{3}\mathcal{L}_T^l + \frac{1}{3}\mathcal{L}_{\mathbf{tr}}^p + \frac{1}{3}\mathcal{L}_{\mathbf{rot}}^p + \frac{1}{3}\mathcal{L}_T^p + 0.01\mathcal{L}_A + 0.99\mathcal{L}_D$$

where $\mathcal{L}_{\mathbf{tr}}^{l}$,$\mathcal{L}_{\mathbf{rot}}^{l}$,$\mathcal{L}_{T}^{l}$ are the losses for the translation, rotation, and torsion of the ligand, respectively. The $\mathcal{L}_{\mathbf{tr}}^{p}$, $\mathcal{L}_{\mathbf{rot}}^{p}$, and $\mathcal{L}_{T}^{p}$ are the losses for the protein residues. The $\mathcal{L}_{A}$ is binding affinity loss and the $\mathcal{L}_{D}$ is contact LDDT loss. The distance difference for computing the ground-turth cLDDT is $d = |d(\mathbf{x}_0^l, \mathbf{x}^{holo}) - d(\mathbf{x}_t^l, \mathbf{x}_t^p)|$ (more details of the cLDDT score calculation can be found in A.2).

Since a rotation vector $\mathbf{u}$ represents the same rotation as another $\mathbf{v}$ if $\mathbf{u}$ and $\mathbf{v}$ have opposite orientation and $\|\mathbf{u}\| + \|\mathbf{v}\| = 2\pi$. So we take the minimum of the forward and opposite orientation losses when computing the rotation loss. The torsion angle losses are computed using the cosine of the angle difference between the predicted value and the added torsion angle noise.

During the inference process, we use the ligand structure with conformations generated by RDKit and the protein structure prediction by AlphaFold as the initial complex conformation. The complex structure is updated with 20 steps. To prevent the final conformation trapped in local minimum, in each step, a small random noise is added to the denoised ligand pose. For each pair, we perform 40 samplings and rank the binding conformations based on the predicted cLDDTs. We also noticed that the weighted averaged of the predicted binding affinity is a more accurate estimator of the experimentally measured affinity. The predicted cLDDT values is used as the weights.

DynamicBind has 63.67 million parameters and was trained for 5 days on eight Nvidia A100 80GB GPUs.

A.2 Evaluation Metrics

- **Contact Local Distance Difference Test (cLDDT) score** To assess the interaction between the protein and the ligand within the predicted complex structure, we determine the extent of intermolecular native contact formation. We adopt a definition similar to that of the Local Distance Difference Test (LDDT) score, previously employed for quantifying the nativeness of predicted protein structures [36]. The Contact-LDDT (cLDDT) score is computed by considering the distances less than 15Å among all pairs of ligand atoms and protein atoms. The distance difference is determined between the ground truth and the predicted complex structure, while accounting for symmetry. The final cLDDT score is derived from the mean fraction of conserved distances across four tolerance thresholds: 0.5, 1, 2, and 4Å.
- **Pocket Root Mean Square Deviation (pocket RMSD)**
  In order to evaluate the deviation of the predicted protein structure from the native protein structure surrounding the binding pocket, we compute the pocket Root Mean Square Deviation (pocket RMSD). This is performed using protein atoms located within 5Å of the reference ligand atoms. Initially, the predicted protein structure is aligned with the crystal protein structure. Subsequently, the RMSD between the predicted pocket atoms and the crystal pocket atoms is determined.
- **Clash score** Similiar to AlphaFill[10], the clash score is the root mean square (RMS) of the van der Waals overlaps [37] across all distances between the ligand atoms and the protein atoms, which are less than 4Å. It is computed as follows:

$$\text{clash score} = \sqrt{\frac{\sum_{i=0}^{N} \text{VdW overlap}_i^2}{N}}$$

  where N is the number of distances considered.

A.3 Dataset construction

Our training and test dataset was built upon the PDBbind2020 [24] database, which includes a curated collection of 19,443 crystal structures of protein-ligand complexes, each paired with an experimentally measured binding affinity. We employed the same time split as previous works [26, 27, 15], using structures deposited before 2019 for training and validation, while those deposited in 2019 were reserved for testing. Each protein was aligned with the AlphaFold-predicted structure that corresponds to the same protein sequence. The aligned AlphaFold structures and the crystal structures are used to generate training samples of the protein part through morph-like interpolation. The Major Drug Targets (MDT) test set was constructed using the following criteria: PDBs deposited in 2020 or later; proteins belonging to one of the four major drug target groups - kinases, GPCRs, nuclear receptors, and ion channels; the AlphaFold predicted protein structures have pocket RMSD above the 2Å (or pocket LDDT below 0.8) with the crystal structure; ligands are drug-like small-molecules with molecular weights between 200 and 650 Dalton; at most 10 PDBs from a single study are included. These criteria ensure that the test set is challenging, with the initial input protein differs from the native conformation, and is representative, covering a wide range of protein targets. In addition, it prevents a few proteins dominating the entire test set, as certain studies deposited significantly more PDBs, structures of the same protein co-crystallized with slightly different ligands, than other studies.

A.4 Baselines.

We performed docking on both PDBbind test set (303 ligand-receptor pairs) and Major drug targets (MDT) test set (599 ligand-receptor pairs) using different docking methods listed below. The docking ligands were extracted from the co-crystalized structures without changing their atomic coordinates and the docking receptor structures were predicted by AlphaFold.

A.4.1 Autodock VINA Rigid

In Autodock Vina[13], ligands were converted from SDF format to PDBQT format by Meeko 2.0.0. Protein preparation was performed by using the 'prepare_receptor' command in ADFR Suite 1.0. The search space was defined using an automatic box around the ligand with the default buffer of 4Å on all 6 sides. And the box center was the center of mass of the ground-truth ligand. Because the boron atom is not a valid AutoDock atom type, ligands with this atom cannot be docked. Therefore, only 301 ligand-receptor pairs in PDBbind dataset and 597 ligand-receptor pairs in MDT dataset had docking output in VINA rigid docking.

A.4.2 Autodock Vina Flex

Comparing to VINA rigid docking, there is an additional flexible receptor preparation step in VINA flexible docking. It was performed by a python script called 'prepare_flexreceptor.py', which is available at `https://github.com/ccsb-scripps/AutoDock-Vina/tree/develop/example/autodock_scripts`. Through this step, the protein PDBQT format file was divided into two PDBQT format files, one for the rigid part and one for the flexible side chains. Residue side chains within 5Å of the ligand atoms were defined as flexible side chains. Ligand preparation and grid box setting were consistent with VINA rigid docking.

A.4.3 GNINA Rigid

Though SDF format can be used as the input ligand files in GNINA[38], the order of atoms in output ligand files will change after docking, making it difficult for us to calculate the RMSD between the ligand pose before and after docking. Therefore, we chose to convert them to PDBQT format by using OpenBabel after adding hydrogens by Rdkit. Protein input files were PDB format files. And grid box setting was consistent with VINA rigid docking. For PDBbind dataset, all of the ligand-receptor pairs had docking output. For MDT dataset, 1 pair had no output because the ligand in the original PDB file (PDB ID: 8UHU) was not completely resolved, which had missing atoms.

A.4.4 GLIDE

GLIDE [12] is a rigid protein docking module in Schrödinger software. Ligands were prepared by using the LigPrep module. Protein preparation was performed by using the Protein Preparation Wizard module. Grid files were generated by the Receptor Grid Generation module with a 10Å inner box and an automatic outer box around the ligand with the default buffer of 4Å on all 6 sides centered on the center of mass of the ligand. Then, the SP precision docking was performed. Some of the ligands in PDBbind dataset are polypeptides, which cannot be processed by LigPrep module. In addition, ligands with severe clashes with pocket atoms had no output pose during docking. Therefore, 266 ligand-receptor pairs in PDBbind dataset and 472 ligand-receptor pairs in MDT dataset had docking output in GLIDE rigid docking.

B Further Experiments

B.1 Benchmark results for PDBbind and MDT test set individually.

As shown in Fig. S1a, the protein structures predicted by DynamicBind have significantly lower pocket RMSD in comparison with the input AlphaFold structures. The success rate of DynamicBind, 0.317, is 25% higher than the second best method (DiffDock, 0.247) under stringent criteria, as shown in Fig.S1b. The area under curve for plot of the true positive rate vs false positive rate under RMSD 2Å threshold is 0.85, Fig.S1c. The success rate of DynamicBind sees a sharp increase initially as the number of generated samples rises. This growth begins to plateau for rank 1 results once the sample count surpasses 20, yet it continues to ascend for the best sampled structures, Fig.S1d. Similar patterns can be observed in the MDT test set as well, Fig.S2.

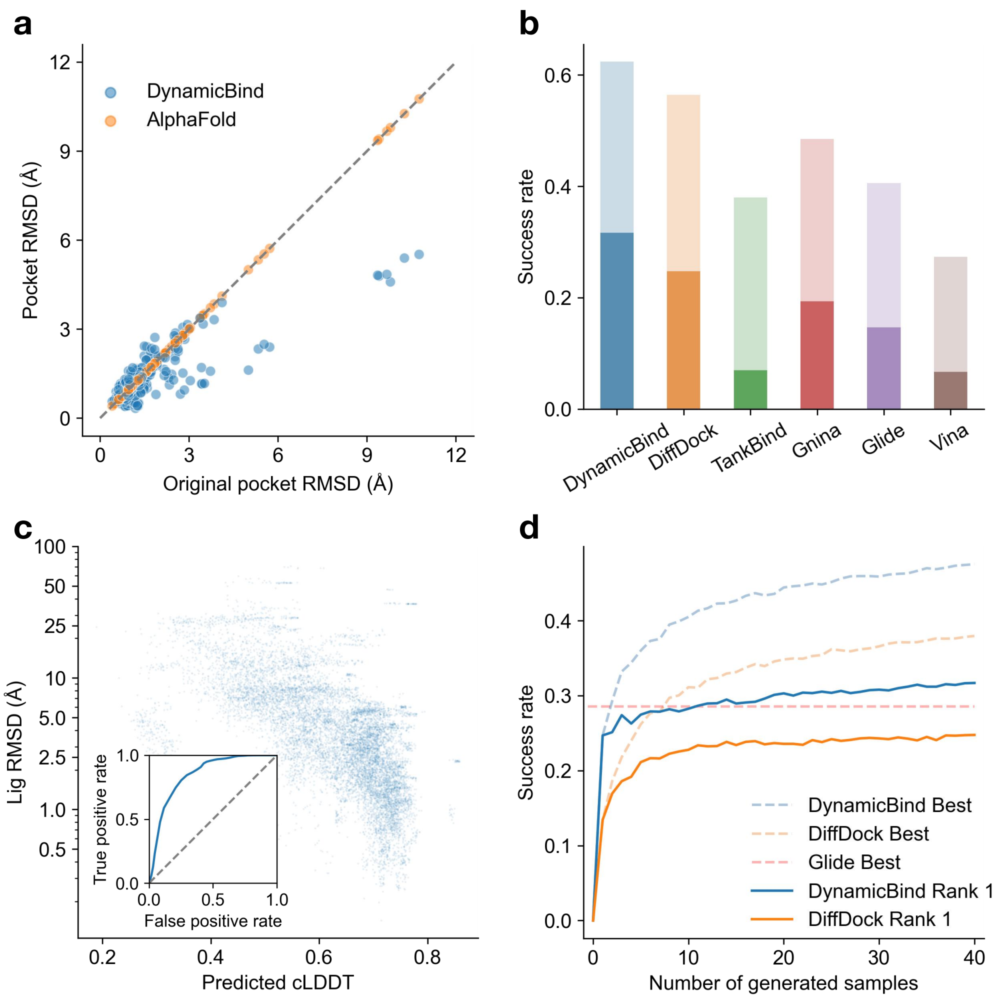

Figure S1: Same as Fig. 2, but evaluated only with PDBbind test set.

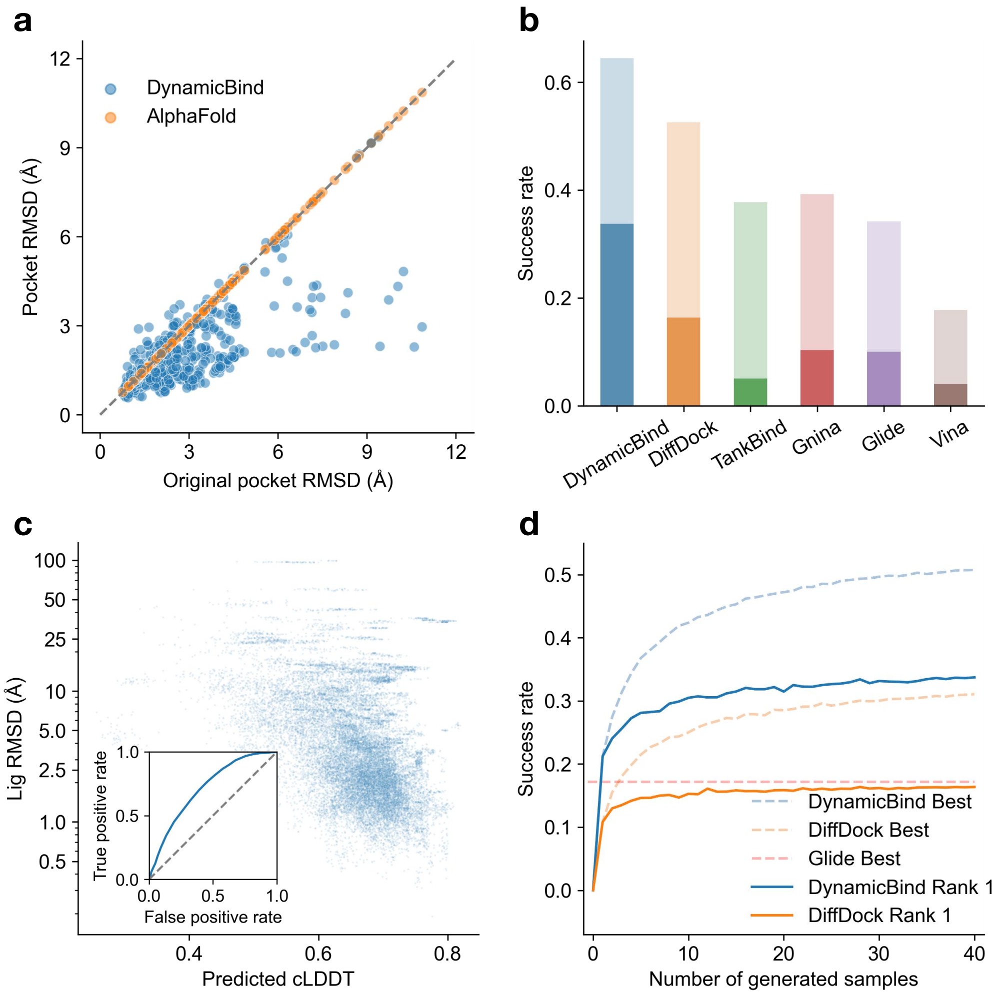

Figure S2: Same as Fig. 2, but evaluated only with MDT test set.

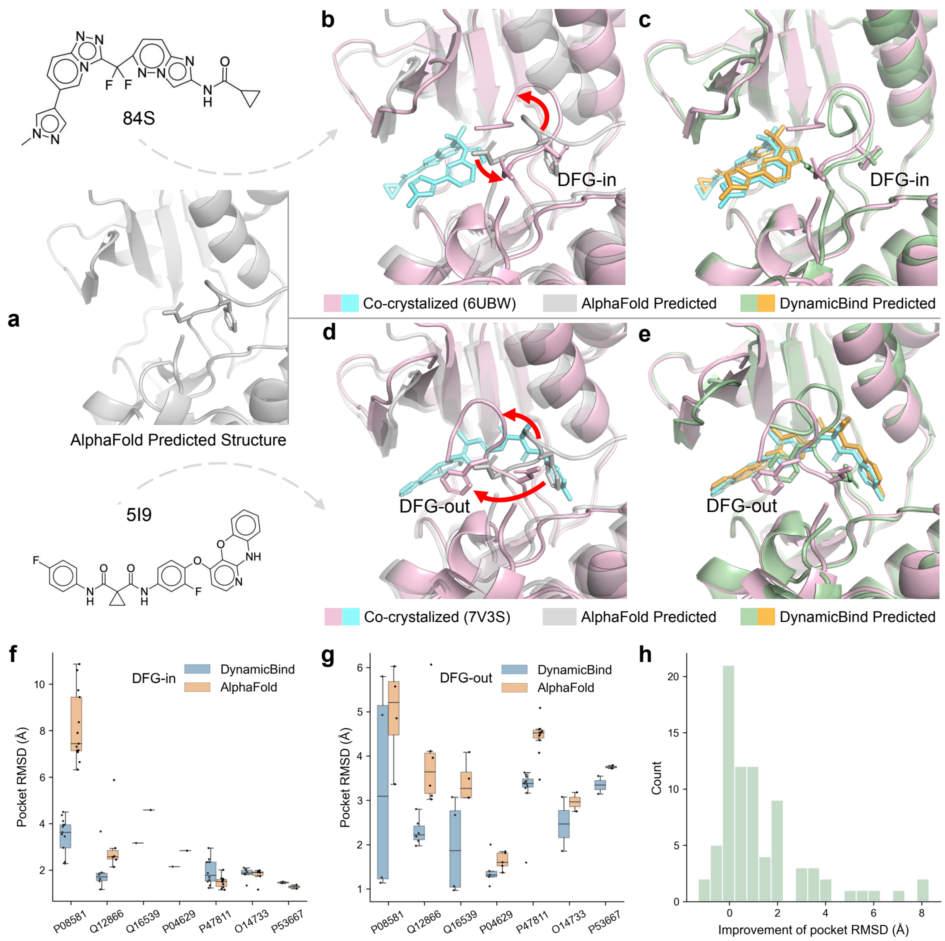

Figure S3: AlphaFold predicted structures are depicted in white, the crystal structure with protein and ligand in pink and cyan, respectively. Our model's predictions are shown in green and orange, for the protein and ligand respectively. The side chains of the Asp-Phe-Gly (DFG) residues are shown in stick. Red arrows highlight significant conformational changes of the crystal structure from the AlphaFold structure. The input conformation is the AlphaFold predicted conformation **a**. When the ligand 84S (**b**) binds to c-Met protein, the protein adopts a DFG-in conformation. When the ligand 5I9 (**d**) binds to the same protein, the protein adopts a DFG-out conformation. Our prediction for both ligands, **c** and **e**, agrees well with the crystal structure. Ligand RMSD is 0.49Å and 0.51Å. Improvement of Pocket RMSD from initial AlphaFold is 7.47Å and 4.83Å for DFG-in and DFG-out respectively. Among the test set, seven proteins (identified by their UniProt IDs), contains both DFG-in and DFG-out crystallized holo conformations, their pocket RMSD of both initial AlphaFold and predicted structures are shown in **f** and **g** for DFG-in holo conformations and DFG-out holo conformations separately. **h**, The histogram of the improvement in pocket RMSD from AlphaFold for all 79 PDBs.

B.2 DynamicBind can capture ligand-specific protein conformational changes

In traditional docking methods, the sampling of protein conformations is decoupled from the docking step. In many instances, however, two distinct ligands may fit into mutually exclusive protein conformations. For example, c-Met kinase can adopt two different conformations, corresponding to active and inactive states, typically referred to as the Asp-Phe-Gly (DFG)-in and DFG-out conformations (Fig. S3b, d). The DFG motif can flip out, subsequently blocking or opening up different regions of the protein. In previous docking models, the protein must be preset to the correct conformation to have a chance of identifying the appropriate binding pose for the ligand[16]. In contrast, DynamicBind, utilizing the protein conformation predicted by AlphaFold (Fig. S3a), can dynamically adjust the

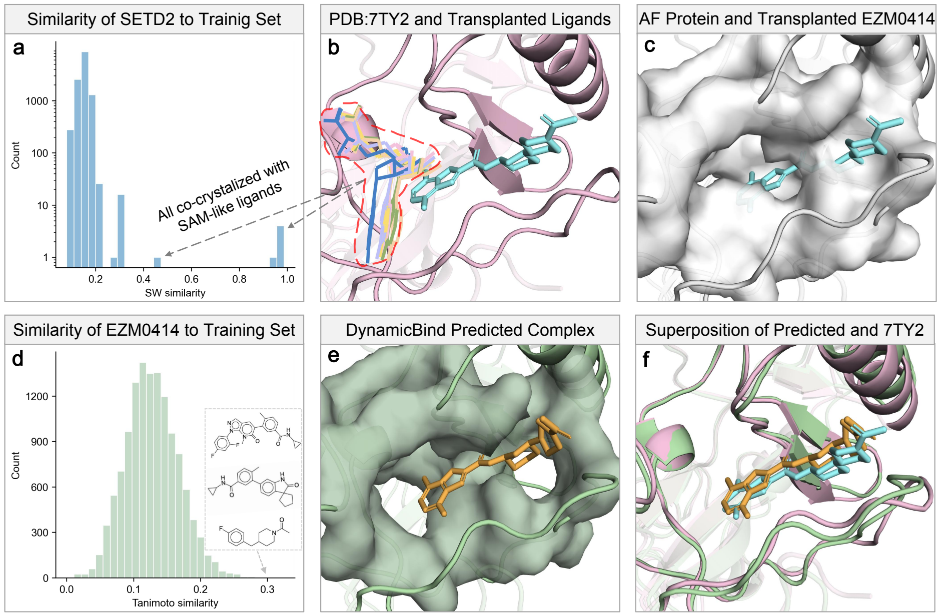

Figure S4: **a**, Only six PDBs in the training set have a protein Smith–Waterman similarity greater than 0.4 with the SETD2 protein, and all are co-crystallized with SAM-like ligands, also shown in lines in **b**, The ligand of PDB 7TY2, EZM0414, is displayed in cyan sticks, with the protein shown in pink. **c**, The binding pocket for EZM0414 is absent in the AlphaFold structure, depicted in white. **d**, This panel shows the Tanimoto similarity of ligands in the training set compared to EZM0414, and the top three most similar ligands are drawn out. **e**, The protein-ligand complex structure as predicted by DynamicBind, with the protein represented in green and the ligand in orange. **f**, The superposition of the complex as predicted by DynamicBind and the corresponding crystal structure.

protein conformation to find the optimal conformation that accommodates the ligand of interest. As a representative case, for PDB 6UBW, the predicted ligand RMSD is 0.49Å, and pocket RMSD is 1.97Å while the pocket RMSD for the AlphaFold structure is 9.44Å. For PDB 7V3S, the predicted ligand RMSD is 0.51Å, and the pocket RMSD is 1.19Å, (AlphaFold 6.02Å). Neither of the two ligands have been seen before in the training set (Fig. S3c, e). In our quantitative analysis, only seven proteins from the test set, represented in 79 PDB structures, were found to adopt both DFG-in and DFG-out conformations, as annotated by the Kinase–Ligand Interaction Fingerprints and Structures (KLIFS) web server [39]. Figs. S3f and g demonstrate how these proteins (denoted by their UniProt IDs), starting from the same initial structure, move progressively towards the DFG-in conformation upon type-I inhibitor binding, and incline towards the DFG-out conformation when interacting with a type-II inhibitor. Further, Fig. S3h reveals that the majority of the predicted protein structures show a lower pocket RMSD compared to the initial AlphaFold structures. These results demonstrate that, DynamicBind, is capable of capturing ligand-specific conformational changes. This feature is critical in preventing the overlooking of potential 'hit' compounds that could bind well with conformations distinct from the initially provided protein structure.

B.3 DynamicBind reveals novel cryptic pockets significant to drug discovery

The dynamic nature of proteins often gives rise to cryptic pockets. These cryptic pockets, which appear during protein dynamics, can reveal novel druggable sites not found in static structures, thus making previously 'undruggable' proteins into potential drug targets. We demonstrate the utility of DynamicBind in revealing these cryptic pockets using the SET domain-containing protein 2 (SETD2), a histone methyltransferase, as a case study. SETD2, critical for the treatment of multiple myeloma (MM) and diffuse large B-cell lymphoma (DLBCL) [40, 41], has a cryptic pocket targeted by a highly selective compound, EZM0414, currently undergoing Phase I clinical trials. As illustrated in Fig. S4a and b, all SETD2 homologs in the training set, defined by a protein Smith-Waterman similarity [42] over 0.4, are co-crystallized with S-Adenosyl methionine (SAM) or Sinefungin analogs, depicted in

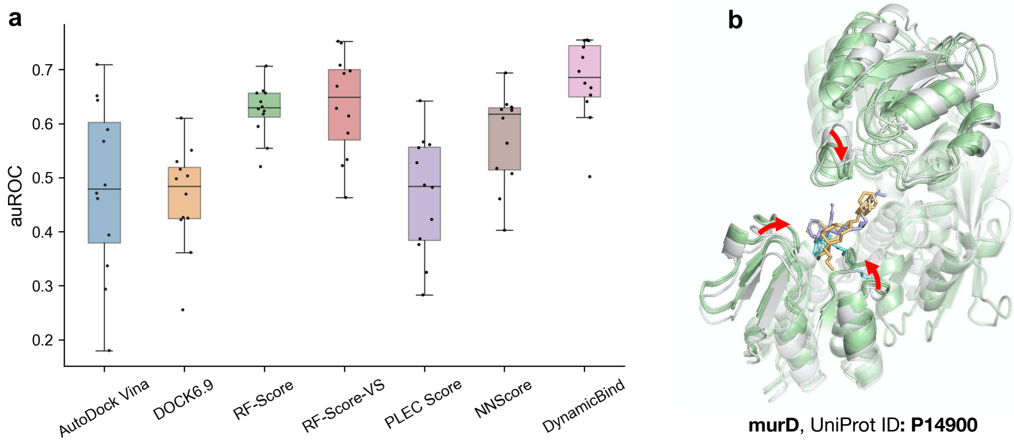

Figure S5: **a**, Comparative evaluation of the virtual screening performance on the antibiotics benchmark by different methods, measured in terms of auROC. **b**, The AlphaFold-predicted protein structure is shown in white, while the protein structures generated by DynamicBind for three active compounds are shown in green. Red arrows indicate the regions where the protein moves closer to the ligand, forming additional interactions.

lines. Sinefungin and its analogs broadly inhibit methyltransferases by occupying the SAM site [43], making the selective inhibition of SETD2 challenging. Before 2019, no structure of SETD2 or its homologs had been crystallized with a compound bound at the site targeted by EZM0414 (depicted in cyan sticks). Consequently, our model had not been trained on any structures with a compound bound to this newly-identified site. In Fig. S4c, the AlphaFold structure and its surface are shown in white. The cryptic site appears blocked, causing substantial clashes with the transplanted EZM0414. Fig. S4d confirms EZM0414 as an unseen ligand, with even the most similar Tanimoto ligands deviating substantially from EZM0414. Fig. S4e displays the protein-ligand complex structure predicted by our model, taking the AlphaFold-predicted structure of SETD2 and the SMILES representation of EZM0414 as inputs. Fig. S4f overlays our prediction with the crystal structure of the SETD2-EZM0414 complex (PDB 7TY2). The resultant ligand RMSD is 1.4Å, and the pocket RMSD is 2.16Å.

B.4 DynamicBind achieves better screening performance in an antibiotics benchmark

In target-based drug discovery, both screening of potential drug candidates and reverse screening, where protein targets are identified for specific compounds, are crucial. These processes require accurate prediction of binding affinities, the measure of the interaction strength between a protein and a compound, at a proteome-level. Therefore, we've added an affinity prediction module to our model, trained using experimentally measured binding affinity data from the PDBbind dataset. To assess DynamicBind in a real-world virtual screening scenario, we used a recently published antibiotic experimental benchmark [44]. This dataset includes a panel of 2616 protein-compound pairs, none of which were encountered during our training phase. It features 12 proteins from the essential proteome of Escherichia coli paired with 218 active antibacterial compounds. Fig. S5a shows that DynamicBind surpasses both common docking methods like VINA and DOCK6.9 and the best machine learning-based re-scoring methods, achieving the mean average area under the receiver operating characteristic curve (auROC) of 0.68. This performance improvement is due to DynamicBind's dynamic docking capability, which refines the AlphaFold structure towards a more native-like state, leading to a more precise binding affinity estimation. As depicted in Fig. S5b, the predicted structures of protein murD conform more closely around the ligand, forming more interactions that were not possible with the initial AlphaFold structure. This evaluation on the antibiotics benchmark agrees with our benchmarks on PDBbind test sets for binding affinity predictions , where DynamicBind consistently outperforms traditional docking methods and deep learning-based rigid docking methods. These results indicate that DynamicBind, with its binding affinity prediction capability, exhibits significant potential for proteome-level virtual screening applications.