ACS OMEGA

Open Access

http://pubs.acs.org/journal/acsodf

Article

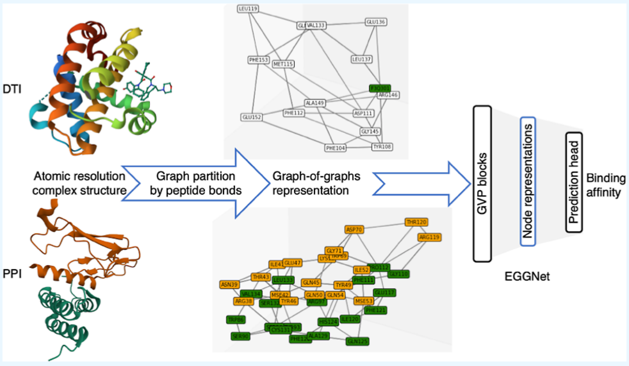
EGGNet, a Generalizable Geometric Deep Learning Framework for Protein Complex Pose Scoring

Zichen Wang,* Ryan Brand, Jared Adolf-Bryfogle, Jasleen Grewal, Yanjun Qi, Steven A. Combs, Nataliya Golovach, Rebecca Alford, Huzefa Rangwala,* and Peter M. Clark*

Cite This: *ACS Omega* 2024, 9, 7471−7479

Read Online

ACCESS | Metrics & More | Article Recommendations



ABSTRACT: Computational prediction of molecule−protein interactions has been key for developing new molecules to interact with a target protein for therapeutics development. Previous work includes two independent streams of approaches: (1) predicting protein−protein interactions (PPIs) between naturally occurring proteins and (2) predicting binding affinities between proteins and small-molecule ligands [also known as drug−target interaction (DTI)]. Studying the two problems in isolation has limited the ability of these computational models to generalize across the PPI and DTI tasks, both of which ultimately involve noncovalent interactions with a protein target. In this work, we developed Equivariant Graph of Graphs neural Network (EGGNet), a geometric deep learning (GDL) framework, for molecule−protein binding predictions that can handle three types of molecules for interacting with a target protein: (1) small molecules, (2) synthetic peptides, and (3) natural proteins. EGGNet leverages a graph of graphs (GoG) representation constructed from the molecular structures at atomic resolution and utilizes a multiresolution equivariant graph neural network to learn from such representations. In addition, EGGNet leverages the underlying biophysics and makes use of both atom- and residue-level interactions, which improve EGGNet's ability to rank candidate poses from blind docking. EGGNet achieves competitive performance on both a public protein−small-molecule binding affinity prediction task (80.2% top 1 success rate on CASF-2016) and a synthetic protein interface prediction task (88.4% area under the precision−recall curve). We envision that the proposed GDL framework can generalize to many other protein interaction prediction problems, such as binding site prediction and molecular docking, helping accelerate protein engineering and structure-based drug development.

■ INTRODUCTION

Physical interactions between a protein and other molecules are key to many fundamental biological processes. Proteins mostly perform their functions via noncovalent interactions with three kinds of molecules, including proteins, nucleotides, and small molecules. The mechanisms of action for most drugs involve interacting with protein targets to modulate their biological functions and activities. Being able to design drugs, either small molecules or biologics, to selectively bind a protein target with a desirable affinity is critically important for structure-based drug design.

The ability to modulate protein-involved molecular interactions with small molecules or synthetic peptides is a core component of therapeutics development. There exist four classes of problems in structure-based drug design for which molecular interactions can be tackled by machine learning (ML),[1] including (i) protein complex property prediction, (ii) binding site/interface identification, (iii) docking (binding pose generation), and (iv) de novo design. Property prediction for protein complexes is a task that can also serve as an integral component for other tasks, such as evaluating binding poses generated by docking algorithms or predicting the affinity for computationally designed novel drug candidates. Many ML methods have been developed for two popular groups of property prediction tasks, protein−protein interactions (PPIs) and drug−target interactions (DTIs), with the structure of the protein complex provided as input. However, to the best of our knowledge, none of the existing methods can be used across both problems. In this work, we developed a unifying geometric deep learning (GDL) framework for protein complex pose

Received: July 21, 2023
Revised: January 19, 2024
Accepted: January 23, 2024
Published: February 9, 2024

https://doi.org/10.1021/acsomega.3c04889

scoring that encompasses both DTI affinity prediction and PPI interface prediction.

The proposed approach provides a generalizable and scalable representation of macromolecular complexes that can efficiently represent both protein−small-molecule complexes and protein−protein complexes. We also desire the representation to be capable of handling synthetic peptides composed of both natural and noncanonical amino acids. 3D graphs are commonly used to represent the 3D structures of individual proteins and protein−small-molecule complexes.[1] To increase its generalizability and scalability, we extend the 3D graph to a 3D graph of graphs (GoG) representation.

GoG, also known as Network of Networks, is a special graph where the nodes in the top-level graph are also graphs.[2,3] GoGs can be constructed naturally by connecting independent lower level graphs by predefined associations,[4] or similarity can be measured by a graph kernel.[5] GoG can also emerge from partitioning a large graph into different subgraphs based on topology or predefined rules. Message-passing GNNs have also been extended to operate on both levels of graphs in GoG.[4−6] The GoG data structure has been used to represent drug−drug interaction networks and metabolite networks.[4,6] However, it has not been applied to model the 3D structures of macromolecules.

In this study, our contributions are 2-fold: (i) we developed a featurization procedure to use the GoG data structure to efficiently unify the representations of all types of molecules, including small molecules, intermediate molecules such as peptides, and macromolecules. The 3D GoG representation retains both atomic- and residue-level information on molecular complexes. (ii) We developed EGGNet, an end-to-end GDL architecture based on an equivariant graph neural network (GNN) to learn from the GoG representations of the 3D structures of protein complexes, optionally integrating a physics-informed inductive bias to learn atomic-level interactions.[7] Our architecture can be used to predict both DTIs and PPIs. Notably, we achieved state-of-the-art protein−small-molecule binding affinity predictive performance. We further analyzed the effects of different choices of lower level molecule graph models and evaluated the potential of transfer learning between the DTI and PPI prediction tasks. We also showed that our model improves the outputs of the blind docking models.

RELATED WORK

Prior approaches for structure-based prediction of DTIs and PPIs with ML represent 3D structures of protein complexes using three common representations of protein structures based on 3D grids, 3D surfaces, and 3D graphs.[1] Among them, the 3D graph representation is the only one that preserves all of the information from the input protein structure. However, representing macromolecules with 10,000 atoms using 3D molecular graphs at all-atom resolution is computationally inefficient. Therefore, researchers decomposed the all-atom 3D graphs to graphs of frames, where individual nodes (frames) represent monomers such as amino acid residues. For instance, AlphaFold2's Invariant Point Attention (IPA)[8] models amino acids as residue gas, which is composed of backbone frames and χ angle for the side chain. However, residue gas is only a coarse-grained representation that does not preserve all of the covalent bonds in the residue compared to all-atom molecular graphs.

For the DTI prediction, all the three types of representations and some of their combinations have been explored. For instance, KDEEP[9] estimates binding affinities by representing the protein−ligand complex as a 3D grid and learns from this representation using a 3D convolutional neural network. 3D graph representations of protein−ligand complexes can be used to extract topological and spectral features for traditional ML models,[10−13] achieving state-of-the-art performance on scoring benchmarks.[10] On the other hand, 3D graphs can also be leveraged by graph ML approaches. For instance, PotentialNet[14] builds 3D graphs for the protein−ligand complex using their atoms as nodes and chemical bonds and noncovalent interactions as edges and then leverages message passing graph neural networks (GNNs)[15] to learn from the 3D heterogeneous graph. PIGNet[7] extended PotentialNet by adding the physics-informed inductive bias to a gate-augmented graph attention network.[16] The physics-informed inductive bias is encoded by parameterized energy equations calculating a few noncovalent forces based on the corresponding interatomic distances. HoloProt[17] combines the graph and surface representations of proteins. These studies decompose amino acid residues into atoms and bonds, making it difficult to incorporate potentially useful residue-level features for the ML model to learn from, such as embeddings from protein language models.[18−21]

3D graph representations that capture the structural information at atomic resolution are computationally more expensive for PPIs than DTIs; therefore, 3D grids are the most prevalent representation of 3D protein complexes in protein−protein interface prediction. For instance, DeepRank[22] is a 3D CNN-based deep learning method that first maps the amino acid residues at the protein−protein interface to a 3D grid centered on the interface. More recently, deep local analysis[23] extends such 3D grid representations to an ensemble of 3D grids.

METHODS

Overview of the EGGNet Approach. Overall, EGGNet takes the 3D structure of a protein complex (also known as pose) as the input and makes predictions about the global properties of the pose, such as the binding affinity between two molecules in the protein complex. We refer to this problem as protein complex pose scoring. The input protein complex is composed of a protein molecule and an interaction partner that takes one of the three types: (1) small molecules, (2) synthetic peptides, and (3) natural proteins. EGGNet first converts the input protein complex pose into a GoG to unify the representations of different types of molecules. Next, EGGNet's architecture can learn from GoG representations by coupling an equivariant message-passing GNN with different lower level molecule-to-vector methods. Further, EGGNet incorporates physics-informed energy-inductive biases.

GoG Representation of Molecule Complexes. Formally, a GoG is a higher level graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set is composed of lower level graphs $\mathcal{V} = \{G_1, G_2, ..., G_n\}$. The lower level graph is used to represent the molecule's building blocks at atomic resolution; hence, $G_i = (V_i, E_i)$ is a graph with atoms as nodes, *atom* $\in V$, and covalent bonds as edges, *bond* $\in E$. For convenience, we represent all notations used in this paper in Table 1.

Given a molecular structure, one can directly construct an atomic resolution graph for the entire molecule by connecting atoms with covalent bonds. We denote this graph as $G = (V, E)$. To construct a GoG from G, we first perform edge-cut graph partitioning to partition V into disjoint subsets $V_1 \cup \cdots \cup V_n = V$ of all of the atoms, $v_u \in V$, in graph G, resulting in n subgraphs, each

Table 1. Table of Notations

symbol	definition
$G = (V, E)$	an atomic graph of a molecule, with atoms as nodes, *atom* $\in V$, and covalent bonds as edges, *bond* $\in E$
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	a GoG where the node set is composed of lower level graphs $\mathcal{V} = \{G_1, G_2, ..., G_n\}$
$f_\theta : G \to \mathbb{R}^d$	a residue featurizer parameterized by θ, mapping an atomic graph to a vector
$\mathbf{s}^G, \mathbf{s}^\mathcal{G} \in \mathbb{R}^d, \mathbb{R}^m$	scalar features on a node a graph
$\mathbf{x}^\mathcal{G} \in \mathbb{R}^{\nu \times 3}$	vector features on a node of a graph
$\mathbf{h} = (\mathbf{s}, \mathbf{x})$	features on a node or an edge of a graph
$\mathbf{H}^\mathcal{V} \in \mathbb{R}^{n\times(m+d)}, \mathbb{R}^{n\times\nu\times3}$	all the node features on a GoG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$\mathbf{H}^\mathcal{E} \in \mathbb{R}^{\lvert\mathcal{E}\rvert\times m'}, \mathbb{R}^{\lvert\mathcal{E}\rvert\times\mu\times3}$	all the edge features on a GoG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$\mathbf{Z} \in \mathbb{R}^{n\times d'}$	node embeddings on the higher level graph of a GoG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$f_\phi : \mathcal{G} \to \mathbb{R}^{n\times d'}$	a GVP-GNN parameterized by ϕ
$E^{(\cdot)} \in \mathbb{R}$	energy

of which corresponds to a lower level graph, $G_i = (V_i, E_i)$. We performed the graph partition by cutting the molecular graph at peptide bonds (Figure 1). Next, we construct the higher level graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, using the k-nearest neighbor (kNN) algorithm to connect $k = 30$ nearest residues following prior works.[24,25] The distance between subgraphs is defined as the Euclidean distance between the C-alpha or the geometric centroids of heavy atoms if the subgraph is not an amino acid residue.

With this procedure, we can unify the representations of small molecules and macromolecules: a small molecule is a special case of GoG where the higher level graph is composed of a single node (i.e., $\mathcal{G} = (\mathcal{V} = \{G_1\}, \mathcal{E})$). The GoG representation has the advantage over one-hot encoding with fixed vocabulary commonly used in protein language models[18−21] because it can generalize to noncanonical amino acids such as penicillamine used in synthetic peptides.

We also use the same GoG construct to represent a protein in complex with other molecules including small molecules, peptides, or proteins. To do that, we simply apply the edge-cutting graph partition procedure to every polymer chain to derive the lower level graphs and then construct the higher level graph by using kNN to connect spatially proximal lower level graphs (Figure 2).

Model Architectures for Protein Complex Pose Scoring. This section describes the novel deep learning architectures for protein complex pose scoring. The overall architectures for EGGNet are shown in Figure 3.

We formulate the protein complex pose scoring problem as a supervised graph-level prediction task. The goal of this task is to learn a function, $\hat{y} = f(\mathcal{G})$, mapping a GoG, $\mathcal{G}$, representing the protein complex's structure (pose) to a scalar value, y, representing the global property of the pose such as the binding affinity between the molecules.

EGGNet first computes feature vectors $\mathbf{s}_i^G \in \mathbb{R}^d$ from the lower level graphs G_i representing residues by $\mathbf{s}_i^G = f_\theta(G_i)$. There are many possible choices for the residue featurizer, $f_\theta : G \to \mathbb{R}^d$, including (1) chemical fingerprints such as MACCS and ECFP/Morgan,[26] (2) GNN models trained on small-molecule graphs, and (3) language models trained on the SMILES strings of small molecules such as MolT5.[27]

The residue feature vector is then concatenated with the node features on the higher level graph, $\mathbf{s}_i^\mathcal{G} \leftarrow concat(\mathbf{s}_i^G, \mathbf{s}_i^\mathcal{G}) \in \mathbb{R}^{m+d}$. We denote the node features on higher level graph as $\mathbf{h}_i^\mathcal{G} = (\mathbf{s}_i^\mathcal{G}, \mathbf{x}_i^\mathcal{G}) \in \mathbb{R}^{m+d}, \mathbb{R}^{\nu\times3}$; for simplicity, we denote the node and edge features as $\mathbf{H}^\mathcal{V} \in \mathbb{R}^{n\times(m+d)}, \mathbb{R}^{n\times\nu\times3}$ and $\mathbf{H}^\mathcal{E} \in \mathbb{R}^{\lvert\mathcal{E}\rvert\times m'}, \mathbb{R}^{\lvert\mathcal{E}\rvert\times\mu\times3}$, respectively.

GVP-GNN[25] is a SE(3)-equivariant GNN in which all node and edge embeddings are tuples $(\mathbf{s}, \mathbf{x})$ of scalar features $\mathbf{s}$ and

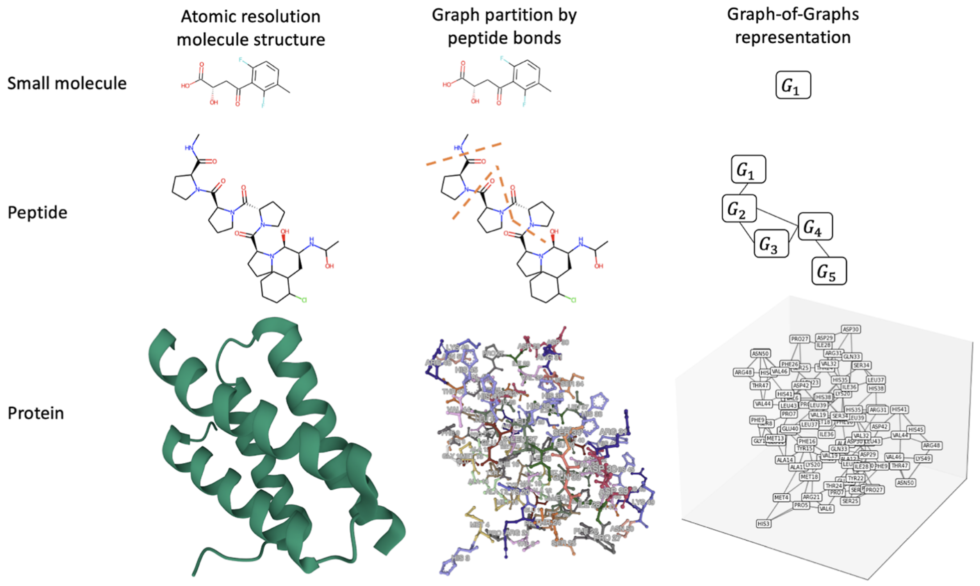

Atomic resolution molecule structure
Graph partition by peptide bonds
Graph-of-Graphs representation
Small molecule
Peptide
Protein
G_1
G_1
G_2
G_3
G_4
G_5

Figure 1. GoG representation of molecular structures. To construct GoG from molecular structure (left column), we first perform graph partition by cutting peptide bonds (middle column) and then connect subgraphs using kNN, leading to a higher level graph (right column). The figure shows how GoG representation can generalize to small molecules, peptides with noncanonical amino acid residues, and proteins.

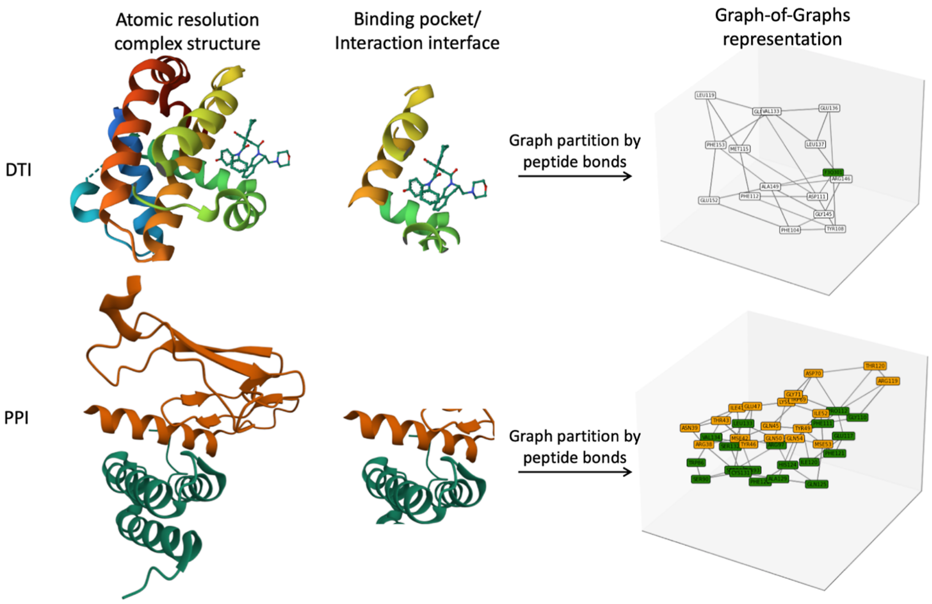
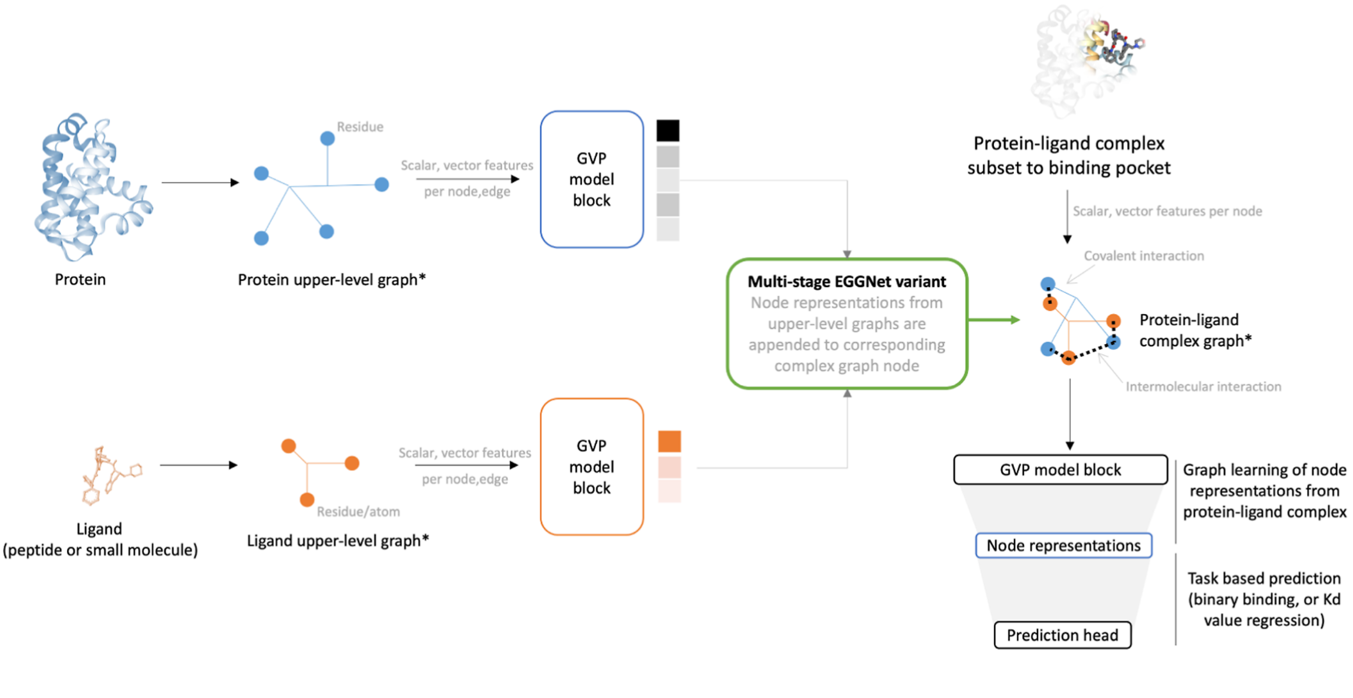


Figure 2. GoG representation of protein complexes. To construct GoG from the structures of protein complexes (left column), we first identify the binding pocket or interaction interface between molecules (middle column), then perform graph partition by cutting peptide bonds, followed by connecting subgraphs using kNN, leading to a higher level graph (right column). The nodes in the higher level graphs of GoGs are colored by the origin of the molecules. The figure shows how GoG representation can generalize to DTI and PPI.


Protein
Residue
Protein upper-level graph*
Scalar, vector features per node,edge
GVP model block
Protein-ligand complex subset to binding pocket
Scalar, vector features per node
Covalent interaction
Multi-stage EGGNet variant
Node representations from upper-level graphs are appended to corresponding complex graph node
Protein-ligand complex graph*
Intermolecular interaction
Ligand (peptide or small molecule)
Residue/atom
Ligand upper-level graph*
Scalar, vector features per node,edge
GVP model block
GVP model block
Node representations
Prediction head
Graph learning of node representations from protein-ligand complex
Task based prediction (binary binding, or Kd value regression)

* Simplified view for representation purposes only

Figure 3. EGGNet model architecture. The multistage EGGNet model learns GVP embeddings from graph representations of the protein and ligand structures and appends these to the corresponding nodes of the protein−ligand complex graph representation. The single-stage EGGNet model does not append node embeddings from these upper level graphs and only trains for the energy objective with the complex graph.

geometric vector features **x**. GVP-GNN operates on the node and edge features from the higher level graph

$$\mathbf{Z} = f_{\phi}(\mathbf{H}^{\mathcal{V}}, \mathbf{H}^{\mathcal{E}}) \tag{1}$$

Next, we use a readout network/prediction head to take the learned equivariant node representations, $\mathbf{Z} \in \mathbb{R}^{n\times d'}$, to make graph-level prediction

$$\hat{y} = READOUT(\mathbf{Z}) \tag{2}$$

We also design a multistage variant of our model architecture to have two additional GVP-GNNs to learn from two GoGs representing the two interacting molecules, $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$, which are two subgraphs of the complex GoG, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$ and $\mathcal{E} = \mathcal{E}_1 \cup \mathcal{E}_2 \cup \mathcal{E}_{\text{int}}$. $\mathcal{E}_{\text{int}}$ denotes the edges representing intermolecular interactions.

https://doi.org/10.1021/acsomega.3c04889

$$\mathbf{Z}_1 = f_{\phi_1}(\mathbf{H}^{\mathcal{V}_1}, \mathbf{H}^{\mathcal{E}_1}) \tag{3}$$

$$\mathbf{Z}_2 = f_{\phi_2}(\mathbf{H}^{\mathcal{V}_2}, \mathbf{H}^{\mathcal{E}_2}) \tag{4}$$

The second stage of GVP-GNN then computes node representations

$$\mathbf{Z} = f_{\phi_3}(concat(\mathbf{Z}_1, \mathbf{Z}_2), \mathbf{H}^{\mathcal{E}_{\text{int}}}) \tag{5}$$

Objective Functions and Model Training. We use the mean-squared error and binary cross-entropy for binding affinity regression and binary interaction prediction tasks, respectively. Additionally, we adopted the physics-informed energy-inductive biases from PIGNet.[7] Briefly, we use an energy decoder as the readout function to approximate four types of noncovalent interaction energies (van der Waals interactions, hydrogen bonds, metal−ligand interactions, and hydrophobic interactions) from a protein complex using parameterized equations. The energy decoder is a set of two-layered multilayer perceptrons approximating the four individual noncovalent interaction energies with the same setting from the PIGNet model.[7]

$$\hat{E} = Energy\ Decoder(\mathbf{Z}) \tag{6}$$

$$\mathcal{L} = \text{MSE}(\hat{E}, y) \tag{7}$$

All the parameterized equations take the atom-level representation, $\mathbf{z}_i$, $\mathbf{z}_j \in \mathbb{R}^{d'}$, as inputs. For instance, van der Waals interaction takes the following form

$$E^{\text{vdW}} = \sum_{i,j} c_{ij}\left[\left(\frac{d'_{ij}}{d_{ij}}\right)^{12} - 2\left(\frac{d'_{ij}}{d_{ij}}\right)^{6}\right] \tag{8}$$

where $d_{ij} = \|\mathbf{z}_i - \mathbf{z}_j\|$ and $d'_{ij} = r_i + r_j + \text{MLP}([\mathbf{z}_i,\mathbf{z}_j])$. r_i denotes the radii or the ith node.

We also decompose the energy from the whole protein complex to intramolecular (monomer) and intermolecular energies $E = E^{(1)} + E^{(2)} + E^{\text{int}}$ as a novel physics-inductive bias, where $E^{(1)}$, $E^{(2)}$ denote intramolecular energy from the two molecules and E^{int} denotes the intermolecular energy. Intramolecular energy can be computed by selecting atoms within a molecule and plugging into eq 8, whereas intermolecular energy can be computed by enumerating atom pairs spanning two molecules.

We use stochastic gradient descent (SGD) to learn parameters ϕ, and optionally residue featurizer parameter θ, to optimize the objective function. When using SGD to learn both ϕ and θ, we essentially allow the end-to-end training of the lower level residue featurizer and the higher level GVP-GNN. Similar end-to-end joint training of models operating at different data modalities has also been used for protein function prediction.[28]

Datasets and Experiments. We use PDBbind,[29] ProtCID,[30] MANY,[31] and DC[32] datasets for training and validation of our models. These datasets cover two otherwise distinctive tasks, DTI and PPI prediction, that can be unified by our approach.

For the DTI binding affinity regression task, we used the same splits of the PDBbind and CASF-2016[33] datasets as the PIGNet study.[7] Briefly, the PDBbind 2019 refined set provides quantified binding affinity data (in pK_d, where K_d is the experimentally measured dissociation constant) and corresponding structure of protein−ligand complexes deposited in the protein data bank (PDB).[34] We used 4514 samples for the training set, which is the PDBbind 2019 refined set after removing the redundant samples from the core set of PDBbind 2016. We randomly sampled 20% from the training set as the validation set for early stopping. For hold-out evaluation, we used the CASF-2016 benchmark dataset, which originated from the PDBbind 2016 core set. We used 283 samples for the scoring and ranking and 22,340 samples for the docking benchmark. All protein−ligand complexes were processed to focus on the pocket-ligand structure, which removes amino acid residues whose minimum distance between the ligand is greater than 5 Å to remove amino acid residues distant to the protein's binding pocket. We adopt the evaluation metrics recommended in the CASF-2016 dataset.[33] Briefly, we calculate Spearman's correlation coefficient ρ as the evaluate metric for the ranking setting in CASF-2016. For the docking setting, we used the predicted binding affinities to rank candidate poses from each protein−ligand pair and then define a success docking if the top 1 ranked pose has the lowest root-mean-square deviation (rmsd) among candidate poses (decoys). We then compute the top 1 success rate across all protein−ligand pairs as the fraction of successful docking from all docking experiments in the CASF-2106 benchmark.

ProtCID dataset[30] is used for the binary binding classification task. Here, the aim is to distinguish physiological binders with nonphysiological ones. Physiological binders were defined as those homo- and heterodimers which had at least five crystal forms. If an interface is only seen in one crystal form of this UniProt ID in addition to not being in any common cluster, then it is likely to be nonphysiological. For data preprocessing, we remove amino acid residues whose C-beta distance is greater than 6 Å from any amino acids from the protein chain of the binding partner. We also removed large interfaces with more than 10^5 atoms, leading to 15,736 and 3889 protein interfaces for training set and hold-out test set, respectively. The train/test split was performed by sequence clusters resulting from MMseq2[35] with an identity cutoff of 30%.

Similarly, the MANY[31] and DC[32] datasets contain physiological and nonphysiological (crystal) dimers in balanced proportions. We download the datasets from the SBGrid data repository, https://data.sbgrid.org/dataset/843/, and follow the same experimental setup as in refs 22 and [36]. Briefly, we train our model using 80% of the MANY dataset, with 20% as the validation set, and test the model performance on the hold-out DC dataset. For data preprocessing, we experimented a range of cropping threshold from 6 to 14 Å to when removing amino acid residues outside of the interfaces (Figure 4).

We evaluate the model performance on the binary classification task by using the area under the receiver operating characteristic curve (AUROC) and the area under the precision−recall curve.

For EGGNet models trained on the above datasets, we set GVP-GNN to have three GVP convolutional layers with node- and edge-hidden dimensions to be (200, 32) and (64, 2), respectively. In the single-stage EGGNet, only one three-layered GVP-GNN is used as the higher level GNN, whereas the multistage EGGNet contains three independent three-layered GVP-GNNs without weight sharing. We used the following hyperparameters for model training: learning rate of 1×10^{-4}, per-GPU batch size of 16. To avoid overfitting, we employed an early stopping criterion with patience = 50 epochs and trained for a maximum of 1000 epochs. ADAM[37] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ is used for optimizing the learnable parameters.

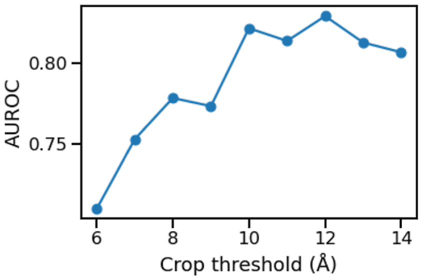


Figure 4. Performance of EGGNet on the biological versus crystal interface classification task from the DC dataset. AUROCs are shown across a range of cropping thresholds used to isolate the protein−protein interfaces when constructing the GoGs. Experiments were performed using GIN as the lower level GNN within the single-stage EGGNet architecture without joint training. We found 12 Å to be the optimal threshold for isolating the interface.

All models are implemented using the Pytorch deep learning library,[38] and training is performed using the Pytorch-Lightning library with 16 bit mixed precision training using four NVIDIA V100 GPUs with 16GB of memory each on Amazon SageMaker.

RESULTS AND DISCUSSION

GoG Representation of Molecules Allows Flexible Transfer Learning of Molecule-to-Vector (Mol2Vec) Methods. In this work, we leverage the GoG data structure to represent the 3D structures of small- and macromolecules. Macromolecules, such as proteins and nucleic acids, are composed of small-molecule building blocks (amino acids and nucleotides) connected by covalent bonds. Inspired by this observation, we view a macromolecule as a graph of small-molecule residues, which can be represented by graph of atoms connected by chemical bonds. The GoG representation of molecules provides the flexibility in modeling macromolecules with out-of-vocabulary residues such as post-translational modifications and noncanonical amino acids. It also unifies the chemical space of small- and macromolecules by applying Mol2Vec methods to the lower level graphs.

We first demonstrate that EGGNet is able to take advantage of any Mol2Vec representation approaches on the GoG representation of proteins in complex with small-molecule ligands. We experimented with three classes of Mol2Vec approaches: (i) chemical fingerprint approaches including MACCS keys and Morgan fingerprints,[26] (ii) GNNs pretrained on a small-molecule chemical library using structures and properties, and (iii) language models pretrained on SMILES strings of small molecules. We evaluated the performance of single-stage EGGNet with different Mol2Vec methods for the PDBbind/CASF-2016 binding affinity regression task. Our results show that Mol2Vec methods based on deep learning such as Graph Isomorphism Network (GIN)[39] and MolT5[27] achieved better performance than chemical fingerprints (Table 2) by 0.9% in docking capability while underperformed the latter in ranking setting. We also observe that EGGNet with the MolT5 model achieves an improved docking success rate than with GIN by 1.3%, suggesting that the more powerful pretrained Mol2Vec model can further boost the overall model performance. However, MolT5 models (MolT5-small: ∼77 M parameters; MolT5-base: ∼250 M parameters) are significantly larger than GIN (∼1.8 M parameters). As such, we use GIN for later experimentation due to the training efficiency.

Table 2. Residue-Level Representation Models Improve Binding Affinity Prediction[a]

Mol2Vec methods	docking top 1 success rate	ranking ρ
baseline (zero vector)	0.802	0.675
MACCS	0.820	**0.727**
Morgan/ECFP4	0.806	0.670
GIN	0.816	0.715
MolT5-small	0.760	0.702
MolT5-base	0.813	0.715
MolT5-small with joint training	**0.827**	0.722

[a] The following experiments use GVP backbone with energy-inductive biases and identical training hyperparameters and setups, except for the residue-level model.

It worth noting that chemical fingerprints (MACCS) and pretrained molecule GNNs are competitive feature representations for amino acid residues and small molecules to be utilized by the higher level GNN in EGGNet. However, this does not suggest that GIN's representation lacks advantages because it can be jointly optimized with the higher level GNN in EGGNet.

Performance of EGGNet on Protein−Molecule Binding Prediction Benchmarks. Next, we evaluated EGGNet on two distinctive protein complex pose scoring tasks: (i) protein−small molecule binding affinity prediction and (ii) protein−protein interface classification.

Protein−Small Molecule Binding Affinity Regression. On the binding affinity prediction task, we trained our model on the PDBbind training set and evaluated on the CASF-2016 ranking and docking benchmarks. Our results show that EGGNet with joint training and energy-inductive biases outperforms PIGNet[7] in the same setting on both docking and ranking benchmarks (Table 3). We found that our best performing model on the docking setting achieved an 80.2% top 1 success rate over 77.4% reported in PIGNet.

Interestingly, we found that the joint training of the GNNs at both the lower level and higher level graphs of the GoG improves the models' performance on the ranking task while slightly decreasing performance on the docking task. The ranking task examines a model's ability to generalize to different combinations of different protein−ligand pairs, which corresponds to different compositions and geometries of the GoGs. Meanwhile, the docking task probes the models' ability to rank candidate poses of the same protein−ligand pair, which corresponds to GoGs with the same composition of lower level graphs but different geometries. Therefore, we reason that the model with the ability to adjust the embeddings of the lower level residue representations is beneficial only to tasks that prioritize in distinguishing the composition rather than the geometry of the GoGs. This also explains the observation that the multistage variant of EGGNet is more performant on the docking setting (Table 3). The multistage variant has more learnable parameters that model the intermolecular geometries.

We also confirmed the findings from Moon et al.[7] that using the noncovalent interaction energy as an inductive bias for the deep learning model can significantly improve binding affinity prediction. Without the energy-inductive bias, the model can only achieve a 22.9% top 1 success rate, compared to 75.8% with the energy-inductive bias (Table 3). As the energy, E, was calculated for the entire protein−ligand complex, we hypothesized that the inductive bias is more informative to the model if we decompose it to $E = E^{(1)} + E^{(2)} + E^{\text{int}}$. That is, for a two-molecule complex, the noncovalent interaction energy can be

Table 3. Model Performances on the CASF-2016 Binding Affinity Regression Task

model	lower level GNN	model backbone	energy-inductive bias	docking top 1 success rate	ranking ρ
Ours	GIN	GVP	none	0.230 ± 0.005	0.653 ± 0.036
	GIN	MS-GVP	none	0.148 ± 0.019	0.566 ± 0.031
	GIN joint training	GVP	none	0.205 ± 0.009	0.715 ± 0.019
	GIN joint training	MS-GVP	none	0.200 ± 0.023	0.690 ± 0.009
	GIN	GVP	E	0.759 ± 0.020	0.719 ± 0.025
	GIN	MS-GVP	E	**0.802 ± 0.020**	0.658 ± 0.032
	GIN joint training	GVP	E	0.779 ± 0.021	**0.765 ± 0.023**
	GIN joint training	MS-GVP	E	0.788 ± 0.007	0.760 ± 0.015
3D GNN	none	GNN	E	0.299	0.604
PIGNet	none	GNN	E	0.774	0.672

decomposed as the sum of intramolecule energy ($E^{(1)}$, $E^{(2)}$) and intermolecule energy (E^{int}). However, our results on the PDBbind/CASF-2016 experiments do not support this hypothesis (Table 4). This is probably due to the inaccurate estimation of the protein's intramolecular energy, $E^{(1)}$, because we used only the binding pocket rather than the entire protein structure to construct the GoG to make the training memory efficient.

Table 4. Decomposing Energy Terms Do Not Improve the Model Performances on the CASF-2016 Binding Affinity Regression Task[a]

energy-inductive bias	docking top 1 success rate	ranking ρ
E	**0.827**	**0.722**
$\mathbf{W}([E^{(1)}, E^{(2)}, E^{\text{int}}])$	0.583	0.638
$\mathbf{W}(\mathit{BatchNorm}([E^{(1)}, E^{(2)}, E^{\text{int}}]))$	0.442	0.677
$\mathbf{W}(\mathit{concat}([E^{(1)}, E^{(2)}, E^{\text{int}}]))$	0.474	0.694
$\mathbf{W}(\mathit{BatchNorm}(\mathit{concat}([E^{(1)}, E^{(2)}, E^{\text{int}}])))$	0.378	0.670

[a] $E = [E^{\text{vdW}}, E^{\text{hbond}}, E^{\text{metal}}, E^{\text{hydrophobic}}]$.

Protein–Protein Interface Classification. Next, we evaluated that EGGNet on another common type of protein complex pose prediction task to distinguish whether a protein–protein interaction interface is physiological or an artifact of the crystallization process. This task can be useful to deduce solution-state quaternary states (dimers, trimers, etc.) and/or to identify novel functions of protein families that were originally deemed crystal artifacts.[40] In addition, models trained for this task could be fine-tuned for detection of binders and non/weak binders for protein interface design. We trained and validated our models on the ProtCID[30] dataset. We found that different variants of our models are able to significantly outperform CAMP,[41] which only takes the sequence information on the protein pairs (Table 5). We also found that the joint training of GNNs at different levels of the GoGs improves the predictive performance over freezing the lower level GIN by at least 5.8% in AUROC.

Next, we evaluated whether the GoG representation leveraged by EGGNet is comparable to other types of representations of protein–protein interfaces. We trained and evaluated EGGNet on the publicly available MANY/DC datasets for protein–protein interface classification (Table 6). We found that our model outperforms traditional ML approaches including PISA[42] and PRODIGY-crystal[43,44] by 8.8 and 1.9%, respectively. Both PISA and PRODIGY-crystal operate on tabular features computed from the interfaces. EGGNet is also competitive with DeepRank-GNN,[36] a recently developed GNN-based method. We noted that approaches using graph representations underperform DeepRank,[22] which represents the interfaces as 3D grids. It is worth pointing out that the featurization process and representation of EGGNet are generalizable to any molecular interaction interfaces, whereas competing methods including DeepRank[22] and DeepRank-GNN[36] rely on features specific to proteins such as precomputed position-specific scoring matrices.

Table 5. Binary Protein–Protein Interaction Prediction on ProtCID

model	lower level GNN	model backbone	AUROC	AUPR
Ours	GIN	GVP	0.673	0.858
	GIN	MS-GVP	0.671	0.857
	GIN joint training	GVP	0.734	0.875
	GIN joint training	MS-GVP	**0.745**	**0.890**
CAMP	none	CNN	0.522	0.800

The evaluation of interface classification is similar to the ranking setting of PDBbind/CASF-2016, where the model is tasked to classify different protein–protein interaction pairs rather than the same protein–protein pair with different poses. Therefore, we observed that the energy-inductive bias is not useful in improving the binary classification performance (Table 7).

Thanks to the uniform GoG representation of the protein complexes, our model architectures used for the protein–protein interface prediction and the protein–small-molecule binding affinity prediction are identical. We next evaluated if a model trained on the binding affinity prediction task can improve protein–protein interface prediction via transfer learning. However, our results show that transfer learning from the model trained to predict small-molecule binding affinity does not help with this task compared to a model trained from scratch (Table 8). We speculate that this is due to the distributions of the input GoGs, and the labels between these datasets are disparate.

■ CONCLUSIONS

In this work, we first developed a biologically inspired data structure, GoG, to represent the structures of molecules and molecular complexes. We construct GoGs by applying edge-cutting graph partition on the atomic graphs of molecules, which is inspired by the fact that macromolecules are composed of small-molecule residues connected by common covalent bonds such as peptide bonds for proteins and phosphodiester bonds for nucleic acids. Next, we developed EGGNet, an equivariant GNN to learn from the GoG representations of molecular complexes to predict their properties. The unifying representa-

Table 6. Protein Interface Classification Performances of Models Trained on the MANY Dataset and Tested on the DC Dataset[a]

model	interface representation	AUROC	AUPR	accuracy
PRODIGY-crystal	tabular			0.74
PISA	tabular			0.79
DeepRank	3D grid			**0.86**
DeepRank-GNN[1]	3D graph	0.865	**0.871**	0.820
EGGNet + GIN joint training	3D GoG	0.861	0.826	0.805
EGGNet + MolT5-small joint training	3D GoG	**0.869**	0.863	0.805

[a] Underlined accuracy scores are reported in ref 36.

Table 7. Energy-Inductive Bias Does Not Help with the ProtCID Task

energy-inductive bias	AUROC	AUPR
none	**0.681**	**0.875**
E	0.405	0.719

Table 8. Transferring Model Weights Learned from PDBBind to ProtCID[a]

model training	model init	AUROC	AUPR
trained with lr = 1 × 10^{-4}	random	0.736 ± 0.005	0.879 ± 0.008
no training	PDBBind	0.387 ± 0.014	0.680 ± 0.005
trained with lr = 1 × 10^{-4}	PDBBind	0.734 ± 0.004	0.881 ± 0.003

[a] Experiments were performed using GIN as the lower level GNN within the single-stage EGGNet architecture. Average and standard deviations are shown in the table from three independent runs with different random seeds. No significant differences (*t*-test *p*-value > 0.6) were observed between transfer learning and training from scratch.

tion allows EGGNet to perform both DTI and PPI prediction tasks, achieving competitive performances.

■ ASSOCIATED CONTENT

Data Availability Statement

The PDBBind/CASF-2016 data can be downloaded from https://zenodo.org/record/6047984. The MANY/DC datasets can be downloaded from the SBGrid data repository, https://data.sbgrid.org/dataset/843/. The source code for this study is available for research and noncommercial use at https://github.com/aws-samples/eggnet-equivariant-graph-of-graph-neural-network.

■ AUTHOR INFORMATION

Corresponding Authors

Zichen Wang − *Amazon Web Services, Amazon, Seattle, Washington 98109-5210, United States;* orcid.org/0000-0002-1415-1286; Email: zichewan@amazon.com

Huzefa Rangwala − *Amazon Web Services, Amazon, Seattle, Washington 98109-5210, United States;* Email: rhuzefa@amazon.com

Peter M. Clark − *Janssen Biotherapeutics, Janssen Pharmaceutical Companies of Johnson & Johnson, Titusville, New Jersey 08560-1504, United States;* Present Address: Computational Drug Design, Novo Nordisk; Email: pmvc@novonordisk.com

Authors

Ryan Brand − *Amazon Web Services, Amazon, Seattle, Washington 98109-5210, United States*

Jared Adolf-Bryfogle − *Janssen Biotherapeutics, Janssen Pharmaceutical Companies of Johnson & Johnson, Titusville, New Jersey 08560-1504, United States*

Jasleen Grewal − *Amazon Web Services, Amazon, Seattle, Washington 98109-5210, United States;* orcid.org/0000-0001-7213-5930

Yanjun Qi − *Amazon Web Services, Amazon, Seattle, Washington 98109-5210, United States*

Steven A. Combs − *Janssen Biotherapeutics, Janssen Pharmaceutical Companies of Johnson & Johnson, Titusville, New Jersey 08560-1504, United States*

Nataliya Golovach − *Janssen Biotherapeutics, Janssen Pharmaceutical Companies of Johnson & Johnson, Titusville, New Jersey 08560-1504, United States*

Rebecca Alford − *Janssen Biotherapeutics, Janssen Pharmaceutical Companies of Johnson & Johnson, Titusville, New Jersey 08560-1504, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c04889

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors thank Sandeep Somani from Janssen and Tong Wang and Nkechinyere Agu from AWS for insightful discussions.

■ REFERENCES

(1) Isert, C.; Atz, K.; Schneider, G. Structure-based drug design with geometric deep learning. *Curr. Opin. Struct. Biol.* **2023**, 79, 102548.

(2) D'Agostino, G.; Scala, A. *Networks of networks: the last frontier of complexity*; Springer, 2014; Vol. *340*.

(3) Ni, J.; Tong, H.; Fan, W.; Zhang, X. Inside the atoms: ranking on a network of networks. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014; pp 1356−1365.

(4) Harada, S.; Akita, H.; Tsubaki, M.; Baba, Y.; Takigawa, I.; Yamanishi, Y.; Kashima, H. Dual graph convolutional neural network for predicting chemical networks. *BMC Bioinf.* **2020**, *21*, 94.

(5) Wang, Y.; Zhao, Y.; Shah, N.; Derr, T. Imbalanced Graph Classification via Graph-of-Graph Neural Networks. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022; pp 2067−2076.

(6) Wang, H.; Lian, D.; Zhang, Y.; Qin, L.; Lin, X. GoGNN: graph of graphs neural network for predicting structured entity interactions. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021; pp 1317−1323.

(7) Moon, S.; Zhung, W.; Yang, S.; Lim, J.; Kim, W. Y. PIGNet: a physics-informed deep learning model toward generalized drug−target interaction predictions. *Chem. Sci.* **2022**, *13*, 3661−3673.

(8) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.;

Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583−589.

(9) Jiménez, J.; Skalic, M.; Martinez-Rosell, G.; De Fabritiis, G. K_{DEEP}: Protein−Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. *J. Chem. Inf. Model.* **2018**, *58*, 287−296.

(10) Cang, Z.; Mu, L.; Wei, G.-W. Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. *PLoS Comput. Biol.* **2018**, *14*, No. e1005929.

(11) Nguyen, D. D.; Cang, Z.; Wei, G.-W. A review of mathematical representations of biomolecular data. *Phys. Chem. Chem. Phys.* **2020**, *22*, 4343−4367.

(12) Meng, Z.; Xia, K. Persistent spectral−based machine learning (PerSpect ML) for protein-ligand binding affinity prediction. *Sci. Adv.* **2021**, *7*, No. eabc5329.

(13) Liu, X.; Feng, H.; Wu, J.; Xia, K. Dowker complex based machine learning (DCML) models for protein-ligand binding affinity prediction. *PLoS Comput. Biol.* **2022**, *18*, No. e1009943.

(14) Feinberg, E. N.; Sur, D.; Wu, Z.; Husic, B. E.; Mai, H.; Li, Y.; Sun, S.; Yang, J.; Ramsundar, B.; Pande, V. S. PotentialNet for molecular property prediction. *ACS Cent. Sci.* **2018**, *4*, 1520−1530.

(15) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural message passing for quantum chemistry. In *International conference on machine learning*, 2017; pp 1263−1272.

(16) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In *International Conference on Learning Representations*, 2018.

(17) Somnath, V. R.; Bunne, C.; Krause, A. Multi-scale representation learning on proteins. *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2021; Vol. *34*, pp 25244−25255.

(18) Alley, E. C.; Khimulya, G.; Biswas, S.; AlQuraishi, M.; Church, G. M. Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods* **2019**, *16*, 1315−1322.

(19) Bepler, T.; Berger, B. Learning protein sequence embeddings using information from structure. In *International Conference on Learning Representations*, 2018.

(20) Rives, A.; Meier, J.; Sercu, T.; Goyal, S.; Lin, Z.; Liu, J.; Guo, D.; Ott, M.; Zitnick, C. L.; Ma, J.; et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. U.S.A.* **2021**, *118*, No. e2016239118.

(21) Elnaggar, A.; Heinzinger, M.; Dallago, C.; Rehawi, G.; Wang, Y.; Jones, L.; Gibbs, T.; Feher, T.; Angerer, C.; Steinegger, M.; Bhowmik, D.; Rost, B. ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 7112−7127.

(22) Renaud, N.; Geng, C.; Georgievska, S.; Ambrosetti, F.; Ridder, L.; Marzella, D. F.; Réau, M. F.; Bonvin, A. M.; Xue, L. C. DeepRank: a deep learning framework for data mining 3D protein-protein interfaces. *Nat. Commun.* **2021**, *12*, 7068.

(23) Mohseni Behbahani, Y.; Crouzet, S.; Laine, E.; Carbone, A. Deep Local Analysis evaluates protein docking conformations with locally oriented cubes. *Bioinformatics* **2022**, *38*, 4505−4512.

(24) Ingraham, J.; Garg, V.; Barzilay, R.; Jaakkola, T. Generative models for graph-based protein design. *Advances in neural information processing systems*; Curran Associates, Inc., 2019; Vol. 32.

(25) Jing, B.; Eismann, S.; Suriana, P.; Townshend, R. J. L.; Dror, R. Learning from Protein Structure with Geometric Vector Perceptrons. In *International Conference on Learning Representations*, 2021.

(26) Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754.

(27) Edwards, C.; Lai, T.; Ros, K.; Honke, G.; Cho, K.; Ji, H. Translation between Molecules and Natural Language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Abu Dhabi, United Arab Emirates*, 2022; pp 375−413.

(28) Wang, Z.; Combs, S. A.; Brand, R.; Calvo, M. R.; Xu, P.; Price, G.; Golovach, N.; Salawu, E. O.; Wise, C. J.; Ponnapalli, S. P.; et al. Lm-gvp: an extensible sequence and structure informed deep learning framework for protein property prediction. *Sci. Rep.* **2022**, *12*, 6832.

(29) Wang, R.; Fang, X.; Lu, Y.; Wang, S. The PDBbind database: Collection of binding affinities for protein- ligand complexes with known three-dimensional structures. *J. Med. Chem.* **2004**, *47*, 2977−2980.

(30) Xu, Q.; Dunbrack, R. L. ProtCID: a data resource for structural information on protein interactions. *Nat. Commun.* **2020**, *11*, 711.

(31) Baskaran, K.; Duarte, J. M.; Biyani, N.; Bliven, S.; Capitani, G. A PDB-wide, evolution-based assessment of protein-protein interfaces. *BMC Struct. Biol.* **2014**, *14*, 22.

(32) Duarte, J. M.; Srebniak, A.; Schärer, M. A.; Capitani, G. Protein interface classification by evolutionary analysis. *BMC Bioinf.* **2012**, *13*, 334.

(33) Su, M.; Yang, Q.; Du, Y.; Feng, G.; Liu, Z.; Li, Y.; Wang, R. Comparative assessment of scoring functions: the CASF-2016 update. *J. Chem. Inf. Model.* **2019**, *59*, 895−913.

(34) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The protein data bank. *Nucleic Acids Res.* **2000**, *28*, 235−242.

(35) Steinegger, M.; Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* **2017**, *35*, 1026−1028.

(36) Réau, M.; Renaud, N.; Xue, L. C.; Bonvin, A. M. DeepRank-GNN: a graph neural network framework to learn patterns in protein−protein interfaces. *Bioinformatics* **2023**, *39*, btac759.

(37) Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

(38) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*; Curran Associates, Inc., 2019; Vol. 32.

(39) Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*, 2018.

(40) Xu, Q.; Malecka, K. L.; Fink, L.; Jordan, E. J.; Duffy, E.; Kolander, S.; Peterson, J. R.; Dunbrack, R. L., Jr. Identifying three-dimensional structures of autophosphorylation complexes in crystals of protein kinases. *Sci. Signaling* **2015**, *8*, rs13.

(41) Lei, Y.; Li, S.; Liu, Z.; Wan, F.; Tian, T.; Li, S.; Zhao, D.; Zeng, J. A deep-learning framework for multi-level peptide−protein interaction prediction. *Nat. Commun.* **2021**, *12*, 5465.

(42) Krissinel, E.; Henrick, K. Inference of macromolecular assemblies from crystalline state. *J. Mol. Biol.* **2007**, *372*, 774−797.

(43) Jiménez-García, B.; Elez, K.; Koukos, P. I.; Bonvin, A. M.; Vangone, A. PRODIGY-crystal: a web-tool for classification of biological interfaces in protein complexes. *Bioinformatics* **2019**, *35*, 4821−4823.

(44) Elez, K.; Bonvin, A. M.; Vangone, A. Distinguishing crystallographic from biological interfaces in protein complexes: role of intermolecular contacts and energetics for classification. *BMC Bioinf.* **2018**, *19*, 438.