JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

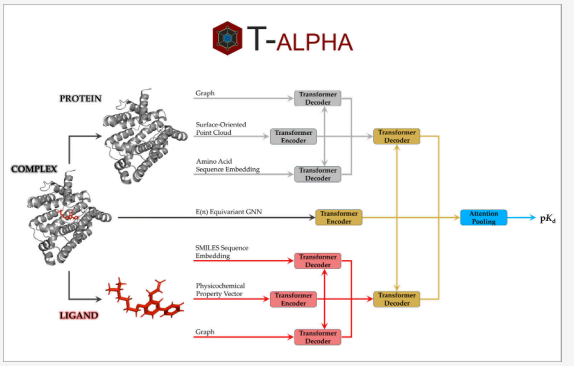
T-ALPHA: A Hierarchical Transformer-Based Deep Neural Network for Protein−Ligand Binding Affinity Prediction with Uncertainty-Aware Self-Learning for Protein-Specific Alignment

Gregory W. Kyro,* Anthony M. Smaldone, Yu Shee, Chuzhi Xu, and Victor S. Batista*

Cite This: https://doi.org/10.1021/acs.jcim.4c02332 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: There is significant interest in targeting disease-causing proteins with small molecule inhibitors to restore healthy cellular states. The ability to accurately predict the binding affinity of small molecules to a protein target in silico enables the rapid identification of candidate inhibitors and facilitates the optimization of on-target potency. In this work, we present T-ALPHA, a novel deep learning model that enhances protein−ligand binding affinity prediction by integrating multimodal feature representations within a hierarchical transformer framework to capture information critical to accurately predicting binding affinity. T-ALPHA outperforms all existing models reported in the literature on multiple benchmarks designed to evaluate protein−ligand binding affinity scoring functions. Remarkably, T-ALPHA maintains state-of-the-art performance when utilizing predicted structures rather than crystal structures, a powerful capability in real-world drug discovery applications where experimentally determined structures are often unavailable or incomplete. Additionally, we present an uncertainty-aware self-learning method for protein-specific alignment that does not require additional experimental data and demonstrate that it improves T-ALPHA's ability to rank compounds by binding affinity to biologically significant targets such as the SARS-CoV-2 main protease and the epidermal growth factor receptor. To facilitate implementation of T-ALPHA and reproducibility of all results presented in this paper, we made all of our software available at https://github.com/gregory-kyro/T-ALPHA.



1. INTRODUCTION

There is growing scientific and societal interest in developing therapeutic interventions targeting diseases that disproportionately affect humans and remain inadequately addressed by current medical treatments.[1] Protein dysregulation, including overexpression and aberrant post-translational modifications, is a fundamental factor in the pathogenesis of numerous human diseases. For instance, in neurodegenerative disorders like Alzheimer's and Parkinson's diseases, abnormal protein modifications can lead to aggregation and neuronal death.[2] Similarly, in cancer, the overexpression of certain proteins disrupts cellular homeostasis, contributing to uncontrolled cell proliferation.[3] Targeting these dysregulated proteins with small molecules (i.e., ligands) offers a promising therapeutic strategy to restore normal cellular function and transition cells from a disease state to a healthy state.[4]

A typical pipeline to develop therapeutic compounds consists of several sequential stages: target identification (selecting a biological target linked to disease),[5] target validation (confirming the target's role in disease progression and suitability for therapeutic intervention),[6] hit identification (screening for compounds with initial activity against the target),[7] lead optimization (refining compounds for potency, selectivity, and desirable pharmacokinetics),[8] preclinical testing (evaluating safety and efficacy in nonhuman models),[9] and clinical development (testing for safety and efficacy in human trials).[10] In this work, we are focused on the hit identification and lead optimization stages, specifically, predicting protein−ligand binding affinity.

Machine learning (ML) has been widely applied to protein−ligand binding affinity prediction and has become central to computer-aided drug design more broadly, with applications including protein structure prediction,[11,12] molecular docking,[13] small molecule property prediction,[14,15] and others.[16] Traditional ML approaches for protein−ligand binding affinity prediction, such as random forests[17−20] and shallow neural networks,[21] are increasingly being replaced by deep learning methods that are better suited for learning geometric representations of molecular structures.[22] Examples include

Received: December 12, 2024
Revised: January 30, 2025
Accepted: February 7, 2025

convolutional neural networks (CNNs), which utilize convolutional operations to capture local spatial features from voxelized grids,[23−35] graph neural networks (GNNs), which employ message passing to model relational information in molecular graphs,[33,35−58] and, more recently, transformers, which utilize self- and cross-attention to model long-range dependencies within and between embeddings, respectively.[59−82] Moreover, it has been demonstrated that transformer-based multimodal feature representation learning of proteins is effective for extracting features from the protein that are important for predicting protein−ligand binding affinity,[73] a finding that has inspired multiple components of our work.

We present T-ALPHA, a novel deep learning model for protein−ligand binding affinity prediction designed to comprehensively integrate multimodal data representations and leverage hierarchical transformer mechanisms to capture the intricate physicochemical, structural, and spatial dynamics governing protein−ligand binding interactions. T-ALPHA processes input data through three distinct channels, corresponding to the protein, ligand, and protein−ligand complex. Each channel independently learns a rich and optimized feature representation for its specific component, ensuring that the critical properties of proteins, ligands, and their interactions are captured.

Within the protein and ligand channels, cross-attention mechanisms are employed to integrate complementary information between diverse feature representations. The protein channel combines features derived from the protein's surface topography and curvature using point cloud-based quasi-geodesic convolutions, connectivity-based structural information from an E(n) equivariant graph neural network (EGNN), and sequence-derived evolutionary and structural embeddings derived from a pretrained transformer-based model. Similarly, the ligand channel integrates molecular-level physicochemical descriptors, graph-based structural features from an E(n) EGNN, and relational information extracted from SMILES strings using a pretrained transformer encoder. The protein−ligand complex channel focuses exclusively on modeling the interactions between the protein and ligand through an E(n) EGNN, capturing spatial relationships and interaction-specific features that are essential for binding affinity prediction. After processing through these channels, an additional layer of cross-attention integrates the outputs from the protein, ligand, and protein−ligand complex channels, enabling the model to combine their complementary perspectives into a unified, hierarchical representation. This design ensures that T-ALPHA effectively models the complex dependencies between proteins and ligands, resulting in state-of-the-art performance across multiple benchmarks.

Typically, deep learning models for protein−ligand binding affinity prediction are trained and benchmarked on datasets containing many different proteins to assess generalization across diverse targets. While this is valuable, in most practical applications, the focus is on a single, disease-relevant protein where high accuracy for that specific target is paramount. Current methods for target-specific alignment, such as active learning-based approaches,[83] require acquisition of additional experimental data which can be resource-intensive and slow. In this work, we introduce a novel uncertainty-aware self-learning method that enables protein-specific alignment without the need for new experimental data. Applied to two distinct protein targets, namely, SARS-CoV-2 main protease (Mpro) and the epidermal growth factor receptor (EGFR), our approach improves the model's target-specific ranking of compounds by binding affinity, offering a resource-efficient strategy for real-world applications that require high accuracy on defined protein targets.

The ability of T-ALPHA to effectively model protein−ligand interactions, combined with the proposed self-learning method for protein-specific alignment, underscores the utility of this work for both broad and target-focused applications in protein−ligand binding affinity prediction.

2. DATA

2.1. Data Curation. *2.1.1. Protein−Ligand Binding Affinity Data.* T-ALPHA is trained on experimentally determined protein−ligand complex structures deposited in the Protein Data Bank, each labeled with a binding affinity measurement. The PDBbind database,[84−86] widely regarded as the primary data source for evaluating deep learning models for protein−ligand binding affinity prediction, contains 19443 protein−ligand complex structures in its latest update (v2020), the majority of which are X-ray crystal structures, with a smaller portion determined via Nuclear Magnetic Resonance (NMR) spectroscopy. Each binding affinity value is reported as an inhibition constant (K_i), dissociation constant (K_d), or half-maximal inhibitory concentration (IC_{50}).

Because the biochemical assays performed to obtain these measurements operate within specific dynamic ranges, some measurements are reported with inequalities (“>”, “<”) rather than exact (“=”) or approximate (“∼”) values. While most protein−ligand binding affinity prediction work treats all labels as exact values (i.e., assuming the operator to be “=” for each data point), we consider the original operators via a custom loss function that we constructed to more accurately penalize model predictions during training and validation (for more details, see Section 4.1).

The complete v2020 release of the PDBbind database, referred to as the *general set*, is filtered into a *refined set* (5316 entries),[87] which comprises relatively high-quality crystal structures with relatively reliable binding affinity values. The refined set is further filtered into the *core set* (290 entries), representing the highest quality data points, with 58 proteins, each co-crystallized with five different ligands. Moreover, each binding affinity measurement is reported as an exact value (“=”). The core set serves as the basis for the Comparative Assessment of Scoring Functions (CASF) 2016 benchmark,[88] which is the most widely used benchmark for assessing protein−ligand binding affinity scoring functions.

While the CASF 2016 benchmark is widely regarded as the gold standard of the field, it is unable to adequately assess the ability of models to generalize to protein−ligand complexes with low similarity to those found in the rest of the PDBbind database (i.e., the corresponding training and validation data).[89] Specifically, the majority of the data points in the core set contain either identical proteins or chemically similar ligands with entries in the rest of the PDBbind v2020 general set, introducing data leakage that inflates performance estimates on the CASF 2016 benchmark with respect to generalizability to low-similarity protein−ligand complexes. In this work, we utilize the CASF 2016 test set exclusively to compare the T-ALPHA architecture to the many models reported in the literature that also benchmark on this dataset.

To address some of the limitations of the CASF 2016 benchmark for assessing protein−ligand binding affinity scoring functions, Leak Proof PDBbind (LP-PDBbind) was

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

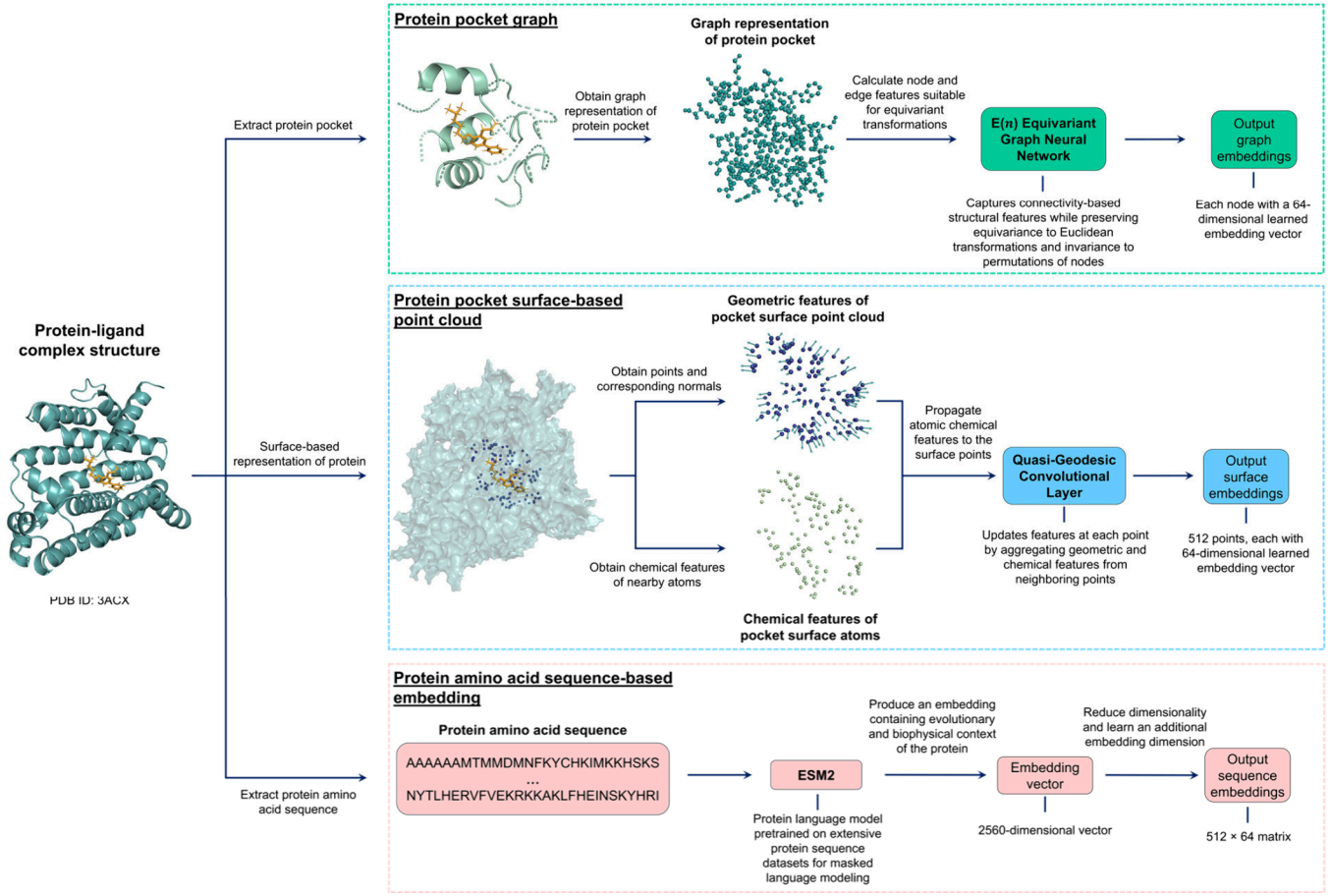


Figure 1. Protein channel of the T-ALPHA pipeline. The protein is represented and processed in three distinct ways: (1) a graph representation of the protein pocket (top, green box), where nodes and edges are annotated with atomic and chemical features, is processed by an E(n) Equivariant Graph Neural Network to generate graph embeddings; (2) a surface-oriented point cloud (middle, blue box) obtained from the entire protein is processed via a quasi-geodesic convolution layer to capture the geometric and chemical properties of the protein's surface; and (3) the protein amino acid sequence (bottom, pink box) is processed by ESM2 to produce sequence-based embeddings that contain evolutionary and functional information about the protein. Together, these three featurization approaches encode complementary information for integration in T-ALPHA, enabling comprehensive modeling of the protein for downstream protein−ligand binding affinity prediction. The protein−ligand complex structure shown corresponds to PDB ID 3ACX.[112]

developed to minimize protein sequence and ligand structure similarities across training, validation, and test sets, while also ensuring distinct protein−ligand structural interaction patterns across these sets.[90] Ligand similarity was calculated as the Dice similarity[91] between 1024-bit Morgan fingerprints, and protein similarity was calculated as the sequence identity between Needleman−Wunsch-aligned[92] amino acid sequences. The splitting approach utilized ensures that no training data points have a protein similarity or ligand similarity greater than 0.5 or 0.99, respectively, to any data point in the validation or test sets, and that no validation data points have a protein similarity or ligand similarity greater than 0.9 or 0.99, respectively, to any data point in the test set. Additionally, proteo-chemometric interaction fingerprints[93] were used to evaluate interaction patterns. These fingerprints extend the Morgan fingerprint by incorporating the spatial interactions between ligand atoms and adjacent protein residues, mapping interaction patterns to a fixed-size integer vector with a length of 256. Pairwise interaction fingerprint similarities were then calculated using a weighted Tanimoto similarity score.[94] The data splitting protocol ensures distinct separation of training, validation, and test data in terms of these interaction fingerprints.

In addition to the LP-PDBbind split, the same research group developed a test set for benchmarking protein−ligand binding affinity scoring functions that is independent of the data contained in the PDBbind database, referred to as the BDB2020+ dataset. This dataset consists of data points from the BindingDB database[95] that were deposited after 2020, reducing the potential for data leakage with v2020 of the PDBbind. While BindingDB provides experimental binding affinity data for protein−ligand complexes, it does not include the corresponding experimentally determined structures. To address this, binding affinity data from BindingDB was cross-referenced with matching experimental structures in the RCSB Protein Data Bank.[96]

Additionally, two protein-specific test sets were developed by the same research group and employed in this work that focus on the SARS-CoV-2 main protease (Mpro)[97] and the epidermal growth factor receptor (EGFR),[98] a receptor tyrosine kinase implicated in multiple types of cancer. The Mpro dataset contains 40 data points, each with a unique ligand co-crystallized with Mpro, while the EGFR dataset comprises 23 data points, each with a unique ligand co-crystallized with EGFR; all data points are labeled with experimentally determined binding affinity measurements. For

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

full details on the preparation and validation of the LP-PDBbind splits, the BDB2020+ test set, as well as the Mpro and EGFR test sets, please refer to the original paper.[90]

2.1.2. Ligand SMILES Transformer Pretraining Data. In T-ALPHA, one of the ligand representations is a feature vector that is extracted from a transformer encoder pretrained for masked token prediction on a dataset of SMILES strings. For pretraining, we utilize a comprehensive dataset that we have previously curated,[99] which combines all of the SMILES strings from ChEMBL 33 (∼2.4 million bioactive molecules with drug-like properties),[100] GuacaMol v1 (∼1.6 million molecules derived from ChEMBL 24 that have been synthesized and tested against biological targets),[101] MOSES (∼1.8 million molecules selected from ZINC 15 to maximize internal diversity and suitability for medicinal chemistry),[102] BindingDB (∼1.2 million unique small molecules bound to proteins),[95] and the v2020 release of the PDBbind general set (15710 unique small molecules bound to proteins).[84−86]

2.1.3. Predicted Protein−Ligand Complex Structures. For each of the test sets employed in this work, we use Chai-1, a state-of-the-art multimodal foundation model for molecular structure prediction, to predict the protein−ligand complex 3D structure from each protein amino acid sequence and ligand SMILES string pair. In the context of this work, we utilize predicted structures to assess the robustness of T-ALPHA in scenarios where experimentally determined structures are unavailable or incomplete. We provide full implementation details of Chai-1 in Section 9.

2.2. Data Preparation. *2.2.1. Protein−Ligand Complex Structures.* We preprocessed all protein−ligand complex structures, including both the experimentally determined and predicted structures. Solvent molecules were removed using Biopython[103] to allow the model to implicitly learn solvent effects. Each protein was corrected for missing heavy atoms and residues using PDBFixer and OpenMM.[104] We then added missing hydrogen atoms to the proteins and ligands with Open Babel.[105] For each data point, we extracted the protein pocket by taking any residue in the protein that contains at least one atom that is within 8 Å of at least one ligand nonhydrogen atom.

2.2.2. Ligand SMILES Strings. To prepare the pretraining dataset for the SMILES-based transformer encoder that we use as a ligand feature extractor, we processed each SMILES string using RDKit[106] by creating a mol object and canonicalizing. After removing duplicate canonical SMILES and those that failed processing, 5131118 valid and unique SMILES strings remained out of the original 5791565 entries. Tokenization resulted in a vocabulary of 379 unique tokens, of which only 132 occur in the PDBbind v2020 general set. We removed any SMILES string that contains at least one token that does not occur in any data point in the PDBbind v2020 general set, resulting in 4810575 remaining SMILES strings and a significant reduction of the vocabulary. In order to significantly increase computational efficiency, we removed any SMILES string with more than the 95th percentile number of tokens, reducing the block size from 385 to 155 tokens. This entire preparation yielded a pretraining dataset of 4778512 data points.

2.3. Data Featurization. *2.3.1. Protein Featurization.* In T-ALPHA, there are three channels of data processing, corresponding to the protein, ligand, and protein−ligand complex. For the protein channel, we represent and process the protein in three distinct ways: (1) a sparse graph derived from the protein pocket which is processed by an E(n) EGNN to learn connectivity-based structural features, (2) a point cloud of the surface of the protein pocket which is processed via quasi-geodesic convolutions to learn features pertaining to the surface topography and curvature, and (3) an amino acid sequence-derived embedding which is processed by a multilayer perceptron (MLP) to learn complementary global evolutionary and structural information (Figure 1).

2.3.1.1. Protein Pocket Graph. For the protein pocket graph representation, each nonhydrogen atom of the protein pocket is represented as a node, and each covalent bond between these atoms is represented as an edge. The nodes and edges of this graph are richly annotated with features that capture both atomic properties and chemical interactions that are suitable for equivariant transformations.

The node features are described in Table 1, and the edge features are described in Table 2.

Table 1. Node Features Used for the Protein Pocket Graph Representation[a]

node feature	description	data type
atom type	C, O, N, S, F, P, Cl, Br, B, I, or other	one-hot encoding
amino acid type	hydrophobic, polar, basic, or acidic	one-hot encoding
hydrophobic indicator	indicates if atom is hydrophobic	binary
aromaticity indicator	indicates if atom is part of an aromatic ring	binary
hydrogen bond acceptor indicator	indicates if atom is a H-bond acceptor	binary
hydrogen bond donor indicator	indicates if atom is a H-bond donor	binary
ring membership indicator	indicates if atom is part of a ring	binary
chirality indicator	indicates whether atom is a chiral center	binary
formal charge	formal charge value	integer
hybridization state	sp^2, sp^3, etc.	integer
total degree	number of bonded neighbors	integer
heavy atom degree	number of bonded heavy atoms	integer
heteroatom degree	number of bonded heteroatoms	integer
hydrogen degree	number of bonded hydrogen atoms	integer
van der Waals radius	as reported by Los Alamos National Lab	float
partial charge	partial atomic charge	float
electronegativity	based on the Pauling scale	float
static dipole polarizability	for neutral atoms	float

[a]van der Waals radius values were obtained from Los Alamos National Lab.[107] Partial charge values were calculated using Open Babel's GetPartialCharge() method,[105] which assigns partial charges based on the Gasteiger method.[108] Electronegativity values were retrieved from PubChems' periodic table resource.[109] Static dipole polarizability values for neutral atoms were obtained from The New Zealand Institute for Advanced Study and the Institute for Natural and Mathematical Sciences.[110]

2.3.1.2. Protein Pocket Surface-Based Point Cloud. In order to capture the detailed geometry and chemical properties of the protein−ligand interface, T-ALPHA employs an approach largely based on the dMaSIF[111] method for describing the curvature of the protein pocket. This approach involves converting atomic-level information into an oriented point cloud that represents the surface of the protein pocket and then processing this point cloud via quasi-geodesic

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 2. Edge Features Used for the Protein Pocket Graph Representation[a]

edge feature	description	data type
bond order	interatomic bond order	integer
aromaticity indicator	indicates if bond is part of an aromatic ring	binary
ring membership indicator	indicates if bond is part of a ring	binary
interatomic distance	measured in angstroms (Å)	float
electronegativity difference	absolute difference in electronegativity between the two bonded atoms (based on the Pauling scale)	float
electrostatic interaction energy	coulombic interatomic electrostatic interaction energy	float

[a]Electrostatic interaction energy is calculated as $q_i q_j / r_{ij}^2$, where q_k denotes the partial charge of atom k and r_{kl} represents the interatomic distance (Å) between atoms k and l.

convolutions. In our implementation, the point cloud is obtained from an disconnected atomic graph created identically to that described in Section 2.3.1.1 but with a few exceptions: (1) it is derived from the entire protein rather than just the protein pocket, (2) it also considers hydrogen atoms as nodes in addition to nonhydrogen atoms, and (3) it does not contain any edges.

This unconnected graph representation is processed through a multistep pipeline designed to accurately capture the relevant protein pocket surface geometry and chemical characteristics for each protein−ligand complex. First, random points are placed around each atom of the protein, and their positions are iteratively refined using a soft distance metric which evaluates how close each sampled point is to the actual surface boundary, taking into account both the atomic radii and local geometric constraints. Through gradient-based optimization, these points converge to positions where the distance metric aligns with a defined threshold representing the protein's surface. This results in a dense collection of points that accurately reflects the topography of the protein surface. Once these initial points have converged, a cubic grid clustering method is applied to produce a more uniform representation.[111] Next, surface normals are computed for each remaining point, where the normal vectors are derived from the gradient of the soft distance function used to position the points on the protein surface. Each normal vector therefore indicates the outward-facing direction of the surface at the respective point, capturing orientation-based information for downstream transformations (Section 3.1.2).

2.3.1.3. Protein Amino Acid Sequence-Based Embedding. In addition to the information derived from the protein's 3D structure, T-ALPHA also integrates embeddings obtained from ESM2 — a large-scale transformer model pretrained on extensive protein sequence databases.[113] ESM2 processes each amino acid sequence using self-attention mechanisms to capture complex relationships and dependencies between amino acid residues, thereby encoding evolutionary information, structural context, and functional insights from the sequence. The output is a high-dimensional embedding vector that encapsulates relevant sequence-based information such as residue conservation, proximity to active sites, secondary structure elements, and overall fold constraints, rooted in the extensive evolutionary and structural patterns learned by ESM2 during its large-scale pretraining. By leveraging ESM2, we can extract complementary features that are not directly inferable from the protein pocket graph or the protein pocket surface-based point cloud. For example, certain amino acid residues might be critical for function and highly conserved evolutionarily, which ESM2 can highlight even if these residues do not have immediately obvious structural patterns.

2.3.2. Ligand Featurization. For the ligand channel in T-ALPHA, we employ a featurization strategy analogous to that used for the protein channel, capturing multiple facets of the ligand's chemical and structural properties that we later demonstrate are important for predicting protein−ligand binding affinity. Specifically, we represent and process the ligand in three distinct ways: (1) a sparse graph which is processed by an E(n) EGNN to learn connectivity-based structural features, (2) a molecular-level descriptor vector which is processed by an MLP to encapsulate global physicochemical properties, and (3) a SMILES sequence-based embedding which is further refined by an MLP to capture local and global chemical context such as atom-level interactions, substructures, and stereochemical details—information that is not inferable from the graph or global descriptor representations. This multifaceted approach ensures a comprehensive representation of the ligand for downstream tasks.

2.3.2.1. Ligand Graph. The ligand is represented as a graph where each nonhydrogen atom is a node and each covalent bond between these atoms is an edge. The node and edge features are calculated similarly to those in the protein pocket graph (Section 2.3.1.1), with the only distinction being that amino acid type is not one of the node features.

2.3.2.2. Ligand Physicochemical Property-Based Embedding. In addition to the graph representation, we compute a molecular-level descriptor vector for each ligand using RDKit.[106] This descriptor vector encapsulates a range of physicochemical and topological properties that provide a comprehensive overview of the ligand's chemical characteristics; all of the calculated descriptors are listed in Table S1 of the Supporting Information. These descriptors include information about molecular size, polarity, flexibility, electronic properties, and other relevant features that influence binding affinity. By characterizing the ligand's global properties, this descriptor vector complements the detailed local structural information captured by the graph representation, ensuring that both global and local features contribute to the overall ligand representation.

2.3.2.3. Ligand SMILES Sequence-Based Embedding. As an additional feature representation of the ligand, we utilize a transformer encoder pretrained on a large dataset of SMILES strings to extract a context-rich embedding that contains information about complex chemical patterns and substructures. This pretrained model utilizes self-attention mechanisms to capture both local chemical contexts such as functional groups and ring systems, as well as global structural information.

In this approach, the ligand SMILES string is first tokenized into individual symbols representing atoms, bonds, and structural features. These tokens include elements like C, N, and O; bond types denoted by symbols such as "=", and "#"; branching symbols like "(", and ")"; and stereochemical indicators such as "@", "/", and "\". Each token is mapped to a learnable high-dimensional embedding vector, and each positional index in the input is encoded into a learnable

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

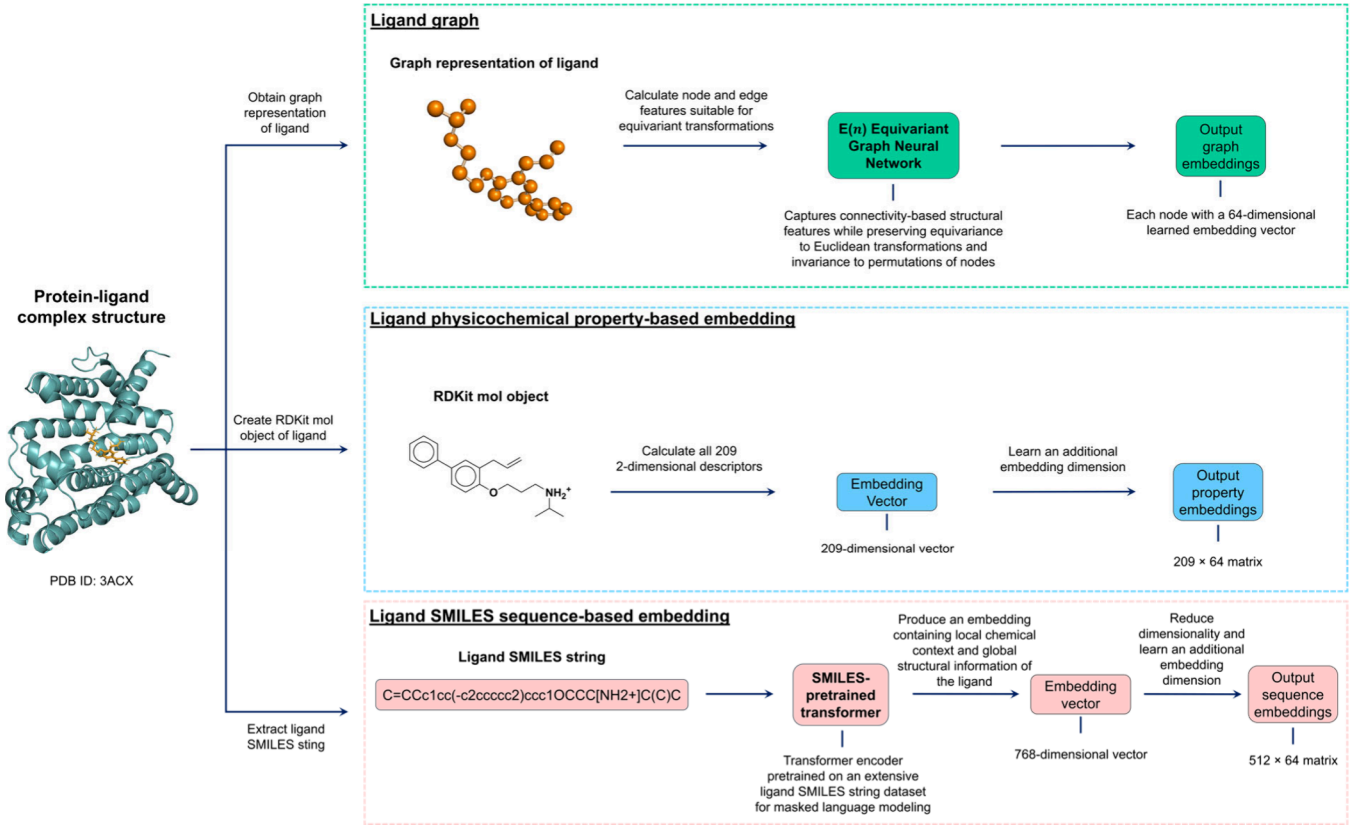


Figure 2. Ligand channel of the T-ALPHA pipeline. The ligand is represented and processed in three distinct ways: (1) a graph representation (top, green box), where nodes and edges are annotated with atomic and chemical features, is processed by an E(n) Equivariant Graph Neural Network to generate graph embeddings; (2) physicochemical property-based embeddings (middle, blue box) capture molecular-level features that contribute to binding affinity such as size, polarity, and flexibility; and (3) the ligand SMILES string (bottom, pink box) is processed by a pretrained transformer to obtain embeddings that capture local substructures, stereochemistry, and long-range dependencies. Together, these three featurization approaches encode complementary information for integration in T-ALPHA, enabling comprehensive modeling of the ligand for downstream protein−ligand binding affinity prediction. The protein−ligand complex structure shown corresponds to PDB ID 3ACX.[112]

high-dimensional vector. Rather than using fixed sinusoidal encodings, we employ learned positional embeddings, where each position is assigned a trainable embedding vector. This strategy allows the model to learn an optimal representation of positional dependencies directly from data rather than imposing a handcrafted inductive bias. The token embeddings capture semantic information about the chemical symbols, while the positional encodings provide the transformer with information about the order of the tokens in the sequence. This is critical because the self-attention mechanism in transformers is inherently permutation-invariant and does not consider token order unless explicitly encoded. The final input embeddings are obtained by element-wise summing the token embeddings and positional encodings, producing a combined representation for each token that integrates both semantic and positional information.

These combined embeddings are then passed through a series of transformer encoder blocks. Each block consists of a multi-head self-attention sublayer followed by a residual connection and layer normalization, and a position-wise feed-forward network sublayer which is also followed by a residual connection and layer normalization. The self-attention mechanism allows the model to compute a weighted representation of all tokens in the sequence relative to a given token, capturing intricate patterns and dependencies within the molecule. For each token, self-attention computes three vectors: a query vector, a key vector, and a value vector, all derived from learned linear transformations of the input embedding. The attention score between any two tokens is calculated as the scaled dot product of the query vector of one token with the key vector of another, normalized using a softmax function to ensure that the scores sum to one. These scores are then used to compute a weighted sum of the value vectors, resulting in an output embedding that reflects the token's context in the sequence. By attending to all tokens in the sequence, the model effectively captures both local interactions, such as adjacent atoms in a functional group, and long-range dependencies, such as conjugated systems or distant functional groups that influence each other. In the multi-head self-attention mechanism, multiple attention heads process the embeddings in parallel, with each head learning to focus on different types of relationships among tokens. This parallel processing enhances the model's ability to capture diverse aspects of chemical interactions within the molecule.

Following the multi-head self-attention sublayer, each transformer block applies a residual connection and layer normalization to the output of the attention mechanism. Next, a position-wise feed-forward sublayer is applied to each token individually. This feed-forward sublayer consists of two linear transformations separated by a Gaussian Error Linear Unit (GELU) activation function. The feed-forward sublayer is followed by another residual connection and layer normal-

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

ization. This setup enables the effective modeling of intricate patterns and long-range dependencies across sequences, with residual connections promoting stable gradient flow and layer normalization ensuring numerical stability to facilitate efficient training of deep architectures. The transformer that we use contains 10 sequential transformer blocks.

After the input SMILES string is processed by the transformer blocks, we obtain contextualized embeddings for each token in the sequence, where each embedding now contains information about the token itself and its relationships with other tokens in the sequence. In our implementation, we extract the embedding corresponding to the start token added at the beginning of the sequence. This start token aggregates information from all positions in the sequence during the self-attention computations, effectively capturing a global representation of the molecule.

For each data point, the start token embedding extracted from the pretrained transformer is passed through an MLP to optimize it for predicting protein−ligand binding affinity. This step allows the model to refine the embedding, capturing subtle chemical features and intricate patterns that might not be inferable from the ligand's graph representation or molecular-level descriptors (Figure 2).

2.3.3. Protein−Ligand Complex Featurization. In T-ALPHA, the protein−ligand complex is represented as a unified graph that integrates both the protein pocket and ligand into a single structure designed to capture the intricate interactions between the two molecules that are crucial for accurate binding affinity prediction.

The nodes in this graph represent nonhydrogen atoms from both the protein and the ligand. The node features are identical to those previously described for the protein (Section 2.3.1.1) and ligand (Section 2.3.2.1) graphs, with the addition of a source identifier that indicates whether a given node belongs to the protein or ligand, enabling the model to learn source-specific patterns and interactions.

Edges in the graph represent both intramolecular and intermolecular interactions. Intramolecular edges are established based on covalent bonds within the protein and within the ligand, as defined in their respective individual graphs. These edges capture the internal connectivity of each molecule. Intermolecular edges are introduced between protein and ligand nonhydrogen atoms that are within 4.5 Å to capture potential noncovalent interactions critical for binding, such as hydrogen bonds, hydrophobic contacts, $\pi-\pi$ stacking, and electrostatic interactions. We chose 4.5 Å as the distance threshold because it is commonly used as a hydrophobic distance threshold in protein−ligand interaction modeling.[114] Edge features are identical to those used for the protein and ligand graphs, with an additional interaction feature to indicate whether a given edge represents an intramolecular connection or an intermolecular interaction. This labeling enables the model to distinguish between bonds that define molecular structures and interactions that contribute to binding affinity that are not apparent when considering the protein and ligand separately. By integrating the protein and ligand into a single graph with enriched node and edge features, T-ALPHA is able to consider both the structural details of the individual molecules and the critical interactions between them.

2.3.4. Data Scaling. To ensure that our model effectively learns from features of varying scales and to enhance training stability, we apply standardization to all continuous features across our dataset. Specifically, we transform these features to have a mean of zero and a standard deviation of one. This preprocessing step is crucial for preventing any single feature from disproportionately influencing the learning process due to differences in magnitude. Additionally, given that our architecture integrates multiple components, some of which do not include built-in normalization mechanisms, scaling ensures numerical consistency across all submodules. Empirically, we observe that presacaling stabilizes training, mitigates gradient-related instabilities, and contributes to improved model performance.

Across our graph representations (i.e., those for the protein pocket, the entire protein used in the dMaSIF-based module, the ligand, and the protein−ligand complex), we standardize continuous node features including van der Waals radius, partial charge, electronegativity, and polarizability. For the edge features, we standardize interatomic distance, electronegativity difference, and Coulombic electrostatic interaction energy. In addition to the graph-based features, we standardize the protein amino acid sequence-based embeddings, ligand physicochemical property-based embeddings, and ligand SMILES sequence-based embeddings.

2.3.5. Training and Validation Set Construction. For each training−validation−test split, the test set is predefined and excluded entirely from the training and validation data. To partition the remaining data, we used a structured and reproducible approach. For CASF 2016, 95% of the remaining PDBbind general set is randomly selected for training, while the remaining 5% is used for validation. For LP-PDBbind, the predefined splits provided by the authors are used directly to ensure consistency with previous studies. For the BDB2020+, Mpro, and EGFR test sets, a single model is trained with 95% of the remaining PDBbind general set randomly selected for training, while the remaining 5% is used for validation.

3. ARCHITECTURE

T-ALPHA employs a hierarchical multimodal transformer-based architecture designed to integrate multiple complementary feature representations from three channels (protein, ligand, and protein−ligand complex) to accurately predict binding affinity.

The protein channel contains three architectural components: (1) an E(n) EGNN that processes a graph representation of the protein pocket, (2) an adaptation of the dMaSIF[111] model based on quasi-geodesic convolutions that processes a point cloud representation of the protein surface, and (3) an MLP that processes the protein amino acid sequence-based embeddings. The output of the dMaSIF-based model is passed to a transformer encoder, and the outputs of the E(n) EGNN and the MLP are passed to separate transformer decoders. Each of the decoders engages in cross-attention with the encoder output, enabling the integration of geometric features of the protein pocket surface into both the graph-based connectivity features and the sequence-based evolutionary and functional features. The outputs from each of the three transformers are then concatenated and reduced in dimensionality to match the output dimensionality of the protein−ligand complex E(n) EGNN, ultimately resulting in a single embedding to describe the protein channel.

For the ligand channel, an analogous architectural framework is employed, containing three components tailored specifically to capture the multifaceted characteristics of the ligand: (1) an E(n) EGNN that processes a graph

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

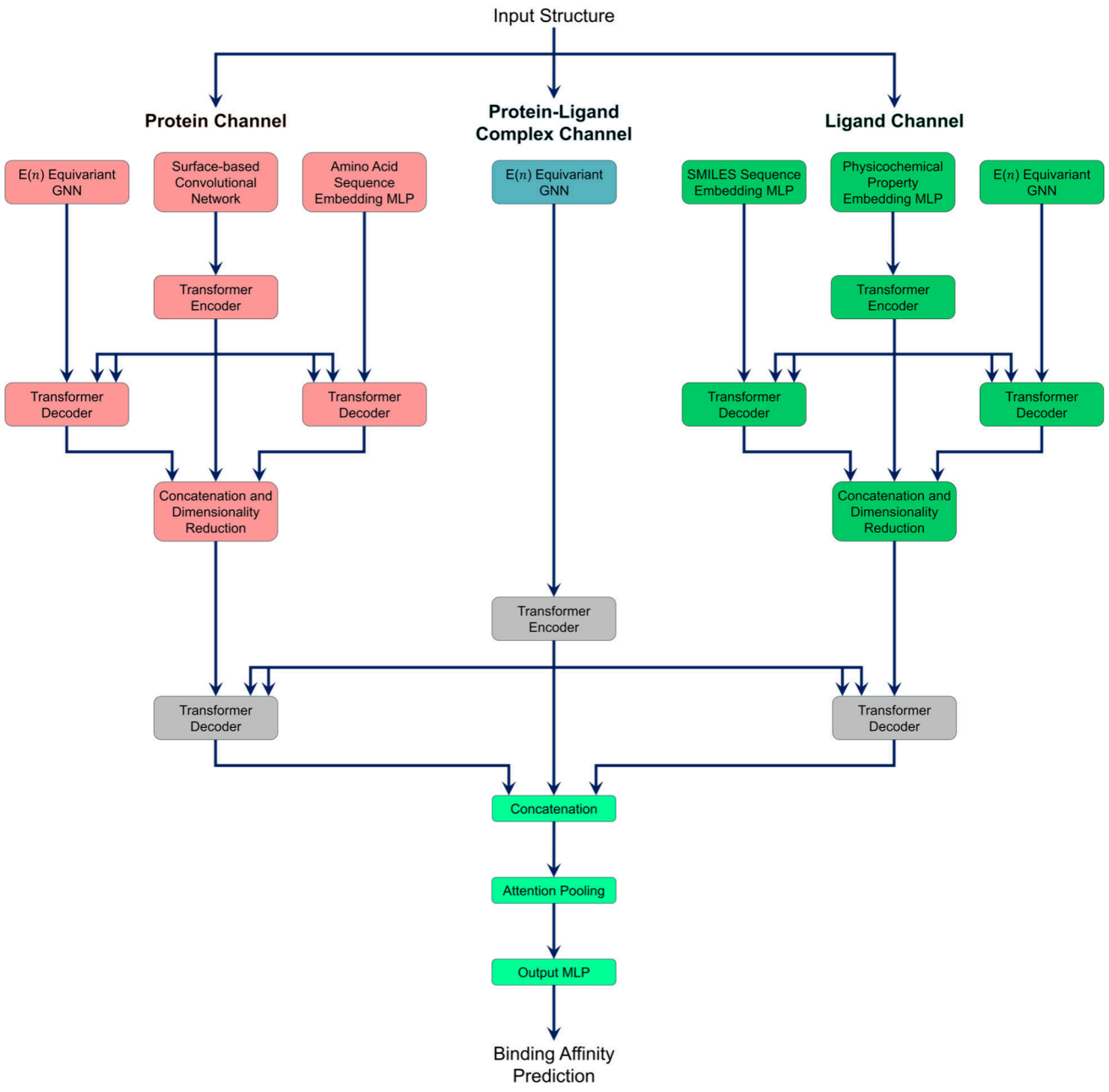


Figure 3. Overview of the hierarchical multimodal transformer-based architecture of T-ALPHA for protein−ligand binding affinity prediction. A given input protein−ligand complex structure is first processed by protein, ligand, and protein−ligand complex channels. The outputs of each of the three channels are then further processed by transformers, the outputs of which are then concatenated, pooled with an attention-based method, and passed to a final MLP to generate the output prediction.

representation of the ligand, (2) an MLP that processes the ligand physicochemical property-based embeddings, and (3) an MLP that processes the ligand SMILES sequence-based embeddings. The output of the MLP that processes the property-based embeddings is passed to a transformer encoder, and the outputs of the E(n) EGNN and the MLP that processes the sequence-based embeddings are passed to separate transformer decoders. Each decoder engages in cross-attention with the encoder output, enabling the integration of molecular-level physicochemical features into both the graph-based connectivity features and the sequence-based chemical and structural features. Similarly to as is done for the protein channel, the outputs from each of the three transformers are then concatenated and reduced in dimensionality to match the output dimensionality of the protein−ligand complex E(n) EGNN, ultimately resulting in a single embedding to describe the ligand channel.

The protein−ligand complex channel focuses exclusively on processing the graph of the bound complex using an E(n) EGNN. This approach captures the spatial relationships and interactions between the protein and ligand atoms that are important for accurately predicting binding affinity.

The outputs from all three channels are integrated using the proposed hierarchical transformer framework (Figure 3). The output of the protein−ligand complex channel is passed to a transformer encoder, and the outputs of the protein and ligand channels are passed to separate transformer decoders. Each decoder engages in cross-attention with the encoder output,

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

enabling the integration of detailed binding interaction information from the protein−ligand complex with the rich representations of the individual protein and ligand. The outputs from the three transformer layers are then concatenated and pooled with an attention mechanism. Specifically, a linear layer computes an attention score for each position across the concatenated outputs, which are normalized using a softmax function to produce attention weights. These weights are applied to the corresponding embeddings at each position, and a weighted sum is computed over all positions. The resulting vector is projected through a linear layer to produce the pooled representation. This representation is then passed to a final MLP to produce the output prediction.

3.1. Protein Channel. *3.1.1. E(n) Equivariant Graph Neural Network.* To capture the connectivity-based structural features of the protein pocket while preserving E(n) equivariance (i.e., equivariance under Euclidean transformations including rotations, translations, reflections, and permutations within n-dimensional Euclidean space), we employ an adaptation of the E(n) EGNN presented by Satorras et al.[115] This architectural component ensures that the learned graph representation accurately reflects the spatial relationships and geometric structure of the protein pocket, enabling the model to capture critical interactions and patterns that are independent of the molecular orientation or position.

In our E(n) EGNN, each node i represents a nonhydrogen atom in the protein pocket, characterized by initial features $\mathbf{h}_i$ that encode atomic properties (Table 1), and coordinates $\mathbf{x}_i$ representing the atom's 3D position. Edges (i,j) connect covalently bonded atoms and include edge features $\mathbf{e}_{ij}$ (Table 2).

The E(n) EGNN updates both the node features and coordinates through iterative message passing, where messages ($\mathbf{m}_{ij}$) are computed along the edges:

$$\mathbf{m}_{ij} = \phi_e(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_i - \mathbf{x}_j\|^2, \mathbf{e}_{ij}) \tag{1}$$

where $\|\mathbf{x}_i - \mathbf{x}_j\|^2$ is the squared Euclidean distance between nodes i and j, and ϕ_e is an MLP that computes edge-specific messages $\mathbf{m}_{ij}$ by integrating node features, edge features, and geometric distance. In addition, we apply a learned attention mechanism that assigns a weight to each edge, allowing the model to focus on more important interactions. For the protein pocket graph, each atom's feature vector $\mathbf{h}_i$ has a dimensionality of 31, and each edge feature vector $\mathbf{e}_{ij}$ has a dimensionality of 6.

The node coordinates are updated based on the messages:

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + \frac{1}{|N(i)|}\sum_{j\in N(i)} (\mathbf{x}_i - \mathbf{x}_j) \odot \phi_x(\mathbf{m}_{ij}) \tag{2}$$

where $N(i)$ denotes the neighbors of node i, ϕ_x is a coordinate-based MLP, and $\odot$ denotes element-wise multiplication. This update mechanism ensures that coordinate transformations depend on relative positions and learned messages, thus maintaining E(n) equivariance. We aggregate the coordinate updates by taking the mean over the neighbors.

The node features are updated using the aggregated messages:

$$\mathbf{h}_i \leftarrow \mathbf{h}_i + \phi_h\left(\mathbf{h}_i, \sum_{j\in N(i)} \mathbf{m}_{ij}\right) \tag{3}$$

where ϕ_h is an MLP that integrates the incoming messages to refine the node features. We incorporate residual connections by adding the original node features $\mathbf{h}_i$ to the output of ϕ_h, thus stabilizing the training.

The protein pocket E(n) EGNN consists of four layers, with each layer applying updates to the node features and coordinates to progressively refine the protein pocket graph representation. Each node's final feature vector has a dimensionality of 64, capturing the spatial and relational information necessary for accurately predicting protein−ligand binding affinity.

3.1.2. Quasi-Geodesic Convolutional Layer. To capture the detailed geometry and curvature of the protein pocket surface, we employ a dMaSIF[111]-based quasi-geodesic convolutional layer to process a surface-based point cloud representation of the protein pocket.

The protein pocket surface is represented as a point cloud of 512 points, where each point $\mathbf{x}_i \in \mathbb{R}^3$ lies on the molecular surface of the protein. The selection of 512 surface points is performed by identifying the closest points to the ligand rather than uniformly sampling across the entire protein surface. As a result, this selection process is independent of the total protein size and instead ensures that the extracted surface representation consistently focuses on the binding pocket across different proteins. For each point $\mathbf{x}_i$, we compute a smoothed normal vector $\mathbf{n}_i$ by averaging the normals of neighboring points $\mathbf{n}_j$ within a Gaussian kernel:

$$\mathbf{n}_i = \frac{\sum_j w_{ij}\mathbf{n}_j}{\|\sum_j w_{ij}\mathbf{n}_j\|} \tag{4}$$

where each weight w_{ij} is defined as

$$w_{ij} = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2s^2} \tag{5}$$

and s is a scale parameter controlling the spread of the Gaussian smoothing window. We implement five different scale parameter values: 1.0, 2.0, 3.0, 5.0, and 10.0, for computing five geometric features at each point. Using the smoothed normal vector $\mathbf{n}_i$, we compute two orthogonal tangent vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ to form an orthonormal basis $[\mathbf{n}_i,\mathbf{u}_i,\mathbf{v}_i]$ at each point, thus providing a local coordinate system at each point on the surface.

We estimate the mean curvature H_i and Gaussian curvature K_i at each scale for each point using a local quadratic approximation of the surface. Specifically, we compute the shape operator $\mathbf{S}_i$ by solving:

$$\mathbf{S}_i = (\mathrm{Cov}(\mathbf{P}_i, \mathbf{P}_i) + \lambda^2\mathbf{I})^{-1}\mathrm{Cov}(\mathbf{P}_i, \mathbf{Q}_i) \tag{6}$$

where $\mathbf{P}_i$ and $\mathbf{Q}_i$ are matrices of coordinate differences and normal differences, respectively, that have been projected into their local tangent planes, λ is a regularization parameter that is set to 0.1 Å, and $\mathbf{I}$ is the identity matrix. Cov($\cdot$, $\cdot$) is an element-wise-weighted covariance matrix where the weights are derived from the Gaussian smoothing window. The mean and Gaussian curvatures are then given by

$$H_i = \mathrm{trace}(\mathbf{S}_i),\ K_i = \det(\mathbf{S}_i) \tag{7}$$

In addition to geometric features, we incorporate chemical information by employing a separate module with message passing, where chemical properties of atoms in proximity to a given surface point are propagated to that point. Each atom a_j is associated with a feature vector $\mathbf{f}_j$ encoding its chemical

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

properties. In our implementation, each atom's feature vector has a dimensionality of 32, incorporating the features listed in Table 1 with the addition of a one-hot encoding for the Hydrogen atom type. We apply a neural network ϕ_f to transform these features:

$$\mathbf{h}_j = \phi_f(\mathbf{f}_j) \tag{8}$$

We then perform message passing among atoms to update their features based on their 16 nearest neighboring atoms:

$$\mathbf{h}'_j = \mathbf{h}_j + \sum_{k \in N(j)} \phi_{\text{atom}}(\mathbf{h}_j, \mathbf{h}_k, d_{jk}) \tag{9}$$

where ϕ_{atom} is a neural network function, $N(j)$ denotes the 16 nearest neighboring atoms to atom j, and d_{jk} is the distance between atoms $j$and k. We use three message passing layers to iteratively update the atomic features.

We then propagate the updated atomic features to each surface point by considering the 16 nearest neighboring atoms:

$$\mathbf{g}_i = \sum_{j \in M(i)} \phi_{\text{surface}}(\mathbf{h}'_j, d_{ij}) \tag{10}$$

where ϕ_{surface} is a neural network function, $M(i)$ denotes the 16 nearest atoms to surface point i, and d_{ij} is the distance between surface point i and atom j. The resulting chemical feature vector $\mathbf{g}_i$, which has a dimensionality of 32, is concatenated with the geometric features (i.e., the mean and Gaussian curvatures computed at five scales). This results in a combined feature vector $\mathbf{c}_j$ with a dimensionality of 42.

To aggregate these combined geometric and chemical features over the surface, we apply a quasi-geodesic convolutional layer that updates the combined feature vector $\mathbf{c}_j$ at each point by aggregating information from neighboring points using a learned kernel:

$$\mathbf{c}'_i = \text{LeakyReLU}\left(\sum_j w_{ij} \cdot \psi(\mathbf{P}_{ij}) \odot \mathbf{c}_j\right) \tag{11}$$

where w_{ij} is a scalar weight based on a pseudo-geodesic distance, and ψ is a neural network acting on the local coordinate differences $\mathbf{P}_{ij}$. The weighting function used to obtain w_{ij} considers both spatial proximity and normal similarity:

$$w_{ij} = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2 \cdot [2 - (\mathbf{n}_i \cdot \mathbf{n}_j)]^2 / 2r^2} \tag{12}$$

where r is a scale parameter that controls the sensitivity of the weighting function to spatial proximity and normal similarity, which we set to 9.0 Å.

For each of the 512 surface points, the final embedding $\mathbf{c}'_i$ is a vector of dimensionality 64 that encapsulates rich geometric and chemical characteristics of the protein pocket surface.

3.1.3. Amino Acid Sequence-Based Embedding Neural Network. To incorporate global evolutionary and biophysical characteristics of the protein that can be derived from its amino acid sequence, we utilize ESM2, a protein language model trained for masked token prediction, to produce a fixed-dimensional embedding vector $\mathbf{e}_{\text{seq}} \in \mathbb{R}^{2560}$. We reduce the dimensionality of each embedding from 2560 to 512 using a linear layer, followed by batch normalization and ReLU activation:

$$\mathbf{h}_{\text{seq}} = \text{ReLU}(\text{BatchNorm}(\mathbf{W}_1 \mathbf{e}_{\text{seq}} + \mathbf{b}_1)) \tag{13}$$

where $\mathbf{W}_1 \in \mathbb{R}^{512 \times 2560}$ is a weight matrix, and $\mathbf{b}_1 \in \mathbb{R}^{512}$ is a bias vector. The resulting vector $\mathbf{h}_{\text{seq}} \in \mathbb{R}^{512}$ is then reshaped into a sequence by treating each of its 512 elements as individual tokens with scalar features:

$$\mathbf{Z}_{\text{seq}} = \mathbf{h}_{\text{seq}} \otimes \mathbf{1}^{\text{T}} \in \mathbb{R}^{512 \times 1} \tag{14}$$

where $\otimes$ denotes the tensor product operation and $\mathbf{1}$ is a vector of ones. We then expand the feature dimension of each token of the resulting vector $\mathbf{Z}_{\text{seq}} \in \mathbb{R}^{512 \times 1}$ from 1 to 64 using another linear layer, where

$$\mathbf{H}_{\text{seq}}[i, :] = \mathbf{W}_2 \mathbf{Z}_{\text{seq}}[i] + \mathbf{b}_2, \quad \text{for} \quad i = 1, \ldots, 512 \tag{15}$$

where $\mathbf{W}_2 \in \mathbb{R}^{64 \times 1}$ and $\mathbf{b}_2 \in \mathbb{R}^{64}$. The resulting representation $\mathbf{H}_{\text{seq}} \in \mathbb{R}^{512 \times 64}$ is now suitable for processing by a transformer decoder.

3.1.4. Protein Transformer. To effectively integrate the diverse protein representations obtained from the E(n) EGNN, the quasi-geodesic convolutional layer, and the amino acid sequence-based embedding neural network, we employ a protein transformer architecture composed of a transformer encoder and two transformer decoders, enabling different feature modalities to interact through cross-attention mechanisms. The key technical difference between a transformer encoder and a decoder lies in their attention mechanisms. While the encoder applies only self-attention, the decoder incorporates both self-attention and cross-attention. Specifically, the decoder in T-ALPHA follows the formulation originally introduced by Vaswani et al.[116]

The processed surface features from the dMaSIF-based model, denoted as $\mathbf{H}_{\text{surf}} \in \mathbb{R}^{512 \times 64}$ (where 512 is the number of surface points and 64 is the feature dimensionality), are passed through a transformer encoder to capture contextual relationships among the learned surface point embeddings:

$$\mathbf{Z}_{\text{enc}} = \text{TransformerEncoder}(\mathbf{H}_{\text{surf}}) \tag{16}$$

The outputs of the E(n) EGNN ($\mathbf{H}_{\text{graph}} \in \mathbb{R}^{N_{\text{nodes}} \times 64}$) and the amino acid sequence-based embedding neural network ($\mathbf{H}_{\text{seq}} \in \mathbb{R}^{512 \times 64}$) are each passed to separate transformer decoders, each of which incorporates the transformer encoder output via cross-attention:

$$\mathbf{Z}_{\text{graph}} = \text{TransformerDecoder}(\mathbf{H}_{\text{graph}}, \mathbf{Z}_{\text{enc}}) \tag{17}$$

$$\mathbf{Z}_{\text{seq}} = \text{TransformerDecoder}(\mathbf{H}_{\text{seq}}, \mathbf{Z}_{\text{enc}}) \tag{18}$$

The cross-attention mechanism allows the transformer decoders to selectively focus on relevant parts of the encoder's output. Specifically, for a decoder input $\mathbf{H}_{\text{dec}}$ (either $\mathbf{H}_{\text{graph}}$ or $\mathbf{H}_{\text{seq}}$), cross-attention is computed as

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\text{T}}}{\sqrt{d_k}}\right)\mathbf{V} \tag{19}$$

where queries $\mathbf{Q} = \mathbf{H}_{\text{dec}}\mathbf{W}_Q$, keys $\mathbf{K} = \mathbf{Z}_{\text{enc}}\mathbf{W}_K$, values $\mathbf{V} = \mathbf{Z}_{\text{enc}}\mathbf{W}_V$, and $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are learned projection matrices. The dimensionality of each of the key vectors, d_k, is used to scale the dot products.

The graph decoder output $\mathbf{Z}_{\text{graph}}$ for each protein has a uniform length within the batch during decoding, achieved by padding all graphs to match the number of nodes in the largest

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

graph in the batch. After decoding, the padding is removed, resulting in an effective length of $\mathbf{Z}_{\text{graph}}$ that matches the number of nodes in the respective graph. To obtain a fixed size representation for each protein graph, we apply a masked mean pooling over the node dimension:

$$\mathbf{h}_{\text{graph}} = \text{MaskedMeanPool}(\mathbf{Z}_{\text{graph}}, \text{mask}) \in \mathbb{R}^{B\times 64} \tag{20}$$

where the mask accounts for the variable node lengths in the batch. Here, B denotes the batch size. The pooled graph representation $\mathbf{h}_{\text{graph}} \in \mathbb{R}^{B\times 64}$ is reshaped and expanded via a learned linear layer to match the dimensionality of the other modalities, resulting in $\mathbf{Z}'_{\text{graph}} \in \mathbb{R}^{512\times 64}$.

We then concatenate the encoder output $\mathbf{Z}_{\text{enc}}$, the processed graph decoder output $\mathbf{Z}'_{\text{graph}}$, and the sequence decoder output $\mathbf{Z}_{\text{seq}}$ along the sequence length dimension to obtain a unified representation of the protein:

$$\mathbf{Z}_{\text{concat}} = \text{Concat}(\mathbf{Z}_{\text{enc}}, \mathbf{Z}'_{\text{graph}}, \mathbf{Z}_{\text{seq}}) \in \mathbb{R}^{1536\times 64} \tag{21}$$

We permute the dimensions of $\mathbf{Z}_{\text{concat}}$ to obtain $\mathbf{Z}'_{\text{concat}} \in \mathbb{R}^{B\times 64\times 1536}$, and then apply a linear projection to reduce the concatenated feature dimension from 1536 to 512. After permuting back to the original dimensions, we obtain the final protein representation $\mathbf{Z}'_{\text{protein}} \in \mathbb{R}^{512\times 64}$. This representation encapsulates the integrated information from all three modalities and is prepared for downstream processing.

3.2. Ligand Channel. *3.2.1. Ligand E(n) Equivariant Graph Neural Network.* To capture the structural and connectivity-based characteristics of the ligand, we represent each ligand as a graph and process it using the same E(n) EGNN architecture described in Section 3.1.1. The only distinction between the ligand graph and the protein pocket graph is that the ligand graph has one fewer node feature, corresponding to the amino acid indicator in the protein pocket graph.

3.2.2. Physicochemical Property-Based Embedding Neural Network. To incorporate molecular-level physicochemical properties of the ligand, we compute a vector of descriptors based on the 2D molecular structure using RDKit, resulting in a fixed-length vector $\mathbf{d}_{\text{prop}} \in \mathbb{R}^{209}$. We reshape and expand the vector using a learned linear layer, where

$$\mathbf{D}_{\text{prop}}[i, :] = \mathbf{W}_1 d_{\text{prop},i} + \mathbf{b}_1, \quad \text{for} \quad i = 1, \ldots, 209 \tag{22}$$

with $\mathbf{W}_1 \in \mathbb{R}^{64\times 1}$ and $\mathbf{b}_1 \in \mathbb{R}^{64}$. The resulting output $\mathbf{D}_{\text{prop}} \in \mathbb{R}^{209\times 64}$.

3.2.3. SMILES Sequence-Based Embedding Neural Network. To capture complementary structural and chemical features of the ligand, we utilize a transformer encoder pretrained on a large dataset of SMILES strings to extract a contextual embedding $\mathbf{e}_{\text{seq}} \in \mathbb{R}^{768}$. To integrate these embeddings into our architecture, we reduce the embedding dimensionality from 768 to 512:

$$\mathbf{h}_{\text{seq}} = \text{ReLU}(\text{BatchNorm}(\mathbf{W}_2\mathbf{e}_{\text{seq}} + \mathbf{b}_2)) \in \mathbb{R}^{512} \tag{23}$$

where $\mathbf{W}_2 \in \mathbb{R}^{512\times 768}$ and $\mathbf{b}_2 \in \mathbb{R}^{512}$. The resulting vector $\mathbf{h}_{\text{seq}}$ is then reshaped into a sequence of length 512, with each element representing a token, and its feature dimension is expanded using a learned linear layer:

$$\mathbf{S}_{\text{seq}} = \text{Linear}(\mathbf{h}_{\text{seq}} \otimes \mathbf{1}^{\text{T}}) \in \mathbb{R}^{512\times 1} \tag{24}$$

Each element of the reshaped sequence $\mathbf{S}_{\text{seq}}$ is projected into a higher-dimensional feature space by using a learned linear transformation:

$$\mathbf{S}_{\text{seq}}[i, :] = \mathbf{W}_3 h_{\text{seq}} + \mathbf{b}_3, \quad \text{for } i = 1, \ldots, 512 \tag{25}$$

with $\mathbf{W}_3 \in \mathbb{R}^{64\times 1}$ and $\mathbf{b}_3 \in \mathbb{R}^{64}$.

3.2.4. Ligand Transformer. To integrate the various ligand representations, we employ a transformer architecture analogous to that described in Section 3.1.4. The output of the physicochemical property-based embedding neural network serves as input to a transformer encoder. The outputs from the E(n) EGNN and the SMILES sequence-based embedding neural network are input into separate transformer decoders that incorporate the encoder output via cross-attention. The outputs from the three transformer components are then combined in the same manner as described for the protein transformer in Section 3.1.4. This combined ligand representation encapsulates integrated structural, physicochemical, and interaction-relevant substructural features.

3.3. Protein−Ligand Complex E(n) Equivariant Graph Neural Network. To model the critical interactions between the protein and ligand, we construct a unified graph representation of the protein−ligand complex and process it using an E(n) EGNN, as described previously in Section 3.1.1. An important distinction is that the protein and ligand E(n) EGNNs are each four layers, while the protein−ligand complex EGNN is eight layers. This increased depth allows the protein−ligand complex E(n) EGNN to capture intricate interactions between the protein and ligand atoms which are essential for accurately predicting binding affinity.

3.4. Meta Transformer. The output feature representations from each of the three channels are integrated via a meta transformer architecture that leverages cross-attention mechanisms to capture interdependencies among the protein, ligand, and complex that are critical for accurately predicting binding affinity. The output of the protein−ligand complex channel serves as the input to a transformer encoder. The outputs of the protein and ligand channels are each passed to separate transformer decoders, each of which uses the encoder output as the memory input in a cross-attention mechanism, allowing the protein and ligand modalities to attend to relevant parts of the complex encoding. The outputs of the protein−ligand complex encoder ($\mathbf{Z}_{\text{complex}}$), protein decoder ($\mathbf{Z}_{\text{protein}}$), and ligand decoder ($\mathbf{Z}_{\text{ligand}}$) are then concatenated along the sequence dimension to create a unified representation that aggregates the learned features from all three modalities:

$$\mathbf{Z}_{\text{meta}} = \text{Concat}(\mathbf{Z}_{\text{complex}}, \mathbf{Z}_{\text{protein}}, \mathbf{Z}_{\text{ligand}}) \tag{26}$$

where $\mathbf{Z}_{\text{meta}} \in \mathbb{R}^{1536\times 64}$. Attention pooling is then applied to distill the concatenated representation into a fixed-size vector $\mathbf{v}_{\text{meta}}$ that captures the most relevant information across the sequence:

$$\mathbf{v}_{\text{meta}} = \mathbf{W}_{\text{proj}}(\boldsymbol{\alpha} \odot \mathbf{Z}_{\text{meta}}) \tag{28}$$

where $\mathbf{W}_{\text{proj}}$ projects the weighted sum of features to a fixed-size output vector $\mathbf{v}_{\text{meta}} \in \mathbb{R}^{512}$, and the attention weights $\boldsymbol{\alpha} \in \mathbb{R}^{1536\times 1}$ are calculated as

$$\boldsymbol{\alpha} = \text{softmax}(\mathbf{W}_{\text{attn}}\mathbf{Z}_{\text{meta}}) \tag{27}$$

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

where $\mathbf{W}_{attn}$ is a learnable weight matrix for the attention mechanism. The pooled representation $\mathbf{v}_{meta}$ is then passed to an MLP to produce the final binding affinity prediction.

4. MODEL TRAINING DETAILS

4.1. Experimental Dynamic Range-Aware Custom Loss Function. The data from PDBbind contains target values y_i, each associated with an operator o_i from the set {=, ∼, >, <}, indicating exact equality, approximate equality, greater than, or less than relationships, respectively. To account for these relational constraints, we designed a custom loss function $\mathcal{L}$ that adjusts the penalization based on the operator associated with each target value. The loss for a given sample is defined as

$$\text{for} \quad o_i \in \{ =, \sim \}: \mathcal{L}_i = (\hat{y}_i - y_i)^2 \tag{29}$$

$$\text{for} \quad o_i \in \{ > \}: \mathcal{L}_i = \begin{cases} (\hat{y}_i - y_i)^2 & \text{if} \quad \hat{y}_i \le y_i \\ 0 & \text{if} \quad \hat{y}_i > y_i \end{cases} \tag{30}$$

$$\text{for} \quad o_i \in \{ < \}: \mathcal{L}_i = \begin{cases} (\hat{y}_i - y_i)^2 & \text{if} \quad \hat{y}_i \ge y_i \\ 0 & \text{if} \quad \hat{y}_i < y_i \end{cases} \tag{31}$$

where $\hat{y}_i$ is the model's prediction for the i-th sample. The overall loss is computed as the mean of the individual $\mathcal{L}_i$ values over the batch:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_i \tag{32}$$

where N is the number of samples in the batch. This custom loss function ensures that the model respects the dynamic ranges of the experiments used to obtain binding affinity measurements rather than assuming all labels to represent exact equalities.

4.2. Parameter Optimization. Optimization was performed using the AdamW optimizer,[117] with an initial learning rate $n_{initial} = 3 \times 10^{-4}$ and a weight decay coefficient of 1×10^{-5}. Gradient clipping with a maximum value of 0.1 was applied to ensure training stability.

We used an adaptive learning rate scheduler comprising two phases. The first phase is a warm-up period over the first $T_{warmup} = 30$ epochs, during which the learning rate increases linearly from $0.1 \times n_{initial}$ to the initial learning rate $n_{initial}$. The learning rate at epoch t during the warm-up phase is given by

$$n_t = 0.1 \times n_{initial} + \left(\frac{t}{T_{warmup}} \right) (n_{initial} - 0.1 \times n_{initial}) \tag{33}$$

Following the warm-up phase, the learning rate is adjusted using a cosine annealing schedule over the remaining epochs, decreasing the learning rate from $n_{initial}$ to a minimum learning rate $n_{min} = 3 \times 10^{-5}$ following a cosine curve:

$$n_t = n_{min} + \frac{1}{2}(n_{initial} - n_{min}) \left(1 + \cos \left(\frac{(t - T_{warmup})\pi}{T_{total} - T_{warmup}} \right) \right) \tag{34}$$

where the total number of training epochs $T_{total} = 120$.

4.3. Distributed Training and Scaling. We leveraged the Fully Sharded Data Parallel (FSDP) strategy implemented in PyTorch Lightning,[118] to shard model parameters and optimizer states across four NVIDIA A100 GPUs. This approach significantly reduces memory consumption and scales the training process efficiently, enabling an effective batch size of 128 (batch size of 32 per GPU).

4.4. Model Selection. We select the model parameters corresponding to the epoch with the lowest validation loss to be used inference. Learning curves for each of the trainings performed in this work are provided in Figures S2−S4 in the Supporting Information.

5. UNCERTAINTY-AWARE SELF-LEARNING METHOD FOR PROTEIN-SPECIFIC ALIGNMENT

To improve the ranking ability of compounds by binding affinity for specific protein targets without requiring additional experimental data, we propose a self-learning method that leverages uncertainty estimation and chemical similarity. We apply this approach to two protein targets, Mpro and EGFR, each associated with a set of ligands for which binding affinity data is available. These datasets serve as test sets for evaluating the proposed method.

The manifold hypothesis posits that high-dimensional data lies on low-dimensional manifolds embedded within the higher-dimensional space.[119] We hypothesize that structurally similar compounds are expected to cluster together on these manifolds, sharing relevant properties that contribute to protein−ligand binding affinity. To exploit this hypothesis, we select compounds with similar ECFP4s to those in the test set, thereby constructing a pseudo-training set focused on the specific regions of the chemical manifold relevant to the test set.

Using Monte Carlo dropout, we estimate the uncertainty of model predictions for these pseudo-training compounds and then update the model parameters using a weighted loss function that emphasizes low-uncertainty pseudo-labels. This approach successfully aligns the model to relevant regions of the chemical manifold, enhancing its ability to rank test set compounds by binding affinity for both Mpro and EGFR (Section 6.4).

5.1. Bayesian and Statistical Learning Theoretical Inspiration. Our method is inspired by foundational principles from Bayesian inference and statistical learning theory. In Bayesian statistics, when dealing with observations of varying uncertainty, each observation should be weighted according to its precision (inverse variance) during parameter estimation. Given a pseudo-labeled input-output pair (x_i, y_i) with associated predictive uncertainty σ_i^2, the likelihood function for the data point can be calculated as

$$p(y_i|x_i, \theta) = \mathcal{N}(\hat{y}_i(x_i; \theta), \sigma_i^2) \tag{35}$$

where $p(y_i|x_i,\theta)$ represents the probability of observing y_i given the input x_i and the model parameters θ. This probability is modeled as a Gaussian distribution $\mathcal{N}$ where the model's prediction $\hat{y}_i(x_i;\theta)$ is the mean and the predictive uncertainty σ_i^2 is the variance.

The negative log-likelihood over the dataset D composed of M points is

$$-\log p(D|\theta) = \sum_{i=1}^{M} \frac{1}{2\sigma_i^2} (y_i - \hat{y}_i(x_i; \theta))^2 + \frac{1}{2} \sum_{i=1}^{M} \log \sigma_i^2 + \text{constant} \tag{36}$$

where $p(D|\theta)$ represents the joint likelihood of all observations in D given the model parameters θ. Assuming the σ_i^2 are known

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 3. Performance of T-ALPHA and the Highest-Performing Models in the Literature on the CASF 2016 Benchmark[a]

model	RMSE	MAE	r^2	Pearson r	Spearman ρ
T-ALPHA	**1.112**	**0.875**	**0.738**	**0.869**	**0.860**
T-ALPHA†	1.134	0.893	0.721	0.857	0.843
EHIGN[51]	1.150	N/R	N/R	0.854	N/R
TopoFormer-Seq[60]	1.151	N.R	N/R	0.864	N/R
MFE[73]	1.151	0.882	N/R	0.851	N/R
LGN[54]	1.177	0.936	N/R	0.842	N/R
GIGN[39]	1.190	N/R	N/R	0.840	N/R
HAC-Net[33]	1.205	0.971	0.692	0.846	0.843
TopBP[121]	1.210	N/R	N/R	0.861	N/R
CurvAGN[45]	1.217	0.930	N/R	0.8305	N/R
AEScore[122]	1.22	N/R	N/R	0.83	0.64
AK-score[32]	1.22	N/R	N/R	0.812	0.670
PLANET[38]	1.226	0.924	N/R	0.830	N/R
DeepAtom[123]	1.232	0.904	N/R	0.831	N/R
GraphscoreDTA[37]	1.249	0.981	N/R	0.831	N/R
EGNA[43]	1.258	0.980	N/R	0.842	N/R
PerSpect ML[124]	1.265	N/R	N/R	0.840	N/R
GIaNt[40]	1.269	0.999	N/R	0.814	N/R
K_{DEEP}[30]	1.27	N/R	N/R	0.82	0.82
AGL-Score[125]	1.272	N/R	N/R	0.833	N/R
OnionNet[25]	1.278	0.984	N/R	0.816	N/R
PSH-GBT[126]	1.280	N/R	N/R	0.835	N/R
ELGN[47]	1.285	1.013	N/R	0.810	N/R
FAST[35]	1.308	1.019	0.638	0.810	0.807
BAPA[127]	1.308	1.021	N/R	0.819	0.819
SIGN[36]	1.316	1.027	N/R	0.797	N/R
TopologyNet[23]	1.34	N/R	N/R	0.81	N/R
DockingApp RF[20]	1.35	1.09	N/R	0.83	N/R
DeepDTAF[128]	1.355	1.073	N/R	0.789	N/R
DLSSAffinity[129]	1.40	N/R	N/R	0.79	N/R
DeepBindGCN[49]	1.41	N/R	N/R	0.75	N/R
Pafnucy[31]	1.42	1.13	N/R	0.78	N/R
Pair[130]	1.44	N/R	N/R	0.75	N/R
GraphBAR[41]	1.542	1.241	N/R	0.726	N/R
PointTransformer[69]	1.58	1.29	N/R	0.753	0.751
MGNN[131]	N/R	N/R	N/R	0.85	N/R
SE-OnionNet[27]	N/R	N/R	N/R	0.83	N/R
PLEC-NN[132]	N/R	N/R	N/R	0.817	N/R

[a]The table reports Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (r^2), Pearson correlation coefficient (r), and Spearman rank correlation coefficient (ρ) for each model. T-ALPHA† represents the performance of T-ALPHA evaluated on protein−ligand complex structures generated by Chai-1 rather than the crystal structures. The best value for each metric is shown in bold. Models are sorted by RMSE value in ascending order. Error metrics (RMSE and MAE) are reported in units of pK_i/pK_d. N/R indicates not reported in the literature. Values are reported with the number of significant figures provided in the original work.

and fixed and ignoring constant terms, the loss function simplifies to a precision-weighted mean squared error:

$$L(\theta) = \sum_{i=1}^{M} w_i \cdot (y_i - \hat{y}_i(x_i; \theta))^2 \tag{37}$$

This weighting scheme ensures that data points with lower uncertainty have greater influence on the loss, thereby contributing more significantly to parameter updates.

Monte Carlo dropout approximates Bayesian inference in neural networks by interpreting dropout as a variational approximation to the posterior distribution $P(\theta|D)$.[120] For each data point x_i, the predictive mean $\hat{y}_i$ and variance σ_i^2 are estimated as

$$\hat{y}_i = \frac{1}{T} \sum_{t=1}^{T} \hat{y}_i^{(t)} \tag{38}$$

$$\sigma_i^2 = \frac{1}{T} \sum_{t=1}^{T} (\hat{y}_i^{(t)} - \hat{y}_i)^2 \tag{39}$$

where $T = 100$ is the number of stochastic forward passes and $\hat{y}_i^{(t)}$ is the model prediction at iteration t.

To implement a numerically stable approximation of the precision-weighted loss (eq 37), we adopt a smoothed weighting scheme that transforms variance values σ_i^2 into weights to be incorporated into the loss function during parameter optimization:

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

$$w_i = 1 - \frac{1}{1 + e^{-s\left(\frac{\sigma_i^2 - \sigma_{\min}^2}{\sigma_{\max}^2 - \sigma_{\min}^2} - 0.5\right)}} \quad (40)$$

where $\sigma_{\min}^2$ and $\sigma_{\max}^2$ are the minimum and maximum variances in the pseudo-training dataset, respectively, and $s = 10$ is a scaling factor. This transformation ensures that weights smoothly decrease with increasing variance, promoting stability and preventing any single data point from disproportionately influencing the optimization process (Figure S5 in the Supporting Information).

5.2. Implementation Details. For each of the test ligands for a given protein target, we compute the Tanimoto similarity between the corresponding Extended Connectivity Fingerprint with a diameter of 4 (ECFP4) and those of compounds in the pretraining set (~5 million entries) used for the SMILES-based transformer encoder (Section 2.3.2.3). The Tanimoto coefficient T between two ECFP4s A and B is calculated as

$$T(A, B) = \frac{A \cdot B}{\|A\| + \|B\| - A \cdot B} \quad (42)$$

where $A \cdot B$ denotes the dot product of the binary vectors, and $\|C\|$ represents the sum of the elements in C. For each test set ligand, we select the 10 compounds from the pretraining set with the highest Tanimoto similarity, resulting in a pseudo-training set of $N' = 10 \times N$ compounds, where N is the number of data points in the test set.

For each pseudo-training compound, we generate a protein−ligand complex structure using Chai-1 (see Section 9 for Chai-1 implementation details). We then compute the pseudo-label and uncertainty-based weight for each pseudo-training data point as described in Section 5.1.

5.3. Experimental Setup and Validation. We implemented the self-learning method via two training strategies: fine-tuning a pre-trained T-ALPHA model and training the T-ALPHA architecture from scratch. Fine-tuning is performed with a learning rate of 3×10^{-5} for 200 epochs to ensure convergence. When training from scratch, we utilize the same procedure described in Sections 4.2 and 4.3, with the distinction that the model is trained for 200 epochs.

To validate the method, we performed control experiments where the uncertainty-based weights w_i were excluded from the loss function (eq 37), allowing us to directly assess the contribution of uncertainty-based weighting to improved ranking performance on test set compounds.

Model parameters for inference are selected based on validation performance. Specifically, a validation set is created using the original N test compounds. Pseudo-labels and weights are computed for each validation data point using the procedures described in Section 5.1. The model's performance is then evaluated on this validation set using the weighted loss function (eq 37), and the parameters from the epoch achieving the lowest validation loss are chosen for inference.

6. RESULTS

T-ALPHA was evaluated across multiple benchmarks to demonstrate its robustness in predicting protein−ligand binding affinity. We benchmarked on the CASF 2016 test set, widely regarded as the standard dataset for protein−ligand binding affinity scoring function assessment. While its popularity enables straightforward comparisons to existing models, we acknowledge the dataset's flaws, such as significant data leakage between training and test sets, which artificially inflates performance metrics. To address these shortcomings, we also benchmarked T-ALPHA on datasets designed to mitigate data leakage and improve generalizability assessments: LP-PDBbind and BDB2020+ test sets. Additionally, we evaluated T-ALPHA on two protein-specific test sets corresponding to Mpro and EGFR to assess its applicability in scenarios that require high accuracy for specific protein targets.

6.1. Comparative Performance on the CASF 2016 Benchmark. T-ALPHA was benchmarked on the CASF 2016 test set and demonstrated superior performance across all evaluated metrics compared with every model reported in the literature to date (Table 3). T-ALPHA achieves the lowest Root Mean Square Error (RMSE: 1.112), the lowest Mean Absolute Error (MAE: 0.875), the highest Pearson correlation coefficient (r: 0.869), the highest coefficient of determination (r^2: 0.738), and the highest Spearman rank correlation coefficient (ρ: 0.860). These results establish T-ALPHA as the current state-of-the-art deep learning model for predicting protein−ligand binding affinity (Figure S6 in the Supporting Information).

In real-world drug discovery applications, experimental structures are often unavailable or incomplete. To evaluate the robustness of T-ALPHA in such scenarios, we assessed its performance on protein−ligand complex structures generated by Chai-1 (T-ALPHA†). The results indicate that T-ALPHA† maintains excellent performance (RMSE: 1.134, MAE: 0.893, r^2: 0.721, Pearson r: 0.857, Spearman ρ: 0.843), outperforming all existing models in the literature that were evaluated using crystal structures (Figure S7 in the Supporting Information). Moreover, we observe that the predictive accuracy of T-ALPHA† is not significantly dependent on the confidence score of the Chai-1-generated structures (Figure S8 in the Supporting Information).

6.2. Assessing Generalizability Using LP-PDBbind and BDB2020+ Test Sets. To evaluate the ability of T-ALPHA to generalize to protein−ligand complexes significantly distinct from those in the training and validation distributions, we benchmarked its performance on two complementary test sets developed to assess the generalizability of protein−ligand binding affinity scoring functions: LP-PDBbind, which was designed to evaluate internal generalizability by minimizing overlap between training, validation and test sets constructed from PDBbind data, and BDB2020+, which was curated to assess external generalizability to data obtained independently of PDBbind and collected after the training and validation data (more details in Section 2.1.1).

On the LP-PDBbind test set, T-ALPHA outperforms all previously evaluated models (Table S9 and Figure S10 in the Supporting Information).

On the BDB2020+ test set, T-ALPHA achieves an RMSE of 0.969, significantly outperforming all other models that have been evaluated (Table 4; Figure S11 in the Supporting Information). When using the Chai-1-generated structures of the test set protein−ligand complexes rather than the crystal structures, the RMSE is also lower than that of any model previously evaluated on the crystal structures (Table 4; Figure S12 in the Supporting Information).

6.3. Protein-Specific Benchmarking Using Mpro and EGFR Test Sets. In many real-world drug discovery scenarios, ranking compounds by binding affinity against a specific protein target is a critical step in prioritizing candidates for

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 4. Performance of T-ALPHA and Competing Models on the BDB2020+ Test Set[a]

model	RMSE	MAE	r^2	Pearson r	Spearman ρ
T-ALPHA	0.969	0.768	0.259	**0.683**	0.531
T-ALPHA†	**0.939**	**0.740**	**0.310**	0.681	**0.544**
AutoDock Vina	1.54	N/R	N/R	0.29	N/R
IGN	1.01	N/R	N/R	0.54	N/R
RF-Score	1.18	N/R	N/R	0.51	N/R
DeepDTA	1.26	N/R	N/R	0.26	N/R
MMPD-DTA	1.206	0.969	N/R	0.488	N/R

[a]The table reports Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (r^2), Pearson correlation coefficient (r), and Spearman rank correlation coefficient (ρ) for each model. T-ALPHA† represents the performance of T-ALPHA evaluated on protein−ligand complex structures generated by Chai-1 rather than the crystal structures. The best value for each metric is shown in bold. Error metrics (RMSE and MAE) are reported in units of pK_i/pK_d. N/R indicates not reported in the literature. Values are reported with the number of significant figures provided in the original work.

further investigation. In this context, a key metric is the Spearman rank correlation coefficient (ρ), which quantifies the monotonic relationship between the rank values of the predicted and experimentally determined binding affinities.

To assess T-ALPHA's effectiveness in protein-specific applications, we independently tested its performance for two highly relevant protein targets: SARS-CoV-2 main protease (Mpro) and epidermal growth factor receptor (EGFR). For the Mpro test set, T-ALPHA achieved a Spearman ρ of 0.737, outperforming all other models that have been evaluated (Table 5; Figure S13 in the Supporting Information). Even with the Chai-1-generated structures (T-ALPHA†), the model maintained a Spearman ρ of 0.733 (Figure S14 in the Supporting Information).

Table 5. Performance of T-ALPHA and Competing Models on the Mpro Test Set[a]

model	RMSE	MAE	r^2	Pearson r	Spearman ρ
T-ALPHA	**0.650**	**0.511**	**0.397**	**0.715**	**0.737**
T-ALPHA†	0.741	0.574	0.172	0.668	0.733
AutoDock Vina	0.86	N/R	N/R	0.66	0.68
IGN	1.06	N/R	N/R	0.61	0.65
RF-Score	1.20	N/R	N/R	0.52	0.58
DeepDTA	**0.65**	N/R	N/R	0.64	0.65

[a]The table reports Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (r^2), Pearson correlation coefficient (r), and Spearman rank correlation coefficient (ρ) for each model. T-ALPHA† represents the performance of T-ALPHA evaluated on protein−ligand complex structures generated by Chai-1 rather than the crystal structures. The best value for each metric is shown in bold. Error metrics (RMSE and MAE) are reported in units of pK_i/pK_d. N/R indicates not reported in the literature. Values are reported with the number of significant figures provided in the original work.

For the EGFR test set, T-ALPHA achieved a Spearman ρ of 0.791, significantly exceeding that of all other models evaluated (Table 6; Figures S15 and S16 in the Supporting Information).

6.4. Enhancing Target-Specific Performance with the Self-Learning Method. We assessed the effectiveness of the proposed self-learning method by comparing T-ALPHA's performance on the Mpro and EGFR test sets before and after its application. For the Mpro test set, the newly trained model and fine-tuned model demonstrate significant improvements in Spearman ρ compared to the baseline, with increases of 9.91% and 5.43%, respectively (Table S17 in the Supporting Information). Importantly, the control experiments exhibit minimal change compared to the baseline, confirming that the observed improvements are attributable to the self-learning method rather than other factors. Similar trends are observed for application of the method to EGFR, with increases in Spearman ρ of 3.41% for the newly trained model and 1.14% for the fine-tuned model (Table S18 in the Supporting Information). These results highlight the effectiveness of the proposed method in enhancing the protein-specific ranking of compounds by binding affinity, although the magnitude of improvement varies depending on the specified target.

Table 6. Performance of T-ALPHA and Competing Models on the EGFR Test Set[a]

model	RMSE	MAE	r^2	Pearson r	Spearman ρ
T-ALPHA	**0.694**	**0.572**	**0.403**	**0.702**	**0.791**
T-ALPHA†	0.842	0.670	0.093	0.593	0.665
AutoDock Vina	1.17	N/R	N/R	0.38	0.36
IGN	0.70	N/R	N/R	0.65	0.62
RF-Score	0.71	N/R	N/R	0.52	0.45
DeepDTA	0.77	N/R	N/R	0.44	0.43

[a]The table reports Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (r^2), Pearson correlation coefficient (r), and Spearman rank correlation coefficient (ρ) for each model. T-ALPHA† represents the performance of T-ALPHA evaluated on protein−ligand complex structures generated by Chai-1 rather than the crystal structures. The best value for each metric is shown in bold. Error metrics (RMSE and MAE) are reported in units of pK_i/pK_d. N/R indicates not reported in the literature. Values are reported with the number of significant figures provided in the original work.

An interesting and somewhat unintuitive observation is that although Spearman ρ and Pearson r values increase for the applications of the method to both Mpro and EGFR, the RMSE and MAE values also increase. This discrepancy arises due to the systematic biases inherent in the pseudo-labels generated by the model. As absolute metrics, RMSE and MAE are sensitive to these biases, whereas correlation metrics such as Spearman ρ and Pearson r, which measure relative relationships, are largely unaffected.

6.5. Architecture Component Contributions to Performance. To validate the design choices underpinning the T-ALPHA architecture, we conducted an ablation study in which we systematically removed individual components and characterized the relative contribution of each component to the overall performance on the CASF 2016 benchmark (Table 7).

Removing the protein−ligand complex E(n) EGNN led to the most significant decline in performance among all components, with an RMSE of 1.247 (Table 7). This component explicitly models the interactions between the protein and ligand atoms, and the performance decline due to its exclusion demonstrates that capturing these intermolecular relationships is critical for accurately predicting binding affinity.

Excluding the protein quasi-geodesic convolutional layer resulted in an RMSE of 1.238 (Table 7), underscoring the importance of capturing the topography and curvature of the protein binding pocket. This layer provides critical insights into

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Table 7. Impact of Excluded Components on T-ALPHA's Performance on the CASF 2016 Benchmark[a]

excluded component	RMSE	MAE	r^2	Pearson r	Spearman ρ
baseline	1.112	0.875	0.738	0.869	0.860
protein amino acid sequence-based embedding neural network	1.132	0.888	0.728	0.866	0.858
ligand E(n) equivariant graph neural network	1.159	0.897	0.715	0.858	0.851
ligand SMILES sequence-based embedding neural network	1.212	0.933	0.688	0.866	0.852
protein E(n) equivariant graph neural network	1.233	0.967	0.678	0.865	0.860
ligand physicochemical property vector neural network	1.235	0.970	0.676	0.849	0.845
protein quasi-geodesic convolutional layer	1.238	0.940	0.675	0.842	0.830
protein−ligand complex E(n) equivariant graph neural network	1.247	0.973	0.670	0.854	0.842

[a]The table reports Root Mean Square Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (r^2), Pearson correlation coefficient (r), and Spearman rank correlation coefficient (ρ) for each model. Models are sorted by RMSE value in ascending order. Error metrics (RMSE and MAE) are reported in units of pK_i/pK_d.

potential interaction hotspots and steric compatibility, which inform the other components of the protein channel through cross-attention mechanisms. Removing the ligand physicochemical property-based embedding neural network led to an RMSE of 1.235, demonstrating that molecular-level properties of the ligand are informative of its binding behavior. Moreover, in the baseline model, the output of this component informs the other components of the ligand channel through cross-attention mechanisms.

The other components, including the protein and ligand E(n) EGNNs, as well as the protein and ligand sequence-based neural networks, each contributes meaningfully to the model's performance (Table 7). The observed declines in performance due to their removals highlight the importance of connectivity-based structural features of both the protein pocket and the ligand, patterns describing chemical and functional relationships of the ligand, and global evolutionary and functional information about the entire protein for predicting binding affinity.

The ablation study reveals that all components contribute to the overall performance of T-ALPHA, with certain modules having a more substantial impact when excluded. These results confirm that the multimodal feature representations and hierarchical transformer architecture of T-ALPHA are critical to its state-of-the-art performance. Each component captures distinct aspects of protein−ligand interactions, and their integration enables the model to account for the diverse factors that determine binding affinity.

7. DISCUSSION

In this work, we introduced T-ALPHA, a novel deep learning model designed to predict protein−ligand binding affinity by integrating multimodal feature representations through a hierarchical transformer framework. Our extensive evaluations demonstrate that T-ALPHA achieves state-of-the-art performance across multiple benchmarks, highlighting its applicability to early-stage drug discovery workflows.

On the widely recognized CASF 2016 benchmark, T-ALPHA outperforms all existing models reported in the literature, underscoring the effectiveness of our architectural approach in capturing the characteristics and interactions that determine binding affinity. Notably, even when using predicted protein−ligand complex structures rather than crystal structures, the model maintains a performance superior to that of existing models. This robustness is particularly important in real-world drug discovery projects, where experimentally determined structures are often unavailable or incomplete.

In addition, T-ALPHA demonstrates effective generalizability via the LP-PDBbind and BDB2020+ benchmarks, which were specifically designed to evaluate performance on protein−ligand complexes outside of the training distribution, outperforming all of the models that were previously evaluated.

T-ALPHA also achieves state-of-the-art performance on protein-specific test sets corresponding to SARS-CoV-2 main protease (Mpro) and the epidermal growth factor receptor (EGFR), effectively ranking compounds by binding affinity to each of the respective targets—a key requirement in prioritization and lead optimization in drug discovery pipelines. Moreover, we proposed an uncertainty-aware self-learning method for protein-specific alignment that does not require additional experimental data and demonstrated that it enhances the ability of T-ALPHA to rank compounds by binding affinity to both of the targets.

The ablation study performed revealed that each component of the architecture contributes to the overall accuracy of the model, validating the architectural design of T-ALPHA and highlighting the importance of multimodal feature integration for state-of-the-art performance.

8. OUTLOOK AND FUTURE DIRECTIONS

While T-ALPHA advances the field of protein−ligand binding affinity prediction, several challenges and opportunities remain. One of the foremost challenges is the availability of high-quality, standardized datasets that comprehensively cover the vast chemical and biological space. Current datasets suffer from inconsistencies in experimental techniques used to obtain binding affinity measurements, and limited coverage of diverse chemical structures and protein targets. Improving data quality through standardized experimental protocols, as well as expanding datasets to include a wider range of chemical entities and protein families, will significantly improve the capabilities of these models.

Despite progress, models often struggle to generalize to chemical and biological spaces beyond those represented in the training data. Developing methods to enhance generalizability, including leveraging transfer learning, zero-shot learning, and incorporating domain knowledge to guide predictions, is critical for advancing the field.

Reproducibility remains an area of ongoing improvement in the field, as many published models lack accessible or functional code, making it difficult to validate or build upon prior work. To address this gap, researchers should release fully functional code with clear documentation alongside their publications. To promote transparency and reproducibility, we have made all of our code and trained models openly available at https://github.com/gregory-kyro/T-ALPHA, enabling researchers to run T-ALPHA and reproduce all of the results presented in this paper.

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

Additionally, although T-ALPHA has been extensively validated for binding affinity scoring and ranking, its applicability to protein−ligand docking and virtual screening has not been explored in this study. These tasks are distinct from affinity prediction but are critical components of early-stage drug discovery pipelines. Future work could extend T-ALPHA to incorporate structure-based docking methodologies or adapt its ranking capabilities for virtual screening applications.

9. SOFTWARE AND IMPLEMENTATION

The implementation of T-ALPHA was conducted using PyTorch (v2.4.1+cu121)[133] and PyTorch Geometric (v2.6.0).[134] Training was performed with PyTorch Lightning.[118] Model training employed features such as *ModelCheckpoint* for saving the best-performing models and *CSVLogger* for logging training metrics. Data handling leveraged PyTorch Geometric's *DataListLoader* to batch and process graph-based datasets efficiently.

The E(n) EGNNs utilized PyTorch's core modules (*torch* and *torch.nn*) to define custom neural network layers. The dMaSIF-based component combined PyTorch, PyTorch Geometric, and KeOps.[135] The meta transformer architecture was implemented using PyTorch's *TransformerEncoder* and *TransformerDecoder* modules, with graph-level pooling operations handled by PyTorch Geometric's *global_mean_pool*. Training optimization incorporated learning rate schedulers including a combination of *LinearLR* for warm-up and *CosineAnnealingLR* for gradual learning rate decay. Metrics such as Pearson correlation coefficient were computed using SciPy[136] for model evaluation.

Chai-1 was run with three trunk cycles and 200 diffusion steps. The predicted structure with the best score was selected for downstream processing.

■ ASSOCIATED CONTENT

Data Availability Statement

All of our code and trained models are openly available at https://github.com/gregory-kyro/T-ALPHA, enabling researchers to run T-ALPHA and reproduce all of the results presented in this paper.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.4c02332.

> Details regarding the training and validation learning curves, custom loss function, CASF 2016 benchmark, LP-PDBbind benchmark, BDB2020+ benchmark, protein-specific Mpro and EGFR benchmarks, and the proposed uncertainty-aware self-learning method (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Gregory W. Kyro − *Department of Chemistry, Yale University, New Haven, Connecticut 06511, Unites States;* orcid.org/0000-0002-0095-8548; Email: gregory.kyro@yale.edu

Victor S. Batista − *Department of Chemistry, Yale University, New Haven, Connecticut 06511, Unites States;* orcid.org/0000-0002-3262-1237; Email: victor.batista@yale.edu

Authors

Anthony M. Smaldone − *Department of Chemistry, Yale University, New Haven, Connecticut 06511, Unites States;* orcid.org/0009-0008-7265-0017

Yu Shee − *Department of Chemistry, Yale University, New Haven, Connecticut 06511, Unites States;* orcid.org/0000-0002-3728-0021

Chuzhi Xu − *Department of Chemistry, Yale University, New Haven, Connecticut 06511, Unites States;* orcid.org/0009-0003-6622-4752

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.4c02332

Author Contributions

G.W.K., A.M.S., Y.S., C.X., and V.S.B. conceived the idea; G.W.K., A.M.S., Y.S., and C.X. designed the research; G.W.K., A.M.S., Y.S., and C.X. developed the software; G.W.K., A.M.S., Y.S., and C.X. performed the research; G.W.K., A.M.S., Y.S., C.X., and V.S.B. analyzed the data; G.W.K., A.M.S., Y.S., and C.X. wrote the paper; V.S.B. provided feedback on the paper. All authors have given approval to the final version of the manuscript.

Funding

National Science Foundation Graduate Research Fellowship: Grant DGE-2139841; National Science Foundation Engines Development Award − Advancing Quantum Technologies (CT): Award Number 2302908; CCI Phase I: National Science Foundation Center for Quantum Dynamics on Modular Quantum Devices (CQD-MQD): Award Number 2124511.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We acknowledge financial support from the National Science Foundation Graduate Research Fellowship under Grant DGE-2139841 [to G.W.K.], from the National Science Foundation Engines Development Award: Advancing Quantum Technologies (CT) under Award Number 2302908 [to V.S.B.], and from the CCI Phase I: National Science Foundation Center for Quantum Dynamics on Modular Quantum Devices (CQD-MQD) under Award Number 2124511 [to V.S.B.]. Additionally, we acknowledge high-performance computer time from the National Energy Research Scientific Computing Center and from the Yale University Faculty of Arts and Sciences High Performance Computing Center.

■ ABBREVIATIONS

ML, machine learning; CNNs, convolutional neural networks; GNNs, graph neural networks; EGNN, equivariant graph neural network; Mpro, SARS-CoV-2 main protease; EGFR, epidermal growth factor receptor; NMR, nuclear magnetic resonance; K_i, inhibition constant; K_d, dissociation constant; IC_{50}, half-maximal inhibitory concentration; CASF, Comparative Assessment of Scoring Functions; LP-PDBbind, Leak Proof PDBbind; MLP, multilayer perceptron; GELU, Gaussian Error Linear Unit; FSDP, fully sharded data parallel; ECFP4, extended connectivity fingerprint with a diameter of 4; RMSE, root-mean-square error; MAE, mean absolute error; r, Pearson correlation coefficient; r^2, coefficient of determination; ρ, Spearman rank correlation coefficient

REFERENCES

(1) Vos, T.; Lim, S. S.; Abbafati, C.; Abbas, K. M.; Abbasi, M.; Abbasifard, M.; Abbasi-Kangevari, M.; Abbastabar, H.; Abd-Allah, F.; Abdelalim, A.; et al. Global burden of 369 diseases and injuries in 204 countries and territories, 1990−2019: a systematic analysis for the Global Burden of Disease Study 2019. *Lancet* **2020**, *396* (10258), 1204−1222.

(2) Li, W.; Li, H.-L.; Wang, J.-Z.; Liu, R.; Wang, X. Abnormal protein post-translational modifications induces aggregation and abnormal deposition of protein, mediating neurodegenerative diseases. *Cell & Bioscience* **2024**, *14* (1), 22.

(3) Hanahan, D.; Weinberg, R. A. Hallmarks of cancer: the next generation. *Cell* **2011**, *144* (5), 646−674.

(4) Santos, R.; Ursu, O.; Gaulton, A.; Bento, A. P.; Donadi, R. S.; Bologa, C. G.; Karlsson, A.; Al-Lazikani, B.; Hersey, A.; Oprea, T. I.; et al. A comprehensive map of molecular drug targets. *Nat. Rev. Drug Discovery* **2017**, *16* (1), 19−34.

(5) Chan, J. N. Y.; Nislow, C.; Emili, A. Recent advances and method development for drug target identification. *Trends Pharmacol. Sci.* **2010**, *31* (2), 82−88.

(6) Smith, C. Drug target validation: Hitting the target. *Nature* **2003**, *422* (6929), 342−345.

(7) Ashraf, S. N.; Blackwell, J. H.; Holdgate, G. A.; Lucas, S. C. C.; Solovyeva, A.; Storer, R. I.; Whitehurst, B. C. Hit me with your best shot: Integrated hit discovery for the next generation of drug targets. *Drug Discovery Today* **2024**, *29* (10), No. 104143.

(8) Hughes, J.; Rees, S.; Kalindjian, S.; Philpott, K. Principles of early drug discovery. *Br. J. Pharmacol.* **2011**, *162* (6), 1239−1249.

(9) Zhang, D.; Luo, G.; Ding, X.; Lu, C. Preclinical experimental models of drug metabolism and disposition in drug discovery and development. *Acta Pharmaceutica Sinica B* **2012**, *2* (6), 549−561.

(10) Mohs, R. C.; Greig, N. H. Drug discovery and development: Role of basic biological research. *Alzheimer's & Dementia: Translational Research & Clinical Interventions* **2017**, *3* (4), 651−657.

(11) Abramson, J.; Adler, J.; Dunger, J.; Evans, R.; Green, T.; Pritzel, A.; Ronneberger, O.; Willmore, L.; Ballard, A. J.; Bambrick, J.; et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* **2024**, *630* (8016), 493−500.

(12) Boitreaud, J.; Dent, J.; McPartlon, M.; Meier, J.; Reis, V.; Rogozhonikov, A.; Wu, K. Chai-1: Decoding the molecular interactions of life. *bioRxiv* **2024**, na.

(13) Corso, G.; Deng, A.; Fry, B.; Polizzi, N.; Barzilay, R.; Jaakkola, T. Deep Confident Steps to New Pockets: Strategies for Docking Generalization. *arXiv:2402.18396 [q-bio.BM]* **2024**, na.

(14) Walters, W. P.; Barzilay, R. Applications of Deep Learning in Molecule Generation and Molecular Property Prediction. *Acc. Chem. Res.* **2021**, *54* (2), 263−270.

(15) Kyro, G. W.; Martin, M. T.; Watt, E. D.; Batista, V. S. CardioGenAI: A Machine Learning-Based Framework for Re-Engineering Drugs for Reduced hERG Liability. *arXiv:2403.07632 [cs.LG]* **2024**, na.

(16) Niazi, S. K.; Mariam, Z. Computer-Aided Drug Design and Drug Discovery: A Prospective Analysis. *Pharmaceuticals* **2024**, *17* (1), 22.

(17) Ballester, P. J.; Mitchell, J. B. A machine learning approach to predicting protein−ligand binding affinity with applications to molecular docking. *Bioinformatics* **2010**, *26* (9), 1169−1175.

(18) Boyles, F.; Deane, C. M.; Morris, G. M. Learning from the ligand: using ligand-based features to improve binding affinity prediction. *Bioinformatics* **2020**, *36* (3), 758−764.

(19) Zilian, D.; Sotriffer, C. A. Sfcscore rf: a random forest-based scoring function for improved affinity prediction of protein−ligand complexes. *J. Chem. Inf. Model.* **2013**, *53* (8), 1923−1933.

(20) Macari, G.; Toti, D.; Pasquadibisceglie, A.; Polticelli, F. DockingApp RF: a state-of-the-art novel scoring function for molecular docking in a user-friendly interface to AutoDock Vina. *International Journal of Molecular Sciences* **2020**, *21* (24), 9548.

(21) Durrant, J. D.; McCammon, J. A. NNScore: a neural-network-based scoring function for the characterization of protein− ligand complexes. *J. Chem. Inf. Model.* **2010**, *50* (10), 1865−1871.

(22) Wang, H. Prediction of protein−ligand binding affinity via deep learning models. *Briefings in Bioinformatics* **2024**, *25* (2), na.

(23) Cang, Z.; Wei, G.-W. TopologyNet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions. *PLoS computational biology* **2017**, *13* (7), No. e1005690.

(24) Öztürk, H.; Özgür, A.; Ozkirimli, E. DeepDTA: deep drug−target binding affinity prediction. *Bioinformatics* **2018**, *34* (17), i821−i829.

(25) Zheng, L.; Fan, J.; Mu, Y. Onionnet: a multiple-layer intermolecular-contact-based convolutional neural network for protein−ligand binding affinity prediction. *ACS omega* **2019**, *4* (14), 15956−15965.

(26) Wang, Z.; Zheng, L.; Liu, Y.; Qu, Y.; Li, Y.-Q.; Zhao, M.; Mu, Y.; Li, W. OnionNet-2: a convolutional neural network model for predicting protein-ligand binding affinity based on residue-atom contacting shells. *Frontiers in chemistry* **2021**, *9*, No. 753002.

(27) Wang, S.; Liu, D.; Ding, M.; Du, Z.; Zhong, Y.; Song, T.; Zhu, J.; Zhao, R. SE-OnionNet: a convolution neural network for protein−ligand binding affinity prediction. *Frontiers in Genetics* **2021**, *11*, No. 607824.

(28) Wang, X.; Liu, D.; Zhu, J.; Rodriguez-Paton, A.; Song, T. CSConv2d: a 2-D structural convolution neural network with a channel and spatial attention mechanism for protein-ligand binding affinity prediction. *Biomolecules* **2021**, *11* (5), 643.

(29) Gomes, J.; Ramsundar, B.; Feinberg, E. N.; Pande, V. S. Atomic convolutional networks for predicting protein-ligand binding affinity. *arXiv:1703.10603 [cs.LG]* **2017**, na.

(30) 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* (2), 287−296.

(31) Stepniewska-Dziubinska, M. M.; Zielenkiewicz, P.; Siedlecki, P. Development and evaluation of a deep learning model for protein−ligand binding affinity prediction. *Bioinformatics* **2018**, *34* (21), 3666−3674.

(32) Kwon, Y.; Shin, W.-H.; Ko, J.; Lee, J. AK-score: accurate protein-ligand binding affinity prediction using an ensemble of 3D-convolutional neural networks. *International journal of molecular sciences* **2020**, *21* (22), 8424.

(33) Kyro, G. W.; Brent, R. I.; Batista, V. S. Hac-net: A hybrid attention-based convolutional neural network for highly accurate protein−ligand binding affinity prediction. *J. Chem. Inf. Model.* **2023**, *63* (7), 1947−1960.

(34) Wang, Y.; Wei, Z.; Xi, L. Sfcnn: a novel scoring function based on 3D convolutional neural network for accurate and stable protein−ligand affinity prediction. *BMC Bioinformatics* **2022**, *23* (1), 222.

(35) Jones, D.; Kim, H.; Zhang, X.; Zemla, A.; Stevenson, G.; Bennett, W. D.; Kirshner, D.; Wong, S. E.; Lightstone, F. C.; Allen, J. E. Improved protein−ligand binding affinity prediction with structure-based deep fusion inference. *J. Chem. Inf. Model.* **2021**, *61* (4), 1583−1592.

(36) Li, S.; Zhou, J.; Xu, T.; Huang, L.; Wang, F.; Xiong, H.; Huang, W.; Dou, D.; Xiong, H. Structure-aware Interactive Graph Neural Networks for the Prediction of Protein-Ligand Binding Affinity. Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, Aug 14−18, 2021, ACM, 2021.

(37) Wang, K.; Zhou, R.; Tang, J.; Li, M. GraphscoreDTA: optimized graph neural network for protein−ligand binding affinity prediction. *Bioinformatics* **2023**, *39* (6), na DOI: 10.1093/bioinformatics/btad340.

(38) Zhang, X.; Gao, H.; Wang, H.; Chen, Z.; Zhang, Z.; Chen, X.; Li, Y.; Qi, Y.; Wang, R. PLANET: A Multi-objective Graph Neural Network Model for Protein−Ligand Binding Affinity Prediction. *J. Chem. Inf. Model.* **2024**, *64* (7), 2205−2220.

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX

(39) Yang, Z.; Zhong, W.; Lv, Q.; Dong, T.; Yu-Chian Chen, C. Geometric Interaction Graph Neural Network for Predicting Protein−Ligand Binding Affinities from 3D Structures (GIGN). *J. Phys. Chem. Lett.* **2023**, *14* (8), 2020−2033.
(40) Li, S.; Zhou, J.; Xu, T.; Huang, L.; Wang, F.; Xiong, H.; Huang, W.; Dou, D.; Xiong, H. GIaNt: Protein-Ligand Binding Affinity Prediction via Geometry-Aware Interactive Graph Neural Network. *IEEE Transactions on Knowledge and Data Engineering* **2024**, *36* (5), 1991−2008.
(41) Son, J.; Kim, D. Development of a graph convolutional neural network model for efficient prediction of protein-ligand binding affinities. *PLoS One* **2021**, *16* (4), No. e0249404.
(42) Mastropietro, A.; Pasculli, G.; Bajorath, J. Learning characteristics of graph neural networks predicting protein−ligand affinities. *Nature Machine Intelligence* **2023**, *5* (12), 1427−1436.
(43) Xia, C.; Feng, S.-H.; Xia, Y.; Pan, X.; Shen, H.-B. Leveraging scaffold information to predict protein−ligand binding affinity with an empirical graph neural network. *Briefings in Bioinformatics* **2023**, *24* (1), na DOI: 10.1093/bib/bbac603.
(44) Knutson, C.; Bontha, M.; Bilbrey, J. A.; Kumar, N. Decoding the protein−ligand interactions using parallel graph neural networks. *Sci. Rep.* **2022**, *12* (1), 7624.
(45) Wu, J.; Chen, H.; Cheng, M.; Xiong, H. CurvAGN: Curvature-based Adaptive Graph Neural Networks for Predicting Protein-Ligand Binding Affinity. *BMC Bioinformatics* **2023**, *24* (1), 378.
(46) Shen, H.; Zhang, Y.; Zheng, C.; Wang, B.; Chen, P. A Cascade Graph Convolutional Network for Predicting Protein−Ligand Binding Affinity. *International Journal of Molecular Sciences* **2021**, *22* (8), 4023.
(47) Yi, Y.; Wan, X.; Zhao, K.; Ou-Yang, L.; Zhao, P. Equivariant Line Graph Neural Network for Protein-Ligand Binding Affinity Prediction. *IEEE Journal of Biomedical and Health Informatics* **2024**, *28* (7), 4336−4347.
(48) Nikolaienko, T.; Gurbych, O.; Druchok, M. Complex machine learning model needs complex testing: Examining predictability of molecular binding affinity by a graph neural network. *J. Comput. Chem.* **2022**, *43* (10), 728−739.
(49) Zhang, H.; Saravanan, K. M.; Zhang, J. Z. H. DeepBindGCN: Integrating Molecular Vector Representation with Graph Convolutional Neural Networks for Protein−Ligand Interaction Prediction. *Molecules* **2023**, *28* (12), 4691.
(50) Yang, Y.; Zhang, R.; Lin, Z. Enhancing protein-ligand binding affinity prediction through sequential fusion of graph and convolutional neural networks. *J. Comput. Chem.* **2024**, *45* (32), 2929−2940.
(51) Yang, Z.; Zhong, W.; Lv, Q.; Dong, T.; Chen, G.; Chen, C. Y. C. Interaction-Based Inductive Bias in Graph Neural Networks: Enhancing Protein-Ligand Binding Affinity Predictions From 3D Structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2024**, *46* (12), 8191−8208.
(52) Mqawass, G.; Popov, P. graphLambda: Fusion Graph Neural Networks for Binding Affinity Prediction. *J. Chem. Inf. Model.* **2024**, *64* (7), 2323−2330.
(53) Jiao, Q.; Qiu, Z.; Wang, Y.; Chen, C.; Yang, Z.; Cui, X. Edge-Gated Graph Neural Network for Predicting Protein-Ligand Binding Affinities. *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* **2021**, 334−339.
(54) Guo, J. Improving structure-based protein-ligand affinity prediction by graph representation learning and ensemble learning. *PLoS One* **2024**, *19* (1), No. e0296676.
(55) Nguyen, T.; Le, H.; Quinn, T. P.; Nguyen, T.; Le, T. D.; Venkatesh, S. GraphDTA: predicting drug−target binding affinity with graph neural networks. *Bioinformatics* **2021**, *37* (8), 1140−1147.
(56) Gale-Day, Z. J.; Shub, L.; Chuang, K. V.; Keiser, M. J. Proximity Graph Networks: Predicting Ligand Affinity with Message Passing Neural Networks. *J. Chem. Inf. Model.* **2024**, *64* (14), 5439−5450.
(57) Yang, Z.; Zhong, W.; Zhao, L.; Yu-Chian Chen, C. MGraphDTA: deep multiscale graph neural network for explainable drug−target binding affinity prediction. *Chemical Science* **2022**, *13* (3), 816−833.
(58) Jiang, M.; Wang, S.; Zhang, S.; Zhou, W.; Zhang, Y.; Li, Z. Sequence-based drug-target affinity prediction using weighted graph neural networks. *BMC Genomics* **2022**, *23* (1), 449.
(59) Monteiro, N. R. C.; Oliveira, J. L.; Arrais, J. P. DTITR: End-to-end drug−target binding affinity prediction with transformers. *Computers in Biology and Medicine* **2022**, *147*, No. 105772.
(60) Chen, D.; Liu, J.; Wei, G.-W. Multiscale topology-enabled structure-to-sequence transformer for protein−ligand interaction predictions. *Nature Machine Intelligence* **2024**, *6* (7), 799−810.
(61) Monteiro, N. R. C.; Oliveira, J. L.; Arrais, J. P. TAG-DTA: Binding-region-guided strategy to predict drug-target affinity using transformers. *Expert Systems with Applications* **2024**, *238*, No. 122334.
(62) Rose, T.; Monti, N.; Anand, N.; Shen, T. PLAPT: Protein-Ligand Binding Affinity Prediction Using Pretrained Transformers. *bioRxiv* **2024**, na.
(63) Shen, C.; Zhang, X.; Deng, Y.; Gao, J.; Wang, D.; Xu, L.; Pan, P.; Hou, T.; Kang, Y. Boosting Protein−Ligand Binding Pose Prediction and Virtual Screening Based on Residue−Atom Distance Likelihood Potential and Graph Transformer. *J. Med. Chem.* **2022**, *65* (15), 10691−10706.
(64) Zhou, C.; Li, Z.; Song, J.; Xiang, W. TransVAE-DTA: Transformer and variational autoencoder network for drug-target binding affinity prediction. *Computer Methods and Programs in Biomedicine* **2024**, *244*, No. 108003.
(65) Han, K.; Shi, C.; Wang, Z.; Liu, W.; Li, Z.; Wang, Z.; Lei, L.; Dai, R.; Wang, M.; Zhang, Z.; et al. Innovative Mamba and graph transformer framework for superior protein-ligand affinity prediction. *Microchemical Journal* **2024**, *206*, No. 111444.
(66) Amine, A. M. E.; Fadila, A. Transformer neural network for protein-specific drug discovery and validation using QSAR. *Journal of Proteins and Proteomics* **2023**, *14* (4), 253−262.
(67) Vasan, A.; Gokdemir, O.; Brace, A.; Ramanathan, A.; Brettin, T.; Stevens, R.; Vishwanath, V. High Performance Binding Affinity Prediction with a Transformer-Based Surrogate Model. *2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* **2024**, 571−580.
(68) Tian, C.; Wang, L.; Cui, Z.; Wu, H. GTAMP-DTA: Graph transformer combined with attention mechanism for drug-target binding affinity prediction. *Computational Biology and Chemistry* **2024**, *108*, No. 107982.
(69) Wang, Y.; Wu, S.; Duan, Y.; Huang, Y. A point cloud-based deep learning strategy for protein−ligand binding affinity prediction. *Briefings in Bioinformatics* **2021**, *23* (1), na.
(70) Wang, J.; Hu, J.; Sun, H.; Xu, M.; Yu, Y.; Liu, Y.; Cheng, L. MGPLI: exploring multigranular representations for protein−ligand interaction prediction. *Bioinformatics* **2022**, *38* (21), 4859−4867.
(71) Tang, X.; Zhou, Y.; Yang, M.; Li, W. TC-DTA: Predicting Drug-Target Binding Affinity With Transformer and Convolutional Neural Networks. *IEEE Transactions on NanoBioscience* **2024**, *23* (4), 572−578.
(72) Li, Q.; Zhang, X.; Wu, L.; Bo, X.; He, S.; Wang, S. PLA-MoRe: A Protein−Ligand Binding Affinity Prediction Model via Comprehensive Molecular Representations. *J. Chem. Inf. Model.* **2022**, *62* (18), 4380−4390.
(73) Xu, S.; Shen, L.; Zhang, M.; Jiang, C.; Zhang, X.; Xu, Y.; Liu, J.; Liu, X. Surface-based multimodal protein−ligand binding affinity prediction. *Bioinformatics* **2024**, *40* (7), na DOI: 10.1093/bioinformatics/btae413.
(74) Yan, X.; Liu, Y. Graph−sequence attention and transformer for predicting drug−target affinity. *RSC Adv.* **2022**, *12* (45), 29525−29534.
(75) Wu, H.; Liu, J.; Jiang, T.; Zou, Q.; Qi, S.; Cui, Z.; Tiwari, P.; Ding, Y. AttentionMGT-DTA: A multi-modal drug-target affinity prediction using graph transformer and attention mechanism. *Neural Networks* **2024**, *169*, 623−636.
(76) Liu, Y.; Xing, L.; Zhang, L.; Cai, H.; Guo, M. GEFormerDTA: drug target affinity prediction based on transformer graph for early fusion. *Sci. Rep.* **2024**, *14* (1), 7416.

(77) Saadat, M.; Behjati, A.; Zare-Mirakabad, F.; Gharaghani, S. Drug-Target Binding Affinity Prediction Using Transformers. *bioRxiv* **2022**, na.
(78) Sun, X.; Huang, J.; Fang, Y.; Jin, Y.; Wu, J.; Wang, G.; Jia, J. MREDTA: A BERT and transformer-based molecular representation encoder for predicting drug-target binding affinity. *FASEB J.* **2024**, *38* (19), No. e70083.
(79) Li, Z.; Ren, P.; Yang, H.; Zheng, J.; Bai, F. TEFDTA: a transformer encoder and fingerprint representation combined prediction method for bonded and non-bonded drug−target affinities. *Bioinformatics* **2023**, *40* (1), na.
(80) Hu, F.; Hu, Y.; Zhang, J.; Wang, D.; Yin, P. Structure Enhanced Protein-Drug Interaction Prediction using Transformer and Graph Embedding. *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* **2020**, 1010−1014.
(81) Quan, L.; Wu, J.; Jiang, Y.; Pan, D.; Qiang, L. DTA-GTOmega: Enhancing Drug-Target Binding Affinity Prediction with Graph Transformers Using OmegaFold Protein Structures. *J. Mol. Biol.* **2024**, No. 168843.
(82) Liu, S.; Wang, Y.; Deng, Y.; He, L.; Shao, B.; Yin, J.; Zheng, N.; Liu, T.-Y.; Wang, T. Improved drug−target interaction prediction with intermolecular graph transformer. *Briefings in Bioinformatics* **2022**, *23* (5), na DOI: 10.1093/bib/bbac162.
(83) Gorantla, R.; Kubincová, A.; Suutari, B.; Cossins, B. P.; Mey, A. S. J. S. Benchmarking Active Learning Protocols for Ligand-Binding Affinity Prediction. *J. Chem. Inf. Model.* **2024**, *64* (6), 1955−1965.
(84) 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* (12), 2977−2980.
(85) Wang, R.; Fang, X.; Lu, Y.; Yang, C. Y.; Wang, S. The PDBbind database: methodologies and updates. *J. Med. Chem.* **2005**, *48* (12), 4111−4119.
(86) Liu, Z.; Li, Y.; Han, L.; Li, J.; Liu, J.; Zhao, Z.; Nie, W.; Liu, Y.; Wang, R. PDB-wide collection of binding data: current status of the PDBbind database. *Bioinformatics* **2015**, *31* (3), 405−412.
(87) Liu, Z.; Su, M.; Han, L.; Liu, J.; Yang, Q.; Li, Y.; Wang, R. Forging the Basis for Developing Protein−Ligand Interaction Scoring Functions. *Acc. Chem. Res.* **2017**, *50* (2), 302−309.
(88) 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* (2), 895−913.
(89) Volkov, M.; Turk, J.-A.; Drizard, N.; Martin, N.; Hoffmann, B.; Gaston-Mathé, Y.; Rognan, D. On the Frustration to Predict Binding Affinities from Protein−Ligand Structures with Deep Neural Networks. *J. Med. Chem.* **2022**, *65* (11), 7946−7958.
(90) Li, J.; Guan, X.; Zhang, O.; Sun, K.; Wang, Y.; Bagni, D.; Head-Gordon, T. Leak Proof PDBBind: A Reorganized Dataset of Protein-Ligand Complexes for More Generalizable Binding Affinity Prediction. *arXiv:2308.09639 [physics.bio-ph]* **2024**, na.
(91) Dice, L. R. Measures of the Amount of Ecologic Association Between Species. *Ecology* **1945**, *26* (3), 297−302.
(92) Needleman, S. B.; Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* **1970**, *48* (3), 443−453.
(93) Wang, D. D.; Xie, H.; Yan, H. Proteo-chemometrics interaction fingerprints of protein-ligand complexes predict binding affinity. *Bioinformatics* **2021**, *37* (17), 2570−2579.
(94) Bajusz, D.; Rácz, A.; Héberger, K. Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of Cheminformatics* **2015**, *7* (1), 20.
(95) Liu, T.; Lin, Y.; Wen, X.; Jorissen, R. N.; Gilson, M. K. BindingDB: a web-accessible database of experimentally determined protein−ligand binding affinities. *Nucleic Acids Res.* **2006**, *35* (suppl_1), D198−D201.
(96) Burley, S. K.; Bhikadiya, C.; Bi, C.; Bittrich, S.; Chao, H.; Chen, L.; Craig, P. A.; Crichlow, G. V.; Dalenberg, K.; Duarte, J. M.; et al. RCSB Protein Data Bank (RCSB.org): delivery of experimentally-determined PDB structures alongside one million computed structure models of proteins from artificial intelligence/machine learning. *Nucleic Acids Res.* **2023**, *51* (D1), D488−D508.
(97) Gao, K.; Wang, R.; Chen, J.; Tepe, J. J.; Huang, F.; Wei, G.-W. Perspectives on SARS-CoV-2 Main Protease Inhibitors. *J. Med. Chem.* **2021**, *64* (23), 16922−16955.
(98) Herbst, R. S. Review of epidermal growth factor receptor biology. *Int. J. Radiat Oncol Biol. Phys.* **2004**, *59* (2 Suppl), 21−26.
(99) Kyro, G. W.; Morgunov, A.; Brent, R. I.; Batista, V. S. ChemSpaceAL: An Efficient Active Learning Methodology Applied to Protein-Specific Molecular Generation. *J. Chem. Inf. Model.* **2024**, *64* (3), 653−665.
(100) Mendez, D.; Gaulton, A.; Bento, A. P.; Chambers, J.; De Veij, M.; Félix, E.; Magariños, M. P.; Mosquera, J. F.; Mutowo, P.; Nowotka, M. ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research* **2019**, *47* (D1), D930−D940.
(101) Brown, N.; Fiscato, M.; Segler, M. H.; Vaucher, A. C. GuacaMol: benchmarking models for de novo molecular design. *J. Chem. Inf. Model.* **2019**, *59* (3), 1096−1108.
(102) Polykovskiy, D.; Zhebrak, A.; Sanchez-Lengeling, B.; Golovanov, S.; Tatanov, O.; Belyaev, S.; Kurbanov, R.; Artamonov, A.; Aladinskiy, V.; Veselov, M.; et al. Molecular sets (MOSES): a benchmarking platform for molecular generation models. *Frontiers in Pharmacology* **2020**, *11*, No. 565644.
(103) Cock, P. J.; Antao, T.; Chang, J. T.; Chapman, B. A.; Cox, C. J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **2009**, *25* (11), 1422−1423.
(104) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L.-P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; et al. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Computational Biology* **2017**, *13* (7), No. e1005659.
(105) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *Journal of Cheminformatics* **2011**, *3* (1), 33.
(106) RDKit: Open-Source Cheminformatics Software. https://www.rdkit.org (accessed 2024).
(107) Los Alamos National Laboratory. Periodic Table of Elements. https://periodic.lanl.gov/index.shtml (accessed 2024).
(108) Gasteiger, J.; Marsili, M. A new model for calculating atomic charges in molecules. *Tetrahedron Lett.* **1978**, *19* (34), 3181−3184.
(109) NCBI. Electronegativity in the Periodic Table of Elements, 2025. https://pubchem.ncbi.nlm.nih.gov/periodic-table/electronegativity (accessed January 26, 2025).
(110) Schwerdtfeger, P.; Nagle, J. K. 2018 Table of static dipole polarizabilities of the neutral elements in the periodic table*. *Mol. Phys.* **2019**, *117* (9−12), 1200−1225.
(111) Sverrisson, F.; Feydy, J.; Correia, B. E.; Bronstein, M. M. Fast end-to-end learning on protein surfaces. *bioRxiv* **2020**, na.
(112) Lin, F.-Y.; Liu, C.-I.; Liu, Y.-L.; Zhang, Y.; Wang, K.; Jeng, W.-Y.; Ko, T.-P.; Cao, R.; Wang, A. H.-J.; Oldfield, E. Mechanism of action and inhibition of dehydrosqualene synthase. *Proc. Natl. Acad. Sci. U. S. A.* **2010**, *107* (50), 21337−21342.
(113) Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; Verkuil, R.; Kabeli, O.; Shmueli, Y.; et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **2023**, *379* (6637), 1123−1130.
(114) Bouysset, C.; Fiorucci, S. ProLIF: a library to encode molecular interactions as fingerprints. *Journal of Cheminformatics* **2021**, *13* (1), 72.
(115) Satorras, V. c. G.; Hoogeboom, E.; Welling, M. E(*n*) Equivariant Graph Neural Networks. Proceedings of the 38th International Conference on Machine Learning, Proceedings of Machine Learning Research, ICML 2021, Virtual Event, July 18−24, 2021, DBLP, 2021.
(116) Vaswani, A.; et al. Attention is all you need. *arXiv:1706.03762 [cs.CL]* **2017**, na.

(117) Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv:1711.05101 [cs.LG]* **2017**, na.
(118) PyTorch Lightning, 2019. https://www.pytorchlightning.ai.
(119) Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2013**, *35* (8), 1798−1828.
(120) Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv:1506.02142v6 [stat.ML]* **2015**, na.
(121) Cang, Z.; Mu, L.; Wei, G.-W. Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. *PLOS Computational Biology* **2018**, *14* (1), No. e1005929.
(122) Meli, R.; Anighoro, A.; Bodkin, M. J.; Morris, G. M.; Biggin, P. C. Learning protein-ligand binding affinity with atomic environment vectors. *Journal of Cheminformatics* **2021**, *13* (1), 59.
(123) Li, Y.; Rezaei, M. A.; Li, C.; Li, X. DeepAtom: A Framework for Protein-Ligand Binding Affinity Prediction. *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* **2019**, 303−310.
(124) Meng, Z.; Xia, K. Persistent spectral−based machine learning (PerSpect ML) for protein-ligand binding affinity prediction. *Science Advances* **2021**, *7* (19), No. eabc5329.
(125) Nguyen, D. D.; Wei, G.-W. AGL-Score: Algebraic Graph Learning Score for Protein−Ligand Binding Scoring, Ranking, Docking, and Screening. *J. Chem. Inf. Model.* **2019**, *59* (7), 3291−3304.
(126) Liu, X.; Feng, H.; Wu, J.; Xia, K. Persistent spectral hypergraph based machine learning (PSH-ML) for protein-ligand binding affinity prediction. *Briefings in Bioinformatics* **2021**, *22* (5), No. bbab127.
(127) Seo, S.; Choi, J.; Park, S.; Ahn, J. Binding affinity prediction for protein−ligand complex using deep attention mechanism based on intermolecular interactions. *BMC Bioinformatics* **2021**, *22*, 1−15.
(128) Wang, K.; Zhou, R.; Li, Y.; Li, M. DeepDTAF: a deep learning method to predict protein−ligand binding affinity. *Briefings in Bioinformatics* **2021**, *22* (5), No. bbab072.
(129) Wang, H.; Liu, H.; Ning, S.; Zeng, C.; Zhao, Y. DLSSAffinity: protein−ligand binding affinity prediction via a deep learning model. *Phys. Chem. Chem. Phys.* **2022**, *24* (17), 10124−10133.
(130) Zhu, F.; Zhang, X.; Allen, J. E.; Jones, D.; Lightstone, F. C. Binding affinity prediction by pairwise function based on neural network. *J. Chem. Inf. Model.* **2020**, *60* (6), 2766−2772.
(131) Li, X.-S.; Liu, X.; Lu, L.; Hua, X.-S.; Chi, Y.; Xia, K. Multiphysical graph neural network (MP-GNN) for COVID-19 drug design. *Briefings in Bioinformatics* **2022**, *23* (4), na DOI: 10.1093/bib/bbac231.
(132) Wójcikowski, M.; Kukiełka, M.; Stepniewska-Dziubinska, M. M.; Siedlecki, P. Development of a protein-ligand extended connectivity (PLEC) fingerprint and its application for binding affinity predictions. *Bioinformatics* **2019**, *35* (8), 1334−1341.
(133) 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. *arXiv:1912.01703 [cs.LG]* **2019**, na.
(134) Fey, M.; Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric. *arXiv:1903.02428 [cs.LG]* **2019**, na.
(135) Charlier, B.; Feydy, J.; Glaunès, J. A.; Collin, F.-D.; Durif, G. Kernel Operations on the GPU, with Autodiff, without Memory Overflows. *arXiv:2004.11127 [cs.LG]* **2020**, na.
(136) Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **2020**, *17* (3), 261−272.

https://doi.org/10.1021/acs.jcim.4c02332
J. Chem. Inf. Model. XXXX, XXX, XXX−XXX